T.R.

GEBZE TECHNICAL UNIVERSITY

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

MONITORING PEPPER/FLOWER RATIO IN GREENHOUSES

EMRE YILMAZ

SUPERVISOR
PROF. YUSUF SINAN AKGÜL

GEBZE
2024

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

MONITORING PEPPER/FLOWER RATIO 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 24/01/2023 by the following jury.

JURY

Member
(Supervisor) : Prof. Yusuf Sinan AKGÜL

Member : Dr. Yakup GENÇ

ABSTRACT

The aim of this project is to develop an artificial intelligence model capable of continuously monitoring the number of peppers and flowers on a pepper plant. This will facilitate easy tracking of plant productivity and allow for early steps to be taken to increase yields.

A method has been developed using the knowledge that, under thermal imaging, the fruits of the plants show a distinct image difference compared to their surroundings. For this, a necessary dataset was initially created in collaboration with Gebze Technical University-affiliated greenhouse, using a phone camera and using a thermal camera. Subsequently, a series of modifications were made to this dataset for use in the training phase, and then the model training phase commenced.

In conclusion, particularly in the pepper detection method using thermal images, significant success was achieved. It was found that thermal technology could yield quantifiable results for use in smart agriculture applications.

Keywords: pepper, flower, detection, greenhouse, agriculture, artifical intelligence.

ÖZET

Bu proje, biber bitkisindeki biber ve çiçek sayısını sürekli olarak izleyebilen bir yapay zeka modeli geliştirmeyi amaçlamaktadır. Bu, bitki verimini kolayca takip etmeyi ve verimi artırmak için erken adımlar atmayı mümkün kılacaktır.

Bitkilerin meyveleri, termal görüntüleme altında, çevrelerine göre belirgin bir görüntü farkı sergilediği bilgisi kullanılarak bir yöntem geliştirilmiştir. Bu amaçla, öncelikle Gebze Teknik Üniversitesi ile işbirliği halinde olan seradan gerekli veri seti oluşturulmuşur. Bu veri seti termal kamera ve telefon kamerası kullanılarak elde edilmiştir. Daha sonra, bu veri setinde, eğitim aşamasında kullanmak için çeşitli değişiklikler yapılmış ve model eğitim aşaması başlatılmıştır.

Sonuç olarak, özellikle termal görüntüler kullanarak biber tespiti yönteminde önemli başarı elde edilmiştir. termal görüntü teknolojisinin, akıllı tarım uygulamalarında kullanılmak üzere anlamlı ve ölçülebilir sonuçlar verdiği sonucuna ulaşılmıştır.

Anahtar Kelimeler: biber, çiçek, 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, who assisted me throughout the project, and to Muhammet Sefa Cahyir and Doğukan Taştan, who tried to help me with transportation to the greenhouse. 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)
PCA	:	Principal Component Analysis (A Dimensionality Reduction Technique)
RGB	:	Red-Green-Blue (Color Model)

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. PROJECT DESCRIPTION

In commercial greenhouses, tracking the growth of plants, especially the ratio of peppers to flowers, is a key challenge for growers. This pepper/flower ratio is crucial because it shows how well the plants are producing fruit. Monitoring this ratio helps in maintaining the quality of the produce.

The goal of this project is to constantly monitor this ratio in large greenhouses. By using artifical intelligence and deep learning methods, we aim to provide farmers with detailed information about how their plants are doing. This means they can see not just how many peppers and flowers there are, but also how the ratio changes over time. Using this information, farmers can make better decisions about how to take care of their plants. They can find out what is working well and what needs to be improved. This way, they can make sure their plants are as healthy and productive as possible.

In short, this project is about using artifical intelligence to give greenhouse growers a clear picture of their plants' health and productivity. By keeping a close eye on the pepper/flower ratio, farmers can get a better harvest and improve the quality of their crops. This is important for making sure they can keep growing good produce in the future.

2. MAIN APPROACH TO PROBLEM

2.1. Hypothesis

While working on pepper detection, it became apparent that this problem was not as simple as initially thought. Since peppers and leaves are the same color, especially in insufficient light, distinguishing the pepper from the plant becomes challenging. Similarly, the detection of peppers is complicated when only a part of the pepper is visible, with the majority hidden behind leaves. You can see an example of this problem below Figure 2.1.

