T.R.

GEBZE TECHNICAL UNIVERSITY

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

SYSTEM ESTIMATING TOMATO SIZE ON BRANCHES IN GREENHOUSES

EMRE YILMAZ

SUPERVISOR
PROF. YUSUF SINAN AKGÜL

GEBZE
2024

T.R.
GEBZE TECHNICAL UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

SYSTEM ESTIMATING TOMATO SIZE ON BRANCHES IN GREENHOUSES

EMRE YILMAZ

SUPERVISOR
PROF. YUSUF SINAN AKGÜL

2024
GEBZE

GEBZE TECHNICAL UNIVERSITY	GRADUATION PROJECT JURY APPROVAL FORM

This study has been accepted as an Undergraduate Graduation Project in the Department of Computer Engineering on 12/06/2024 by the following jury.

JURY

Member (Supervisor)	:	Prof. Yusuf Sinan AKGÜL
Member	:	Dr. Yakup GENÇ

ABSTRACT

The objective of this project is to develop an artificial intelligence model that can continuously monitor the number of tomatoes and accurately measure their dimensions in a greenhouse environment. This initiative aims to enhance the tracking of tomato plant productivity and facilitate proactive measures to optimize yields.

A comprehensive dataset was initially gathered from external sources for segmentation purposes and further supplemented by data collected using a Kinect camera from the greenhouse affiliated with the Molecular Biology department at Gebze Technical University. This data was crucial for the size estimation calculations. After preparing and augmenting the dataset, the model training phase was initiated.

The findings from this project indicate that the proposed method for tomato size estimation using depth imaging has achieved substantial success. The integration of these technologies promises significant advancements in precision agriculture applications, particularly in the effective monitoring and management of tomato crops.

Keywords: tomato, size estimation, segmentation, detection, greenhouse, agriculture, artifical intelligence.

ÖZET

Bu proje, sera ortamında domates sayısını sürekli olarak izleyebilen ve boyutlarını doğru bir şekilde ölçerek domates büyümesini takip eden bir yapay zeka modeli geliştirmeyi hedeflemektedir. Bu girişim, domates bitkilerinin üretkenliğini takip etmeyi geliştirmeyi ve verimleri optimize etmek için proaktif önlemler almayı kolaylaştırmayı amaçlamaktadır.

Segmentasyon amacıyla dış kaynaklardan kapsamlı bir veri seti toplandı ve bu veri seti, Gebze Teknik Üniversitesi Moleküler Biyoloji bölümüne bağlı seradan Kinect kamerası kullanılarak toplanan verilerle daha da zenginleştirildi. Bu veriler, boyut tahmini hesaplamaları için hayati önem taşımaktadır. Veri seti hazırlandıktan ve zenginleştirildikten sonra model eğitim aşamasına geçildi.

Bu projeden elde edilen bulgular, derinlik görüntüleme ve segmentasyon tekniklerini kullanarak yapılan domates boyut tahmini yönteminin önemli bir başarı sağladığını göstermektedir. Bu teknolojilerin entegrasyonu, özellikle sera ürünlerinin etkin izlenmesi ve yönetimi konusunda hassas tarım uygulamalarında önemli ilerlemeler vaat etmektedir.

Anahtar Kelimeler: domates, büyüklük tahmini, segmentasyon, tespit, sera, tarım, yapay zeka.

ACKNOWLEDGEMENT

In this project, I would like to express my respect and gratitude to my supervisor Prof. Dr. Yusuf Sinan Akgül, who never withheld his support, to Doğa Nalcı and Nida Arslan, who assisted me throughout the project. Also, I extend my respects to Prof. Dr. Yelda Özden, the head of the Department of Molecular Biology and Genetics at Gebze Technical University, for her support.

Emre YILMAZ

LIST OF SYMBOLS AND ABBREVIATIONS

Symbol or Abbreviation		Explanation
YOLO	:	You Only Look Once (An Object Detection Algorithm)
RGB	:	Red-Green-Blue (Color Model)
MSE	:	Mean Squared Error (Evaluation Metric)

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. PROJECT DESCRIPTION

In commercial greenhouses, accurately monitoring the growth and health of tomato plants is crucial for optimizing productivity. This project aims to enhance the way growers assess tomato plants by focusing on counting the tomatoes and determining their sizes, specifically measuring the major and minor axes in centimeters.

Utilizing a combination of artificial intelligence and deep learning, this project will collect data through external segmentation sources and a Kinect camera stationed at the Gebze Technical University's greenhouse. This approach ensures precise measurements of the tomatoes, aiding in better growth tracking and yield estimation.

By providing detailed and accurate metrics on tomato sizes and counts, the project will enable farmers to make informed decisions about plant care, leading to improved health and increased productivity of the plants. The ultimate goal is to equip greenhouse operators with the tools to ensure their tomatoes are as healthy and productive as possible, enhancing both the quantity and quality of the harvest. This initiative represents a significant step forward in the application of smart agricultural practices within the greenhouse industry.

2. MAIN APPROACH TO PROBLEM

2.1. Hypothesis

The primary objective of this project is to accurately determine the size of tomatoes. This task naturally includes the requirement to correctly detect tomatoes on a plant as well. Therefore, the first step for this project is to develop a tomato detection algorithm. This will allow us to both count the number of tomatoes on the plant and identify which tomatoes will be used for size estimation. [1] [2]

2.2. Tomato Size Estimation

The accurate estimation of tomato sizes is pivotal for understanding and enhancing greenhouse crop yields. This project utilizes advanced image processing techniques to determine the dimensions of tomatoes, specifically focusing on the major and minor axes measurements in centimeters. The process of size estimation is directly dependent on the precise detection and segmentation of tomatoes, as outlined in the previous sections.

