FRUIT DETECTION AND RIPENESS CLASSIFICATION PROJECT

Team Bravo: Julia Baardse, Eleni Gemintzi, Griselda Subashaj, Christos Vlassis
Student IDs: f2822201, f2822206, f28222016, f2822204

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS MSc in Business Analytics

CONTENTS

INTRODUCTION

This is a project regarding the development of a machine learning program, which uses computer vision algorithms to categorize fruit based on ripeness. We will use images of fruit as a dataset to create a prototype model, which is able to locate, recognize and classify the fruit based on type and ripeness. In this report, we will present our idea, its potential, its implementation and the possibilities for improvement. We will describe in detail all technical aspects of development and present our results.

This paper begins by introducing the motivation behind our idea, presenting our vision and discussing our goals. A business model outline highlights the feasibility of our vision, by examining what are the opportunities to monetize this idea (possible market segments and revenue streams), how it has already been implemented (competition analysis), and what are the means to realize our vision, including possible partnerships, key activities and cost analysis.

Then, the report explains in depth our methods for developing the prototype including data, algorithms for the initial model and its alternatives, all results and future work. More specifically, the methodology part begins by describing the data we use, including its source, description and any preprocessing activities. Then it presents the algorithms used for the initial simple model and our efforts to optimize the results by increasing complexity of the tasks. Finally we present our results, what we have achieved and what requires future development.

To conclude, we present our team, our collaboration and our takeaway from this project. We document what additional resources we would require to achieve better results and a time plan for activities, containing a rough timeline of our work and time estimates for further work and improvement.

OUR PROJECT

Our project aims to develop a model which is capable of identifying the ripeness level of a fruit, using an image. To achieve this, we use a binary classification methodology (ripe/not ripe) for 6 distinct fruit categories.

Our approach for this project is to work progressively, starting from a basic machine learning model and using it as a basis for further experimentation. This way we can record step by step our progress and potential, while having the ability to pinpoint areas for improvement.

More specifically, we aim to begin from a simple CNN model, which categorizes specific fruit as ripe or not ripe (e.g. "given that this is a lemon, it is not ripe"). Our second target is to create a model that can understand what fruit is depicted in the image ("This is a lemon"). Thirdly, we propose to use bounding boxes to build a model that can identify fruit in an image. By combining these three methodologies, a complete program can

1. identify fruit in the image
2. classify what type of fruit it is
3. decide if it is ripe or not ripe

Lastly, we want to explore combination possibilities, to see if some steps can be performed simultaneously.

At the same time, we create a business plan, not only to understand what capabilities are important to a potential client and the needs our models need to address, the resources we need and the potential for development, but also to gain insight into current developments and technological milestones in this area.

It is important to emphasize that, throughout our process, there has been a significant feedback loop between business planning and model experimentation. Business concepts, shaped through an analysis of prevailing industry trends, guide the trajectory of development, influencing decisions regarding data selection and the objectives of the model. Then, the model's performance reveals its limitations and capabilities, leading to adjustments or enhancements in the business plan. This process fosters a more precise comprehension of what can be produced, which customer segment it will primarily serve, which gaps need to be delegated to partners and how costs will be structured.

This report aims to present our most mature model and the accompanying business plan, and pinpoint strengths, weaknesses, potentials and means to achieve them.

OUR VISION/GOALS

Our vision is to utilize machine learning to reduce food waste and maximize resource efficiency in agricultural and food processing enterprises. We believe that a machine learning model that deploys neural networks to distinguish ripe from not ripe fruit based on image recognition, can effectively solve many modern industry issues.

As the agriculture industry, including wholesale, food processing and retail of fresh produce, begins to adopt Artificial Intelligence, we believe that it is the ideal timing to propose our solution. We are enthusiastic to enter a niche but rapidly growing sector, to introduce our idea and allow it to evolve and take shape in a rapidly changing environment. We want to take advantage of this momentum to unlock the potential of AI in identifying fruit quality.

BUSINESS MODEL

THE PRODUCT AND THE NEEDS ADDRESSED

We propose a machine learning model, which uses live images of fresh fruit to categorize it according to ripeness. Such a technology has wide application in businesses that deal with fresh produce in any form, like wholesale, processing or retail of fruit. Adopting this model will enable companies in these fields to reduce waste, thus simultaneously increasing sustainability and minimizing costs.

In 2023, WRAP (Waste and Resource Action Program) estimates that 16% of the global food waste comes from manufacturers and 2% from the retail industry, with supermarkets being the leading contributor (WRAP, 2023). To put that in perspective, according to Eurostat, in 2020 and in Europe alone, there were 12.000.000 tons of fresh food wasted during manufacturing and 4.000.000 tons were lost during distribution and retail processes. These numbers also signify great annual losses for companies in these sectors, when the volume of lost food is translated into wasted raw materials cost (Eurostat, 2023).

As the importance of eco-friendly products and the adoption of green tactics has accelerated through the recent years, businesses are faced with high sustainability targets. We strongly believe that the technology we propose, based on computer vision, will result in food waste reduction and proper resource allocation before production begins. This will help our manufacturing customers achieve their sustainability goals and greatly reduce cost. In addition, it will optimize warehouse and storage processes, to eliminate food waste at the distribution level, both for manufacture and retail clients.

MARKET RESEARCH - COMPETITION ANALYSIS

The application of machine learning for quality control in the food industry is growing exponentially. There are multiple innovations using different data sources and various models to evaluate a multitude of parameters concerning food and fresh produce. In this section we will present the technologies that already exist in the market as a form of indirect competition, the offered products that are similar to ours (direct competition) and an overall image of the current market state.

There are various ways to inspect ripeness and overall quality in fruit. Some of the most widespread are hyperspectral Imaging, electronic noses and computer vision. Hyperspectral cameras capture images across a wide range of wavelengths, allowing for detailed analysis of fruit composition. Companies like Specim and Resonon provide hyperspectral imaging solutions for ripeness detection. Electronic nose devices, used predominantly by Aryballe, utilize gas sensors to detect volatile compounds emitted by ripening fruits. Other companies, like AgroFresh solutions, utilize various sensors to estimate the ripeness of fresh produce. These technologies, while dissimilar to our computer vision models, have applications in the same market, thus posing indirect competition.

On the other hand, companies like AgShift and ImpactVision, use computer vision systems to analyze color changes in fruits as they ripen. These systems capture images and use machine learning algorithms to determine ripeness based on color patterns. These companies are prime examples of direct competition our model faces. By analyzing the competitors' products and operating strategy, we can adopt their successful strategies, improve upon their weaknesses, and innovate to stand out in the market. This analysis informs our product enhancement, marketing approaches, and overall business strategy for a competitive edge.

Understanding competition is also crucial for the initial stages of the development, as it highlights the milestones we need to reach to create a product that is viable in the market. For instance, we know that AgShift's Hydra AI is able to perform in 2 minutes with 90% accuracy. This is important information when accessing our model's performance and execution time.

Another important takeaway is that the market right now is far from saturated. On the contrary, it is still new, with few businesses entering, limited products being offered and no established monopolies. This not only guarantees ease of entry into the market, it also limits competition to product design instead of price competition. This will become more apparent in our cost structure analysis, but it is important to take notice of the conditions that create the ideal environment for a model such as ours to become a competitive product.

