Identification of Rotten Fruits using Deep Learning Techniques

MSc Research Project
MSc in Data Analytics

Janush Prajosh
Student ID: 22132732

School of Computing
National College of Ireland

Supervisor: Prof. Arjun Chikkankod

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Janush Prajosh		
Student ID:	22132732		
Programme:	Msc in Data Analytics	**Year:**	2023-2024
Module:	Msc Research Project		
Supervisor:	Arjun Chikkankod		
Submission Due Date:	14-12-2023		
Project Title:	Identification of Rotten Fruits using Deep Learning Techniques		
Word Count:	6864 words	**Page Count**:	18

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Janush Prajosh

Date: 14-12-2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	☑
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☑
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☑

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Identification of Rotten Fruits using Deep Learning Techniques

Janush Prajosh
22132732

Abstract

Determining whether a fruit is fresh or rotten has traditionally been a challenging task as it has always been done manually. However, using deep learning models offers a more accurate and automated approach to solving this problem. This study evaluates the performance of four deep learning models such as Inception Net V3, SGD-CNN, ResNet50, and Efficient Net V2B0, on three types of datasets that include unbalanced, majority undersampled, and minority oversampled datasets. The food wastage issue is a significant problem globally, and automating fruit quality assessment using deep learning can enhance the efficiency of the supply chain and reduce waste. This research contributes to the growing field of agricultural AI by displaying the efficacy of different models in determining the fruit's quality. Additionally, it emphasizes the importance of dataset composition in the model's performance and lays the foundation for future advancements in the field.

Keywords – Rotten fruits Detection, Deep Learning, Supply chain Management, Food waste reduction, Class balancing techniques.

1 Introduction

The field of food technology and agriculture has been transformed by the emergence of deep learning (Dandavate and Patodkar, 2020). One of the greatest challenges in this field has been identifying the quality of food, particularly distinguishing between fresh and rotten fruits. Previously, human inspectors were responsible for this task, which was both difficult and prone to errors (Hossain et al., 2018). However, deep learning techniques can now be used to make this process more accurate and automated.

Deep learning is a type of machine learning that employs neural networks with multiple layers to learn from vast amounts of data. When assessing the fruit quality, deep learning models can accurately determine the quality status of fruit images (Hossain et al., 2018). This technological advancement is crucial for various stakeholders in the supply chain such as farmers, retailers, and consumers because it enhances food safety, quality control, and reduces waste.

Food waste is a worldwide problem with around one-third of all food grown for consumption being lost or wasted. One significant contributor to this problem is the need for more time to identify and discard rotten fruits in time. To overcome this issue and improve the efficiency of the supply chain, deep learning is being used to evaluate the quality of fruit. By automating the detection process, it becomes possible to identify and discard spoiled fruits early in the supply chain resulting in less waste and ensuring consumers receive only fresh food.

This study aims to determine the effectiveness of deep learning models in identifying rotten fruits from pictures using different types of data representation. The study examines how well various models perform on unbalanced, majority undersampled, and minority oversampled datasets to determine the best approach for evaluating agricultural products. The results of this research not only benefit academia by providing insights into the use of deep learning in agriculture and providing farmers with valuable information to improve their food quality checks, reduce waste, and ensure consumer safety.

The potential impact of deep learning in identifying rotten fruits on food safety, supply chain efficiency and economic viability is significant. Implementing such systems can lead to faster and more precise sorting processes, compared to manual inspection. It also supports the broader goals of sustainable agriculture by minimizing waste and maximizing resource efficiency. From a technological standpoint, this study contributes to the evolving field of agricultural AI, demonstrating practical applications of advanced machine learning techniques in real-world scenarios.

The study contributes to the field by implementing and evaluating four different deep learning models: Inception Net V3, SGD-CNN, ResNet50, and EfficientNet. Each model has unique architectural features and capabilities in processing image data making it suitable for fruit quality assessment.

1.1 Research Question

The primary research question guiding this study is: *"How do deep learning models, specifically Inception Net V3, SGD-CNN, ResNet50, and EfficientNet, perform in the identification of rotten fruits, and how does their performance vary across datasets with unbalanced data, majority undersampled data, and minority oversampled data?"*

This section briefly introduces the readers to the study's topic, motivates to undertake it, explains its significance, and describes its aim. The following section of the study deals with the literature review related to the presented study focusing on the state of the art in the field.

2 Related Work

This literature review performs an in-depth comparison of the techniques used in classification of the fruits using Deep Learning. The section is divided into sub-sections namely, (i) Review on Machine Learning in Fruit Classification (ii) Review on Conventional Deep Learning Techniques (iii) Review on Advanced Deep Learning Architectures.

2.1 Review on Machine Learning in Fruit Classification

Classification of fruits can be an important aspect in the logistics and inventory management for businesses. Various studies have been done in the field of fruit classification using machine learning. Study conducted by Mustafa et. al. (2011) implemented Probabilistic Neural Networks (PNN) to classify the features extracted from the images of the fruits of classes apples, bananas, mangoes, carrots, and oranges. The authors employed colour and geometrical feature extraction techniques prior to the modelling. The study showcased an impressive performance in terms of accuracy with the model achieving 90% score. Accuracy although high, is not satisfactory as the business implications for such systems would be very crucial. Similarly, a study by Dubey et. al. (2012) employed K-means clustering technique to segment the fruits from images. Following the segmentation, the study involved extraction of texture features which were fed to the Support Vector Machine (SVM) model. The model achieved an impressive accuracy of 99% in classification of the fruit images. Though segmentation helps in extracting the image pixels pertaining to fruits, selecting the number of clusters for the same is a tedious task and involves many trials. Also, addition of the segmentation processes tends to increase the processing time for classification (Huang et. al., 2023).

In a similar study, Savakar (2012) investigated colour and texture-based feature extraction technique along with the use of Artificial Neural Network (ANN). The methodology implemented in the study achieved a high accuracy of 94%. The study though achieved a high accuracy involves the images of the fruits to be in bulk which might limit its ability to identify and distinguish individual fruit samples. A study by Zhang et. al. (2014) implemented split and merge algorithm for segmentation purposed followed by extraction of texture and shape features before feeding in the features to a feedforward neural network (FNN). Although, this study involves a comprehensive methodology, it could achieve an accuracy of 89.1% only indicating the difficulty that arises in the task of fruit classification.

Various machine learning modalities have been discussed in the sub-section along with different feature extraction techniques pertaining to the classification of the fruit images. The studies that involve implementation of the conventional deep learning techniques based on Convolutional Neural Networks (CNNs) are discussed in the subsection below.

2.2 Review of Conventional Deep Learning Techniques in Fruit Classification

As seen above, the conventional approach of fruit classification mostly involved extracting features such as texture, colour, and shape etc. These steps require additional computational time to implement in real life. To achieve better performance with a reduced computational requirement, CNNs are found to be a choice (Dandavate and Patodkar, 2020).