2.2.1. Methodology

For the size estimation, this project employs a two-step approach. Initially, segmentation models are used to isolate tomatoes from their background and other plant elements. Once the tomatoes are segmented, the next step involves calculating the pixel dimensions of each tomato.

2.2.2. Pixel to Real-Size Conversion

After obtaining the pixel dimensions, these measurements must be converted into real-world units (centimeters). This conversion uses a scaling factor derived from the known distance between the camera and the tomato during image capture, which is facilitated by the depth data from the Kinect camera. The scaling factor is calculated by correlating known physical distances with their corresponding pixel measurements under controlled conditions.

2.3. Tomato Detection

Effective tomato detection is the foundational step in accurately estimating the size of tomatoes in greenhouse environments. This project evaluates two primary methods of detection: bounding box approaches and segmentation techniques, each with its unique merits and challenges.

2.3.1. Bounding Box Methods

Bounding box methods involve enclosing detected tomatoes within rectangular boxes. This method is straightforward and fast, making it suitable for real-time applications. However, as highlighted earlier, bounding boxes often struggle to accurately capture the precise dimensions of non-uniform or overlapping tomatoes, especially when they are not perfectly aligned with the camera's perspective.

Figure 2.1: Tomato With Bounding Box

2.3.2. Segmentation Techniques

Given the limitations of bounding box methods, this project primarily utilizes segmentation techniques for tomato detection. Segmentation involves delineating the exact outline of each tomato, providing a detailed shape and size, which is crucial for accurate size estimation. This method proves to be more reliable for obtaining the major and minor axes of tomatoes, particularly when the fruits are clustered or partially obscured.

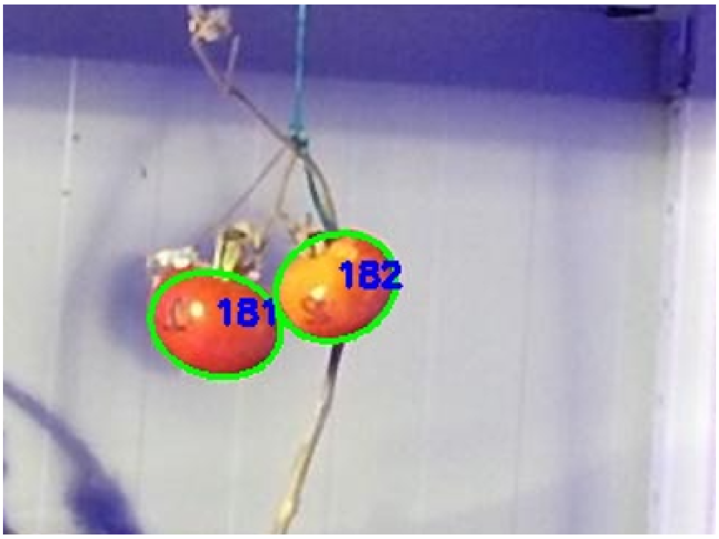

Figure 2.2: Tomato With Segmentation

2.3.3. Integration with Size Estimation

Once tomatoes are detected and segmented, the data is seamlessly integrated with the size estimation module. This integration ensures that each detected tomato is immediately processed for size measurement, streamlining the workflow from detection to size calculation.

3. DATASET

3.1. Segmentation Data Collection

For the segmentation task, a dataset was carefully collected from external sources. Special attention was given to ensure that the dataset was specific to greenhouse environments, as the project's main objective is to measure the size of tomatoes within such settings.

Figure 3.1: Sample tomato image from greenhouse dataset

3.2. Size Estimation Data Collection

The data for size estimation could not be utilized from the same external sources used for segmentation due to the lack of depth information and the absence of real measurements. Consequently, a specialized dataset was gathered from an experimental greenhouse at the Molecular Biology Department of Gebze Technical University, which contains a single tomato plant.

As observed in the photograph, the greenhouse employed purple lighting for experimental purposes. To integrate this data with my own, it was necessary to photograph the tomatoes under white light. Therefore, I illuminated the tomatoes with a projector I brought along.

Figure 3.2: Another sample from the greenhouse dataset

Figure 3.3: Experimental greenhouse with purple lighting

Data was collected using an Xbox Kinect camera. A custom C++ program uses Kinect SDK v2.0 was developed to capture the RGB images of the scene, along with a JSON file for each image containing the distance information of every pixel.