VALUE PROPOSITIONS

Here, we will describe the value proposition of our product. We will demonstrate its strong points regarding newness, convenience, cost reduction and customization. Additionally, in the next section, we will describe what value each customer segment derives from the product.

The utilization of Machine Learning technologies in the food industry is innovative and not yet widespread, especially among smaller produce related businesses. The adoption of a machine learning model in fruit ripeness recognition will aid the companies to stay competitive in the new technological era of artificial intelligence.

The product we propose, compared to other machine learning programs, is also characterized by convenience, due to the computer vision component. This means that, contrary to other fruit classification technologies, this one does not require specialized sensors or scanners. Instead, it can be used with a medium resolution camera, which is inexpensive and easy to replace.

In addition, these models can be retrained if the product of the end user alters and they wish that the program adapts to these changes. As will be presented under revenue streams, we propose a solution where the basic package of the program performs the core processes of fruit detection using bounding boxes and fruit ripeness classification. However, any additional custom process that concerns either the methodology of the classification or the data available, can be purchased as an add on. This way, all customers receive the same standard of care, but simultaneously a custom tailored experience.

CUSTOMER SEGMENTS AND INDIVIDUAL VALUE PROPOSITION

Our machine learning program, designed to separate fruit at different stages of ripeness using computer vision, addresses the needs of manufacturing and retail companies that deal with fresh produce. We can distinguish 3 main customer segments: distributors and wholesalers, food processing companies and retailers (supermarkets).

Firstly, all three of our customer segments are medium to large businesses. We believe that our product is more viable if it caters exclusively to B2B distribution channels, targeting medium to large enterprises as our primary clientele.

Distributors and wholesalers are companies that purchase large quantities of produce and need to maximize the quality of their product. Only in Greece, this sector has a total worth of 3.3 billion and a steady annual increase in the last five years. It is a sector that has multiple companies, covering all sizes, not even including international companies that deal in produce exports.

For wholesalers, with large quantities of fruit, it is challenging to distinguish ripe from unripe fruit and, as a result, fruit that could have been salvaged had it been stored and distributed under the right conditions, is wasted. For these companies, our computer vision variation, which uses bounding boxes to identify bad fruit in a large pile, can be especially useful to segregate batches of fruit. For instance, if a batch has 15% more ripe fruit compared to another batch, maybe it would be wise to prioritize its delivery or distribute it to a channel that is more appropriate.

Food processing companies are the main focus of our project. According to the United States Department of Agriculture estimate, right now and in Greece alone, there are over 15000 food processing companies, collectively creating 11 billion USD in value (Faniadis, 2020). These industries have the budget and the financial incentive to invest in the technology solution we provide.

Food processing enterprises will greatly benefit from identifying in real time the ripeness of their raw material, especially if they produce more than one product based on the same fruit. In any case, the correct allocation of fruit based on their expiration, will prevent the loss of resources and will benefit quality assurance. We expect that in an industrial setting with an automated pipeline, a one-by-one examination of each item, as our first variation of the model offers, will be in more demand.

The final customer segment, retail chains that include fresh produce, are very prevalent in Greece and abroad. There are numerous multinational chains worth billions each, that are currently active in Greece and Europe, most of which offer a produce section in all of their locations. These companies, due to their volume, are not only able to invest in such technology, but can save significant amounts of money by doing so.

In retail, correct storage and refrigeration is crucial for minimizing loss in produce. Similarly to the distribution segment, the retail market will likely benefit more from identification of ripe fruit in a batch, rather than individual item scans.

Finally, a great benefit, which applies to all customer segments and regardless of the specific model variation they adopt, is sustainability. It is always important that optimal resource management does not only yield fiscal rewards, but enables companies to reach their green goals.

REVENUE STREAMS

There are multiple types of possible revenue streams to be considered, like asset sale, usage fee, subscription fees and licensing. Each possible scenario will be described briefly below.

To begin, the asset sale would involve a one time sale. As our product is in development, however, this option is not very convenient for pricing. The main issue is that either the customers will have to pay a high price for features that are not yet available and will be updated later or we will have to charge a price that reflects only the current capabilities. This is primarily why asset sale is not the most attractive revenue stream, right now.

Licensing involves granting the right to use a product or intellectual property for a one-time fee, often allowing the user to use it perpetually. This method addresses the problem above, as the user may choose to upgrade their license tier once it expires, to include new features (methods, fruit category etc). This also allows for customization, in the sense of adding a model feature specifically tailored to this customer (e.g. rare fruit).

Subscription, on the other hand, entails regular payments for ongoing access to a product or service, often with updates, maintenance, and additional features included as part of the package. While this option is the best while the product is in the initial to mid levels of development, it may complicate customization options.

This is why we believe the best option as our preferred revenue stream is time dependent. It is clear that subscription based revenue is better in the beginning, until the product is developed enough to include most of the main features we target. Later, we will consider switching to licensing options, to enable customization of the product through add-ons. Lastly, while asset sales are more rigid as an option, it is early to eliminate this alternative completely.

KEY ACTIVITIES AND PARTNERSHIPS

In this section, we describe the key activities required for the development and continuation of the product. These activities will be complemented by our choice of partners, to complete our vision.

Key Activities

The major key activity is undoubtedly Research and Development. Since we are still in the development phase, designing our product, testing it and repeating the process utilizing the feedback we received from testing is the key to having a reliable and competitive product. This can be further broken into two sections: data and model.

Data wise, right now our model prototype is limited to only recognize 6 fruit. Of course, before entering the market, this number ought to be significantly greater, probably including not only the fruit category, but also their varieties. In addition, we wish to expand the ripeness categories, to adhere to more complicated tasks. Therefore, instead of a binary classification (ripe/ not ripe) we aim to have more levels, which reflect the customer’s needs. Both of these expansions need extensive data collection, augmentation and labeling to cover more fruit categories, their varieties and different levels of ripeness. This data collection, preprocessing and labeling should be executed intensively before entering the market, but should be ongoing as well afterwards, as part of regular development activities.

Model wise, there are multiple deep learning techniques which can be utilized, aside from those we have experimented with for the prototype. This needs to be further developed and we expect that, as R&D progresses, we can unlock new capabilities and value our product can offer.

Software development is another key task. There needs to be a mediator between our model and the hardware components, giving the ability for human participation. This part may be outsourced or executed by partner companies or delegated to a software developer who joins our core team.

It is also important for us to create strong and lasting relationships with our customers. This needs to be achieved by further research of our customer base, their market, their needs and expectations. Therefore, market research and market validation are also key activities.

As seen from the key activities, our processes are focused primarily around developing and optimizing the algorithm. However, for a software product to be viable, there are other key tasks that need to be addressed. This is a segway to introduce how crucial partnership is for our project.

Motivations for partnerships

Forming carefully targeted partnerships is key for the distribution and installation of our product. We believe that the optimal partnership, at least in the beginning, would be experienced automation companies who work with food processing manufacturers.

Our product is a software program and, therefore, needs a complimentary hardware component to function. More specifically, it would need at least a device which is able to scan the fruit and send the image to the machine learning model, to be evaluated for ripeness.

