big data and cognitive computing

Article

Strawberry Ripeness Detection Using Deep Learning Models z[1],

Zhiyuan Mi, Wei Qi Yan

School of Engineering, Computer and Mathematical Sciences, Auckland University of Technology, Auckland 1010 New Zealand

* Correspondence:gxf2674@autuni.ac.nz, weiqi.yan@aut.ac.nz

Abstract: In agriculture, timely and accurate assessment of fruit ripeness is crucial to optimize harvest planning and reduce waste. In this article, we explore the integration of two cutting-edge deep learning models, YOLOv9 and Swin Transformer, to develop a complex model for detecting strawberry ripeness. Trained and tested on a specially curated dataset, our model achieves a mean precision (mAP) 87.3% by using the metric intersection over union (IoU) at threshold 0.5. This performance of this model outperforms the model by using YOLOv9 alone, which achieved a mAP 86.1%. Our model also demonstrated the improved Precision and Recall, with Precision rising to 85.3% and Recall rising to 84.0%, reflecting its ability to accurately and consistently detect different stages of strawberry ripeness.

Keywords: Transformer, YOLOv9, Swin Transformer, Deep Learning, Computer Vision

Citation: To be added by editorial staff during production.

Academic Editor: Firstname Lastname

Received: date
Revised: date
Accepted: date
Published: date

1. Introduction

With global population growth, climate change, and the need for sustainable practices, the agricultural sector is facing unprecedented challenges which is in dire need of innovative solutions. Strawberries are highly perishable, sensitive to environmental conditions, and require precise harvest timing to ensure optimal quality and yield.

Fortunately, the use of AI and deep learning in agriculture is becoming more common. YOLOv9 is well-known for its real-time target detection capabilities, while Swin Transformer excels at processing image data, especially for tasks requiring detailed visual understanding [1, 2]. By integrating these two techniques for strawberry ripeness detection, we aim to significantly improve the accuracy and efficiency of detection, reduce reliance on manual labor, minimize human error, and make more accurate and timely harvest decisions.

The main objective of this article is how to effectively integrate and apply YOLOv9 and Swin Transformer technologies for strawberry ripeness detection, so the main research questions of this article are: How can we effectively combine YOLOv9 and Swin Transformer for strawberry ripeness detection? To solve this research question, we split it into the following questions:

- How can the integrated YOLOv9 and Swin Transformer model be trained, validated?
- What role does the Swin Transformer play in enhancing the accuracy of ripeness detection, and how does it complement the object detection capabilities of YOLOv9?
- How do we evaluate the model and prove that our improvements to the model are effective?

The core concept of this project is to utilize the complementary advantages of YOLOv9 and Swin Transformer to develop a robust, efficient, and highly accurate strawberry ripeness detection system. We train the dataset to get the best results and evaluate the model thoroughly.

The article is structured as follows: Section 2 reviews some of the related works. Section 3 provides an in-depth discussion of the models and methods used in this research. Section 4 presents the results of the research. Section 5 concludes the article.

2. Related Work

Deep learning and computer vision has emerged as powerful tools in the field of agriculture, offering innovative solutions to various challenges faced in the industry. Papers [3, 4, 5, 6, 7] present the status and applications of deep learning and computer vision in agriculture.

The effectiveness of using pretrained deep neural networks (DNN) on agricultural datasets was explored to improve weed identification accuracy in precision agriculture [8].

The study by Sharma et al. [9] proposed the exceptional efficacy of Convolutional Neural Networks (CNNs) in analyzing plant disease images. The Faster R-CNN model they devised achieved a detection rate of 99.39% for chili plants, highlighting the potential of deep learning models to revolutionize agricultural disease management.

A method [10] was proposed to automatically detect unripe tomatoes by using Faster Region-based Convolutional Neural Network (Faster R-CNN) and ResNet-101 model to learn from the COCO dataset through transfer learning. The method performed well on immature and occluded tomatoes that are difficult to detect through traditional image analysis methods.

A CNN model [11] was introduced for automated, lossless classification of mulberry maturity. The method improved the accuracy and efficiency of the sorting process by automatically classifying fruit into different ripeness categories based on visual cues.

Pardede et al. used transfer learning of VGG-16 models for fruit ripeness detection. Their study [12] highlights the effectiveness of deep learning relative to traditional machine learning for this task, with a particular emphasis on the important role of regularization techniques in improving model performance.

A powerful CNN model was proposed by Momeny et al. [13] to detect citrus black spot disease and evaluate fruit ripeness through deep learning. One of their key innovations is the use of a learning augmentation strategy that generates new data from noisy and recovered images to enhance model training. Momeny et al. utilized Bayesian algorithm-optimized noise parameters to create noisy images and then took use of convolutional autoencoders to restore these images, effectively augmenting the training data.