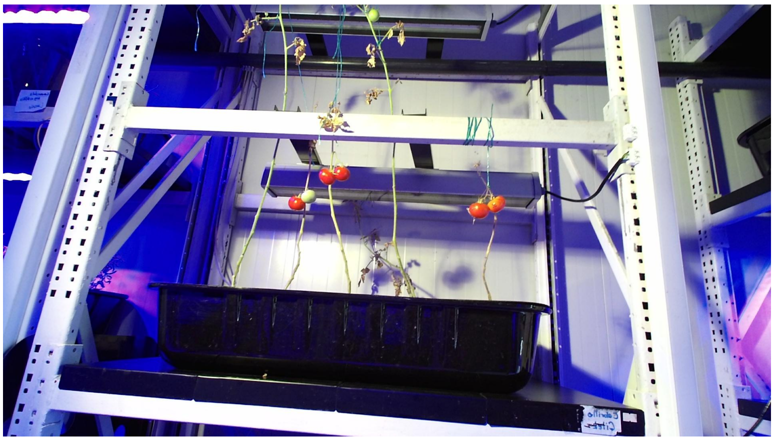

Figure 3.4: RGB photograph of the tomato plant

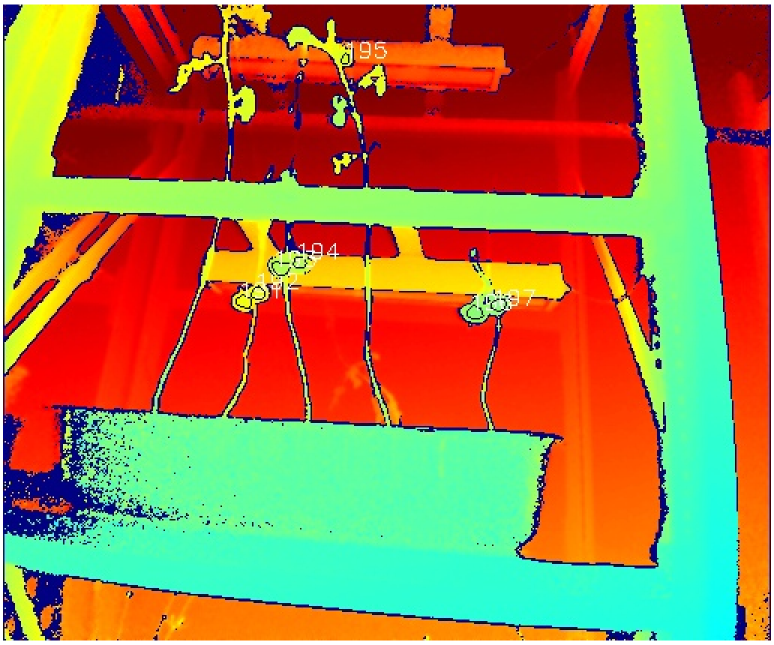

Figure 3.5: Visualized depth information version of Figure 3.4

Five trips were made to the greenhouse to collect this data. During each visit, a caliper was used to take physical measurements of the tomatoes present. To match these measurements with the photographs, the tomatoes were numbered. This method ensured that we gathered accurate information on the RGB photographs, depth details, and the real measurements of the tomatoes from the experimental greenhouse.

Figure 3.6: Numbered Tomatoes

4. DATA PREPARATION

This chapter details the comprehensive process of preparing and processing the dataset required for the project's goals of tomato segmentation and size estimation.

4.1. Segmentation Dataset

The segmentation dataset was meticulously prepared and optimized for immediate use in the model. This preparation included selecting images suitable for training the model to accurately identify and segment tomatoes from the surrounding environment.

4.2. Data Collection for Size Estimation

Size estimation required individual segmentation of tomatoes, collection of depth information, and physical measurements. As discussed in the previous chapter, numerous visits were made to the greenhouse, and extensive data was collected to support this phase.

4.2.1. Segmentation Process

Once the data collection phase was completed, the segmentation phase commenced. Each tomato was segmented individually, and these segmentations were intended to be used for extracting depth information. However, it was discovered during this phase that the program using the Kinect SDK had failed to align the depth and RGB images correctly. This misalignment made it impossible to use the RGB segmentations to capture accurate depth information.

Attempts to align the photographs after data collection were unsuccessful due to time constraints, leading to a decision to segment the visualized depth photographs individually, akin to the RGB photos. This process, although time-consuming, resulted in both depth and RGB photographs of approximately 300 tomatoes being segmented and included in the dataset.