A partner that deals in industrial automation solutions would be able to pair our software the appropriate hardware to fulfill the end user's needs. Since the model has potential to be retrained to address multiple customer needs and have various applications depending on the case, the complementary mechanism should also be customizable for each specific case. This is why we would greatly benefit from the expertise of an automations partner.

Moreover, since our product is new to the market, a partnership with an established enterprise will greatly help us develop a network and extend our understanding of the market and the customer base.

In conclusion, forming partnerships with automation enterprises will greatly aid us in presenting our product as part of a complete solution for an industrial client. In addition, we will benefit by creating lasting relationships with local clients and gaining valuable domain knowledge, thus improving R&D processes and reducing risk and uncertainty

COST STRUCTURE

Value-driven

We believe that a value driven approach would be more suited to our product and customer base, compared to a cost driven approach. This means our business model emphasizes on maximizing the value created by our product, rather than minimizing its cost.

This is because, as will be presented below, the majority of the cost is variable and project dependent. This means that these costs can be easily transferred to the customer as part of the agreed upon price. In addition, since we are operating in fairly new territory and a market that has not yet been segmented, we are not faced with intense price competition. Instead, we need to compete in creating the most appropriate solution for a customer, which will require a lot of adjustments to the product.

Short - term cost analysis

In the short-term, which we estimate to be no more than 24 months, the cost we face can be dissected into fixed and variable. Fixed cost represents all costs that are unavoidable in the short term period and do not depend on whether or not we are developing our product or working with a client. On the other hand, variable costs include everything that accompanies a new project and ceases to exist once a project is completed. For purposes of cohesion, labor costs, including our core team of data scientists and administration staff, will be categorized under variable cost.

Fixed costs:

Due to the nature of the product, the fixed cost is not substantial. The majority of this cost is office space rent and maintenance, and regular administrative expenses. In addition, we can consider research and development costs as fixed, as these are crucial in ensuring a high standard for the product and meeting market expectations.

Variable costs:

As mentioned previously, the majority of cost in the short term is variable, in the sense that it is a function of the type and amount of product produced. In brief, it includes salaries, cloud service charges, data storage, customer support and, most importantly, tailoring the product to meet the various end user's expectations.

Each new customer can potentially require retraining the model on new parameters or additional data. For instance, maybe a production line is such that it can only take panoramic images of the fruit and the training data should be representative of that. Another example would be to change the parameters to minimize type I misclassification error, as that specific mistake may be more expensive for the client than type II. In any case, any new project may require additional cost, which will then be included directly in the price.

Additionally, we face high labor costs, which includes the salaries of our core team and additional personnel we may require, depending on the needs. In summation, this is the salaries of data scientists, data engineers, software developers and administration staff.

We also expect cloud service charges for server usage and computational resources required for real-time image processing. This is also combined with data storage costs for storing training data, customer data, and backups.

Finally, since we need to emphasize on building long lasting relationships with our customer network, we need to offer after care services. This results in variable customer support expenses., which are expected to increase with the growth of the customer base.

On the opposite note, because we address our product only to B2B channels, our machine learning program does not require advertising costs, as much as it requires word of mouth and representation.

Long term cost analysis

When examining the cost structure of our product, it is detrimental to take into account its future viability. This is why, here we examine the effect of the economies of scale and the economies of scope.

As the company grows and acquires new customers, we expect the cost of each unit sold to decrease. More specifically, as more subscriptions are sold, the cost per unit (per license) may decrease due to shared server costs and operational efficiencies. In addition, as the machine learning program's capabilities expand to cover additional image recognition tasks, existing infrastructure and expertise can be leveraged for cost savings.

On the other hand, we expect aftercare services and customer relation costs to increase, to reflect our growing customer base. This includes customer support and troubleshooting costs, as well as public relations events.

Methodology

To achieve our business objectives, we have developed two distinct models. The first model is designed for image classification, enabling us to distinguish between mediocre and high-quality fruits. The second model focuses on fruit detection. Our ultimate goal is to seamlessly integrate these two models in the future. For our image classification model
we used the VGG16 architecture, a member of the keras family as our main Convolutional Neural Network (CNN) model. For the object detection we used faster R-CNN as our archticture. Both models used different part of the data and needed different processes. The detailed description and the setup of our models is written in the chapters below.

DATA COLLECTION

For the purposes of this specific project, we decided to use a ready dataset, "Fruit Quality Classification", that can be found on Kaggle. This decision was made due to time constraints and because we wanted to focus on the model itself, by allocating as much time as possible to development.

DATASET OVERVIEW

The dataset contains images for 6 fruit classes and 3 quality classes, with a total number of 19526 images.

Fruit Classes:
There are 6 fruit classes represented in the dataset:

1. Apple
2. Banana
3. Guava
4. Lemon
5. Orange
6. Pomegranate

Quality Classes:

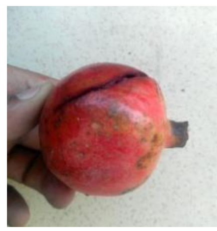
The dataset also categorizes the fruits into 3 quality classes:

1. Bad Fruit
2. Good Fruit
3. Mixed Fruit images:

Sizes:

- Bad Fruit = 256x256 pixels
- Good Fruit = 256x256 pixels
- Mixed Fruit = 256x192 pixels

A sample for the images of each category can be seen below:

	Good	Bad	Mixed
Apple			
Pomegranate			
Orange			
Banana			
Lemon			

Guava			

Table 1: Data Sample

As seen by samples of the images, there are also varying backgrounds and the number of fruit also changes. Moreover, the pictures are also taken from the front ,top, backward and bottom direction, while additionally being rotated 180 degrees. This is true for both good and bad quality fruit for each of the 6 fruit categories and for the mixed images.

A clarification needs to be made for mixed fruit data. Mixed fruit images refer to images of mixed quality but the same type of fruit. For example, mixed oranges contain images with exclusively oranges, multiple oranges in each image, and they range in ripeness. There are images where 100% of the fruit are ripe or unripe, but the majority are mixed.

The non mixed fruit datasets, which only contain 1 class per image (e.g. only good apples), can be easily examined for their frequency, as they do not require additional labeling. The pie chart below depicts the distribution of the 12 classes of the non-mixed images in the data.