In conclusion, it is very feasible to use deep learning for agricultural aspects especially for fruit ripening detection. The potential of deep learning for fruit ripeness detection represents an important step forward in agricultural technology, with the potential to not only reduce labor costs, but also improve efficiency and reduce waste.

3. Materials and Methods

In this section, we outline the research methodology for the development and evaluation of a deep learning-based system for dynamic detection of strawberry ripeness through video analysis. The integration of YOLOv9 and Swin Transformer forms the core of our approach, leveraging their capabilities to achieve real-time, accurate ripeness detection.

3.1. Research Designn

3.1.1. Overall of the Proposed Model

In this article, we propose a strawberry ripeness detection method based on YOLOv9 network and Swin Transformer. The method can automatically detect the position of strawberries from a video with multiple frames and track movement trajectory to mark the strawberries and predict their ripeness. This method will be a great convenience for growing and picking strawberries.

We trained a hybrid model by combining YOLOv9 and Swin Transformer, which enhances the ability of this model to generalize and rely on modeling capabilities at a distance, resulting in better overall performance.

The overall structure of strawberry ripeness detection model is shown in Figure 1. Firstly, YOLOv9 model is trained by using the pre-prepared dataset. This model is improved by combining Swin Transformer, which can better extract the target feature information. Then, the video was processed by using a fusion network of YOLOv9 and Swin Transformer to detect strawberry ripeness with high accuracy. This model will classify the strawberries as the classes "Unripe", "Half-ripe", and "Ripe", outputs detection frames and feature vectors for each frame of the given video.

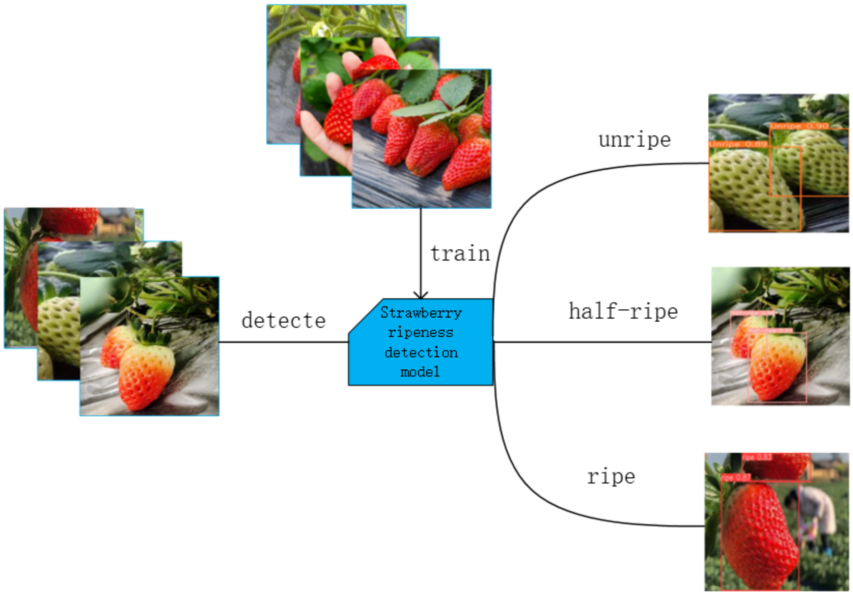

Figure 1. Overall structure of the strawberry ripeness detection model.

3.1.2. Research Design of YOLOv9 Model

YOLOv9 introduces Programmable Gradient Information (PGI), which preserves important data throughout the depth of the proposed network, ensuring more reliable gradient generation and thus improving model convergence and performance. Meanwhile, YOLOv9 designs a new lightweight network structure based on gradient path planning: generalized efficient layer aggregation network (GELAN). By using only conventional convolution, GELAN achieves higher parameter utilization than deeply differentiable convolutional designs based on state-of-the-art techniques, while demonstrating the great advantages of being lightweight, fast, and accurate [14].