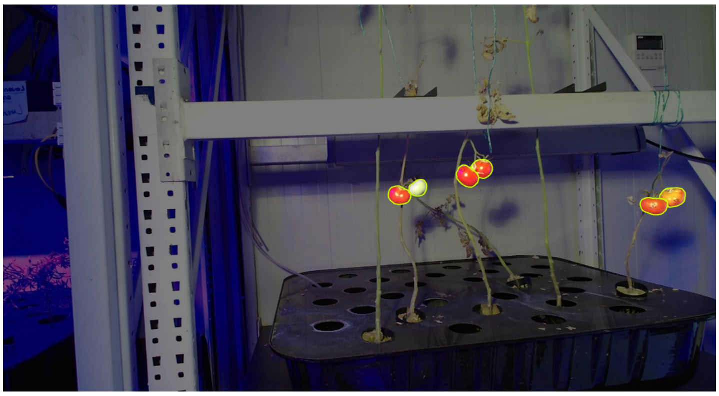

Figure 4.1: Segmented RGB photo of a tomato

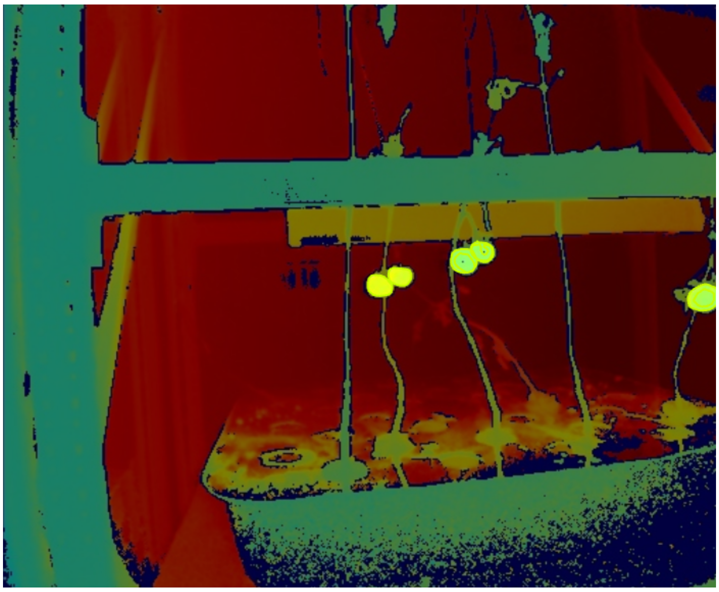

Figure 4.2: Segmented depth photo corresponding to the RGB image

4.2.2. Elliptical Fitting for Axis Measurement

To address the challenges of calculating the major and minor axes from segmented images, an elliptical fitting approach was adopted. This technique involves fitting an ellipse to each segmented tomato image to accurately determine the axes. This method compensates for any irregularities in the segmentation and provides a consistent basis for measuring dimensions, especially in cases where tomatoes are partially occluded or not fully visible.

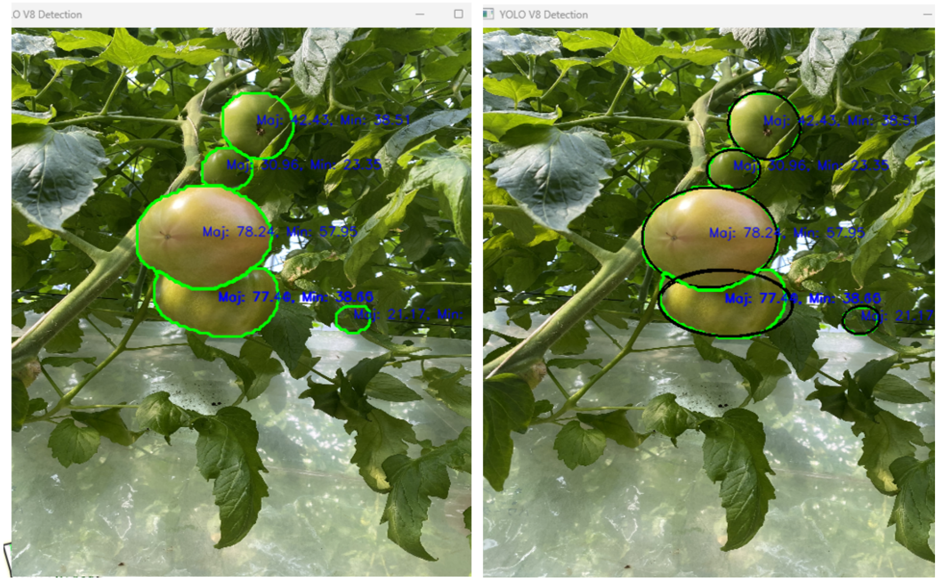

Figure 4.3: Example of Tomato Photo Fitted Ellipse

As you can see in the figure 4.3, by fitting ellipses with a natural margin of error, we can capture the entire tomatoes, even those not fully visible, and obtain their major and minor axes. This method not only provided a more reliable basis for size estimation but also allowed for the inclusion of unseen parts of the tomato through the elliptical approximation.

4.2.3. Data Matching and Creation

A Python script was developed to match each segmented RGB photograph with its corresponding segmented depth photograph. This matching allowed for the collection of depth information for each segmented tomato. These depth details were then combined with measurements taken using a caliper to start the creation of a JSON file for size estimation.

4.2.4. JSON Entry Structure

The JSON file was structured to include detailed entries with pixel-wise major and minor axis lengths, depth information, and the corresponding major and minor axis lengths in centimeters. Each JSON entry also reflects the measurements derived from the elliptical fitting, ensuring the accuracy and consistency of the size data recorded.


```json
[
    {
        "id": 0,
        "major_axis_length": 58.21798324584961,
        "minor_axis_length": 51.89908218383789,
        "mean_depth": 769.5357055664062,
        "real_major_axis_length": 4.0,
        "real_minor_axis_length": 3.2
    },
```


Figure 4.4: Example of a JSON entry for size estimation

5. MODEL BUILDING

This chapter describes the development and training of the segmentation model used in the project.

5.1. Segmentation Model

The segmentation model is a crucial component of this project, designed to identify and segment tomatoes accurately within the images. For this purpose, the YOLOv8 model was employed, utilizing the dataset prepared as described in the previous chapters.

5.1.1. Training the YOLOv8 Model

The YOLOv8 model was trained using the dataset available, and the results obtained are presented in the table below. Although the results were lower than initially expected for this project, they have been deemed experimentally sufficient for the purposes of this study.

Metric	**Value**
Box Precision	0.805%
Box Recall	0.81%
mAP50	0.85.7%
Mask Precision	0.812%
Mask Recall	0.801%
mAP50	0.855%

Table 5.1: Performance metrics of the YOLOv8 model

This section has explained how the segmentation model was configured and trained. The results, while not meeting the optimal projections, are satisfactory and sufficient for continuing the project's further stages.

5.2. Size Estimation

This section describes the methods used for estimating the real dimensions of tomatoes based on segmented images and depth information.

5.2.1. Methodology

Two approaches were adopted for size estimation. The first approach involves using linear regression models to predict the real major and minor axis lengths of the tomatoes.

