Fruit and Vegetable Ripeness detection using deep Learning

Pranava Chanakya Stanam
MS ECE UW Madison
ECE 697 Final Report
stanam@wisc.edu

Abstract

The quality of the food we consume is a matter of significant concern, as we consistently strive to procure the highest standard of food products. However, various issues within the supply chain can contribute to a degradation in the quality of the edibles we consume. These issues encompass multiple stages, including cultivation, transportation, and storage of food items. To optimize the sale of superior quality products while minimizing waste, it becomes imperative to present consumers with a clear indication of the various qualities available, enabling them to select the best options. Consequently, the remaining qualities can be appropriately allocated for alternative consumption methods. Therefore, the objective of this project entails developing a methodology to determine the grading of food products through image analysis and comparing the performance of well-known architectures over the developed model, and analyzing the performance of fine-tuning and pre-training.

1 Introduction

The ripeness of food undergoes several stages, including raw, ripened, and rotten. Usually, we rely on visual cues, smell, and haptic feedback to classify food quality. However, visual classification plays a pivotal role throughout this process. The visual senses are one of the prominent features we consider in deciding the ripeness and type of food item we want to select.
Nowadays, we often hear about the wastage of food resulting from improper storage, miscommunication, and poor logistics, which are often caused by inadequate segregation of food items based on their ripeness. In the current situation, raw food serves as a crucial ingredient for processed food, and its nutritional values must fall within a preferred range to ensure that the final product aligns with desired specifications. Another significant motivation for this project is to provide visual empowerment for visually disabled individuals, enabling them to utilize the proposed solution to make informed decisions.
Based on market requirements, the project entails three primary use cases:

- Application in Food processing centers: This system can be implemented in the initial filtering stages of food processing industries to effectively segregate the food based on their features into different stages of their fruit cycle. This facilitates the maintenance of the good quality of the end products with the slightest error in deviation from the measured nutritional range.
- Implementation in food distribution centers: At food distribution centers, the system enables precise grading of goods by assessing defects and other relevant attributes. Consequently,

this information can be utilized to determine the market value for each grade, streamlining pricing strategies and facilitating efficient allocation of the products.

- Consumer Empowerment: It can be used by a consumer whose visual senses are not able to decide the food quality, then they can use this method to capture an image of the product they want to buy and determine the quality of the food.

In this project, we aim to present a comprehensive solution for food type and ripeness detection by leveraging neural networks. We intend to evaluate and compare the performance of established architectures, namely MobilenetV2 and VGG16 models. Subsequently, we will devise a tailored strategy to effectively address the three use cases.

2 Related Work

Several noteworthy studies have been conducted using deep learning techniques in fruit classification and ripeness assessment. In the "CNN and Data Augmentation Based Fruit Classification Model" paper by R. Dandavate and V. Patodkar, the authors leveraged the efficiency and high-performance attributes of pre-trained VGG16 and MobilenetV2 architectures. Through this approach, they achieved an impressive accuracy of 87% in classifying four different types of fruits.
Similarly, R. E. Saragih and A. W. R. Emanuel explored the realm of banana ripeness classification in their study titled "Banana Ripeness Classification Based on Deep Learning using Convolutional Neural Network." Their work aimed to categorize banana ripeness into three distinct classes. Employing the MobileNet V2 and NASNetMobile architectures, the authors attained the highest accuracy of 90%, thus underscoring the efficacy of the MobileNet V2 architecture in this context.
These studies emphasize the efficacy of pre-trained deep learning architectures for accurate fruit classification and ripeness assessment, showcasing the potential for achieving high performance with reduced model complexity.

3 Methods and Theory

3.1 Data

3.1.1 DataSource

Labeled data is used for this project. The dataset utilized in this endeavor was curated by amalgamating diverse publicly available open-source datasets. Several Kaggle datasets were considered, enabling the collection of labeled data for all classes except for raw orange, potato, and tomato samples.
For the three classes above, namely raw samples of orange, potato, and tomato, approximately 40 samples each were meticulously procured. This was achieved by sourcing images from platforms such as Google Images and other accessible public domains.