Substantial advancements have recently been made in using Convolutional Neural Networks (CNNs) in the agricultural sector for fruit classification. The deep learning models have undergone customization and evaluation on diverse datasets, demonstrating their efficacy in accurately categorizing fruits according to their visual attributes (Vasconez et. al., 2020).

In a study by Lu (2016), a 5-layer Convolutional Neural Network (CNN) model was applied to the ImageNet dataset, which consists of RGB images with dimensions of 128 × 128 × 3. The model demonstrated a performance accuracy of 74%. Nevertheless, by the implementation of data augmentation approaches, the accuracy experienced a significant increase, reaching a notable 90%. The advantage of the model resides in its adaptability to data augmentation. It is important to note that its optimal performance is dependent on the use of augmentation, which may provide a potential constraint in real-world scenarios.

The 13-layer Convolutional Neural Network (CNN) model was evaluated using the VegFru dataset in the study Zhang et. al. (2019), which consists of RGB images with dimensions of 256 × 256 × 3. The results of this study demonstrated a notable accuracy rate of 94.94%. The model's depth, consisting of 13 layers, facilitates detailed feature extraction, resulting in a notable level of accuracy. Nevertheless, this model’s intricacy may present difficulties concerning the availability of computational resources and the risk of overfitting.

In a similar study by Katarzyna and Powell (2019), a 9-layer Convolutional Neural Network (CNN) model was utilized to analyse a proprietary dataset consisting of RGB images with dimensions of 150 × 150 × 3. The model demonstrated an exceptional accuracy rate of 99.78%. Although the performance of the model is praiseworthy, it is important to acknowledge that proprietary datasets may be extensively customized, and the model's performance on datasets with greater diversity has yet to be determined.

The CNN models that have been proposed and evaluated on the Fruits-360 dataset and other datasets have demonstrated diverse levels of performance (Sakib, Ashrafi and Siddique, 2019). As an example, the model attained a flawless accuracy of 100% on the Fruits-360 dataset, which consists of RGB photos with dimensions of 100 × 100 × 3. However, while evaluating on the UEC-FOOD100 dataset, the accuracy decreased to 80.8% for the classification of individual fruits and further declined to 60.9% for the classification of multiple food items. This finding suggests a constraint in the ability to differentiate between closely related entities or many entities within a singular visual representation.

Although CNN-based models for fruit classification have demonstrated promising outcomes, exhibiting high levels of accuracy on datasets, they are not exempt from restrictions. Future study and development can be focused on various factors, including but not limited to model complexity, excessive dependence on data augmentation, and the difficulties encountered in multi-fruit classification.

2.3 Review on Advanced Deep Learning Architectures

As discussed in the previous subsection, convolutional neural networks showed high level of performance in the fruit classification problem. However, data augmentation is one of the reasons for this. This subsection tries to review the studies in the field that involve modern advanced deep learning architectures such as MobileNet, Deep Belief Networks, InceptionNet and ResNet models. These models are used in fruit classification through transfer learning.

A study by Rojas-Aranda et. al. (2020) implemented the MobileNetV2 model through transfer learning on Fruits-360 dataset. The initial model applied with weight transfer achieved a remarkable accuracy of 98% on the training dataset whereas a lower accuracy of 78% was observed for the test dataset. This suggests that the model might be overfitting the training data. Authors of the study improved upon the existing model by including features to the previous feature set. The model is then tested with 3 distinct features viz. Single color, Histogram and K-means Centroid. The model with Single color feature achieved the highest test accuracy of 95% which is exceptional.

A study by Muresan and Oltean (2018) applied deep belief networks through transfer learning on the same dataset. The duo applied and tested the model in three different color spaces viz. grayscale, RGB and HSV. The DBN model applied achieved the highest classification accuracy in the RGB colorspace with an outstanding accuracy of 98.66%. Zeng (2017) applied a modified VGG model on a personalized dataset. The author incorporated image saliency features into the VGG model. This modified model achieved an improved performance of 92.5% related to the vanilla VGG model that showed an accuracy of 89.6%. A study by Zhu et. al. (2018) implemented a modified AlexNet model onto the personalized ImageNet dataset. Authors incorporated a ReLU activation function at the model output to perform classification. The study suggests that it achieved an accuracy of 92.1%.

Hossain et al. (2018) proposed an ensemble of Light Architecture they developed and the use of VGG16 model. The application of the proposed ensemble model on the Supermarket Produce Dataset (SPD) showcased an impressive performance. The model achieved a staggering accuracy of 99.75% in fruit classification. The model that has been implemented in the study has shown an impressive accuracy, but the creation of an ensemble architecture does not allow for transfer learning of VGG16. The VGG16 model is significant and should require extensive training time and many resources for training the ensemble model.

Steinbrener, Posch and Leitner (2019) modified a GoogleNet model to incorporate and classify Pseudo-RGB color space-based fruit classification. The model was applied on the hyperspectral images of different fruits for classification. The model achieved highest performance on the pseudo-RGB model compared to the kernel and linear model.

Xue, Liu and Ma (2020) implemented and compared ResNet50 and DenseNet169 models on the Fruit 26 and Fruit 15 datasets. The models showed an average accuracy of around 92% across both the datasets. Authors also implemented an attention-based densely connected convolutional networks with convolution autoencoder (CAE-ADN) model. This model showed high accuracies of 95.64% and 93.49% on Fruit 26 and Fruit 15 datasets respectively.

From the review conducted, the fruit classification is a tough task. Transfer learning provides a faster method to make the models learn compared to conventional feature extraction and segmentation methods.

Based on the reviewed literature and identified gaps, it can be found out that, fruit classification is a difficult task to achieve with models requiring low computational resources. There is a need to investigate different modern advanced model that can be implemented through transfer learning to classify the fruits. Following section of methodology provides in-depth implementation of the system and the approach used for classification of fruits.

3 Methodology

In this section of the report, we will explore the approach taken for conducting the study that is being presented. The methodology implemented in the study is shown in Figure 1 below.

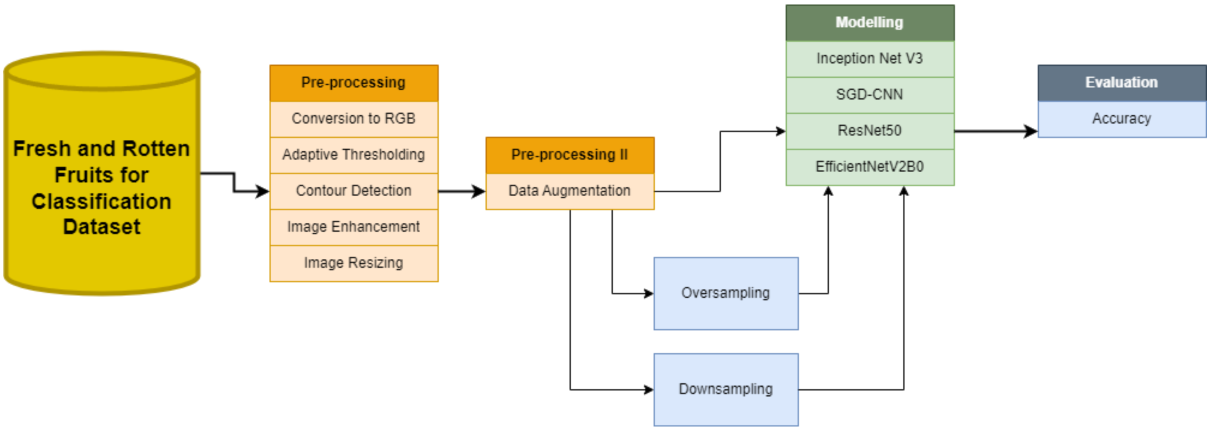

Figure 1: Methodology flow of the Study

3.1 Data Collection

The dataset used in the study, 'Fruits fresh and rotten for classification,' was acquired from the Kaggle repository[1]. It contains 10901 images of fresh and rotten Apples, Bananas, and Oranges. The count of images for each type of fruit is given in Table 1 below.