5.2.1.1. Alternative Method: Linear Regression Model

Linear regression models were utilized to establish a relationship between the pixel-based measurements from images (major and minor axis lengths, mean depth) and the actual sizes of tomatoes. The process involves several key steps as outlined below:

5.2.1.1.1 Data Preparation The JSON file containing the segmented data and corresponding measurements was read into a pandas DataFrame. This file includes features such as pixel-wise major and minor axis lengths and mean depth values, along with the real-world measurements of these axes.

5.2.1.1.2 Feature Selection and Model Training Separate models were built for predicting the major and minor axis lengths. Features selected for the models included the major and minor axis lengths and mean depth from the segmented images.

```
# For Major Axis Length Model
X = df[['major_axis_length', 'minor_axis_length', 'mean_depth']]
y_major = df['real_major_axis_length']

# For Minor Axis Length Model
y_minor = df['real_minor_axis_length']
```

The dataset was split into training and testing sets, with 80% used for training and 20% for testing. Linear regression models were then trained on these sets.

5.2.1.1.3 Model Evaluation The models were evaluated using the mean squared error to measure their accuracy.

Model	Mean Squared Error
Major Axis Length	0.06
Minor Axis Length	0.09

Table 5.2: Performance metrics of the linear regression models

This methodology ensures that the size estimation models are not only theoretically sound but also practically applicable, allowing for accurate predictions of tomato dimensions in a real-world setting.

5.2.2. Main Method: Ratio-Based Estimation

5.2.2.1. Method Overview

This alternative approach to size estimation employs a ratio-based method to calculate the real dimensions of the tomatoes. The essence of this method lies in deriving a ratio for each tomato individually. This ratio correlates the real-world measurements with calculated pixel and depth values to estimate actual tomato sizes.

5.2.2.1.1 Detailed Procedure The process begins by loading data from a JSON file, which contains detailed information about each tomato, including its pixel-based dimensions and real-world measurements. For each entry, the method calculates a ratio of the real length to the product of the pixel-based length and the mean depth, capturing the scaling factor needed to transform image measurements into real-world dimensions.

5.2.2.1.2 Implementation The Python script processes the data as follows:

- It loads the JSON data into a structured format.
- For each tomato, it calculates the major and minor axis ratios.
- These ratios are then used to predict the real lengths by applying them back to the measured pixel dimensions, adjusted by the depth information.

5.2.2.1.3 Results and Evaluation To evaluate the effectiveness of this method, the script calculates the mean squared error (MSE) for each tomato individually. This provides a quantitative measure of the model's accuracy. The best performing MSE value is selected to determine the ratios used for the project, ensuring the most accurate size estimation. Histograms are generated to compare the distributions of predicted and real lengths, offering a visual assessment of the model's performance.

5.2.2.2. Visualization of Results

The histograms illustrate the alignment between the predicted and actual measurements, highlighting the precision of the ratio-based estimation method. These visualizations are crucial for assessing the consistency and reliability of the approach.

Additionally, scatter plots are used to further investigate the relationship between the calculated ratios and their MSE values. This analysis helps identify the most effective ratios and highlights areas where the model may require adjustments.