3.1.2 Data Modification

We employed data augmentation techniques to enhance the dataset's size, mitigate overfitting, and ensure diversity. We applied rotation within -20 to +20 degrees, a 10% zoom range, and horizontal and vertical flips. Additionally, we resized all images to a consistent 224x224 RGB format, resulting in a dataset comprising roughly 1200 samples for each label.
During the training process, we allocated 1000 samples for each class while reserving 100 samples per class for testing. The distribution of the training dataset across the 15 classes is illustrated in Figure 1. A sample of the Dataset is been illustrated in figure 1

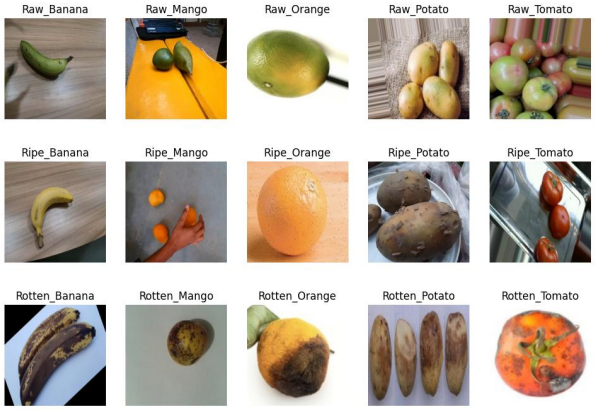

Figure 1: Dataset Sample

3.2 Deep Learning Models

In this project, we have employed a total of three models, among which one is a 4-convolutional layers neural network, and the other two are deep learning models chosen as reference benchmarks due to their renowned standards and state-of-the-art performance on the ImageNet Dataset.
For our first use case, we require a model with modest computational demands capable of operating on edge devices with limited storage and complexity. To address this, we have selected the MobileNetV2 model as a reference. Our decision to adopt the MobileNetV2 model is rooted in its well-established performance on the ImageNet dataset and its publicly available pre-trained model on ImageNet. Furthermore, its efficiency, speed, and compatibility with deployment on resource-constrained mobile phones make it an optimal choice. This architecture employs depthwise separable convolutions, reducing computational overhead while retaining satisfactory accuracy.
Another notable feature of this architecture is its utilization of residual blocks with a bottleneck structure. This innovative design reduces parameters (approximately 5 million) while effectively capturing essential features.
We have also incorporated VGG16 as an additional reference architecture due to its cutting-edge performance on the ImageNet Dataset. VGG16 arrives pre-trained on ImageNet, displaying a more intricate architecture compared to MobileNetV2, encompassing a significant 14 million trainable parameters. With 16 layers, including 13 convolution layers, VGG16 employs 3x3 convolution filters, contributing to an augmented parameter count. Its prominent reputation stems from its remarkable 97% accuracy with the ImageNet dataset, a recognized industry standard for image classification.

3.2.1 Custom 4-Layer model

.

We delved into the impact of pre-trained models versus non-pre-trained ones of comparable complexity. We crafted a 4-layer CNN possessing equivalent trainable parameters (complexity) to that of MobileNetV2. However, unlike MobileNetV2, this model was not pre-trained. We then proceeded to analyze and contrast their performances. The model can be visualized in the figure 2