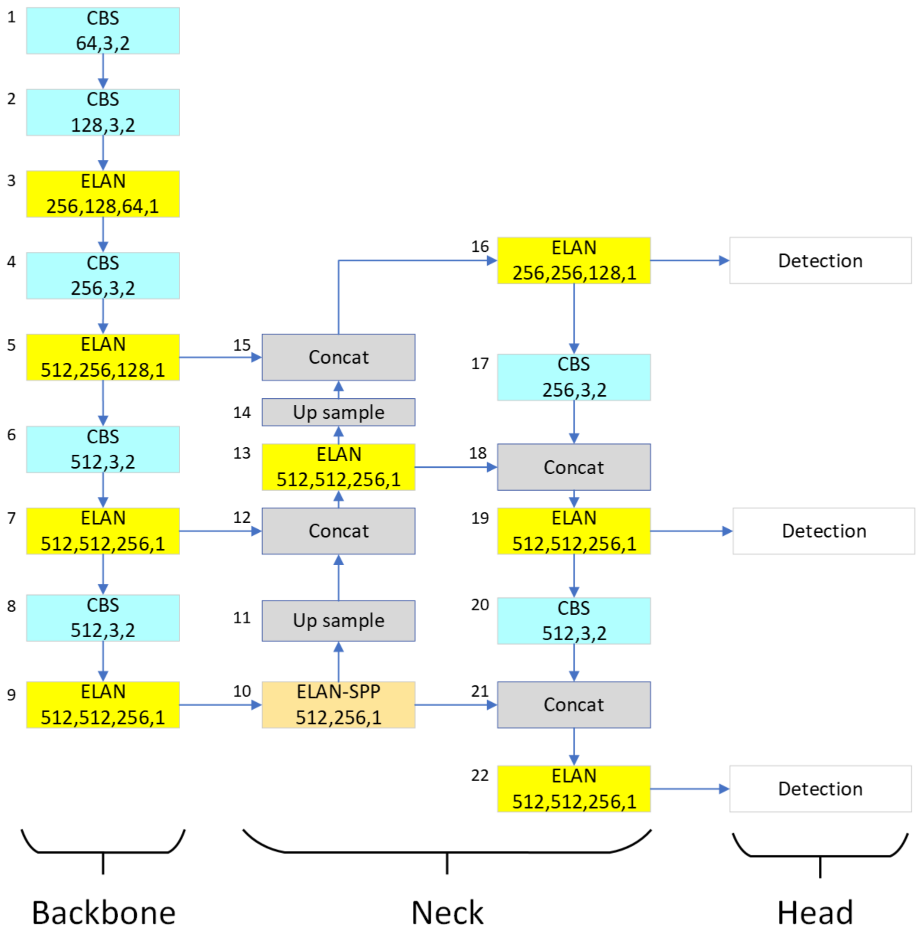

Figure 2. YOLOv9 structure

Figure 2 illustrates the convolutional neural network architecture of YOLOv9 model. This model is divided into three main parts: Backbone, Neck, and Head. The Backbone is the main feature extraction part of this model. It consists of multiple convolutional layers that are responsible for extracting useful features from the input image. Backbone consists of multiple layers that progressively reduce the spatial dimensions and increase the number of channels through different depth and step configurations, which helps in capturing features at different levels of abstraction of the image.

Neck is the part that connects Backbone and Head, which serves to perform feature fusion and realignment for object recognition. This part consists of Up sample and Concat operations, which combine high level, smaller feature maps with low level, larger feature maps, thus preserving spatial information at different scales. This helps to detect objects at different scales of the image. Head is the last part of the model and is responsible for object detection based on the features coming from Backbone and Neck.

3.1.3. Research Design of Swin Transformer

Swin Transformer (Shifted Window Transformer) is a computer vision model based on Transformer. Swin Transformer overcomes the problems of computational inefficiency and difficulty in handling high-resolution images of traditional Transformer models [15].