Figure 2.1: Pepper Indistinguishability Issue

Significant information was discovered in literature research on this topic and was decided to be utilized. In plants, fruits are warmer than other parts (branches, leaves) because certain metabolic events that occur during the fruit ripening process raise the temperature of the fruit.

- Respiration: Like other parts of the plant, fruits produce energy through respiration. During respiration, organic compounds such as glucose are burned with oxygen, releasing carbon dioxide, water, and energy. This process can increase the temperature of the fruit.

- Sugar production: As fruits ripen, they increase sugar production. Sugar is the main energy source of the fruit and also provides its taste. During sugar production, energy is released, which can further increase the temperature of the fruit.

- Color change: During the ripening of fruits, changes in color also occur. These color changes happen due to the production of pigments like carotenoids and anthocyanins. The production of these pigments also releases energy, which can further increase the temperature of the fruit.

Due to these factors, the fruit of a plant can be warmer than its other parts. This temperature difference supports the ripening of the fruit and the spread of its seeds. You can see the color difference between fruit and other part of plant from the image in the next page Figure 2.3.

2.2. Test of Hypothesis

A Flir ONE brand thermal camera was used to test this information in a greenhouse, and the following result was obtained. This result confirmed the accuracy of the examined articles, and it was concluded that it would be helpful to use this approach in the project. By examining the figures below containing photos taken by me in the greenhouse, you can see that under thermal imaging, the pepper fruits are significantly distinguishable from the rest of the plant. Given below Figure 2.3 and Figure 2.4 include same photo.

Figure 2.2: Thermal Image of My Pepper

Figure 2.3: RGB Image of My Pepper

3. DATASET

Although it was planned at the start of the project to obtain dataset support from external sources, this could not be realized due to the unavailability of suitable datasets. For the project, datasets were created using two different devices.

3.1. Android Device Usage

Initially, multiple visits were made to the relevant greenhouse (Figure 3.1) using an Android device camera, resulting in a dataset comprising about 1000 photos. However, since the approach to be used in the project was focused on infrared technology, these photos could not be used in the pepper detection model. In contrast, the photos obtained from the Android device were utilized for flower detection.

Figure 3.1: Example Image from Greenhouse

3.2. Thermal Camera Usage

A Flir One brand thermal camera was used for the thermal imaging. This camera is capable of capturing both thermal and RGB photos simultaneously.

After deciding to use thermal imaging technology for collecting plant photos, an initial visit to the relevant greenhouse was made at the end of November, resulting in a dataset of about 100 photos. When the greenhouse was revisited the following week, it was observed that the temperature had dropped to 15 degrees Celsius, which significantly deteriorated the quality of the thermal images. It was decided to wait until the weather warmed up to the desired temperature. In mid-December, another visit to the greenhouse was made, and nearly 600 thermal images were obtained

Figure 3.2: Thermal Image in High Temperature

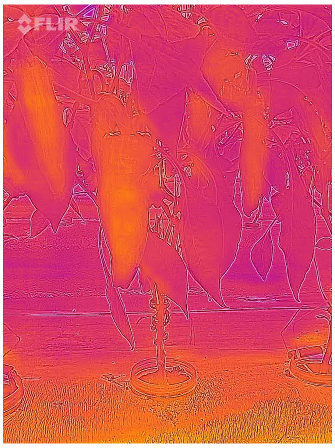

Figure 3.3: Thermal Image in Low Temperature

As seen in Figure 3.2, in warm weather (26 degrees Celsius), it can be clearly seen the distinct separation between peppers and leaves. However, when we look at Figure 3.3, at a lower temperature (15 degrees Celsius), this distinction is not clearly made, indicating that it is not suitable for use in the model. The greenhouse was visited three times to capture thermal photographs, and a necessary dataset comprising approximately 600 photos was created.

3.3. Labelling Process

600 images obtained using the Flir One thermal camera and 300 images from the Android device, after selection to include only those containing flowers, have been labeled using the Roboflow web application for use in the model.