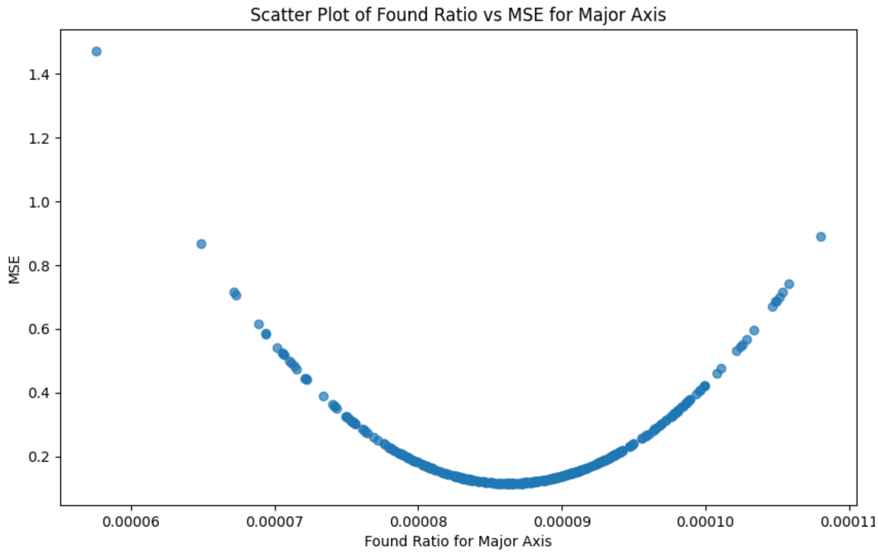

Figure 5.1: Graph of MSE and found ratio to estimate real major axis lengths

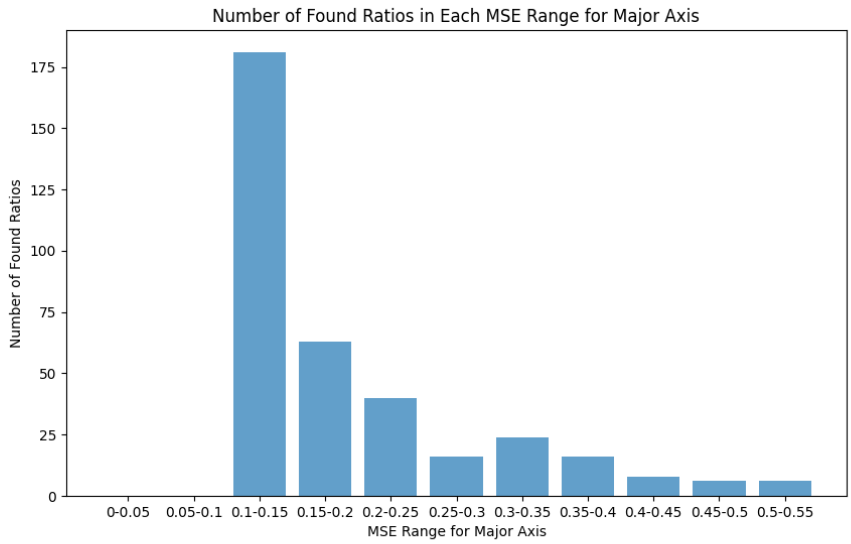

Figure 5.2: MSE Range for Major Axis

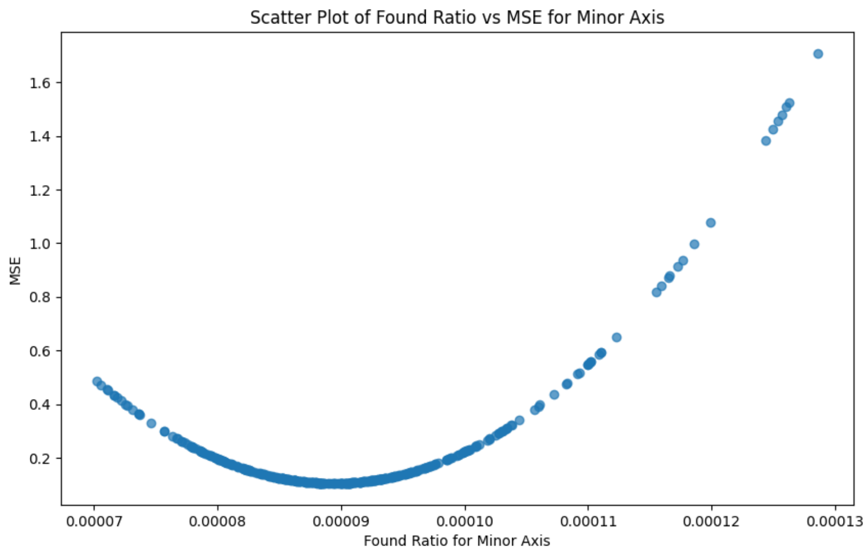

Figure 5.3: Graph of MSE and found ratio to estimate real minor axis lengths

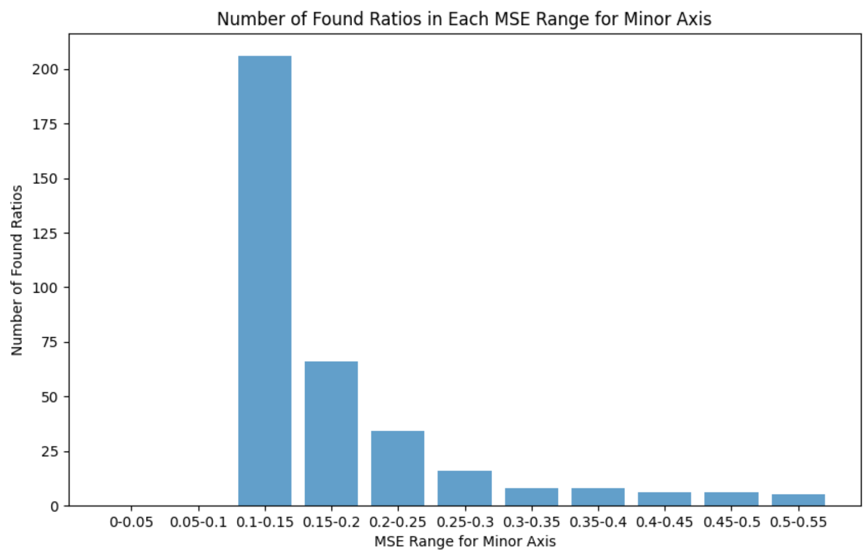

Figure 5.4: MSE Range for Minor Axis

5.2.3. Conclusion

This ratio-based estimation method provides a complementary approach to the linear regression models used in this project. By calculating a specific ratio for each tomato and using the ratio that results in the best mean squared error (MSE), the project benefits from a robust framework for accurately estimating the sizes of tomatoes. Importantly, when examining the graphs and histograms associated with these ratio calculations, the results consistently demonstrate a high level of accuracy and reliability. This visual validation confirms the effectiveness of the ratio-based approach, enhancing the overall precision and utility of the agricultural assessment tools. This consistent pattern seen across graphical representations assures us of the robustness and applicability of the method in real-world agricultural scenarios.

6. TEST AND RESULTS

This chapter outlines the testing procedures used to validate the effectiveness of the developed models and their compliance with the project's success criteria.

6.1. Test

The test code has been developed using C++ and utilizes the Kinect SDK. The process involves capturing RGB images from the Kinect, saving them, and simultaneously retrieving and aligning the depth information related to these images. Once the images are captured and saved, a Python script is executed. This script loads the segmentation model, performs segmentation on the RGB image, and fits an ellipse to the segmentation result. The masks obtained from segmentation are then applied to the depth information to calculate the average depth of the tomatoes. This process results in the collection of essential parameters for size estimation, including pixel-based major and minor axes lengths and depth information.

Three tomatoes were used to conduct two demo tests each with different models. Previously, parameter calculations and linear regression were evaluated using test sets. This section presents the demo results obtained with real data.