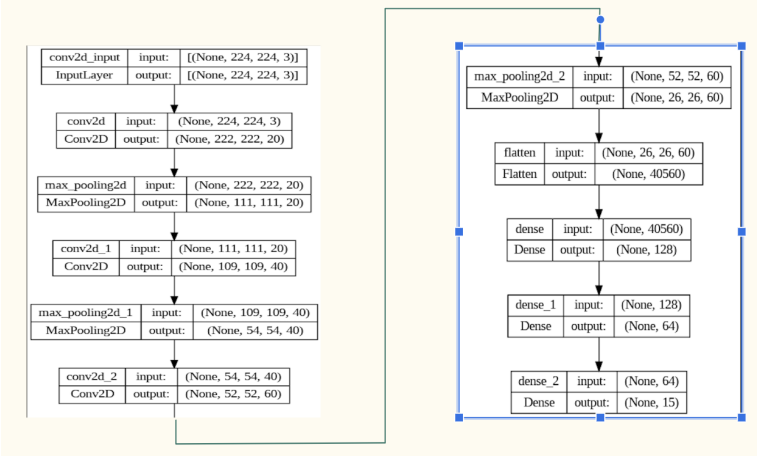

Figure 2: 4-Layer CNN

3.3 Important terminologies

3.3.1 Pretrained Models

We thoroughly explored using pre-trained models, incorporating them into our investigation, and evaluating their performance on the training and test datasets. Specifically, we employed pre-trained versions of MobilenetV2 and VGG16 models, leveraging weights imported from the ImageNet Dataset.

These two models were trained on the extensive ImageNet Dataset, which comprises an impressive 1.4 million images spanning approximately 1000 distinct classes, encompassing categories such as fruits, animals, objects, and materials. Renowned within the image classification domain, this dataset stands as a benchmark, recognized for its high quality and comprehensive coverage.

Our rationale for employing pre-trained models stemmed from their ability to facilitate faster convergence, mitigate data requirements, enhance generalization, and offer benefits in domain adaptation and transfer learning. These advantages collectively led us to consider using pre-trained versions of the MobilenetV2 and VGG16 models.

3.3.2 Fine-tuning

Fine-tuning represents a process in which a pre-trained model is customized to align with a specific use case. This procedure involves importing the pre-trained model and immobilizing all of its layers. Then, supplementary top layers tailored to the use case are appended. Consequently, training solely transpires on these newly introduced layers. Adopting this approach diminishes the count of trainable parameters, fostering improved transfer learning and knowledge transfer capabilities.

The primary distinctions between a solely pre-trained model and a fine-tuned model encompass the following: In pre-trained models, the assumption is that the use case will be a subset of the pre-trained dataset. Hence, initiating the weights with pre-existing imagenet weights yields better performance than initializing with zeros. Conversely, during fine-tuning, the model's weights remain primarily unaltered, leveraging the inherent performance of pre-trained weights. However, success is contingent on the close alignment of non-trainable parameters with the specifics of the new use case.