4. DATA PREPARATION

Figure 4.1: Example of a RGB Image

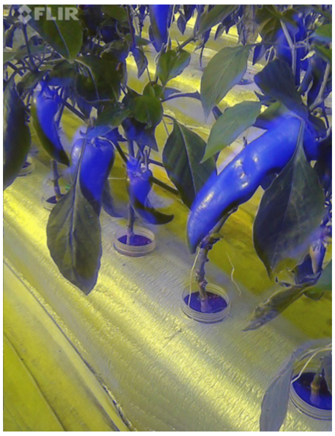

Figure 4.2: Example of a Blue Channel Modified Image

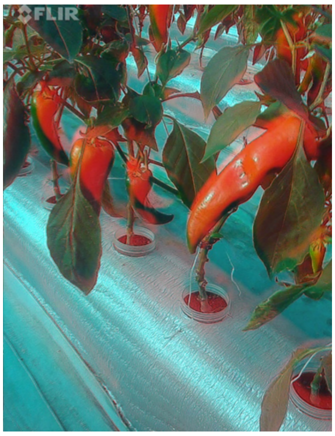

Figure 4.3: Example of a Red Channel Modified Image

When examining image examples, distinguishing peppers in the RGB photo appears to be extremely difficult. However, when looking at images where the channels have been replaced with thermal imaging, the distinction of peppers can be made quite easily.

The obtained thermal images have been attempted to be utilized in various ways. Since we had access to both RGB and thermal images of the captured photos, a series of manipulated new datasets have been created. Each of these datasets embodies different approaches.

Initially, the obtained image's thermal version was converted into a grayscale image, making it one-dimensional (Figure 4.4).

Figure 4.4: Grayscale of Thermal Image

4.1. Alpha Channel Addition

Then, this one-dimensional image was added as the fourth channel (alpha channel) to the RGB version of the picture (Figure 4.5). The following type of image was obtained (Figure 4.6). As expected, it was observed that the relatively cooler areas in the photo were considered as the background. Upon examining this image, it was anticipated that the fourth channel of the obtained image would be extremely meaningful and would be significant for the model to be trained.

Figure 4.5: Original Version Of Alpha Channel Added Image

Figure 4.6: Alpha Channel Added Pepper Image

4.2. PCA Method On Four Channel Image

Since the model trained with the dataset containing 4-channel images resulted in failure (the results will be discussed in detail in the next chapter), the PCA (Principal Component Analysis) method was applied to reduce the dimension count to 3, and a dataset containing images like the ones below has been created (Figure 4.7).

Figure 4.7: PCA Applied Pepper Image

The obtained thermal image was again converted to a one-dimensional grayscale image and then integrated into the fourth channel (alpha channel) of the RGB version. Then, PCA is applied and dimensionality is reducted to three.

4.3. Replacing Red Channel Of Pepper Image

The red channel of the RGB photo has been replaced with the one-dimensional thermal image saved in grayscale, resulting in a dataset consisting of images like the ones below (Figure 4.9).

Figure 4.8: Original Version Of Pepper Image Whose Red Channel Replaced

Figure 4.9: Red Channel Replaced Pepper Image

4.4. Replacing Blue Channel Of Pepper Image

The blue channel of the RGB photo has been replaced with the one-dimensional thermal image saved in grayscale, resulting in a dataset consisting of images like the ones below (Figure 4.11).

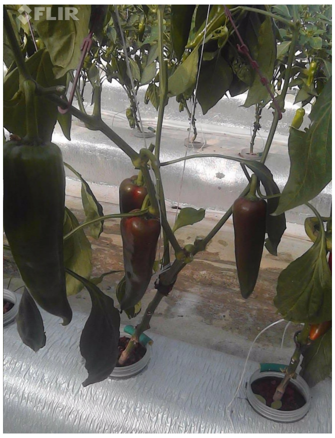

Figure 4.10: Original Version Of Pepper Image Whose Blue Channel Replaced

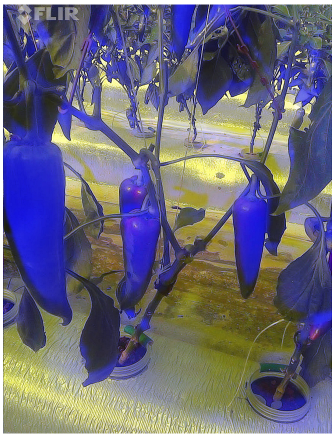

Figure 4.11: Blue Channel Replaced Pepper Image

4.5. Flower Dataset

Photos taken with an Android device were carefully selected to create a flower dataset consisting of approximately 300 photos. Due to seasonal conditions and the early death of bees in our greenhouse, the flower dataset remained limited.