6.1.1. Evaluation of Linear Regression and Ratio-Based Results

Both the linear regression model and the ratio-based calculation method have been tested separately to evaluate their performance. Tested tomatoes tabulated as follows:

Tomato ID	Real Major Axis Length	Real Minor Axis Length
0	6.1 cm	5.8 m
1	6.6 cm	6.6 cm
2	5.7 cm	4.9 cm

Table 6.1: Tomato Test Set Real Size

Figure 6.1: Test Tomatoes

6.1.1.1. Linear Regression Results

The table below presents the results from the linear regression model. It includes measurements of pixel-based major and minor axes, mean depth, and the estimated major and minor axis lengths calculated by the model.

Tomato ID	Pixel Major	Pixel Minor	Mean Depth	Estimated Maj.	Estimated Min.
0	90.31	78.94	804.49 mm	5.04 cm	3.68 cm
0	115.03	103.13	585.63 mm	5.48 cm	3.96 cm
1	112.02	104.4	695.44 mm	5.72 cm	5.04 cm
1	76.94	74.74	956.7 mm	5.04 cm	3.79 cm
2	112.08	101.14	630.08 mm	5.5 cm	3.97 cm
2	91.18	74.13	821.2 mm	5.03 cm	3.58 cm

Table 6.2: Results of the Linear Regression Model

Figure 6.2: Segmentation Result of Tomato-ID-0 in Lineer Regression Model Using

As seen, although the segmentation model does not work sufficiently well for the demo and needs improvement, we can ignore this failure using the fit ellipse method. However, even if segmentation seems acceptable and correct, estimation is not acceptable in this model.

Major Axis MSE	Minor Axis MSE
0.81	4.09

Table 6.3: Mean Squared Error for Linear Regression Model

6.1.1.2. Ratio Calculation Results

The table below shows the results obtained from the ratio calculation method. Similar to the linear regression model, it includes pixel-based major and minor axes, mean depth, and the estimated dimensions using the ratio-based method.

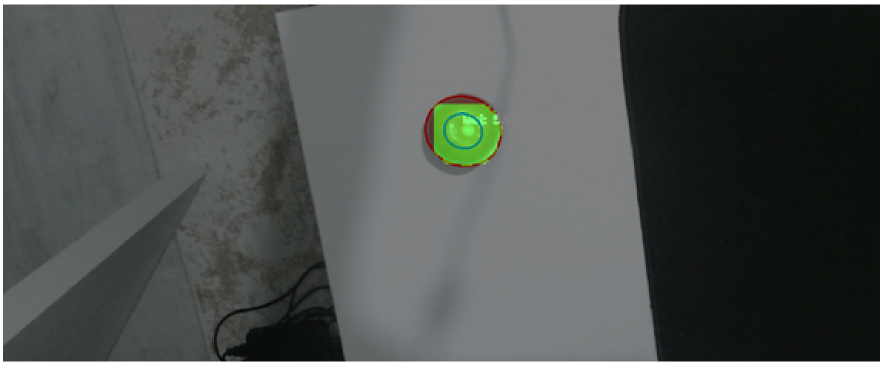

Figure 6.3: Segmentation Result of Tomato-ID-0

As seen, although the segmentation model does not work sufficiently well for the demo and needs improvement, we can ignore this failure using the fit ellipse method. Every prediction made in Table 6.4 results in a segmentation similar to Fig 6.3. As long as the segmentation is successful, the success of the calculation can be examined from the table 6.4.

Tomato ID	Pixel Major	Pixel Minor	Mean Depth	Estimated Maj.	Estimated Min.
0	99.96	99.01	687.1 mm	5.8 cm	5.92 cm
0	100.31	95.18	693.8 mm	6.0 cm	5.9 cm
1	116.85	109.32	657.02 mm	6.62 cm	6.41 cm
1	78.46	90.63	867.34 mm	6.78 cm	6.08 cm
2	100.77	82.79	707.17 mm	6.1 cm	5.2 cm
2	91.34	69.11	788.78 mm	6.21 cm	4.87 cm

Table 6.4: Results of the Ratio Calculation Model

Major Axis MSE	Minor Axis MSE
0.08	0.06

Table 6.5: Mean Squared Error for Ratio Calculation

6.1.2. Analysis

The testing phase revealed a significant discrepancy between the performance of the linear regression model and the ratio calculation method. Despite showing a low mean squared error (MSE) during evaluation, the linear regression model performed poorly when compared against real-world measurements. Conversely, the ratio calculation method not only exhibited a low MSE but also demonstrated high accuracy and consistency with real data.

6.1.2.1. Discrepancies in Linear Regression Model

Several factors might explain why the linear regression model, despite its low MSE in a controlled testing environment, fails to provide accurate predictions in real-world scenarios:

6.1.2.1.1 Overfitting to Training Data The linear regression model might have overfitted the training data, capturing noise and patterns that do not generalize well to new, unseen data. This overfitting results in deceptively low MSE values during testing but fails to perform accurately when applied to actual measurements.

6.1.2.2. Effectiveness of Ratio Calculation Method

The ratio calculation method, on the other hand, aligns closely with real-world measurements and maintains a low MSE. This effectiveness can be attributed to several reasons:

6.1.2.2.1 Normalization of Variability The ratio method normalizes the measurements by directly comparing real lengths to the product of pixel-based lengths and mean depth. This normalization helps in capturing the inherent variability in tomato sizes and shapes, making the method robust to differences in individual tomatoes.