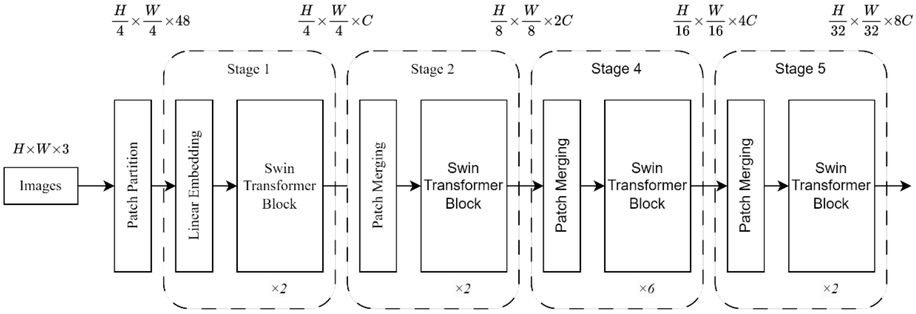

Figure 3. Swin Transformer structure

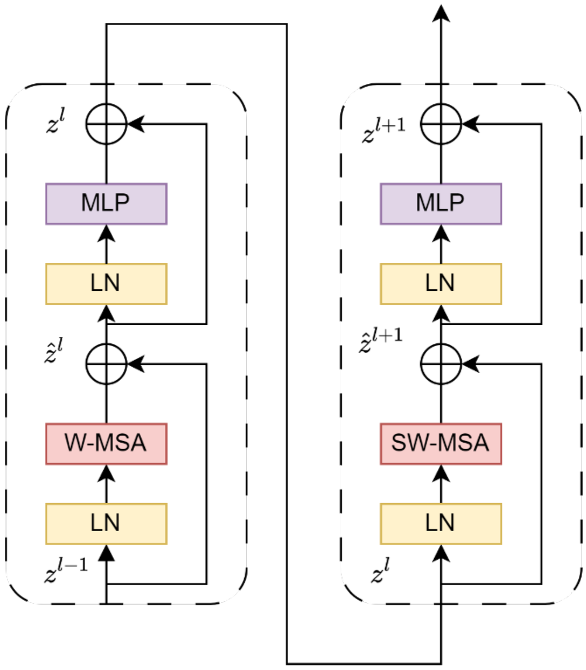

Figure 4. Swin Transformer Blocks

Figure 3 shows the structure of the Swin Transformer. At the beginning, the image is divided into multiple small blocks, each small block is usually a small square. These patches are flattened into vectors and passed into the model for processing. The model adopts a layered design and consists of four stages, each stage will reduce the resolution of the input feature map. The four stages build varying-sized feature maps. The first three stages go via a Patch Merging layer for down sampling, while the first stage goes through a Linear Embedding layer. Swin Transformer Blocks are piled one after the other on each level.

Transformer Block has two structures, as shown in Figure 4. One structure makes use of the W-MSA structure, while the other uses the SW-MSA structure. Furthermore, a W-MSA structure and a SW-MSA structure are employed in pairs when utilizing these two structures.

In this article, we built the strawberry ripeness detection model based on YOLOv9. We propose a method to replace the backbone network in YOLOv9 with Swin Transformer. This hybrid model combines the fast and efficient detection capabilities of YOLOv9 with the powerful and flexible feature representation of Swin Transformer, designed to enhance the system's ability to accurately identify and classify strawberry ripeness from video input.

In the hybrid model, Swin Transformer acts as a powerful feature extractor by capturing the details and variations of strawberry appearance. These details and changes mark different stages of maturity. Swin Transformer ensures that global and local features are effectively captured and used for prediction. This is particularly useful for detecting strawberries under varying lighting, occlusion, and background complexity conditions.

3.2. Data Preparation

In this article, we collected various strawberry images and videos datasets to ensure the quality and accuracy of our models. For the data set, we used pre-processing techniques such as image cropping, resizing, and labeling to ensure that the data set is processed into the form required by the model.

We downloaded test videos of strawberry plantations from the internet and performed images extraction on the videos. We are use of a Python script to assist us in extracting images. This script allows us to split the video into images at set intervals. This is helpful for reducing data redundancy. Additionally, we downloaded strawberry images from the Internet to increase robustness.

As shown in Figure 5, we collected a total of 722 strawberry images. In addition, we also downloaded the strawberry image dataset, open sourced by the StrawDI team and selected images that met our requirements [16]. Finally, we collected a total of 2,000 strawberry images from different regions and under different lighting and weather conditions, which helped to enhance model diversity.

Figure 5. Samples of our dataset

In this article, we are use of EISeg to label the collected strawberry images. Figure 6 illustrates the results after labeling. EISeg (Efficient and Interactive Segmentation) is an efficient interactive image segmentation tool, mainly used in geospatial analysis, remote sensing image processing, medical image processing and other fields. EISeg provides a method to achieve precise segmentation with minimal user interaction, greatly improving the efficiency and accuracy of image segmentation [17].

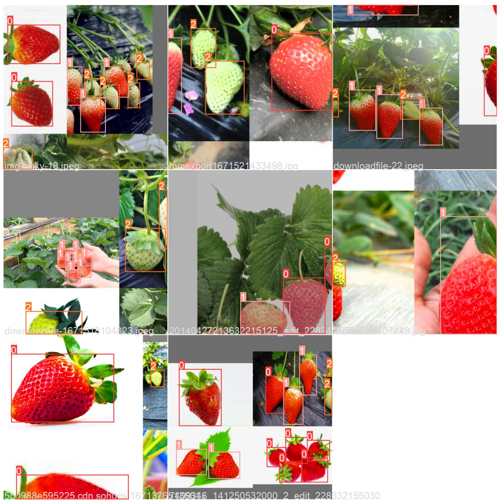

Figure 6. The example of the results after labeling

Data splitting is a fundamental method in machine learning for training models and evaluating model performance. It involves dividing the dataset into separate subsets to provide an honest assessment of the performance of our proposed models on unseen data. The three main subsets commonly used are: Training dataset, validation dataset and test dataset. The training set is the largest part of the dataset used to train the model. The validation set is employed to provide an unbiased assessment of the model which fits on the training data set when adjusting the model's hyperparameters. After the model has been trained and validated, the test set is used for an unbiased evaluation of the final model. The correct data splitting can avoid model overfitting problems and significantly improve the validity and reliability of model evaluation.

In this article, we also split the data. We take use of 80% of the dataset for training, 10% for validation, and 10% for testing. Figure 7 clearly illustrates the dataset splitting.