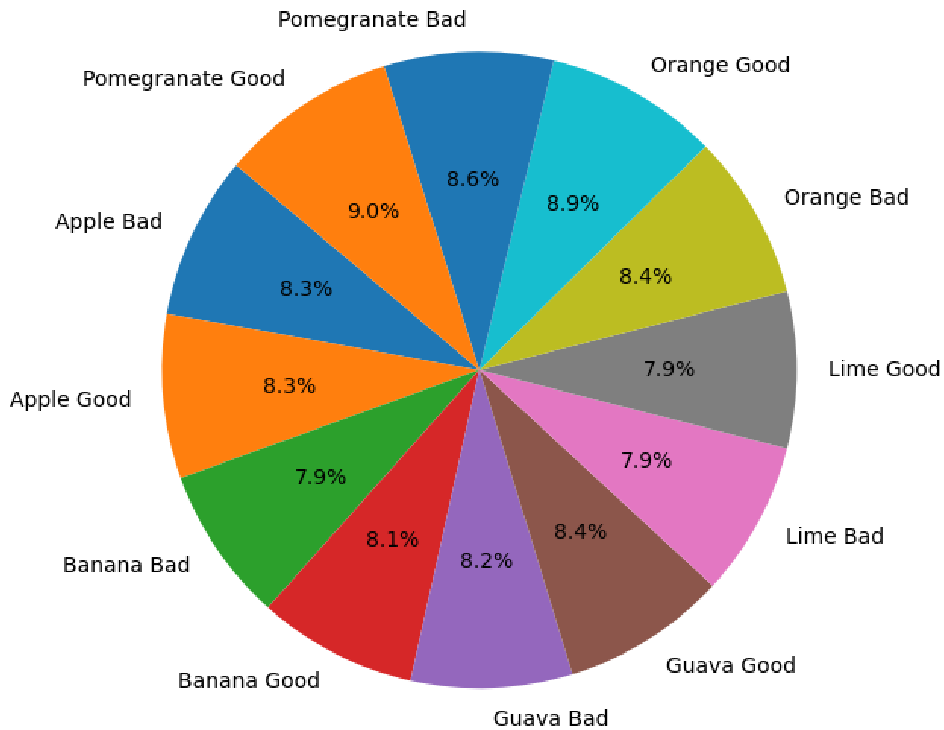


Figure 1: Distribution of the 12 classes of the non-mixed images in the data set

We believe that there is no need for balancing further this distribution, therefore we use them as they are. Any augmentation process will only be applied to the 6 mixed fruit datasets and will be presented right below.

DATA PRE-PROCESSING AND ANNOTATION OF MIXED FRUIT DATA

LABELING

Mixed fruits were annotated and augmented, in order to create the training,val and test datasets that were necessary for our model creation and which contained the bounding boxes for each fruit. The annotation was performed using LabelStudio and augmentation was performed using Python.

Figure 2: Screenshot from one of our labeling jobs

Given the time constraints and the voluminous nature of our dataset, we initially employed a streamlined approach to assess the effectiveness of our object detection model. We strategically narrowed our focus to two fruits—Bananas and Lemons—and annotated 285 images featuring Bananas and another 237 spotlighting Lemons. Upon completing the labeling process, we used Label Studio to export the data in the COCO format, thereby generating annotated images along with a comprehensive .JSON file containing specific annotations.

We initially trained the model on this dataset, only to find that its performance left much to be desired. The model was fraught with misclassifications and inaccurate bounding box placements, prompting us to reconsider our strategy.

To refine the model's predictive accuracy, we turned to a more targeted dataset—images containing exactly five fruits, either all Bananas or all Lemons. This refined dataset consisted of 112 images, a substantially smaller pool compared to our original dataset. Anticipating that this limited dataset might not suffice for the model to effectively learn, we embarked on an image augmentation journey to bolster our dataset's breadth and depth, resulting in a dramatically improved model performance.

It's crucial to mention that we undertook a multi-phased, complex process to arrive at our final model. We commenced with the complete fruit dataset, involving two types of fruits. We then moved to a dataset of images containing a specific number of fruits across all six categories. As a third step, we focused on a combination of two fruits, exploring their entirety before finally narrowing it down to images with a specific count of these two fruits. Each of these phases was an individual project in itself, requiring additional annotation using label studio and training.

AUGMENTATION

Initially, we had considered using a smaller dataset for training, but it quickly became apparent that a larger volume of data was essential for the model to learn effectively.

While we needed to populate the dataset by augmenting the original images, time constraints did not allow us to annotate the entirety of the new increased dataset. This is why, we labeled only the original images and then we used the annotated images for augmentation.

This is why we proceeded with an image augmentation strategy specifically designed to avoid affecting the coordinates of the existing bounding boxes. To accomplish this, we restricted our augmentation techniques to adjustments in contrast and color saturation, bypassing any spatial transformations that could misalign the annotated regions.

For the augmentation process, we created a custom function that modified the image contrast and brightness, thereby enriching the dataset's visual variability. To effectively carry over the bounding box annotations from the original images to their augmented counterparts, we implemented an iterative procedure. This method assigns discrete IDs to the objects within the augmented images, ensuring their representation in the corresponding JSON annotation files. Each augmented image inherits its bounding box annotations from its "parent" image, facilitating a seamless mapping of new IDs onto duplicated bounding box details.

To validate the integrity of this annotation transfer process, we employed random sampling techniques. Specifically, we selected augmented images at random and visualized their bounding boxes based on the amended annotation files. By confirming that the bounding boxes aligned correctly with their corresponding objects in the images, we were able to validate the methodology. Repeating this verification process multiple times provided us with the confidence that our approach for inheriting bounding box annotations from original to augmented images was both accurate and reliable.

The objective of the augmentation was to enrich our mixed fruit dataset, ensuring a more comprehensive and expansive training set for our object detection model. This enhancement not only amplifies the volume of images but also bolsters the model's ability to generalize across diverse scenarios.

MODELS INTRODUCTION

As described before, we developed two models for our project. The first model is an image classification model that uses the VGG16 CNN architecture as its foundation. The second model is an object detection model based on the Faster R-CNN framework.

IMAGE CLASSIFICATION MODEL

ALGORITHMS, NLP ARCHITECTURES/SYSTEMS

We have chosen the VGG16 model from the Keras library as our architecture of the image. VGG16 is a Convolutional Neural Network (CNN) architecture (2019). It is a deep learning model that was introduced by researchers at the University of Oxford in 2014. The "VGG" in VGG16 stands for Visual Geometry Group, the research group where this architecture was developed.

The VGG16 model is renowned for its simplicity and effectiveness in image classification tasks. It comprises 16 weight layers, including 13 convolutional layers and 3 fully connected layers (2023). Its proven track record in various image-related tasks makes it a suitable choice for our project.

To achieve our specific goals, we will use a pre-trained model with weight from ImageNet dataset as our basemodel. The pre-trained weights are up to penultimate layer and we only train the final layer. This strategy is advantageous for us, as it leverages the knowledge learned from a large dataset and adapts it to our task.

We have decided to train the model for a total of 5 epochs. An epoch is a single pass through the entire training dataset, and running too many epochs can lead the model to "memorize" the training data, thereby reducing its generalization capability on unseen data. Therefor the relatively low number of epochs is chosen to prevent overfitting, given that we are only fine-tuning the last layer. Overfitting occurs when the model learns to perform exceptionally well on the training data but fails to generalize to new, unseen data. By restricting the training duration, we aim to strike a balance between adapting the model to our task and preventing overfitting. We also have chosen 5 epochs as the results already revealed promising results.

EXPERIMENTS – SETUP, CONFIGURATION

For the model, we used and divided the full dataset described in data collection into three distinct sets: 70% for training, 15% for validation to fine-tune the model, and another 15% for testing. This partitioning resulted in a total of 9944 training images, 2057 validation images, and 2057 test images, each belonging to one of 12 unique classes.

To enhance the model's ability to generalize and effectively recognize patterns, we employed the ImageDataGenerator class from the Keras library for real-time data augmentation during training. This technique is pivotal in bolstering the model's robustness, mitigating overfitting, and facilitating effective training.

