Elevating Grape Detection Precision and Efficiency with a Novel Deep Learning Model

Xiaoli Geng*, Yaru Huang, Yangxu Wang
Department of Network Technology, Guangzhou Institute of Software Engineering, Conghua, Guangdong, China

***Abstract*—In the domain of modern agricultural automation, precise grape detection in orchards is pivotal for efficient harvesting operations. This study introduces the Grapes Enhanced Feature Detection Network (GEFDNet), leveraging deep learning and convolutional neural networks (CNN) to enhance target detection capabilities specifically for grape detection in orchard environments. GEFDNet integrates an innovative Enhanced Feature Fusion Module (EFFM) into an advanced YOLO architecture, employing a 16x downsampling Backbone for feature extraction. This approach significantly reduces computational complexity while capturing rich spatial hierarchies and accelerating model inference, which is crucial for real-time object detection. Additionally, an optimized dual-path detection structure with an attention mechanism in the Neck enhances the model's focus on targets and robustness against dense grape detection and complex background interference, a common challenge in computer vision applications. Experimental results demonstrate that GEFDNet achieves at least a 3.5% improvement in mean Average Precision (mAP@0.5), reaching 89.4%. It also has a 9.24% reduction in parameters and a 10.35 FPS increase in frame rate compared to YOLOv9. This advancement maintains high precision while improving operational efficiency, offering a promising solution for the development of automated harvesting technologies. The study is publicly available at: https://github.com/YangxuWangamI/GEFDNet.**

Keywords*—Computer vision; deep learning; Convolutional Neural Networks (CNN); real-time object detection; dual-path detection structure

I. INTRODUCTION

Grapes, as deciduous vines of the Vitis genus, are celebrated as the "Queen of Fruits." They are not only rich in nutrients but also possess significant medicinal value, making them one of the most popular fruits globally [1]. In the field of agricultural automation, precise grape detection is key to improving harvesting efficiency and fruit quality. Although manual harvesting is still the mainstream method, it is inefficient and labor-dependent [2], creating an urgent need for automated solutions. Existing vision detection systems face challenges in complex orchard environments, such as changes in lighting, occlusions, and fruit overlapping, which limit their performance. Therefore, a robust detection model is crucial for robots to achieve target perception in complex vineyard scenarios [3].

To enhance the recognition ability and efficiency of deep learning models in orchard grape detection, the goal of this study is to develop a fast, parameter-reduced, and low-miss detection model for dense and occluded grape detection in orchards, named Grapes Enhanced Feature Detection Network (GEFDNet). At the same time, YOLOv9 [4], as the latest generation of the YOLO series, has demonstrated its excellent accuracy and speed in various general object detection tasks through optimized network architecture and detection algorithms. Despite this, applying YOLOv9 directly to grape detection tasks in orchards still faces specific challenges. In response to these challenges, the GEFDNet model targets grapes, innovatively designing a 16x downsampling Backbone network and proposing a new high-efficiency scale fusion module called the Enhanced Feature Fusion Module (EFFM) module, aiming to capture target feature information at a finer granularity. Applied to the main trunk and detection neck networks, it significantly reduces the model's computational burden and parameter volume, enabling GEFDNet to better adapt to the complex and variable agricultural environment.

In the experiments, to objectively and comprehensively evaluate model performance, this study conducted comparative experiments with seven other advanced methods, especially an in-depth performance evaluation against the benchmark model YOLOv9. Performance analysis results show that GEFDNet has increased the mean Average Precision (mAP@0.5) on the test dataset by at least 3.5%. Through visual analysis, the model's advantages in dealing with challenging complex scenes were further revealed. In addition, compared to YOLOv9, GEFDNet has reduced the parameter volume by about 9.24% and increased the frame rate (FPS) by 10.35. This series of data highlights the efficiency of GEFDNet in object detection tasks.

The main contributions of this paper are as follows:

- The design of the EFFM module, which enhances the accuracy and efficiency of target detection in images by providing a powerful feature extraction and fusion mechanism for grape target detection tasks.
- The innovative design of a 16x downsampling Backbone network addresses the prolonged training times and weight redundancy issues associated with YOLOv9's detection neck and auxiliary branch structures.
- Optimization of the main detection neck design overcomes the difficulty of detecting occlusions and dense targets, reduces the model's computational burden and parameter count, and ensures high detection accuracy in resource-constrained environments.
- Comparative experiments with seven other popular detection models demonstrate GEFDNet's advantages in lightweight design, further verifying its effectiveness and feasibility.