5. MODEL BUILDING

YOLOv8_x model was trained with four different datasets and the original RGB dataset that are mentioned in the 'Data Preparation Chapter' (Chapter 4). During the training of these models, the train, valid, and test splits of each preprocessed dataset were completely identical, and the applied augmentations were exactly the same excluding PCA applied dataset. In other words, different versions of a photo were used in training with YOLO. This approach allows us to make a correct and objective comparison of the models' successes.

5.1. Training Of RGB Original Dataset

YOLOv8_x model was trained using the original RGB dataset. The augmentations applied to the training set are as follows:

- HorizontalFlip
- Rotate
- Random Crop
- Random Brightness Contrast

The results after 50 epochs of training are as follows:

Table 5.1: Result of YOLOv8 Model (RGB Dataset

Model	Confidence	Precision	Recall	F1	Ground Truth Total	Found Total	Ratio
rgb-best	0.3	0.93	0.6	0.729	342	219	64.03

When examining the results, we observe that the precision value is 93% but the recall value is only 60%. We see that only 64% of the 363 peppers in the dataset were accurately detected. It is evident that this model is inconsistent and will produce results that are not reflective of reality in practice.

Figure 5.1: Some examples of YOLOv8 Model with RGB Dataset

5.2. Training Of Red Channel Replaced Dataset

YOLOv8_x model was trained using the described dataset in chapter 4.3. Red channel of images is replaced with one dimensional thermal image. Model is trained with this modified dataset. Same augmentation that is applied to original RGB dataset is applied to this dataset.

The results after 50 epochs of training are as follows:

Table 5.2: Result of YOLOv8 Model (Red Channel Replaced Dataset)

Model	Confidence	Precision	Recall	F1	Ground Truth Total	Found Total	Ratio
red-best	0.3	0.91	0.69	0.784	342	259	75.73

When examining the results, we observe that the precision value is 91% but the recall value is only 69%. We see that only 75% of the 363 peppers in the dataset were accurately detected. It is evident that this model is inconsistent and will produce results that are not reflective of reality in practice.

Nevertheless, we can prove that the model trained with the dataset where the red channel is replaced by thermal imaging is more successful than the model trained with the original RGB dataset.

Figure 5.2: Some examples Of YOLOv8 Model With Red Channel Replaced Dataset

5.3. Training Of Blue Channel Replaced Dataset

YOLOv8_x model was trained using the described dataset in chapter 4.4. Blue channel of images is replaced with one dimensional grayscale thermal image. Model is trained with this modified dataset. Same augmentation that is applied to original RGB dataset is applied to this dataset.

The results after 50 epochs of training are as follows:

Table 5.3: Result of YOLOv8 Model (Blue channel replaced dataset)

Model	Confidence	Precision	Recall	F1	Ground Truth Total	Found Total	Ratio
blue-best	0.3	0.94	0.68	0.789	342	244	71.34

When examining the results, we observe that the precision value is 94% but the recall value is only 68%. We see that only 71% of the 363 peppers in the dataset were accurately detected. It is evident that this model is inconsistent and will produce results that are not reflective of reality in practice.

Nevertheless, again, we can prove that the model trained with the dataset where the blue channel is replaced by thermal imaging is more successful than the model trained with the original RGB dataset.