Figure 3: 1-classifier

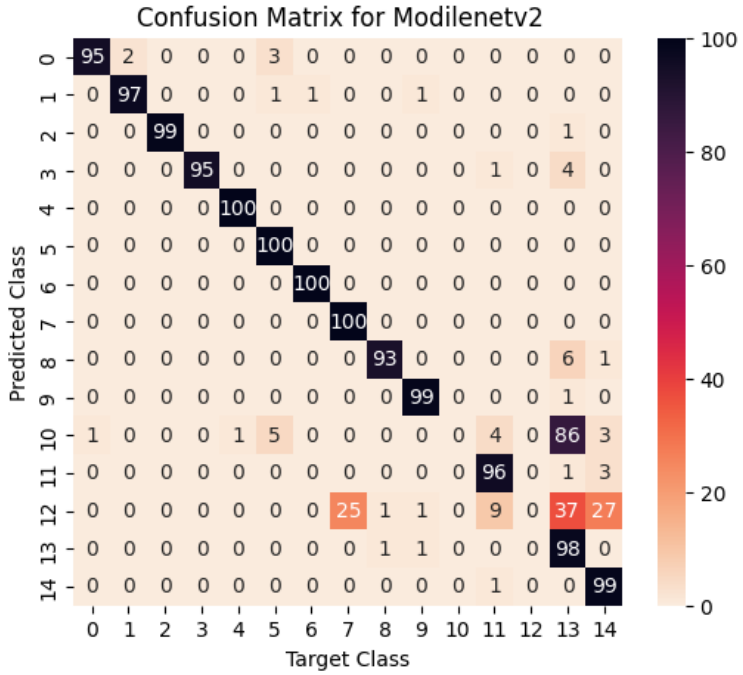

Figure 4: Confusion Matrix for MobilenetV2 model

3.4 Workflow

3.4.1 Training Mobilenetv2 and VGG16 pre-trained models,4-Layer CNN with the dataset .

We developed three models based on the MobileNetV2 pre-trained architecture, VGG16 architecture, and a customized 4-layer CNN. These models were tailored to perform classification across 15 distinct classes, representing the combination of 5 food types and three ripeness levels. The fundamental workflow of this approach is depicted in Figure 3. Here, the classifier's task involves accurately categorizing images into one of the 15 classes. The comprehensive results are presented in Table1.

Throughout this methodology, a recurring pattern of misclassification emerged, particularly between Ripe Oranges and Rotten Oranges, as discerned from the confusion matrix (reference in figure 4.) To address this issue, we proposed an alternative approach involving the implementation of two separate classifiers. The first classifier is responsible for identifying the type of food, while the second one focuses on discerning the level of ripeness. By segmenting the classification process, we aim to enhance our understanding of the underlying reasons for misclassifications.

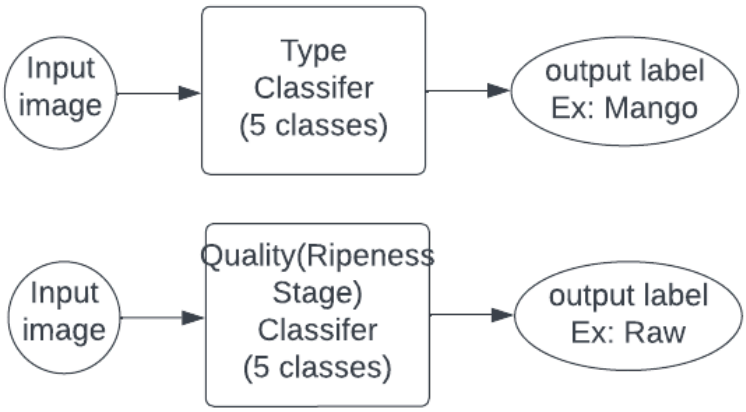

Figure 5: Approach 2

3.4.2 Training Mobilenetv2 and VGG16 pre-trained models,4-Layer CNN with the dataset using approach 2

In this approach, we devised two separate classifiers within the same model – one for determining the food type and the other for assessing the ripeness level. We employed a logical "AND" operation on both test outputs to ascertain the final accuracy. A visual representation of this methodology is illustrated in Figure 5.
This approach facilitates an individualized assessment of each architecture's competence in fruit classification and ripeness determination. The respective accuracies for these tasks are outlined in Table3.

3.4.3 Training Mobilenetv2 and VGG16 pre-trained models with fine tuning.

Our analysis found the performance of pre-trained models to be satisfactory. However, the potential for enhancing accuracy remains a crucial pursuit within this project. To this end, we have chosen finetuning as a viable avenue. This approach is aligned with a reference work where a finetuned version of VGG16 was employed. It is worth noting that a finetuned model pre-trained on the ImageNet Dataset demonstrates superior knowledge transfer when the training dataset showcases a similar diversity to the trained dataset. Given our current circumstances, which parallel this similarity, I have determined that adopting finetuning represents a significant advancement for our use case.
For this purpose, we have taken the pre trained MobileNetV2 model, appended custom top layers, and frozen the default layers. Subsequently, we trained the newly introduced parameters for the 15-class classification task, evaluating the ensuing metrics. Similarly, the pre-trained VGG16 model underwent the same treatment: adding top layers, freezing default layers, and training additional parameters for the 15-class classification. The outcomes of these processes are presented in Table4

4 Results

Table 1: Results for Single classifier (15 classes)

Model	Accuracy(%)	Precision	Recall	F-1 Score
MobilenetV2	85	0.78	0.85	0.8
VGG16	65	0.58	0.65	0.6
4-Layer CNN	64	0.56	0.94	0.59

Analyzing the data presented in Table1, we can derive the conclusion that the accuracies of MobileNetV2 surpass those of the 4-Layer CNN, even though both models share the same level of

complexity. This observation highlights the impact of model pretraining, indicating that it significantly outperforms initializing weights to zero.