Here are the key parameters we configured for the data augmentation process:

- rescale=1./255: This parameter rescaled the pixel values of input images to the range [0, 1], standardizing the input data and promoting training stability.
- rotation_range=40: We randomly rotated images by up to 40 degrees, introducing variability and aiding the model in object recognition from diverse angles.
- width_shift_range=0.2: Randomly shifting the width of images by up to 20% of the total width enabled the model to adapt to variations in object positions.
- height_shift_range=0.2: Similarly, random height shifting by up to 20% of the total height complemented width shifting for improved model robustness.
- shear_range=0.2: Applying shear transformations to images slightly deformed object shapes, enhancing the model's invariance to shear.
- zoom_range=0.2: Random zooming into or out of images by up to 20% enabled the model to recognize objects at different scales.
- horizontal_flip=True: Horizontally flipping images with a 50% probability introduced horizontal symmetry variations to diversify the dataset.
- fill_mode='nearest': To maintain the image's integrity during augmentation, newly created pixels (due to rotations or shifts) were filled with values from nearby pixels.

To use data generators for both the training and validation datasets, we utilized the flow_from_directory method to extract our data from the folder, specifying the target image size as 256x256 pixels. The data was organized into batches of 32 samples for each training iteration. We adopted the categorical class mode, which is well-suited for our multi-class classification task.

RESULTS & QUANTITATIVE ANALYSIS (INCL. VISUALIZATIONS)

our VGG16 model's performance has been evaluated in terms of both overall accuracy and class-specific accuracies for the different categories.

The model achieved an impressive overall accuracy of 87.8%, indicating its capability to make accurate predictions across different classes. However, the relatively high loss value of 36.7% suggests that there may be room for improvement in terms of minimizing the model's prediction errors.

Seen in the epochs results shown below the model's performance improved steadily throughout the training process. In the first epoch, the training accuracy was approximately 47.81%, and the validation accuracy was 82.16%. This initial accuracy difference indicates that the model's initial weights, obtained from pre-training on a large dataset, were not entirely suited to our specific task.

However following epochs show a positive trend in both training and validation accuracy. By the end of the training process, the model achieved a training accuracy of approximately 71.14% and a validation accuracy of 86.58%. This indicates that the model adapted to the task over time and improved its performance.
The loss values decreased over the epochs, indicating that the model learned to make better predictions with each iteration.

```
Epoch 1/5
311/311 [==============================] - 1692s 5s/step - loss: 1.6578 - accuracy: 0.4781 - val_loss: 0.6908 - val_accuracy:
0.8216
Epoch 2/5
311/311 [==============================] - 1912s 6s/step - loss: 1.0745 - accuracy: 0.6188 - val_loss: 0.5068 - val_accuracy:
0.8386
Epoch 3/5
311/311 [==============================] - 2000s 6s/step - loss: 0.9520 - accuracy: 0.6585 - val_loss: 0.4212 - val_accuracy:
0.8658
Epoch 4/5
311/311 [==============================] - 1755s 6s/step - loss: 0.8829 - accuracy: 0.6822 - val_loss: 0.4080 - val_accuracy:
0.8683
Epoch 5/5
311/311 [==============================] - 1740s 6s/step - loss: 0.8149 - accuracy: 0.7114 - val_loss: 0.3839 - val_accuracy:
0.8658
```

Class specific results

In terms of class specific results, the model provides some notable achievements seen in the figure. Analyzing accuracy for each class is crucial, as it offers valuable insights into the model's performance for individual categories. Notably, classes like "Banana_Bad" and "Banana_Good" achieved exceptional accuracies, both surpassing 99%. This underscores the model's exceptional predictive performance for these categories. Moreover, when visualizing the ground truth with the predicted result, the model shows its predictive capabilities see picture below.

Figure 3: predicted pomegranate bad and ground truth

However, other categories, such as "Guava_Bad" (68.75%) and "Apple_Bad," (78.62%) presented lower accuracies, indicating areas with potential for improvement. These variations in accuracy levels reveal the model's diverse performance across different fruit categories, providing valuable information for future refinements and optimizations.

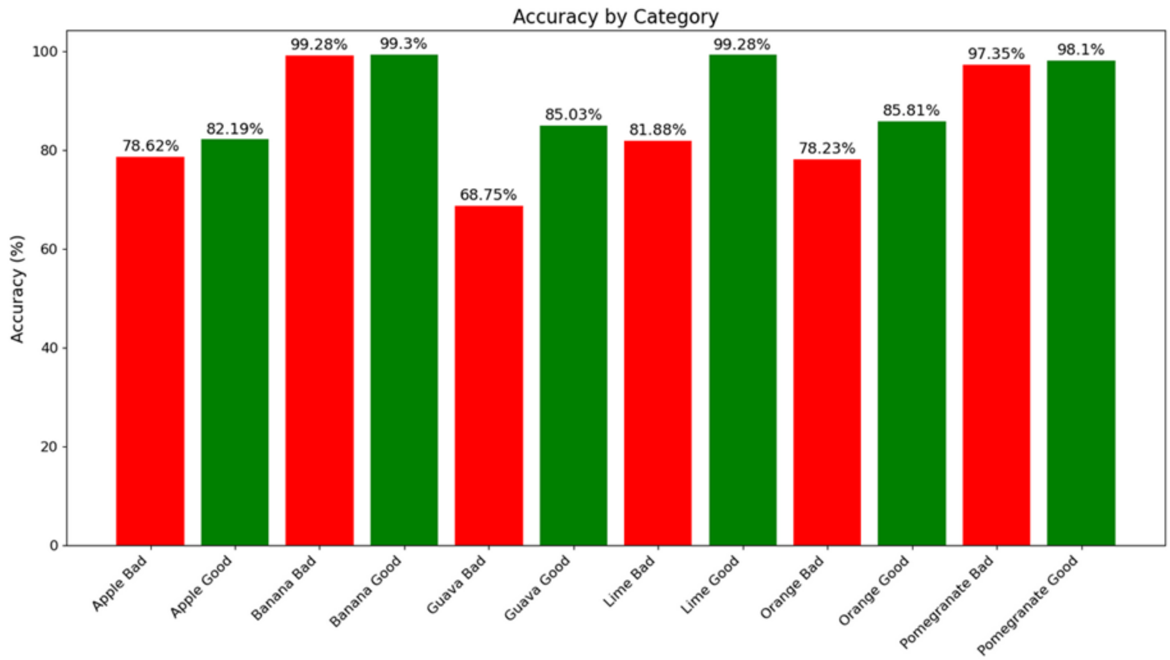


Figure 4: Frequency Barplot

Upon closer examination of "bad apples", “bad oranges” and "guavas," we found an interesting challenge: distinguishing between a bad apple, an orange, and a guava can be quite challenging, even for the human eye (see figure). This observation underscores the visual similarity between these fruit subcategories.