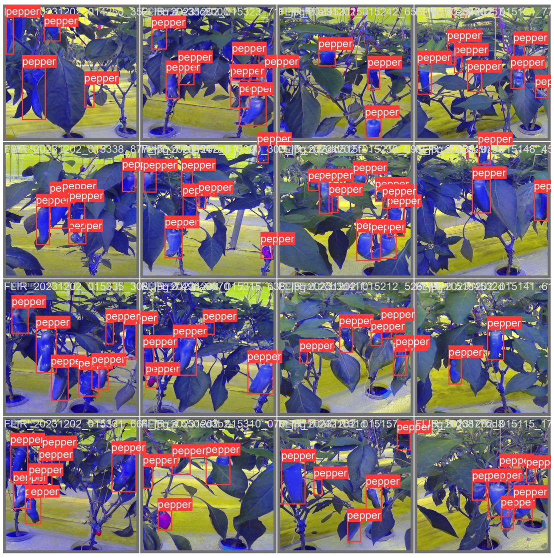

Figure 5.3: Some examples Of YOLOv8 Model With Blue Channel Replaced Dataset

5.4. Training Of Alpha Channel Added Dataset

5.4.1. Modify YOLO to Accept 4-Channel Input

To train the dataset containing a four-channel image with an alpha channel, it is first necessary for the model to accept four-channel input. Therefore, some modifications were made to the model:

- ”ch: 4” line is added to .yaml file
- In autobackend.py file, warmup function’s parameters are edited. Channel size changed as 4.
- In validator.py, while calling warmup function of autobackend.py, channel parameter sent as 4.
- Augmentation flag changed as false in augment.py. Also, augmentations are commented out in this file.
- In every file in ultralytics/data, It is changed cv2.imread(’photo.png’) to cv2.imread(’photo.png’, cv2.IMREAD_UNCHANGED)

5.4.2. Training

YOLOv8_x model was trained using the described dataset in chapter 4.1. One dimensional grayscale thermal image is added as fourth channel (alpha channel) of

RGB image. Then model is trained using this modified dataset. Same augmentation that is applied to original RGB dataset is applied to this dataset. But, YOLO's default augmentation is deactivated. The results after 50 epochs of training are as follows:

Table 5.4: Result of Model (Alpha Channel Added Dataset)

True Positive	False Positive	False Negative
635	178	532

When examining the results, we observe that the precision value is 78% and the recall value is only 59%. It is evident that this model is inconsistent and will produce poor results.

Although we observed very meaningful images in the four-channel photos, the exact reason behind the model's poor performance could not be fully understood. However, it is highly likely that many necessary changes within the YOLO architecture were overlooked. Since this problem could not be resolved, this method was abandoned early on.

5.5. Training Of PCA Applied Dataset

YOLOv8_x model was trained using the described dataset in chapter 4.3. One dimensional grayscale thermal image is added as fourth channel (alpha channel) of RGB image. After that, PCA is applied on this dataset and model is trained with this 3-channel dataset. Same augmentation that is applied to original RGB dataset is applied to this dataset.

The results after 50 epochs of train are as follows:

Table 5.5: Result of YOLOv8 Model (PCA Applied Dataset)

Model	Confidence	Precision	Recall	F1	Ground Truth Total	Found Total	Ratio
pca-best	0.3	0.81	0.74	0.77	255	230	90.19

When examining the results, we observe that the precision value is 81% but the recall value is only 74%. We see that only 90% of the 363 peppers in the dataset were accurately detected.

Looking at these results, we see that the model trained with the PCA-applied dataset has almost the same F1 score as the models trained with datasets where the blue and red channels are replaced with thermal images. However, the notable point is that the precision and recall values are balanced in this model. Consequently, this

Figure 5.4: An Example of Detection on PCA Applied Dataset

model accurately identified the number of peppers in the test dataset up to a 90% rate. Although this prediction seems like a good result, we cannot consider it consistently reliable. An F1 score of 75% could lead to undesirable outcomes in practice. Therefore, a different method was tried to improve success.

5.6. Flower Detection Model Training

Unfortunately, no special approach could be applied for the flower model. With the available dataset, YOLOv5 and YOLOv8 models were trained, and the results were obtained.