*Corresponding Author

The layout of this paper is as follows: Section I (this section) introduces the prominent issues in the research field and the motivation behind the model design. Section II summarizes the research background and the challenges of existing technology. Section III provides a detailed introduction to the characteristics of the dataset and the principles of model design. Section IV includes the experimental process, model performance comparison, and result analysis. Section V discusses the research findings and proposes future research plans. Section VI summarizes the entire paper.

II. Background

With the rapid development of computing power and deep learning techniques [5], convolutional neural networks have attracted attention across various industrial sectors. Object detection models integrated with deep learning are being increasingly applied to agricultural studies, including fruit recognition [6, 7], disease detection [8, 9, 10], and yield estimation [11, 12]. Currently, deep learning-based fruit object detection models are mainly divided into two categories: one category is the region proposal-based two-stage detection models, such as Faster R-CNN [13] and Spatial Pyramid Pooling Network (SPP-Net) [14], which have high detection accuracy but are slower due to their two-stage nature. The other category is the regression-based single-stage detection models, such as SSD [15], YOLO [16, 17], and CenterNet [18], which maintain high detection accuracy while offering faster detection speeds and stronger real-time capabilities. Due to the high demand for detection speed in most tasks, especially in real-time scenarios, single-stage algorithms have more advantages in practical applications.

In recent years, research on deep learning models for grape detection has been continuously emerging. The latest research from Wu et al., 2024 [19], uses Adaptive Training Sample Selection (ATSS) as a label matching strategy to improve the quality of positive samples and address the challenge of detecting grape stems with similar colors. They utilize the Wise-IoU (Sequential Evidence for Intersection over Union) loss function with weighted interpolation to overcome the limitations of CIoU, which does not consider the geometric properties of targets, thus improving detection efficiency. Behera et al., 2023 [20], proposed an FR-CNN algorithm for plant fruit prediction using Intersection over Union (IoU), achieving an 89% accuracy rate in fruit yield estimation. Aguiar et al., 2021 [21], used deep learning models for grape cluster detection with an average accuracy of 66.96%. Pereira et al., 2019 [22], introduced a grape detection method based on the AlexNet neural network architecture, achieving a high average accuracy of 77.30%. Rong et al., 2024 [23], proposed a grape cluster detection method based on Spatial-to-Depth Convolution (STD-Conv) and Simple Attention Mechanism (SimAM), expanding the dataset through data augmentation technology, enabling the improved YOLOX model to achieve an 88.40% average accuracy in grape cluster detection. Marani et al., 2020 [24], proposed a vehicle-mounted RGB-D camera system for grape recognition using a deep learning framework. Sozzi et al., 2021 [25], used the YOLOv4 model for the detection and counting of grape clusters, achieving an accuracy rate of 48.90%. Li et al., 2021 [26], proposed an improved YOLOv4-tiny model, YOLO-Grape, to address the issue of unrecognizable accuracy caused by complex background scenes such as shadows and overlaps.

III. Materials and Methods

This section provides a detailed description of the datasets used in the experiments and elucidates the design principles, innovations, and activation functions of the GEFDNet model.

A. Datasets

To validate the effectiveness and adaptability of the proposed method, the experiments in this study are conducted using the Embrapa Wine Grape Instance Segmentation Dataset (Embrapa WGISD) [27]. This dataset was created for the application of object detection and instance segmentation techniques in image monitoring and field robot vision in vineyards, containing instance images of five different grape varieties. These images were captured under natural field conditions, encompassing various postures, lighting and focus conditions, as well as genetic and phenotypic variations such as shape, color, and compactness.

The images of the dataset were taken using a Canon EOS REBEL T3i DSLR camera and Motorola Z2 Play smartphone at the Guaspari Winery in São Paulo, Brazil. The image resolution was adjusted to a width of 2,048 pixels to balance image detail and processing time. The dataset was annotated with rectangular bounding boxes to identify grape clusters using the LabelImg tool [28], comprising a total of 300 images with 4,432 annotated grape clusters.

In summary, experiments conducted on the Embrapa WGISD dataset will provide a comprehensive evaluation of the universality and effectiveness of the proposed method. However, the orchard environment is challenging, as depicted in Fig. 1, which categorizes the dataset's characteristics and key detection challenges into four types, including densely packed arrangements of grapes, occlusions by leaves or trunks, complex backgrounds, and varying lighting conditions.