Table 1: Count of Images for Each Fruit Type

Fruit	Count of Images
Apples (fresh)	1693
Apples (rotten)	2342
Bananas (fresh)	1581
Bananas (rotten)	2224
Oranges (fresh)	1466
Oranges (rotten)	1595

[1] https://www.kaggle.com/datasets/sriramr/fruits-fresh-and-rotten-for-classification/data

Some of the sample images from the dataset are shown below in Figure 2.

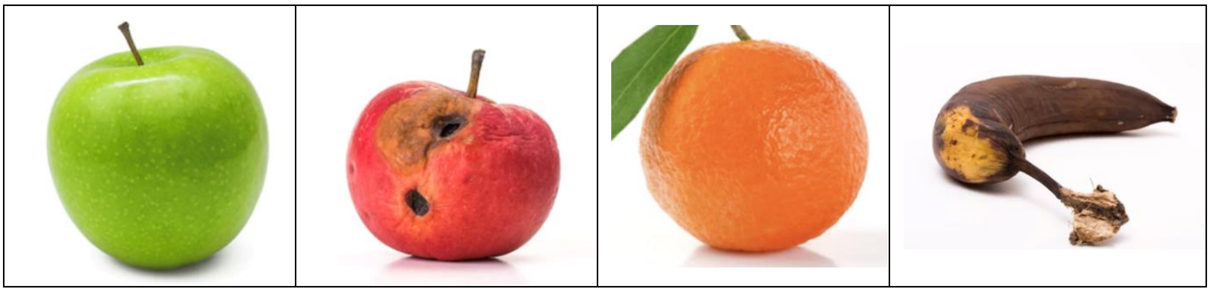

Figure 2: Sample images from the dataset

The size of the dataset and its contents make it a comprehensive resource for developing the system of fruit type classification.

3.2 Data Preparation

The data preparation process for classifying images of fruits as fresh or rotten involves a series of critical steps, each contributing to the effectiveness of the deep learning models:

An initial analysis includes counting the images in each category, essential for understanding the dataset's composition and potential class imbalances.

3.2.1 Conversion from BGR to RGB colorspace.

The CV2 library used in the study reads the images in the Blue-Green-Red (BGR) colorspace. To enable reliable visualization of the data, the images are first converted to the Red-Green-Blue (RGB) colorspace. CV2's cvtColor function helps to convert the image to RGB form.

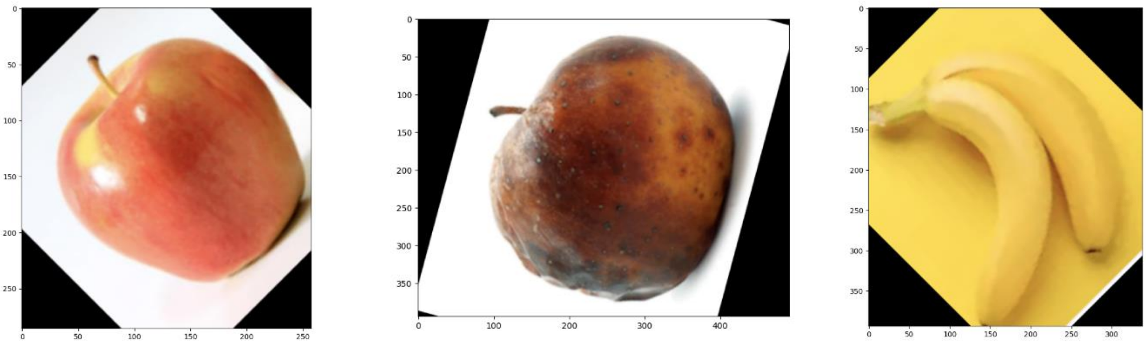

Figure 3: Output of BGR to RGB Conversion

3.2.2 Conversion to Grayscale Image

Conversion of images to grayscale simplifies the analysis. It converts the image containing three data planes viz. Red, Green, and Blue to a single place of grayscale with values ranging from 0 to 255. This helps to analyse the images better and does not require to manipulate all the planes while processing.

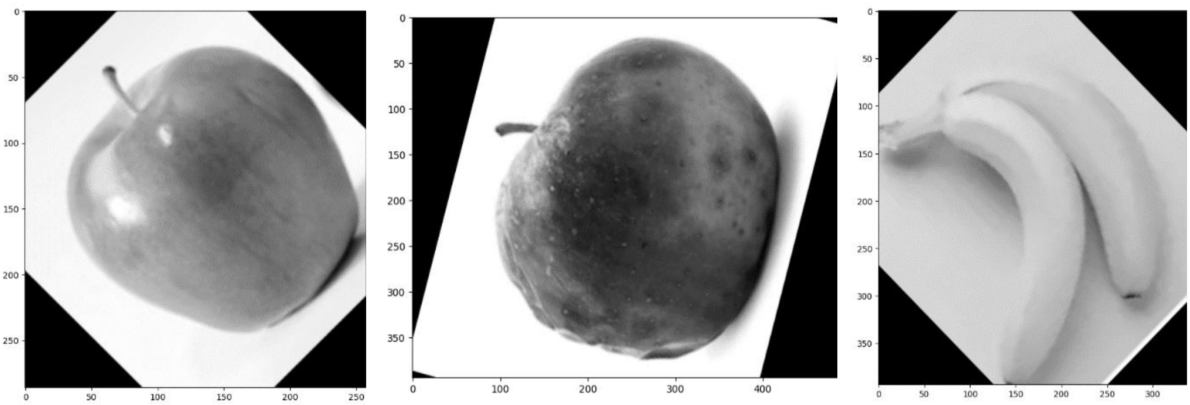

Figure 4: Grayscale version of the original image

3.2.3 Adaptive thresholding

Once the grayscale images are obtained through RGB2Grayscale conversion adaptive thresholding is applied. The application of adaptive thresholding over the image enhances key features such as texture features present in the image. The adaptive thresholding essentially converts the grayscale image into a binary image that can take only two values i.e. either 1 or 0. The adaptive thresholding can be applied through a range of methods in the study the Gaussian Thresholding technique is used to implement the thresholding.