At the end of the YOLOv5_x model training, a true positive rate of 68% and a false negative rate of 32% were achieved. The evaluation with the test set at the end of the training showed a Precision value of 82% and a Recall value of 70%. And here is the YOLOv8 results:

Table 5.6: Result of YOLOv8 Model (Flower Detection

Model	Confidence	Precision	Recall	F1	Ground Truth Total	Found Total	Ratio
flower	0.3	0.85	0.66	0.743	83	64	77.10

In the case of the YOLOv8_x model training, the evaluation with the test set resulted in a Precision value of 85% and a Recall value of 69%. It accurately detected 77% of the total 83 flowers. Both models can be considered unsuccessful.

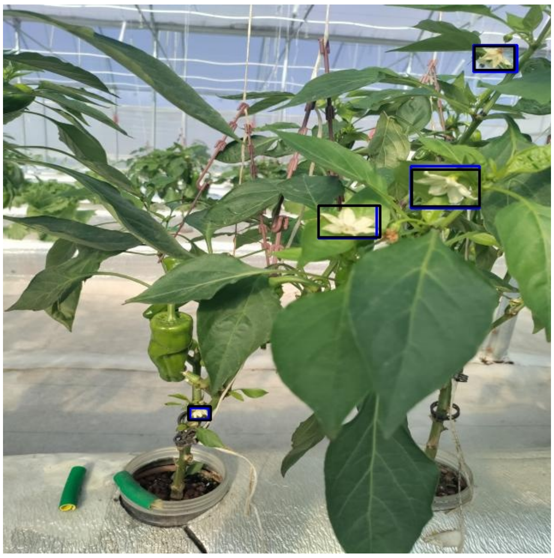

Figure 5.5: An Example of Flower Detection

6. ENSEMBLE MODEL

6.1. Implementation

After training five different pepper detection models, particular attention was focused on three models: the model trained with the dataset where the red channel was replaced by thermal imaging, the model trained with the dataset where the blue channel was replaced by thermal imaging, and the model trained with the original RGB dataset. The results of these models were thoroughly analyzed, and conclusions were drawn. It was observed that all three models had high precision but low recall, meaning they were successful in not falsely identifying non-pepper points as peppers but tended to miss objects labeled as peppers. When examining the test results of these models, it was noticed that a pepper missed by one model could be detected by another model.

Based on this information, the idea of using these three models together seemed logical and likely to yield good results. It was anticipated that this would increase the recall value. An ensemble algorithm was implemented.

Here how it is done:

1. **Load Labels and Image Paths for Each Model:** This step involves loading the necessary labels and paths to the images for each model.

2. **Load the Models:** In this step, all the models that will be used for image analysis are loaded.

3. **Resize Images for Processing:** Each image is loaded and resized to 640x640 pixels, preparing them for processing.

4. **Process Images Through All Models:** The resized images are then fed into all the loaded models for analysis.

5. **Extract Prediction Boxes:** The output from each model is post-processed to retrieve all the predicted bounding boxes as a list. The postprocessing step was modified for this purpose.

6. **Filter Out Low Confidence Predictions:** Any predicted bounding boxes with a confidence score below 0.3 are discarded.

7. **Store Prediction Boxes in a Dictionary:** The prediction boxes from all models are stored in a dictionary. This dictionary is essential for understanding and analyzing which model detected which specific object in the images.

8. **Merge Overlapping Boxes Based on Color and IoU:** All bounding boxes are compared and merged based on their colors and Intersection over Union (IoU) of 30%. For instance, if there is a red box and a blue box overlapping in an area, they are merged into a single purple box. Boxes overlapping more than 30% are combined into one box, covering both areas and blending their colors.

9. **Compare Results with Original Labels and Calculate Precision-Recall:** The final results from the models are compared with the original labels of the images. Precision and recall metrics are then calculated based on this comparison.

10. **Overlay Results on Original Images:** Lastly, the final results, including the prediction boxes and labels, are drawn onto the original images for visual representation and analysis.

6.2. Results

Table 6.1: Ensemble Model Results

Model	Confidence	Precision	Recall	F1	Ground Truth Total	Found Total	Ratio
ensemble-best	0.3	0.88	0.85	0.864	342	329	96.19
ensemble-best	0.4	0.91	0.80	0.851	342	303	88.59
ensemble-best	0.5	0.93	0.77	0.842	342	284	83.04
ensemble-best	0.6	0.94	0.73	0.821	342	264	77.19

When examining the results, the best outcome was achieved with a confidence value of 0.3. The precision value is 88%, and the recall value is 85%. The obtained F1 score is 86%. Of the total 342 peppers, 329, which is 96%, were accurately detected. It is evident that the ensemble model has improved upon the other models and yielded a good result. You can see detection examples (Figure 6.2, Figure 6.3, Figure 6.4).