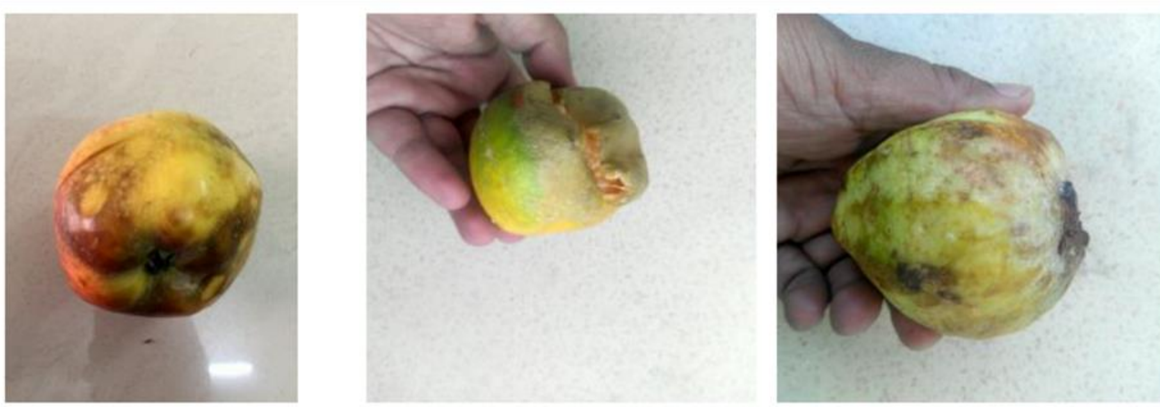


Figure 5: Example of our “bad apple”, “bad orange”, “bad guava” data

QUALITATIVE & ERROR ANALYSIS

Our model performs exceptional accuracy in certain classes, notably "Banana_Bad" and "Banana_Good," indicating its proficiency in recognizing distinguishing features within these categories. However, this high accuracy may be influenced by data imbalance, warranting attention to potential overfitting.

Conversely, the model encounters challenges in distinguishing between "bad" variants of fruits, such as guavas, oranges, and apples. This difficulty stems from the visual similarity between these subcategories, even for human observers.

To address these challenges, future refinements may involve obtaining diverse training data for challenging categories and fine-tuning the model architecture to capture subtle differences. Ongoing efforts aim to enhance the model's robustness and ensure accurate predictions across all fruit variants.

Considering the real-world implications of classification errors, especially in contexts like agriculture and the food industry, underscores the importance of continued model refinement.

FRUIT DETECTION MODEL

ALGORITHMS AND NLP ARCHITECTURES/SYSTEMS

For our fruit detection model, we selected Faster R-CNN (Region-based Convolutional Neural Network), a prominent architecture in the realm of computer vision for object detection. Faster R-CNN is known for its ability to balance high accuracy with computational efficiency, making it well-suited for precisely identifying and localizing diverse fruits in images (Parking Occupancy Detection and Slot Delineation Using Deep Learning: A Tutorial, 2021).

The specific variant of Faster R-CNN we employed is based on ResNet-50 with a Feature Pyramid Network (FPN) backbone. This configuration utilizes a ResNet-50 architecture as the feature extractor backbone, complemented by FPN for effective multi-scale object detection. The backbone's weights are pre-trained on the COCO dataset, imparting valuable knowledge about a wide array of objects and contexts.

Pretraining the backbone on the COCO dataset is a pivotal step as it initializes the model's weights with a foundation in general object recognition. COCO (Common Objects in Context) is a renowned benchmark dataset encompassing object detection, segmentation, and captioning tasks. These pretrained weights capture fundamental image features, including edges and textures, serving as a beneficial starting point for feature extraction.

Faster R-CNN's selection for our fruit detection project is based on its remarkable attributes, combining accuracy and computational efficiency. Its architecture is adept at accurately localizing and identifying fruits within images, making it a compelling choice for our specific task.

EXPERIMENTS - SETUP AND CONFIGURATION

In our project, we focused on training our object detection model using two unique sets of fruit combinations: one featuring bananas and lemons, and another comprising oranges and pomegranates. For the banana-lemon combination, we have designated specific category labels: '0' for the background, '1' for banana, and '2' for lemon. Similarly, in the orange-pomegranate combination, the category labels are set as follows: '0' for the background, '1' for oranges, and '2' for pomegranates.

We initially began with the banana and lemon combination but we wanted to further challenge our model and assess its ability to distinguish between similarly shaped and colored fruits, so we extended our dataset to include a combination of oranges and pomegranates. Both fruits are circular and share a similar orange-red hue, making it an ideal "stress-test" for our model's capability to differentiate and accurately label the objects.

Therefore, we first began our process by augmenting the 112 curated images of Bananas and Lemons with a specific number of five fruits in each image, generating 90 augmented versions for each original image. As a result, our enhanced dataset swelled to a robust size of 10,080 images.

While for the second fruit combination we opted for a slightly different approach this time by selecting images that contain exactly four fruits rather than five. This deviation was intentional; our aim was to gauge the model's robustness and flexibility when subjected to varying conditions, such as a different number of elements within each image.

Initially, our Orange-Pomegranate dataset comprised 84 unique images. So, similarly to our Banana-Lemon dataset, we found that a smaller dataset was insufficient for robust training. To bolster the model's learning potential, we augmented these 84 base images to create again a total of 10,080 images using similar contrast and brightness adjustments as before. This process was crucial to ensure that the model generalizes well and performs reliably under different conditions.

In setting up our experiments, we organized our augmented dataset, splitting it into training (80%), validation (10%), and testing (10%) subsets. Each image in the dataset is meticulously annotated with essential metadata, including unique image IDs, category IDs, and bounding boxes to define the regions of fruits.

By conducting this two-fold experiment—first with the Banana-Lemon combination and then with the Orange-Pomegranate pair—we aim to comprehensively evaluate the model's performance in both straightforward and complex object detection scenarios. These strategically chosen combinations will provide us with insights into the model's strengths and weaknesses, thereby guiding future improvements.

In the training phase, we utilized the Stochastic Gradient Descent (SGD) optimizer from PyTorch's optimization library to refine our model's parameters. To ensure optimal resource allocation and faster computation, we set the batch size to 4. By doing so, the model was able to process four images at a time, providing a good balance between memory usage and gradient estimation.

We intentionally limited the training process to two epochs to limit the risk of overfitting. The SGD optimizer was configured to dynamically identify and update only those model parameters that require gradient-based modification. This ensured that any layers we chose to "freeze" or fix during the training process remained unaffected, thus preserving their pre-trained features.

By marrying efficient batch processing with strategic epoch selection and a dynamic optimization algorithm, we aimed to train a robust and versatile object detection model capable of high accuracy while minimizing the risk of overfitting.

QUANTITATIVE ANALYSIS AND VISUALIZATIONS

The results obtained from our model show accurate labeling of fruit objects within images see figures. However, the metrics indicate suboptimal performance.

Banana-Lemon combination:

- The precision of our model is approximately 32.5%. This indicates that about one-third of the predicted bounding boxes are true positives, showing some effectiveness in our model's predictions.

- Surprisingly, the recall stands at around 3.5%, suggesting that a significant number of actual positive cases were not identified by our model. This discrepancy may be due to limitations in the number of predicted bounding boxes considered for metrics calculation.