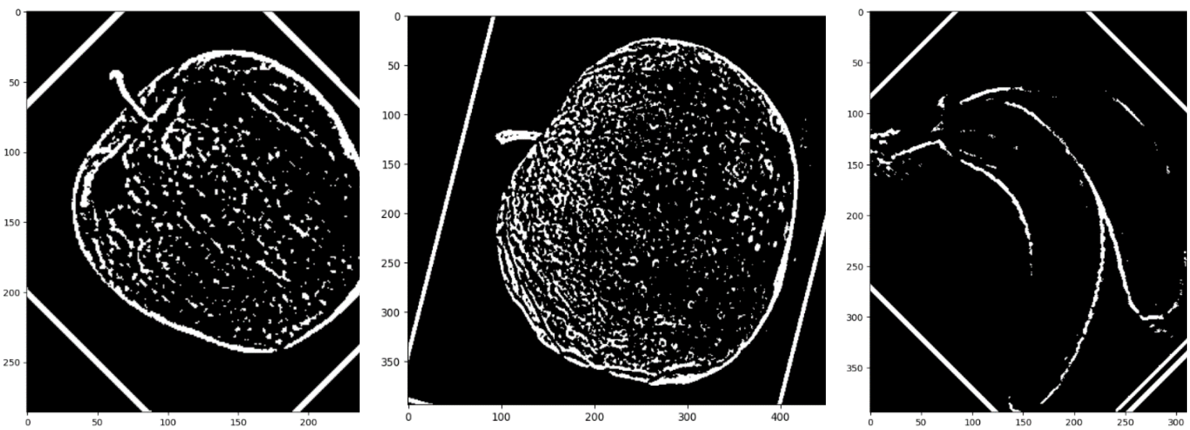

Figure 5: Binary image after Adaptive Thresholding

3.2.4 Contour detection

Once the binary image is obtained through the adaptive thresholding, contours on the images can be identified helping to segment the image based on the variation in the features in image.

Detection of contours helps to identify and isolate regions of interest in the fruits. This method is implemented through two steps, first is the method that needs to be used to get the contours and second one is the identification of the endpoints of the contour that is essentially needed. For the study the contour retrieval is performed using the External method whereas the contour endpoints are identified using the Chain Approximation Simple method. The contours are then applied to the RGB image to view the areas of interest.

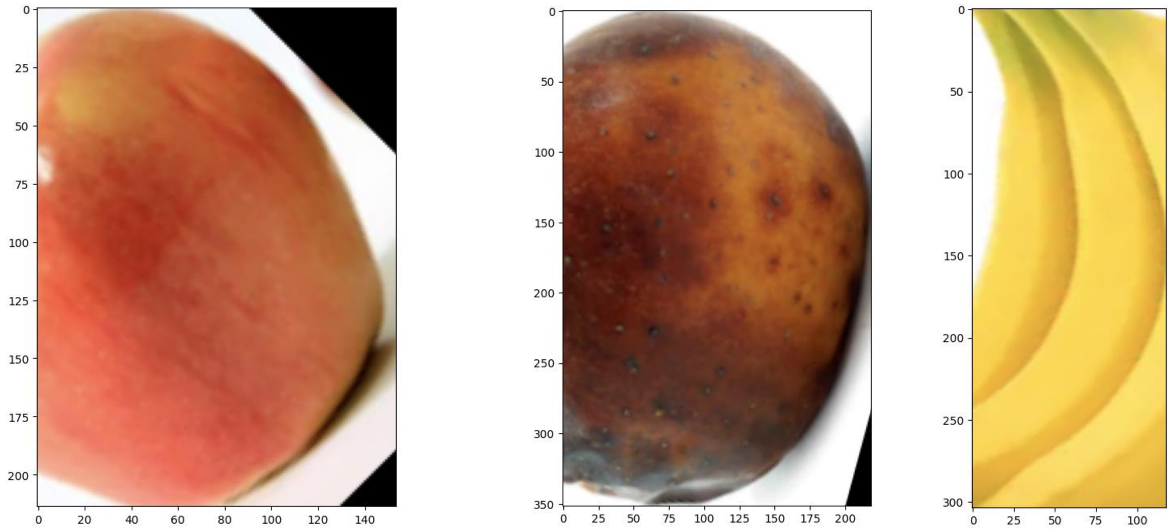

Figure 6: Identified contours in the images for different classes

3.2.5 Image normalisation

The images in the dataset are then normalised using the ImageDataGenerator object in the Keras library. The normalisation process involves dividing the pixel intensities of the images by a factor of 255. This normalisation process ensures the pixel intensities to be in the range of 0 to 1. This helps the deep learning algorithms to learn over the images faster requires less computational resources.

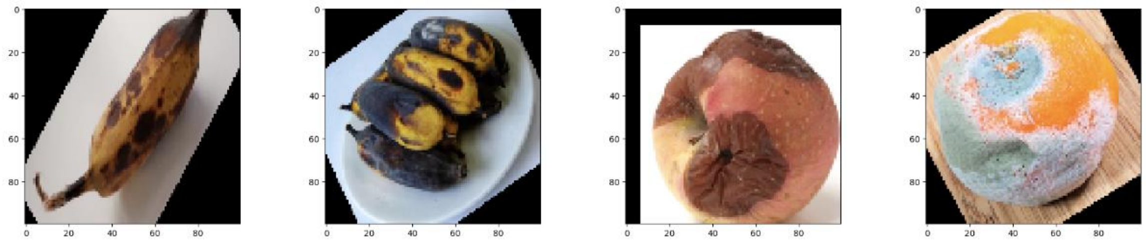

Figure 7: Normalised images from the dataset

3.2.6 Dataset Segregation

Using the same 'ImageDataGenerator' tool, generators for training and testing are created. These play a crucial role in automating the labelling process based on the directory structure. This helps to divide the data into training and testing subsets. The models then can be trained upon the training set whereas they can be tested on the testing subset.

3.2.7 Handling Class Imbalance

This study aims to experiment the deep learning models with three types of datasets that derived from the original dataset.

The first experiment involves using the deep learning models on the original dataset that has class imbalance as can be seen in Figure 8 below. As per the below Figure 8 the rotten bananas have most of the samples, and the rotten apples has the minimum number of sample images.