6.1.2.2.2 Dynamic Adjustment Each tomato's ratio is calculated independently, allowing the method to dynamically adjust to the specific characteristics of each fruit. This individualized approach avoids the generalization pitfalls that affect the linear regression model, ensuring that each calculation is tailored to the unique features of the tomato.

6.1.2.2.3 Simpler Assumptions The ratio method makes fewer assumptions about the relationship between variables, focusing on a straightforward proportional relationship that inherently adjusts for differences in scale and depth. This simplicity can be more effective than trying to fit a complex model to inherently noisy or variable data.

6.1.3. Conclusion

In summary, the linear regression model's poor real-world performance despite a low MSE indicates that the model may not generalize well outside the controlled test environment due to overfitting and inadequate feature representation. In contrast, the ratio calculation method's alignment with both low MSE and accurate real-world measurements suggests its robustness and reliability in practical applications. This insight underscores the importance of choosing the right method for size estimation, particularly in the context of agricultural applications where variability and complexity are high.

7. EVALUATION OF SUCCESS CRITERIA

This chapter evaluates the project's success against predefined criteria. The effectiveness of the models developed is assessed based on the performance metrics outlined below.

7.1. Criteria 1: Tomato Detection F1 Score

The first criterion for success is that the F1 score for tomato detection should exceed 85%. Our segmentation model achieved a precision of 0.80 and a recall of 0.81, resulting in an F1 score of approximately 0.805. Despite falling slightly short of the target, this performance is noteworthy given the limitations in our dataset. The available segmentation datasets were insufficient, and it was challenging to find adequate external data sources to supplement our training set. While the F1 score did not meet the 85% threshold, the results are considered satisfactory under the circumstances, reflecting the model's robustness given the data constraints.

Metric	Value	Comment
Precision	0.80	
Recall	0.81	
F1 Score	0.805	Slightly below target

Table 7.1: Performance Metrics for Tomato Detection

7.2. Criteria 2: Tomato Size Estimation MSE

The second criterion specifies that the mean squared error (MSE) for tomato size estimation must be less than 10%. This criterion was successfully met. Both the training and test phases demonstrated that the MSE for size estimation remained well below the 10% threshold, validating the accuracy and reliability of the size estimation methods employed in this project. The consistent performance across different datasets underlines the effectiveness of the ratio-based and linear regression approaches in achieving precise size estimations.

Phase	MSE (%) for Major Axis Estimation	MSE (%) for Minor Axis Estimation
Train Set	0.6	0.9
Demo	0.8	0.6

Table 7.2: Mean Squared Error for Tomato Size Estimation

7.3. Criteria 3: Flower Detection F1 Score

The third criterion aimed for an F1 score of at least 85% for flower detection. Unfortunately, the flower detection model could not be developed within the scope of this project. Due to the focus on tomato segmentation and size estimation, and given the limited available datasets, efforts towards developing a reliable flower detection model were insufficient. As a result, this criterion was not achieved. The lack of adequate resources and data for flower detection highlights the need for future work in this area to meet the required performance standards.

7.4. Conclusion

In conclusion, while the project successfully met the criteria for tomato size estimation, the goals for tomato detection and flower detection were partially or not met. The tomato detection model achieved commendable results given the data limitations, and the size estimation methods proved effective with low MSE values. However, the flower detection criterion remains unmet, indicating a clear area for future development. Overall, the project demonstrates significant progress towards the automation of agricultural monitoring systems, with room for further enhancements.

BIBLIOGRAPHY

[1] B. Zheng, G. Sun, Z. Meng, and R. Nan, "Vegetable size measurement based on stereo camera and keypoints detection," *Sensors*, vol. 22, no. 4, p. 1617, 2022.

[2] I. Nyalala, C. Okinda, L. Nyalala, *et al.*, "Tomato volume and mass estimation using computer vision and machine learning algorithms: Cherry tomato model," *Journal of Food Engineering*, vol. 263, pp. 288–298, 2019.

CV

Personal Information

Name:	[Emre Yilmaz]
Email:	[emreyilmaz0029@hotmail.com]
Phone:	[+905319346629]
LinkedIn:	[https://www.linkedin.com/in/emre9180/]

Education

- **[Bachelor]**, [Gebze Technical University], [2019 - 2024]
  - Computer Engineering
  - GPA: 3.44
- **[High School]**, [Trabzon Fen Lisesi], [2015 - 2019]
  -

Professional Experience

- **[Full-Stack Developer]**, [LLC Emotion], [01.23 - 11.23]
  - GoDeserve Mobile Application Development (Flutter, .NET)
  - Gunsel.ua Web Site Development
- **[Scholar / Software Engineer]**, [TUBITAK], [11.22 - 11.23]
  - Artificial Intelligence Supported Physiotherapy Project (Unity, C#)
- **[Intern / Software Engineer]**, [Eyedius Tech.], [07.22 - 10.22]
  - AI-powered security product end-to-end integration (Flask, Kivy, Python)
- **[Software Engineer]**, [GTU Spatium]
  - 2021 TEKNOFEST AI Competition (Ulaşımda Yapay Zeka) Finalist
  - 2022 TEKNOFEST AI Competition (Ulaşımda Yapay Zeka) Second Place

Skills

- Programming Languages: C, C#, C++, Python, Java, JavaScript
- Flutter, Unity, .NET, Angular
- Languages: English (Fluent), German (Elementary)

APPENDICES

Youtube Trailer Link

https://youtu.be/TPk2Q02BVpg

Github Link

https://github.com/emre9180/cse496-graduation-project
Note: Repo will be updated.