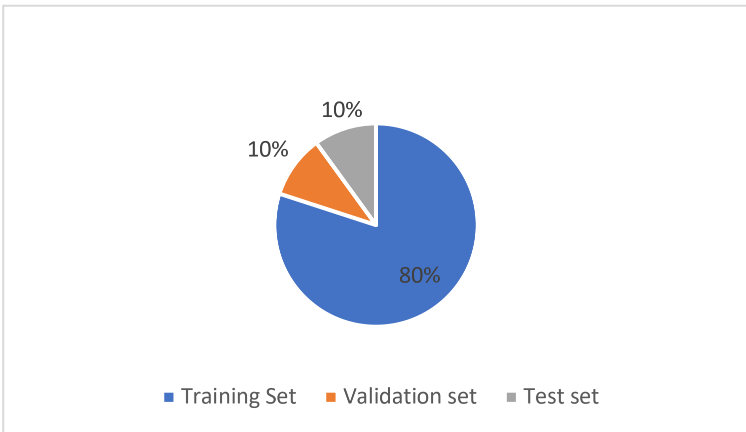

Figure 7. Data splitting pie chart

3.3. Evaluation Methods

Our evaluation is a critical step for computer vision models, which helps measure model performance and guide future improvements. In deep learning, all evaluation methods are based on confusion matrix. Table 1 shows the confusion matrix. In Table 1, True Positive (TP) means that the true category of the sample is a positive sample, and the model predicts a positive sample, therefore the prediction is correct. True Negative (TN) means that the true category of the sample is a negative example, and the model predicts it as a negative example, therefore the prediction is correct. False Positive (FP) is a sample whose true category is a negative sample, but the model predicts it as a positive sample, and therefore the prediction is wrong. False Negative (FN) means that the true category of the sample is a positive example, but the model predicts it as a negative example, so the prediction is wrong [18].

Table 1. Confusion matrix.

	Positive	**Negative**
True	TP	TN
False	FP	FN

IoU (Intersection over Union) is a general evaluation index in the field of computer vision, especially in tasks such as target detection and image segmentation. IoU mainly reflects the degree of overlap between the predicted bounding box and the ground truth bounding box. As shown in Figure 8, the green box is the truth bounding box, which is the box marked while labeling the data set. The red box is the predicted bounding box, which is the prediction box predicted by the trained model. IoU is the result of dividing the overlapping part of two areas by using the part of the two areas [19].

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (1)$$

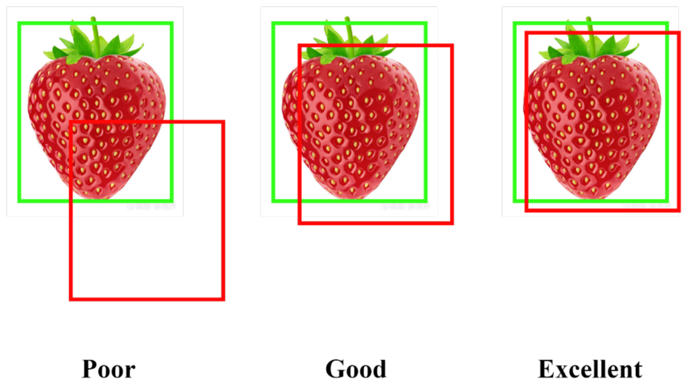

Figure 8. An example of bounding box

Precision is an indicator for evaluating the performance of a classification model. It measures the proportion of items that the model correctly identifies as positive out of all items that the model identifies as positive [20].

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (2)$$

Although precision is an important metric, it does not provide a complete view of model performance on its own. Therefore, precision is often combined with recall.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

where mAP (mean Average Precision) is an indicator widely to evaluate model performance in computer vision tasks, especially in the fields of target detection and image retrieval. The mAP provides a single performance metric to evaluate the overall effectiveness of the model by comprehensively considering the precision and recall of the model under different categories and different detection difficulties. By plotting the curve of Precision versus Recall and calculating the area under the curve (AUC), the AP value of a single category is obtained. The mAP value is the average of the AP values of all categories. The higher the value, the better the performance of the model. The mathematical expressions are shown in Eq. (4) and Eq. (5). In this article, because the prediction results are divided into three classes, namely, k=3.

$$AP = \int_0^1 p(r)dr \tag{4}$$

$$mAP = \frac{\sum_{i=1}^{k} AP_i}{k} \tag{5}$$

where mAP@IoU represents the mAP value calculated under a specific IoU threshold. For example, mAP@0.5 means that the result is considered correct only when IoU ⩾ 0.5. mAP@0.5 is a very popular evaluation metric because it considers the recognition accuracy and positioning accuracy of the model. mAP@[.5:.95] is also an evaluation indicator, which calculates the average of all mAP values with IoU from 0.5 to 0.95 (in steps of 0.05). This approach is more rigorous as it considers a range of different IoU thresholds, providing a more comprehensive perspective on model performance.