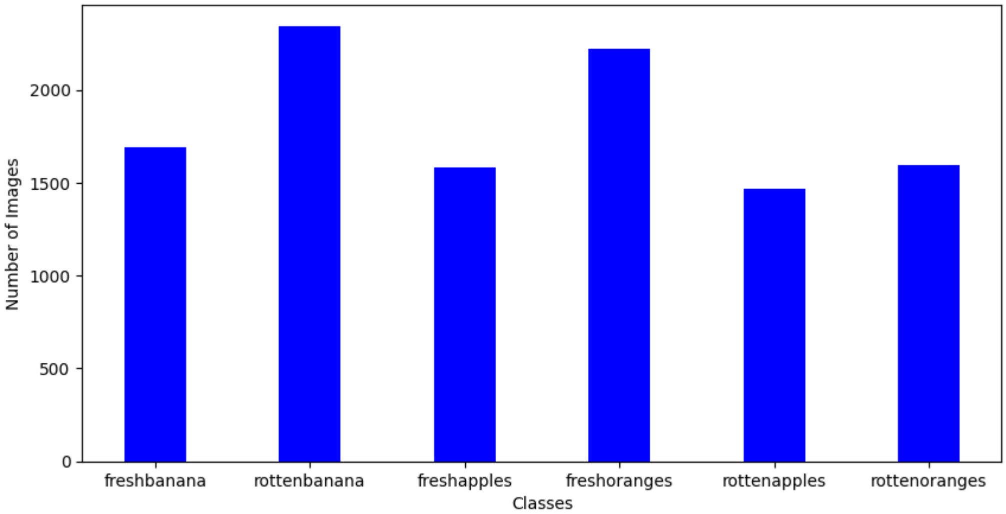

Figure 8: Distribution of Classes in the Dataset

In the second experiment, the class imbalance is addressed by undersampling the classes in the majority. In this experiment, the number of samples belonging to the classes in the majority is reduced such that the number of samples for these classes is made equal to those in the minority. From the figure above, the class ‘rottenapples’ has the least number of samples in the class. The number of samples for this class is 1466. Hence, for this experiment, the number of samples for the other classes is reduced to 1466, making a class-balanced dataset.

In the third experiment, the class imbalance is addressed by generating additional images for the minority class. This is done by first identifying the class with the highest number of samples and getting the number of samples from that class. From the original dataset, it is observed from the figure above that the class ‘rottenbanana’ consists of the highest number of samples i.e. 2342 samples. So, the number of samples for the remaining classes are increased through the process of image augmentation.

In this the image augmentation process, an ImageDataGenerator object is created that performs the horizontal flip and nearest fill on to the images in the classes in minority generating a new set of images. The distribution of the images after this oversampling is shown in figure 9 below.

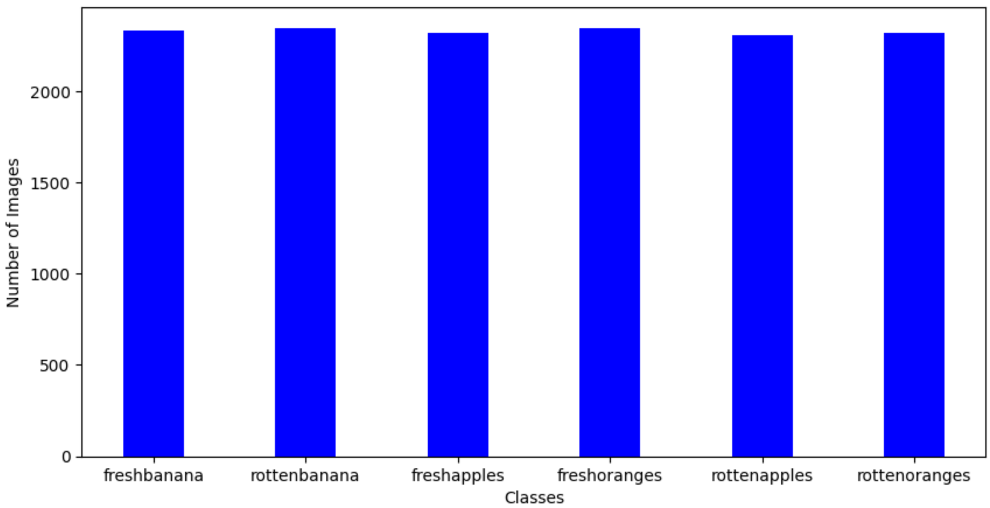

Figure 9: Distribution of Classes after Oversampling

To ensure the modelling part that follows to be effective these data preparation steps are crucial. Once the data is ready, it is then can be modelled using the deep learning algorithms that are to be tested and evaluated.

3.3 Modelling

There are four models that are implemented and evaluated in the presented study. These models are:

1. Inception Net V3
2. SGD-CNN
3. ResNet50
4. Efficient Net V2B0

3.3.1 Inception Net V3

Inception Net V3 is a sophisticated model known for its efficiency and accuracy, Inception Net V3 utilizes a complex architecture with multiple convolutional layers. It's particularly effective in feature extraction due to its varied kernel sizes (Szegedy et al., 2016). The model is designed to be efficient through a good balance between model size and computational cost. The model achieved impressive feature extraction in images because of the inception modules present in the network.

3.3.2 SGD-CNN

This model uses a simple CNN architecture optimized with stochastic gradient descent. It's suitable for image classification tasks due to its straightforward yet effective structure. A simple CNN model is characterised by a set of layers working together to extract high-level features from the images presented to it. These layers include a convolutional layer where a kernel is applied all over to the image to obtain a set of features. This is followed by a Pooling layer that extracts specific parts of the feature map to reduce the size of it. This is followed by dense layer at the output. In the SGD-CNN, the network is compiled using the Stochastic Gradient Descent (SGD) to optimize the weights while training.

3.3.3 ResNet50

A part of the ResNet family known for its deep architectures, ResNet50 stands out for its use of residual connections, which help avoid the vanishing gradient problem in deep networks (He et al., 2016). It is highly effective in image recognition. The model has 50 layers in total hence the name. The residual connections which are the features of this model are a sort of skip connections that connect with layers far apart directly by ignoring the ones in between. This ensures that the model does not overfit with so many hidden layers present.

3.3.4 Efficient Net V2B0

This model is known for balancing model scaling in terms of depth, width, and resolution, making it highly efficient and accurate (Tan and Le, 2019). It is a newer model compared to the others and offers excellent performance, especially considering its computational efficiency.

This network introduces something called compound scaling. In traditional models, the three aspects of the model, depth, width, and resolution are scaled independently whereas in the Efficient Net model, all these aspects are scaled simultaneously. This is done through fixed scaling coefficients. Because of this systematic scaling, the computational requirements of the model reduced.

Figure 10 below depicts the final architecture of the system implemented.