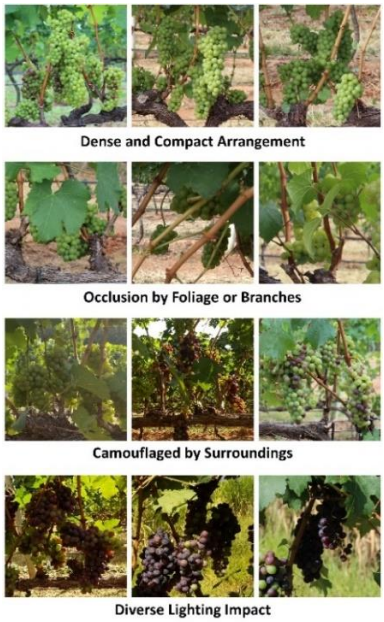

Fig. 1. Four typical challenges in dataset images.

B. Model Construction

When applying neural networks for grape detection in orchard environments, numerous factors must be considered. To address these, this study introduces the Grapes Enhanced Feature Detection Network (GEFDNet), a novel high-precision, low-complexity grape detection model for orchard environments. The model adopts a Backbone-Neck structure and integrates the proposed Enhanced Feature Fusion Module (EFFM). The Backbone is responsible for extracting key features from the input images, employing a deep convolutional neural network to ensure the capture of rich spatial hierarchical information while reducing computational complexity. The Neck features a dual-path detection structure [4], including the Main Branch and the Auxiliary Branch, which further process target features, providing additional feature fusion and contextual information through parallel processing paths, enhancing the model's robustness against complex environmental variations. The architectural framework of GEFDNet is illustrated in Fig. 2, with detailed module structures, including EFFM, presented in Fig. 3. The auxiliary detection components are denoted with dashed lines in the diagrams. The following sections will detail their configuration specifics.

C. Enhanced Feature Fusion Module (EFFM)

Feature fusion is crucial for enhancing the model's generalization capability and detection accuracy in grape target detection tasks. By integrating features from different levels and scales, the model can more comprehensively understand image content, leading to more accurate identification and localization of grapes.

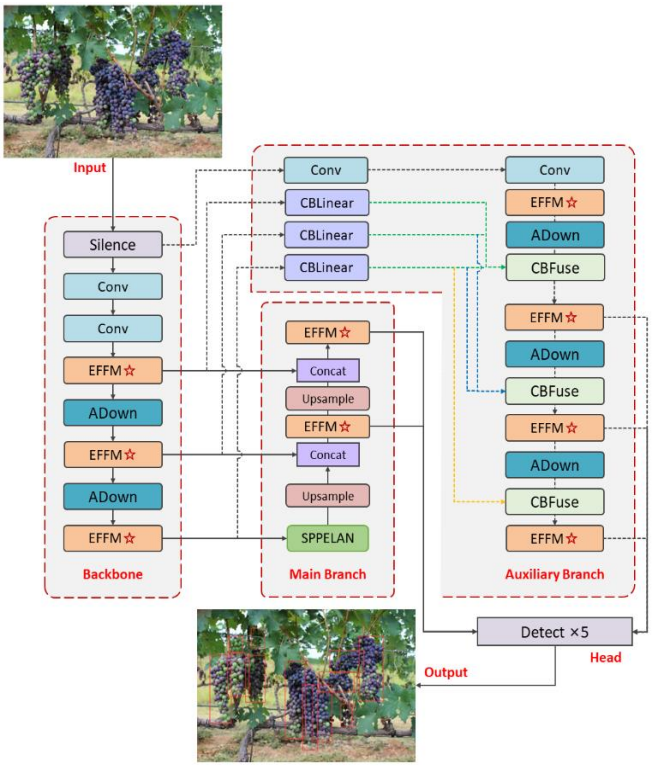

Fig. 2. Architecture of GEFDNet.

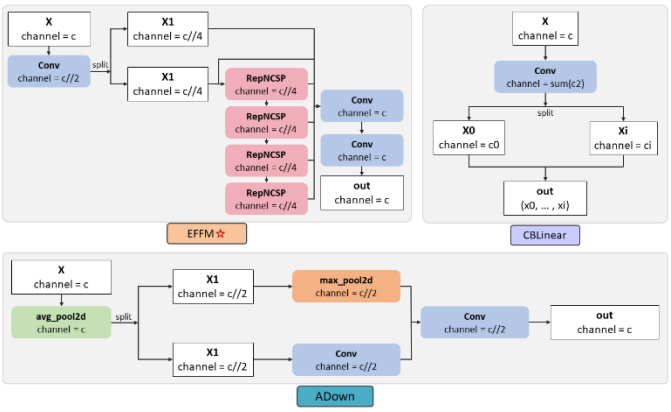

Fig. 3. Detailed module structures.