4. Results

4.1. Performance of Strawberry Ripeness Detection Model

Figure 9 shows the results of our model based on the validation set. In the same validation dataset, our model can detect most strawberries of different sizes, orientations, environments, and ripeness.

Figure 9. Results of our model on the validation set

We plotted the Precision-Recall (PR) curves of the two models as shown in Figure 10 and Figure 11. The PR curve represents the relationship between precision and recall, where the thin line represents the PR curve of each category, the thick line represents the average PR curve of all categories. The area under the PR curve (AUC) can be employed to reflect the performance of the model. Comparing the AUC values can show the performance gap between the two models [21]. Comparing the AUC values can indicate how much the performance of the model has been improved by adding Swin Transformer. For the "ripe", the AUC increased from 0.925 to 0.933. For the "half- ripe", the AUC increased from 0.789 to 0.804. For the "unripe", the AUC increased from 0.868 to 0.882. For the sum of all, the AUC of mAP @0.5 improves from 0.861 to 0.873. It is obvious that the performance of the model with Swin Transformer is improved at all ripeness, with higher precision and recall values. In short, the PR curve of our model performs well, can accurately detect the ripeness of strawberries, and is significantly improved compared to the YOLOv9 model.

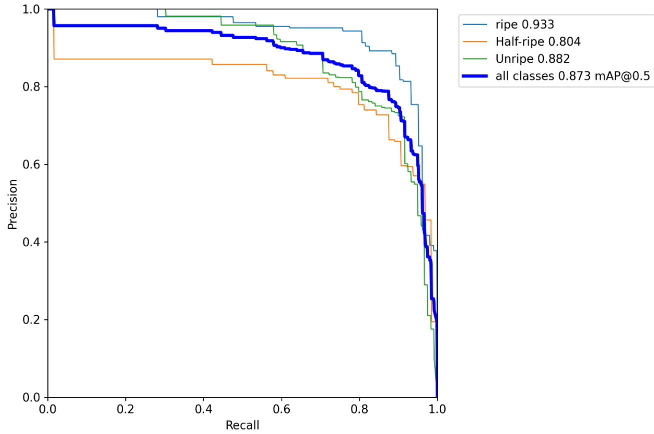

Figure 10. PR curve of YOLOv9+Swin Transformer model

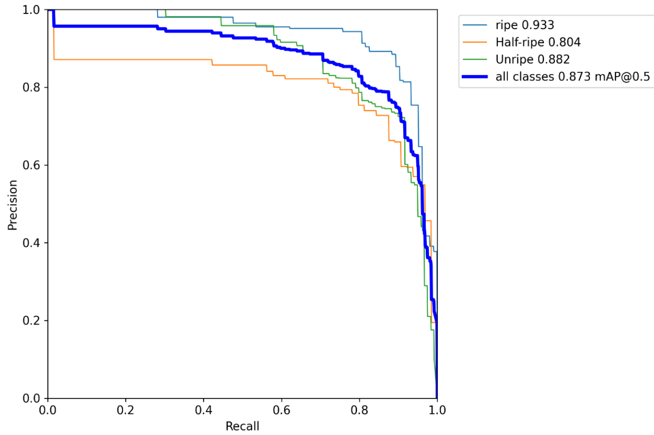

Figure 11. PR curve of YOLOv9 model

Figure 12 shows the evaluation metrics of the proposed model. The Precision and Recall of our model are both high and stable, indicating that the model performs well. mAP@0.5 is the average average accuracy with IoU threshold 0.5. mAP@[.5:.95] This shows the average accuracy calculated over multiple IoU thresholds from 0.5 to 0.95 in steps of 0.05. Figure 4.8 shows that mAP@ and mAP@[.5:.95] continue to increase, indicating that the model performs well under different levels of detection stringency. Overall, the model shows improvement over time in all aspects: Bounding box prediction, object presence confidence, and classification. Precision and recall are both high. mAP is also excellent, reflecting excellent model performance.