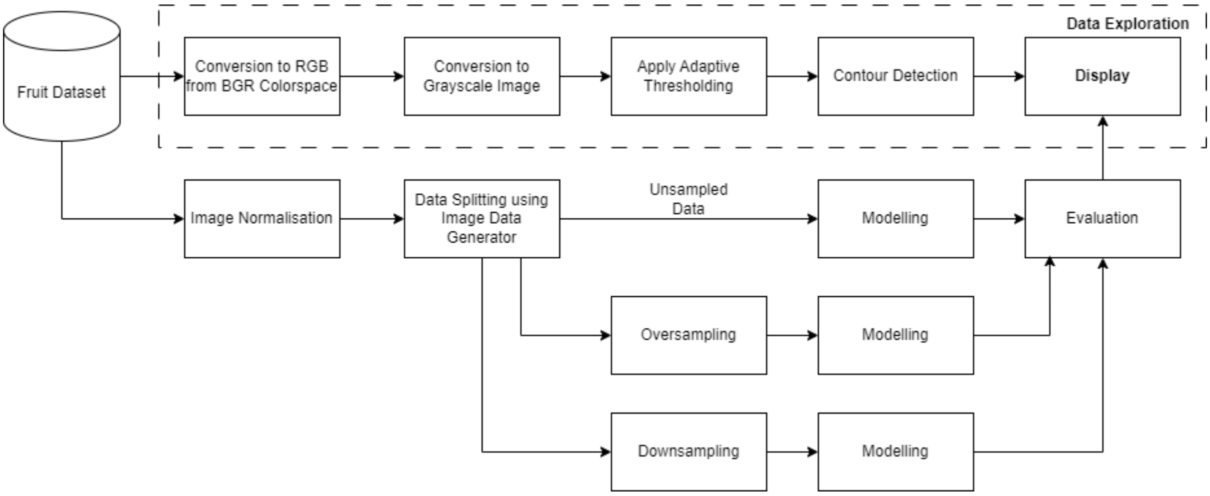

Figure 10: Methodology Architecture Diagram

4 Design Specifications

Figure 11 below depicts the design of the system employed for the study. The design consists of two layers viz., the Presentation Layer and the Business Layer. The data exploration part of the presentation layer and the data collection and pre-processing part of the business are discussed in section 3 above.

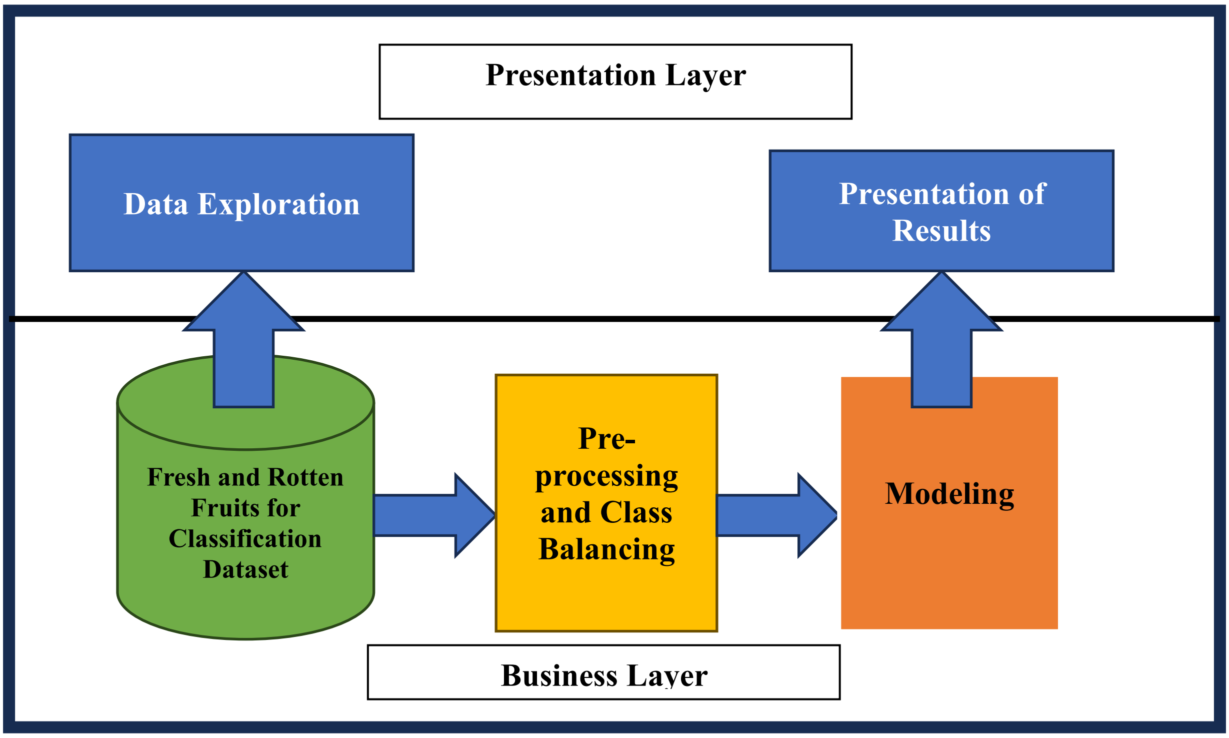

Figure 11: Design architecture of the study

The test data consisting of 2698 images is used for evaluating the models implemented. The Keras library for Python is used to implement the models in the study. The models used in the study are used through transfer learning except for the SGD-CNN model. The models are then trained on the training dataset and are applied on the testing set of the dataset. These models are evaluated using the accuracy and loss metrics through the training and testing phase.

5 Implementation

This section of the report discusses the implementation of the models and presents the results obtained for the experiments conducted. The implementation of the system is done using the Python development language using Google Colab. Google Colab is used for its access to the powerful CUDA GPUs. The dataset used in the study has been first uploaded on Google Drive. The dataset then can be accessed by mounting Google Drive in the Colab environment. We have subscribed to Google Colab pro account and used High Ram V100 as our hardware accelerator to execute our model faster. Keras is the primary library used for the implementation of deep learning models. The models, Inception Net V3, ResNet50 and the Efficient Net are used through transfer learning, an approach in which the models are trained on one task and the knowledge is then used on the other task. The SGD-CNN model used in the study is implemented from scratch through a randomized hyperparameter search.

5.1 Inception Net V3

This model is implemented with its standard architecture but augmented with additional dense layers and dropout layers for fine-tuning. The dense layers increase the model's capacity to learn from the fruit dataset, while dropout layers help in preventing overfitting.

In this implementation, the InceptionV3 model has been modified to suit the presented classification task. To enable the use of custom layers, the top layers of InceptionV3 have been excluded. The model's output is first passed through Global Average Pooling layer, which reduces its size. Then, three dense layers with 64, 16, and 8 neurons respectively are added one after the other. Each dense layer has a ReLU activation function to introduce non-linearity, and dropout layers are included with a rate of 0.05 to prevent overfitting.

The number of neurons for the dense layers are selected through a thorough trial and test method. 10 random combinations of number neurons and activation functions for the dense layers are first created. The model is then training and validated based on these combinations. The number of neurons and the activation functions for which the highest validation accuracy is achieved is selected to be the final combination of hyperparameters. This is done for all the datasets that are created in the study viz. original dataset, minority oversampled dataset and the majority downsampled dataset.

5.2 SGD-CNN

This simpler model, optimized with stochastic gradient descent, includes convolutional layers for feature extraction. Extra layers like Global Average Pooling, Dropout, and Dense layers are added for dimensionality reduction, regularization, and final classification.