- The overall F1 score is 0.0634, highlighting a substantial gap between precision and recall. This imbalance needs to be addressed to improve the model's overall performance.

- When we lowered the IoU threshold to 10%, the precision increased significantly to 97.3%. However, this high precision comes at the cost of a low recall, approximately 9.98%. The F1 score improved to 0.181, indicating room for further refinement.

This less-than-ideal overall performance may be attributed to several factors, although further investigation is required. The low recall may be influenced by the limited number of bounding boxes considered in our evaluation, potentially neglecting correct but lower-ranked predictions.

Despite these seemingly low performance indicators, visual inspection on unseen data (test dataset) told a more optimistic story. For lemons, the visual results are outstanding, with high confidence bounding boxes closely aligning with the actual contours of the fruit (see figure). This suggests that the model is not only reliable but also exceptionally accurate when it comes to identifying and localizing lemons. Evaluating the model's generalization to unseen data the model excels in detecting lemons, achieving prediction confidence scores exceeding 70%. This underscores its robustness and reliability in identifying lemons within previously unseen data.

Figure 6: Visualizing Lemons: Predicted Bounding Boxes with High Confidence.

As for bananas, the model's predictions are generally good (see figure). While the bounding boxes are not as precisely tailored to the fruit's shape as those for lemons, they nevertheless cover the bananas effectively, without including other objects. This implies a more-than-adequate level of accuracy and reliability for tasks requiring banana identification, despite minor room for improvement

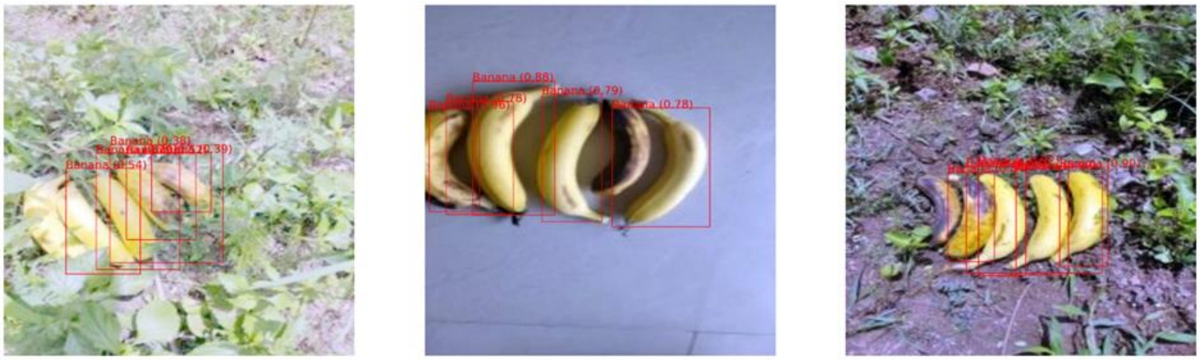


Figure 7: Visualizing bananas: Predicted Bounding Boxes with High Confidence.

Orange-Pomegranate combination:

> -Precision: The model recorded a modest 25% precision rate, indicating that it correctly identifies a quarter of the fruits within the bounding boxes.
>
> -Recall: Alarmingly, the recall hovers at a meager 2.05%, implying that the model fails to capture the majority of actual fruit instances in the images. Again, this may be due to limitation in the number of bounding boxes predicted.
>
> -F1 Score: With an IoU threshold set at 0.4, the model's F1 score comes in at a disappointing 0.037, illustrating the need for substantial improvements in model accuracy.
>
> -When we adjusted IoU threshold at 20% precision soared to an impressive 86.9%, and recall experienced a modest increase to roughly 6.8%. This resulted in a marginally improved F1 score of 0.126

Again the low recall rate suggests that our metrics could be skewed by a limitation in the number of bounding boxes considered, potentially omitting accurate but less conspicuous predictions. Overall, again our results suggest that the model is a promising starting point for object detection in fruit, but it necessitates further adjustments to align its numerical evaluation metrics with its apparent visual successes.

On the other hand, for this combination of fruits again the visual results on new unseen data(test dataset) tell an overwhelmingly positive story for both fruits, standing in sharp contrast to the purely quantitative metrics. The visual evaluations from our test dataset are especially compelling for pomegranates (see figure). The model not only identifies the fruit correctly but does so with a high degree of confidence—exceeding an 85% prediction score. This high degree of accuracy emphasizes the model's robustness and reliability, even when applied to new or previously unseen data sets. Such consistency strongly implies that the model is highly optimized for pomegranate detection tasks, making it a reliable tool for accurate identification and localization of this fruit.

Figure 8: Visualizing pomegranate: Predicted Bounding Boxes with High Confidence

Similarly, in the case of oranges, the model's visual performance is remarkably effective. While the numerical metrics may not fully capture this, the visual plots reveal a high level of precision in the detection and localization of oranges (see figure). This high visual accuracy underscores the model's applicability and reliability in real-world scenarios where orange detection is critical.

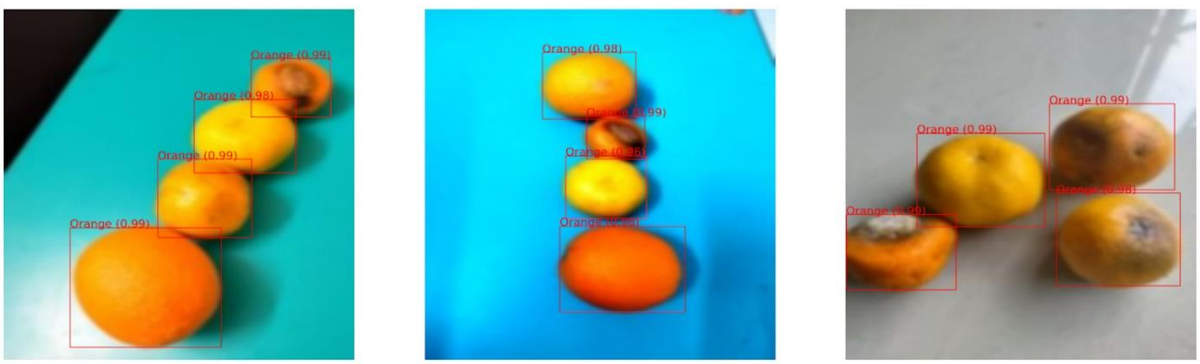


Figure 9: Visualizing oranges: Predicted Bounding Boxes with High Confidence

Overall, the visual outcomes for both fruit combinations are encouraging and suggest a promising avenue for deploying this model in practical applications. The stark contrast between the quantitative metrics and the visual assessments reinforces the need for a more comprehensive evaluation framework, one that integrates both empirical measurements and qualitative assessments for a more holistic understanding of model performance.

QUALITATIVE ASSESSMENT AND ERROR ANALYSIS

Given the underwhelming metrics, a thorough qualitative and error analysis of the model's output is imperative. This analysis is essential to understand the root causes of the suboptimal results. Further investigation is needed to address the issues leading to the relatively low precision and recall, and corrective measures must be explored to enhance the model's performance.

Problematic model performance in fruit detection can arise from a range of factors. These include issues related to data, such as insufficient or imbalanced training data, inaccurate annotations, or the absence of robust data augmentation.