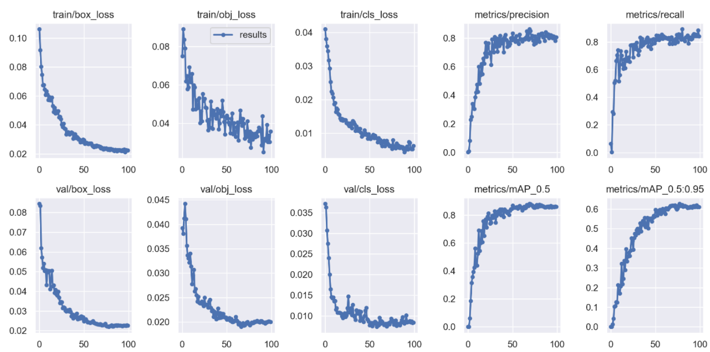

Figure 12. Plots of results of YOLOv9+Swin Transformer model

From Table 2, we analyze the performance of different versions of the YOLO model in terms of Precision, Recall and mAP. The YOLOv9+Swin Transformer model has the highest Precision, reaching 0.853. In comparison, the Precision of the original YOLOv8 and YOLOv9 is slightly lower, with YOLOv9 being 0.777 and YOLOv8 being 0.774. YOLOv9+Swin Transformer reaches 0.840 in Recall, higher than the other three models. YOLOv9+Swin Transformer also has the highest mAP@0.5, reaching 0.873. On the more stringent mAP@[.5:.95], YOLOv9+Swin Transformer also showed the best performance, reaching 0.627. In summary, the YOLOv9+Swin Transformer model we proposed performs optimally on all major performance indicators. This further demonstrates that our method combining YOLOv9 and Swin Transformer is able to improve the performance of the strawberry detection model.

Table 2. Comparison of strawberry ripeness detection models.

Model	Precision	Recall	mAP@0.5	mAP@[.5:.95]
YOLOv9	0.777	0.800	0.861	0.610
YOLOv8	0.774	0.749	0.823	0.552
YOLOv8+Swin Transformer	0.815	0.831	0.861	0.613
YOLOv9+Swin Transformer	**0.853**	**0.840**	**0.873**	**0.627**

All in all, our strawberry ripeness detection model can accurately detect the ripeness of strawberries. All indicators of the model are very good, and our improvements to the model have proven to be very effective.

4.2. Comparison of YOLOv10 modle

YOLOv10 introduces a new approach to real-time target detection. Addressing the shortcomings of previous versions of YOLO in terms of post-processing and model architecture, YOLOv10 achieves state-of-the-art performance while significantly reducing computational overhead [22].

We also run our dataset based on the YOLOv10 model. The results are shown in Table 3.

Table 3. Comparison of YOLOv10 model

Model	Precision	Recall	mAP@0.5	mAP@[.5:.95]
YOLOv9	0.777	0.800	0.861	0.610
YOLOv10	0.817	0.789	0.871	0.620
YOLOv9+Swin Transformer	**0.853**	**0.840**	**0.873**	**0.627**

YOLOv10 shows improved performance compared to YOLOv9, but the combination of YOLOv9 with Swin Transformer still achieves the highest scores. This suggests that the enhancements introduced in YOLOv10 are beneficial, but the additional integration with Swin Transformer provides the best results for real-time target detection.

Notably, the YOLOv10 model significantly outperforms the other two models in terms of training time. This is also in accordance with the improvements made by Wang et al. in their paper on model lightweighting. Combining YOLOv10 with Transformer is possible. The combined model might be able to balance the high efficiency of YOLOv10 with the accuracy of Transformer.

5. Conclusions and Future Work

5.1. Analysis and Discussions

In summary, the model of YOLOv9 and Swin Transformer effectively improves the accuracy and reliability of strawberry ripeness detection. Indicators such as precision, recall, and mAP all show that the hybrid model of YOLOv9 and Swin Transformer has better detection results for strawberries of various ripeness levels.

As shown in the previous sections, our experimental results show that the hybrid model of YOLOv9 and Swin Transformer performs better than the YOLOv9 model. The key factors enabling these advances include:

Firstly, Swin Transformer can capture detailed and subtle features of strawberries, which greatly improves detection rates. This works particularly well in complex scenes where strawberries appear under various lighting and occlusion conditions.

Secondly, the architecture of YOLOv9, especially the integration of Programmable Gradient Information (PGI) and its lightweight and powerful network structure (GELAN), can locate strawberries quickly and accurately within video frames.

5.2. *Conclusion*

In this research project, we successfully demonstrated the integration of YOLOv9 and Swin Transformer models to detect strawberry ripeness with high accuracy. The hybrid model achieved a mean Average Precision (mAP) at an IoU of 0.5 of 87.3%, surpassing the performance of traditional models by using YOLOv9 alone, which registered a mAP of 86.1%. The Precision and Recall are better. This improvement underscores the effectiveness of combining these advanced deep learning technologies to enhance precision in agricultural applications. The ability of this proposed model to accurately categorize strawberries into unripe, half-ripe, and ripe stages can significantly aid in optimizing harvest times, thus reducing waste and increasing yield quality.

5.3. *Limitations*

While the results are promising, our research has several limitations:

Firstly, though the dataset includes images of strawberries from a variety of conditions, they are primarily from one variety. This limitation may affect the applicability of the model to different varieties of strawberries, such as strawberries that are white when ripe.

Secondly, our model has good performance. However, the performance of this proposed model in actual strawberry planting may be affected by external factors such as lighting and camera clarity.

Finally, the strawberry dataset we have proposed is limited in size and variety, and using more datasets may further improve model performance.