Table 2: Results for approach 2 type and ripeness classifier separate

Model	Accuracy(%)	Precision	Recall	F-1 Score
MobilenetV2-Type	90	0.91	0.9	0.9
MobilenetV2-Quality	84	0.88	0.84	0.83
VGG16-Type	70	0.73	0.7	0.69
VGG16-Quality	85	0.85	0.85	0.84
4-Layer-CNN-Type	77	0.78	0.77	0.76
4-Layer-CNN-Quality	72	0.74	0.72	0.7

Table 3: Results for Combined Accuracy for Approach 2

Model	Accuracy(%)
MobilenetV2	78
VGG16	61
4-Layer CNN	58

Based on the outcomes in tables 2 and 3, we can infer that individual accuracies exhibit higher values. In contrast, the combined accuracy for the same set of samples is slightly lower than that of a single classifier. This leads us to deduce that the single classifier can capture subtle nuances within the samples, effectively classifying them into the correct classes. Furthermore, we can interpret the elevated individual accuracies resulting from the models being trained on datasets comprising similar classes.

In this final approach, we can see in the results(Table 4) the use of higher knowledge transfer

Table 4: Results for Single classifier (15 classes) with fine tuning

Model	Accuracy(%)
MobilenetV2	62
VGG16	88

because of pretraining and freezing the weights. The VGG16 finetuned model performed better than the mobilenetv2 model because of its higher trainable parameters and adaptability to new datasets.

5 Conclusions

We have made predictions regarding the optimal model choice depending on the specific use cases identified.

For the use case of mobile utility/consumer empowerment, it is recommended to employ the MobileNetV2 pre-trained model without any further finetuning. This recommendation is based on the understanding that we typically deal with a specific dataset in this scenario, and there will not be any future updates. The model's efficiency is evident, achieving an 85% accuracy with the current dataset.

On the other hand, if the intention is to expand the classification to include additional fruits and vegetables, the VGG16 model is the preferred choice. The fine-tuned version of VGG16 showcases strong knowledge transfer capabilities, rendering it adaptable to new classes. While achieving an efficient 88% accuracy, it is crucial to consider that VGG16 is a larger model, demanding more computational resources to operate effectively.

6 Future Scope

An ideal model is characterized by lower complexity and enhanced adaptability to novel classes. In the future, we propose the development of a 4-Layer CNN that can be trained using a vast dataset like ImageNet. Leveraging its knowledge transfer capabilities, this approach can achieve the coveted scenario of reduced complexity while maintaining increased flexibility.
Our confusion matrix analysis has revealed instances of misclassification for certain classes. To address this, we suggest skewing the training data distribution. This involves introducing additional samples for the misclassified classes, effectively mitigating overfitting arising from insufficient diversity within these classes.

References

[1] 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.

[2] Aherwadi, N.; Mittal, U.; Singla, J.; Jhanjhi, N.Z.; Yassine, A.; Hossain, M.S. Prediction of Fruit Maturity, Quality, and Its Life Using Deep Learning Algorithms. Electronics 2022, 11, 4100. https://doi.org/10.3390/electronics11244100

[3] R. E. Saragih and A. W. R. Emanuel, "Banana Ripeness Classification Based on Deep Learning using Convolutional Neural Network," 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT), Surabaya, Indonesia, 2021, pp. 85-89, doi: 10.1109/EIConCIT50028.2021.9431928.

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