The summary of the implemented model is shown in Figure 12 below.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_94 (Conv2D)          (None, 20, 20, 5)         61445

 global_average_pooling2d_1   (None, 5)                0
 (GlobalAveragePooling2D)

 dropout_2 (Dropout)         (None, 5)                 0

 flatten (Flatten)           (None, 5)                 0

 dense_4 (Dense)             (None, 128)               768

 dense_5 (Dense)             (None, 1)                 129

=================================================================
Total params: 62,342
Trainable params: 62,342
Non-trainable params: 0
_________________________________________________________________
```

Figure 12: Model summary for SGD-CNN model

The SGD-CNN model begins with a 2D convolutional layer that has five filters, a kernel size of 64, a stride of 5, "same" padding, and "tanh" activation. These features are specifically designed to extract the first set of features. The next layer is a Global Average Pooling layer that reduces the size of the space by averaging the feature maps. This is followed by a dropout layer with a rate of 0.5 to prevent the model from overfitting. The output is then flattened and sent through two dense layers. The first dense layer has 128 neurons, and the second has a single neuron. Both layers use "tanh" activation to output the data. This model is designed for the classification tasks as it balances the ability to extract features and make decisions. It is made with a stochastic gradient descent optimizer and categorical cross-entropy as the loss function.

The hyperparameters for the model are number of filters, kernel size, stride, and activation functions. These hyperparameters are selected similarly to that of the Inception Net.

5.3 ResNet50

It is implemented in its standard form but often appended with additional dense layers for classification. The model's residual connections inherently aid in training deeper networks without losing relevant information. Added layers here help tailor the model to the specific classification task. The pre-trained ResNet50 model is downloaded using the Keras library. The top layer of the model which is defined to classify the data into 1000 classes is excluded while downloading making the model available for a custom task.

Once the model's output is obtained, it is flattened so that it can be used for dense layers. There are now two dense layers with 64 or 128 and 32 neurons each. Each has a ReLU activation function for nonlinear transformations after it. To stop overfitting, a dropout layer with a rate of 0.2 is added. The last layer is a dense layer with six neurons and a sigmoid activation function. It is made for a task with six classes. Using categorical crossentropy as the loss function and accuracy as the evaluation metric, the Adam optimiser is used to put together the model. The ResNet50 model is changed in this setup to work with a complicated multi-class classification problem.

All the hyperparameters are selected through a randomised search pertaining to each of the experiments that are conducted in the study. The combination of the hyperparameters giving the best accuracy is then selected as the final combination.

5.4 Efficient Net V2B0

The model's implementation includes additional dense layers that are customized to the specific classification task, such as fruit images. These layers are crucial for adapting the model to the unique characteristics of the data. The downloaded model's output is flattened to convert the multidimensional output to a one-dimensional array. Two dense layers are then added, with 4096 and 1072 neurons, respectively. These layers use the ReLU activation function to introduce non-linearity and enhance the model's capacity to learn. A dropout layer with a 0.2 dropout rate is included to prevent overfitting, deactivating certain neurons during training randomly. Finally, the model's architecture is completed with a dense output layer featuring a single neuron and a SoftMax activation function, which is ideal for binary classification tasks. The model is compiled using the Adam optimizer, binary crossentropy as the loss function, and accuracy as the performance metric. This configuration aligns it well for binary classification challenges.

6 Evaluation

This section of the report presents the results obtained through the experimentation. The models are compared in the section based on the accuracies they have achieved.

6.1 Experiment 1: Performance of the models on an unbalanced dataset

Table 2 below lists the accuracies obtained for the models when applied to an unbalanced dataset.

Table 2: Accuracies of the models on an unbalanced dataset

Model	Accuracy (%)
Inception Net V3	83.33
SGD-CNN	83.33
ResNet50	51.57
Efficient Net V2B0	16.67

For this experiment of the study, InceptionNetV3 and the SGD-CNN model achieved the best accuracies of 83.33%. These accuracies are obtained by training the models on the training dataset and then these models are evaluated based on the testing data.

The Efficient Net model implemented in this experiment achieved the lowest accuracy which was not expected from it given its prowess in working with image data. The accuracy of the said model could not be improved more even through applying a different set of hyperparameter values.

6.2 Experiment 2: Performance of the models on the majority down-sampled dataset

In this experiment, the number of samples belonging to the majority class is reduced to match them to the number of samples of the minority class. For the selected dataset, the number of samples in the class 'rottenapples' is the lowest. Hence the samples from the other classes are reduced in the evaluation. A smaller dataset with 7038 images is obtained for modelling. The testing set of the data contains 2698 samples.

Table 3 below shows the accuracies of the models when applied to the majority downsampled dataset.

Table 3: Accuracies of the models on the majority downsampled dataset

Model	Accuracy (%)
Inception Net V3	83.33
SGD-CNN	83.33
ResNet50	22.13
Efficient Net V2B0	83.33

From the table, all the models except the ResNet50 achieved an equal accuracy of 83.33%.

6.3 Experiment 3: Performance of the models on minority oversampled dataset

In this experiment, the number of samples belonging to the minority classes is increased through image augmentation process to match them to the number of samples of the majority class. For the selected dataset, the number of samples in the class ‘rottenbanana’ is the highest making it the majority class. Hence the samples from the other classes are increased such that a larger dataset with 13957 images is obtained for modeling. The models are then tested on 2698 image samples.

The accuracies of the models for the experiment are shown in Table 4 below.

Table 4: Accuracies of the models on minority oversampled dataset

Model	Accuracy (%)
Inception Net V3	83.33
SGD-CNN	83.33
ResNet50	93.66
Efficient Net V2B0	83.33

From the table, the ResNet50 model achieved the highest accuracy of 93.66%. Whereas the other models achieved an accuracy of 83.33%.

6.4 Discussion

During the experiments, both Inception Net V3 and SGD-CNN models showed the most consistent performance, achieving 83.33% accuracy across all datasets. This demonstrated their flexibility and robustness while handling different types of data - unbalanced, majority down-sampled, and minority oversampled. However, ResNet50's performance was more variable with moderate accuracy of 51.57% on unbalanced data, a drastic drop to 22.13% on majority down-sampled data, and a significant jump to 93.66% on minority oversampled data. This suggests that ResNet50 is better suited for class balanced datasets with large number of samples.