Overfitting, driven by the nature of our augmentation strategy, the complex model architectures or inadequate regularization, can also undermine performance. The tuning of hyperparameters, including learning rates and batch sizes, plays a crucial role. Hardware limitations and real-world environmental variations add additional layers of complexity.

In our case one of the main reasons could be the data imbalance and the hardware limitation. Despite experimenting with various adjustments, including tuning and model architecture changes, we observed no notable improvement in results. Furthermore, our decision to limit training to just 2 epochs was made to prevent overfitting, indicating that it may not be a significant factor.

In the future, conducting a comprehensive analysis of these factors will be crucial for enhancing model accuracy. However, the model does show promising results when looking at the predicted labelings.

MODELS SUMMARY

In summary, our image classification model, employing VGG16 architecture and transfer learning, achieved an 87.8% accuracy. Notably, it excelled in categories like "Banana_Bad" and "Banana_Good". However, visually similar categories presented challenges, indicating the need for data diversity and model refinements.

Our fruit detection model, based on Faster R-CNN architecture, exhibited noteworthy strengths, especially in the accurate identification of lemons, oranges, and pomegranates. However, its performance was less compelling for banana detection. Although the quantitative metrics indicate areas for optimization, the visual evaluations paint a more optimistic picture, hinting at the model's latent capabilities.

To enhance both models, we'll address data imbalances, refine hyperparameters, and conduct comprehensive analyses. Our dual-model approach holds potential for agriculture and the food industry, with ongoing refinement ensuring accuracy across diverse fruit categories.

DISCUSSION, COMMENTS/NOTES AND FUTURE WORK

At the conclusion of our project, we compile some notes detailing opportunities for improvement and the resources required to achieve them. We will discuss this for data selection and handling, and model development, both in theory and practice.

To begin with, we recognize that data quantity is crucial for the performance of our models. This has been proven multiple times during development, as neural networks models performed with increasing precision as data volume per category increased, while they performed significantly worse when the classification levels increased disproportionately to the data.

On the other hand, data quality also has room for improvement, given the fact that all our data were augmentation of a limited original dataset. We believe that introducing more angles, more backgrounds and fruits for each fruit category would increase the models' ability to perform on validation images. In addition, increasing the data size would also introduce the idea of multiple ripeness levels into the equation and move from binary to multilevel classification.

To address the data quality and quantity issue, we ideally would combine multiple datasets. This would require a lot of additional time or additional team members to identify and clean ready made datasets, or better yet, take original pictures to create a curated data source. However, this would significantly increase the time or team members required to capture, augment and label all the data. But, it would undoubtedly yield better results and more options for further development.

In addition, we were hopeful to develop more original models and more variations of each, by finetuning their parameters. However, the long computation time needed did not allow for a lot of experimentation and calibration. For reference, on our equipment, our most complicated model runs for approximately 20 hours. It goes without saying that this was very limiting for our work. This boldly highlighted the need for much better equipment, especially regarding processing capability and memory.

However, aside from the technical development which would benefit tremendously from time and equipment, we can expect better performance on a theoretical level as well if we have more time and expertise. This is because we required a considerable number of hours to understand theory, study existing models and learn from our errors (like attempting to use YOLO in an unsuited problem and trying a number of different fruit combinations). This would have been resolved if we had more time at our disposal to collect knowledge, learn and make mistakes or, instead, have an experienced member on our team to guide us.

Moreover, combining all that is mentioned above, would enable us to create a unified model. Right now, our models tackle different stages of the fruit ripeness recognition, with one model identifying fruit within bounding boxes and another classifying whether it is ripe or not. We would certainly utilize the additional resources to create the pipeline which enables information from the first model to travel to the next.

In conclusion, while we are satisfied with our progress and our results, it is clear that there is significant room for improvement and further development. To achieve this, we would need to increase our team, to include more analysts and at least one experienced data scientist, invest in equipment and secure our most important resource: time. We are positive that, with the combination of human and technological capital, we can greatly expand our potential.

MEMBERS/ROLES

Our diverse student team effectively collaborated on our project by leveraging each member's unique background and skill set.

Julia Baardse, holding a degree in business engineering and skilled in data management, played a pivotal role in sourcing, labeling, and augmenting the data. She also made critical changes to annotations for every model variation, while also contributing to the coding process.

Christos Vlassis, with a background in economics and simultaneously our most tech-savvy member, provided essential guidance on outlining the skeleton of our project. He was primarily involved in the development of the neural network models and was instrumental in debugging efforts.

Eleni Gemintzi, with a bachelor’s degree in economics as well, took charge of designing the business plan. She set project goals, managed time schedules, and organized the team, while ensuring consistency between the business and the technical components of the project.

Griselda Subashaj, a detail-oriented mathematician, was a valuable asset in coding, calibrating, refining and testing our models. Her troubleshooting skills contributed significantly to the project's success.

While we had designated roles, our team fostered a culture of collaboration and constant communication, allowing each member to contribute significantly to our project's success.

TIME PLAN OF ACTIVITIES

Up to date (1.5 months)

Week 1:	Researching data, relevant projects and business information
Weeks 2-3:	Developing CNN model, documenting results, annotating data, developing business plan
Week 4-6:	Developing different models, troubleshooting
Week 7:	Assembling all information, polishing and finalizing

Future work* (3 months)

Weeks 8-9:	Labeling and augmenting new data, resolving existing issues in code.
Weeks 10:	Researching next step, revising business plan
Weeks 11-15:	Developing more complex models
Weeks 16-18:	Creating a pipeline to unify all best models into one.
Week 19:	Reorganizing, evaluating and finalizing.

*Assuming we have gained access to resources mentioned under “notes” section.

REFERENCES

2023. Beginner’s Guide to VGG16 Implementation in Keras. *builtin.com.* [Online] 2023. https://builtin.com/machine-learning/vgg16?fbclid=IwAR1b7bckqrXPoM8pngheSGoMvVyLNcqi7Rcxr0PqPZY0oCyQlZsTuSXOwrM.

2019. Convolutional Neural Networks Explained: Using PyTorch to Understand CNNs. *builtin.com.* [Online] 2019. https://builtin.com/data-science/convolutional-neural-networks-explained?fbclid=IwAR1b7bckqrXPoM8pngheSGoMvVyLNcqi7Rcxr0PqPZY0oCyQlZsTuSXOwrM.

Eurostat. 2023. *Food waste and food waste prevention - estimates.* 2023.

Faniadis, Dimosthenis. 2020. *Food Processing Ingredients.* s.l. : United States Department of Agriculture, 2020.

Parking Occupancy Detection and Slot Delineation Using Deep Learning: A Tutorial. **Debaditya Acharya, Kourosh Khoshelham. 2021.** s.l. : ResearchGate, 2021.

Thakur, Rohit. 2023. Builtin.com. *Beginner’s Guide to VGG16 Implementation in Keras.* [Online] March 2023. https://builtin.com/machine-learning/vgg16?fbclid=IwAR1b7bckqrXPoM8pngheSGoMvVyLNcqi7Rcxr0PqPZY0oCyQlZsTuSXOwrM.

WRAP. 2023. *WRAP Annual Report 2022/23.* 2023.