5.4. *Future Work*

Our future work remains to solidify the findings of this article and address its limitations.

Firstly, we will collect and integrate data from a wider range of climate and geographic regions to improve the model's robustness and applicability in different agricultural settings.

Secondly, we will improve the model according to different varieties of strawberries to improve the general applicability of our model to various varieties of strawberries.

Thirdly, we will combine visual data with input from environmental sensors (e.g., humidity, temperature), which can improve the accuracy of maturity detection under different environmental conditions.

Finally, we will try to train a model that combines YOLOv10 and Transformer to achieve improved detection results while being more lightweight.

Author Contributions:

Funding: This research has no external fundings.

Data Availability Statement: Data sharing is not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang A, Chen H, Liu L, et al. YOLOv10: Real-Time End-to-End Object Detection. arXiv preprint arXiv:2405.14458, 2024.
2. Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021: 10012-10022.
3. Dhanya VG, Subeesh A, Kushwaha NL, et al. Deep learning based computer vision approaches for smart agricultural applications. Artificial Intelligence in Agriculture, 2022; 6: 211-229.
4. Fracarolli JA, Pavarin FFA, Castro W, Blasco J. Computer vision applied to food and agricultural products. Revista Ciência Agronômica, 2020; 51(spe): e20207749.
5. Ghazal S, Munir A, Qureshi WS. Computer vision in smart agriculture and precision farming: Techniques and applications. Artificial Intelligence in Agriculture, 2024.
6. Kamilaris A, Prenafeta-Boldú FX. Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 2018; 147: 70-90.
7. Bharman P, Saad SA, Khan S, et al. Deep learning in agriculture: a review. Asian Journal of Research in Computer Science, 2022; 13(2): 28-47.
8. Espejo-Garcia B, Mylonas N, Athanasakos L, et al. Towards weeds identification assistance through transfer learning. Computers and Electronics in Agriculture, 2020; 171: 105306.
9. Sharma R, Kukreja V, Bordoloi D. Deep learning meets agriculture: A faster RCNN based approach to pepper leaf blight disease detection and multi-classification. In: 2023 4th International Conference for Emerging Technology (INCET), 2023: 1-5. IEEE.
10. Mu Y, Chen TS, Ninomiya S, Guo W. Intact detection of highly occluded immature tomatoes on plants using deep learning techniques. Sensors, 2020; 20(10): 2984.
11. Ashtiani SHM, Javanmardi S, Jahanbanifard M, et al. Detection of mulberry ripeness stages using deep learning models. IEEE Access, 2021; 9: 100380-100394.
12. Pardede J, Sitohang B, Akbar S, Khodra ML. Implementation of transfer learning using VGG16 on fruit ripeness detection. Int J Intell Syst Appl, 2021; 13(2): 52-61.
13. Momeny M, Jahanbakhshi A, Neshat AA, et al. Detection of citrus black spot disease and ripeness level in orange fruit using learning-to-augment incorporated deep networks. Ecological Informatics, 2022; 71: 101829.
14. Bakirci M, Bayraktar I. YOLOv9-Enabled Vehicle Detection for Urban Security and Forensics Applications. In: Proceedings of the 2024 12th International Symposium on Digital Forensics and Security (ISDFS), 2024: 1-6.
15. Liu Z, et al. Video Swin Transformer. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA: IEEE, 2022: 3192-3201.
16. Pérez-Borrero I, Marín-Santos D, Gegúndez-Arias ME, Cortés-Ancos E. A fast and accurate deep learning method for strawberry instance segmentation. Computers and Electronics in Agriculture, 2020; 178: 105736.
17. Xian M, Xu F, Cheng HD, Zhang Y, Ding J. EISeg: Effective interactive segmentation. In: International Conference on Pat-tern Recognition (ICPR), Cancun: IEEE, 2016: 1982-1987.
18. Caelen O. A Bayesian interpretation of the confusion matrix. Ann Math Artif Intell, 2017; 81(3): 429-450. doi: 10.1007/s10472-017-9564-8.
19. Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A. The Pascal Visual Object Classes (VOC) Challenge. Int J Comput Vis, 2010; 88(2): 303-338.
20. Buckland M, Gey F. The relationship between Recall and Precision. Journal of the American Society for Information Science, 1994; 45(1): 12-19.
21. Davis J, Goadrich M. The relationship between Precision-Recall and ROC curves. In: Proceedings of the 23rd International Conference on Machine Learning (ICML), Pittsburgh, Pennsylvania: ACM Press, 2006: 233-240.
22. Wang A, Chen H, Liu L, Chen K, Lin Z, Han J, Ding G. YOLOv10: Real-Time End-to-End Object Detection. arXiv preprint arXiv:2405.14458, 2024
23. Yan, W. Computational Methods for Deep Learning. Springer 2023.
24. Yan, W. Introduction to Intelligent Surveillance. Springer 2019.