- **White boxes:** Peppers detected by all three models.
- **Green boxes:** Peppers detected only by the model trained with the original RGB dataset.
- **Red boxes:** Peppers detected only by the model trained with the dataset where the red channel was replaced by thermal imaging.
- **Blue boxes:** Peppers detected only by the model trained with the dataset where the blue channel was replaced by thermal imaging.

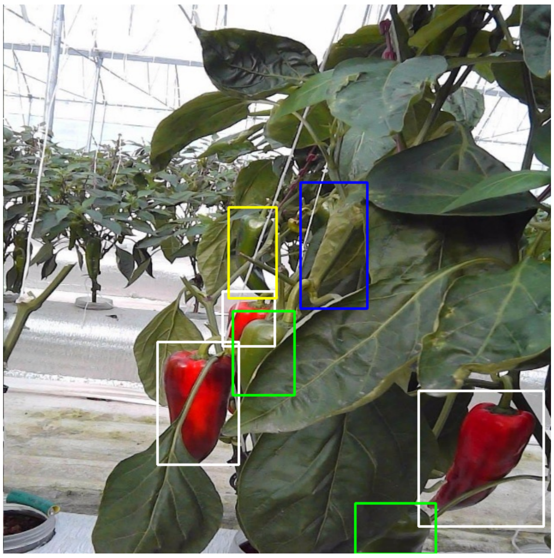

Figure 6.1: An Example of Ensemble Model Result

Figure 6.2: An Example of Ensemble Model Result

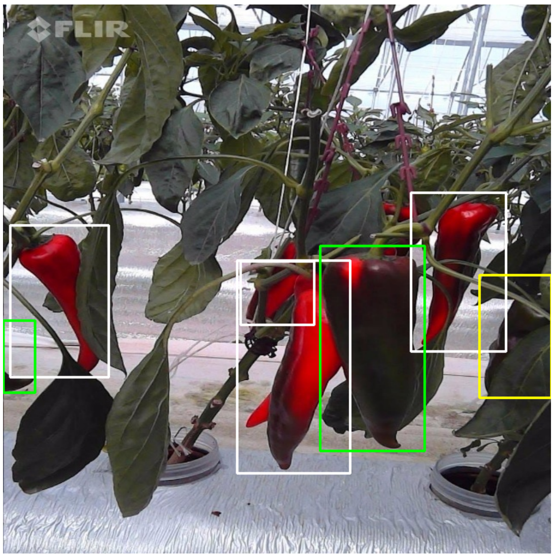

Figure 6.3: An Example of Ensemble Model Result

- **Purple boxes:** Peppers detected by the models trained with datasets where the blue channel was replaced with thermal imaging and the red channel was replaced with thermal imaging.
- **Yellow boxes:** Peppers detected both by the models trained with the red channel replaced by thermal imaging and the model trained with the original RGB dataset.
- **Cyan boxes:** Peppers detected both by the model trained the dataset where the blue channel replaced by thermal imaging and the model trained with the original RGB dataset.

When looking at the results, it can be seen the validity of our hypothesis that led us to implement an ensemble model. While some peppers are accurately detected by only one model, others are correctly identified by two or all three models. Using this method, we were able to maintain the precision value at 88% while increasing the recall value to 85%

7. CONCLUSION

7.1. Problems

7.1.1. Temperature of Greenhouse

The greenhouse temperature significantly affected the success of pepper detection. While peppers could be clearly distinguished from other parts of the plant in thermal images taken in relatively warm weather, this distinction becomes more challenging as the greenhouse temperature drops. When the temperature falls to around 15-16 degrees Celsius, this distinction starts to become meaningless. The dataset was created over three visits to the greenhouse, with the first one occurring in warm weather, yielding data of the desired quality. No data could be collected during another visit due to cold weather, prompting a return the following week. During this last visit, the bulk of the dataset was collected. However, as the temperature during this last visit was not as high as during the first, there was not as significant a temperature difference between the peppers and other parts of the plant as observed during the first visit. Still, meaningful images were obtained and used in model training. It is anticipated that if the dataset could have been created in warmer weather, the model's success would have been much better.