Figure 13 below depicts the comparison of the model performances achieved in various experiments conducted.

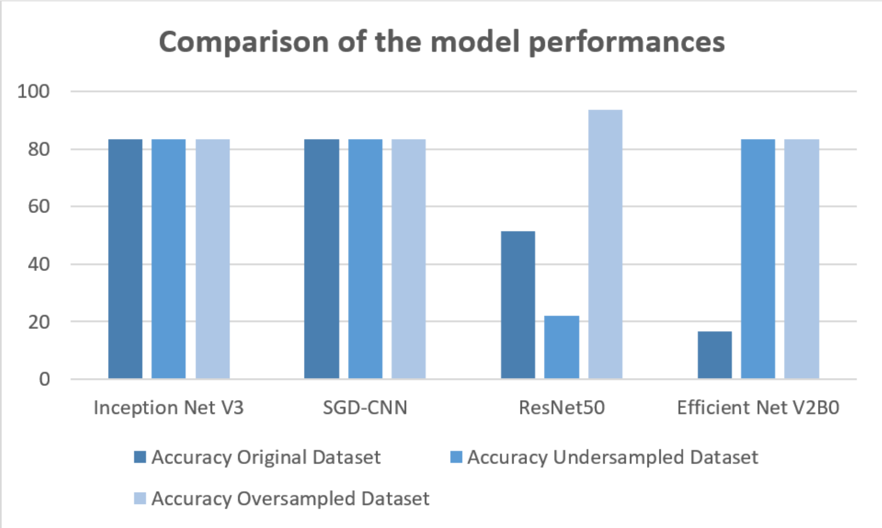

Figure 13: Comparison of the model performances

The results of the study indicate that Efficient Net V2B0 produced inconsistent results, correctly scoring only 16.67% on both unbalanced and minority oversampled datasets. However, it achieved an 83.33% accuracy on the majority down-sampled dataset, which is the same as the best-performing models. Based on these results, it can be inferred that Efficient Net V2B0 might work better when the dataset is balanced. The study highlights the importance of dataset composition in determining the performance of a model since different models have different strengths and weaknesses depending on the dataset used.

Because this task study involved multi-class classification problem, the accuracies achieved by the models are impressive. A multi-class classification problem already involves several levels of difficulties. The hyperparameter tuning for model implementation hence plays a crucial role to extract the best results from the models.

Adding the dropout layers in the models further enhanced their performances by avoiding overfitting the dataset. Overall, the models achieved commendable performance in all the experiments conducted in the study.

7 Conclusion and Future Work

Segregation the fruits based on the quality is an important process in agricultural industry. Current methods of segregating the fruits based on their quality involves manual separation that is both tedious as well as time consuming. This study evaluated various deep learning modalities for the detection of the quality of fruits.

This study evaluates the effectiveness of four deep learning models in distinguishing fresh and rotten fruits: Inception Net V3, SGD-CNN, ResNet50, and Efficient Net V2B0. The research concludes that Inception Net V3 and SGD-CNN are strong performers as they consistently delivered good results on various datasets including unbalanced, majority down-sampled, and minority oversampled. ResNet50 showed varying results and performed better when minority samples were oversampled, while Efficient Net V2B0 performed best on balanced datasets. The study highlights the significance of dataset composition in the performance of deep learning models and offers insights for improving the application of deep learning methods in agriculture.

7.1 Future Work

Further research is needed to improve the efficiency of these models across a broader spectrum of real-life agricultural scenarios. This could involve exploring more sophisticated techniques like unsupervised or semi-supervised learning, which may prove more adept at generalizing unlabelled data, a common occurrence in agricultural settings. Additionally, it would be worthwhile to investigate how these models can be integrated with real-time detection systems in agricultural production and supply chains. To enhance the reliability and usefulness of the model, it would be beneficial to experiment with larger and more diverse datasets, encompassing a range of fruits and spoilage stages.

References

Dubey, S.R. and Jalal, A.S., 2012. Robust approach for fruit and vegetable classification. *Procedia Engineering*, *38*, pp.3449-3453.

He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

Hossain, M.S., Al-Hammadi, M. and Muhammad, G., 2018. Automatic fruit classification using deep learning for industrial applications. *IEEE transactions on industrial informatics*, *15*(2), pp.1027-1034.

R. Dandavate and V. Patodkar, "CNN and Data Augmentation Based Fruit Classification Model," *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, India, 2020, pp. 784-787, doi: 10.1109/I-SMAC49090.2020.9243440.

Huang, Y., Yang, H., Sun, K., Zhang, S., Cao, L., Jiang, G. and Ji, R., 2023. InterFormer: Real-time interactive image segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 22301-22311).

Katarzyna, R. and Paweł, M., 2019. A vision-based method utilizing deep convolutional neural networks for fruit variety classification in uncertainty conditions of retail sales. *Applied Sciences*, *9*(19), p.3971.

Khune, S., Pawale, P., Khune, R., and Ranpise, S., 2015. Fruits quality assessment and classification using image processing. *Int. J. Innov. Res. Creat. Technol*, *2*(4), pp.156-159.

Lu, S., Lu, Z., Phillips, P., Wang, S., Wu, J. and Zhang, Y., 2016, October. Fruit classification by HPA-SLFN. In *2016 8th International Conference on Wireless Communications & Signal Processing (WCSP)* (pp. 1-5). IEEE.

Lu, Y., 2016. Food image recognition by using convolutional neural networks (cnns). *arXiv preprint arXiv:1612.00983*.

Lydia, A.A. and Francis, F.S., 2020, February. Multi-label classification using deep convolutional neural network. In *2020 international conference on innovative trends in information technology (ICITIIT)* (pp. 1-6). IEEE.

Mureşan, H. and Oltean, M., 2017. Fruit recognition from images using deep learning. *arXiv preprint arXiv:1712.00580*.

Mustafa, N.B.A., Arumugam, K., Ahmed, S.K. and Sharrif, Z.A.M., 2011, November. Classification of fruits using Probabilistic Neural Networks-Improvement using color features. In *TENCON 2011-2011 IEEE Region 10 Conference* (pp. 264-269). IEEE.

Rojas-Aranda, J.L., Nunez-Varela, J.I., Cuevas-Tello, J.C. and Rangel-Ramirez, G., 2020. Fruit classification for retail stores using deep learning. In *Pattern Recognition: 12th Mexican Conference, MCPR 2020, Morelia, Mexico, June 24–27, 2020, Proceedings 12* (pp. 3-13). Springer International Publishing.

Sakib, S., Ashrafi, Z. and Siddique, M.A.B., 2019. Implementation of fruits recognition classifier using convolutional neural network algorithm for observation of accuracies for various hidden layers. *arXiv preprint arXiv:1904.00783*.

Savakar, D., 2012. Identification and classification of bulk fruits images using artificial neural networks. *International Journal of Engineering and Innovative Technology (IJEIT)*, *1*(3), pp.35-40.

Steinbrener, J., Posch, K. and Leitner, R., 2019. Hyperspectral fruit and vegetable classification using convolutional neural networks. *Computers and Electronics in Agriculture*, *162*, pp.364-372.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818-2826).

Tan, M. and Le, Q., 2019, May. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105-6114). PMLR.

Vasconez, J.P., Delpiano, J., Vougioukas, S. and Cheein, F.A., 2020. Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation. *Computers and Electronics in Agriculture*, *173*, p.105348.

Xue, G., Liu, S. and Ma, Y., 2020. A hybrid deep learning-based fruit classification using attention model and convolution autoencoder. *Complex & Intelligent Systems*, pp.1-11.

Zhang, Y., Wang, S., Ji, G. and Phillips, P., 2014. Fruit classification using computer vision and feedforward neural network. *Journal of Food Engineering*, *143*, pp.167-177.

Zhang, Y.D., Dong, Z., Chen, X., Jia, W., Du, S., Muhammad, K. and Wang, S.H., 2019. Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation. *Multimedia Tools and Applications*, *78*, pp.3613-3632.

Zeng, G., 2017, October. Fruit and vegetables classification system using image saliency and convolutional neural network. In *2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC)* (pp. 613-617). IEEE.

Zhu, L., Li, Z., Li, C., Wu, J. and Yue, J., 2018. High performance vegetable classification from images based on alexnet deep learning model. *International Journal of Agricultural and Biological Engineering*, *11*(4), pp.217-223.