7.1.2. Quantity Issue in Dataset

A total of approximately 300 photos for the flower dataset and approximately 600 photos for the pepper dataset were obtained. Particularly, the numerical weakness of the flower dataset has significantly affected the model's success. Both datasets need to be enriched in terms of quantity to improve model success. It should also be emphasized that these datasets were labeled by me.

7.2. Success Criteria of Pepper Detection

At the beginning of the project, literature reviews indicated high success rates in pepper detection [1], [2], [3]. Therefore, aiming to surpass these successes, a goal of 95% F1 score was set. However, attempting to solve the problem with a novel approach not found in the literature, an F1 score of 86% was achieved. It is anticipated that if the dataset is enriched with new photos under more favorable seasonal conditions, this success rate could increase. Although the pepper detection model's performance fell below the initially stated success criterion, a consistent and improvable result was obtained.

7.3. Success Criteria of Flower Detection

As mentioned, the quantity of flower photos obtained from the greenhouse is low, and a beneficial approach like the one developed for pepper detection could not be implemented. While the F1 score achieved with the YOLOv8 model is 77%, the targeted success criterion is 80%. This initial success criterion was established as a result of a literature review [4], [5], [6], [7]. It must be acknowledged that the flower detection model development process was unsuccessful.

7.4. Success Criteria of Detection Time

When the developed program was tested, it was capable of predicting both peppers and flowers in 1 photo every 5.99 seconds. The success criterion for this value was initially set at 5 seconds. The model is run on CPU Intel Core i7 9750 of my computer.

```
Results saved to runs\detect\predict152
'FLIR_20231202_125958_821_jpg'
5.990800367702137
```

Figure 7.1: Run time of models

BIBLIOGRAPHY

[1] X. Li, J. Pan, F. Xie, *et al.*, "Fast and accurate green pepper detection in complex backgrounds via an improved yolov4-tiny model," *Computers and Electronics in Agriculture*, vol. 191, p. 106 503, 2021.

[2] Y. Nan, H. Zhang, Y. Zeng, J. Zheng, and Y. Ge, "Faster and accurate green pepper detection using nsga-ii-based pruned yolov5l in the field environment," *Computers and Electronics in Agriculture*, vol. 205, p. 107 563, 2023.

[3] P. Cong, S. Li, J. Zhou, K. Lv, and H. Feng, "Research on instance segmentation algorithm of greenhouse sweet pepper detection based on improved mask rcnn," *Agronomy*, vol. 13, no. 1, p. 196, 2023.

[4] X. Mu, L. He, P. Heinemann, J. Schupp, and M. Karkee, "Mask r-cnn based apple flower detection and king flower identification for precision pollination," *Smart Agricultural Technology*, vol. 4, p. 100 151, 2023.

[5] Y. Zhang, S. He, S. Wa, Z. Zong, and Y. Liu, "Using generative module and pruning inference for the fast and accurate detection of apple flower in natural environments," *Information*, vol. 12, no. 12, p. 495, 2021.

[6] X. A. Wang, J. Tang, and M. Whitty, "Side-view apple flower mapping using edge-based fully convolutional networks for variable rate chemical thinning," *Computers and Electronics in Agriculture*, vol. 178, p. 105 673, 2020.

[7] K. Sun, X. Wang, S. Liu, and C. Liu, "Apple, peach, and pear flower detection using semantic segmentation network and shape constraint level set," *Computers and Electronics in Agriculture*, vol. 185, p. 106 150, 2021.

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: 3.40
- **[High School]**, [Trabzon Fen Lisesi], [2015 - 2019]
  -

Professional Experience

- **Scholar**, [EUREKA Eurostars], [02.24 -]
  -
- **[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/S5NXvXMDrpA

Github Link

https://github.com/emre9180/cse495-graduation-project
Note: Repo will be updated.