This paper introduces an innovative and efficient scale fusion module referred to as the Enhanced Feature Fusion Module (EFFM), as depicted in Fig. 3. After the input feature maps undergo a 1×1 convolution, the channel count is halved. The feature maps are evenly divided into two subsets of the same spatial size, denoted as X1, each with a quarter of the channel count of the input feature maps. One X1 subset is retained as is, while the other is processed through the RepNCSP module, further divided into four feature map subsets. These subsets undergo channel adjustments as they pass through each RepNCSP module, achieving efficient feature processing and fusion. For instance, a feature map may transition from channel count c to c//4, processed through a Conv layer, and further refined by the RepNCSP module, ultimately restoring to the original channel count c at the output. It is notable that each convolution can receive feature information from the preceding features, and for each feature branch after the RepNCSP module, the output has a larger receptive field and richer features compared to the unprocessed branch.

D. GEFDNet

1) Backbone Component: The Backbone component is tasked with feature extraction from input images. The architecture initiates with a Silence Module that serves as the preliminary processing unit, accepting raw image data and performing necessary preprocessing steps to maintain the initial feature information of the image, ensuring that the Main Branch and Auxiliary Branch can fully utilize this information for precise target localization. Subsequently, the model integrates multiple standard convolutional and pooling layers, each equipped with a 3×3 convolutional kernel, and employs stride-2 downsampling to reduce the dimensionality of the feature maps.

To enhance the model's nonlinear feature expression and integration capabilities, the Backbone network incorporates the innovative EFFM module, which facilitates deep integration of cross-layer features and effectively captures complex spatial hierarchies within the image. Notably, the core network of GEFDNet innovates in its downsampling strategy, adopting a 16x downsampling design that significantly reduces the loss of spatial resolution, enabling the model to excel in detecting small-sized targets and under varying lighting conditions.

Upon processing through the Backbone network, the model achieves 16x downsampling through four downsampling convolutional layers and three EFFM feature extraction layers, with the output feature map size being 1/16th of the original image, transitioning to multi-scale, multi-depth feature representations. This aids in reducing the model's computational load while retaining sufficient feature information to support subsequent detection tasks.

2) Neck Component: The Neck serves as the critical link between the Backbone and the detection Head, comprising both the Main Branch and the Auxiliary Branch. The Main Branch receives feature maps of varying downsampling levels output from the Backbone and initially processes them through an SPPELAN module to expand the receptive field, enhancing feature abstraction and expression. To further enhance the model's robustness, an attention mechanism is integrated into the Main Branch, allowing the model to adaptively focus on key image regions, such as grape edges and textures, thereby maintaining high accuracy despite challenges like occlusions and overlaps. Computational efficiency is also a significant consideration in the design of the Main Branch, where lightweight network components and depthwise separable convolutions are employed, effectively reducing the model's parameter count and computational complexity without compromising detection accuracy.

A series of upsampling and concatenation operations then follow, merging deep and shallow features from the Backbone to construct a multi-scale feature representation. Upsampling employs nearest neighbor interpolation to enlarge the feature map size, increasing resolution for more precise small target localization and facilitating fusion with larger feature maps. The concatenation operation integrates features from different levels, importantly, further processed through the innovatively designed EFFM feature extraction layer. Ultimately, the Main Branch outputs feature maps with high semantic information and spatial resolution, providing the Head with high-quality inputs for detection.

In addition to the Main Branch, the model's innovation lies in the design of the Auxiliary Branch, incorporating a reversible auxiliary branch design, utilizing cross-layer connections to directly extract and fuse features from the Backbone with high-level features from the Main Branch. Modules such as CBLinear and CBFuse are employed, unifying feature map sizes through cross-block connections and feature fusion strategies, followed by an addition operation to achieve multi-level auxiliary information fusion. This design not only enhances the model's detection capabilities for small targets and complex scenes but also reduces computational load through parallel processing, balancing computational efficiency with detection accuracy. Furthermore, the Auxiliary Branch serves as a regularization technique to prevent overfitting during model training.

3) Head Component: Upon the completion of the Neck's operations, the Head detection component receives five feature maps with varying spatial resolutions and semantic depths from the Neck. This enables the detection head to generate bounding boxes and aim frames of corresponding scales based on the feature map scales, overlaying the model's inference results onto the input image.

E. Activation Functions

In the realm of deep learning, activation functions play an indispensable role in dictating the performance and convergence rate of a model, determining the network's capacity to learn nonlinear relationships. Commonly utilized activation functions include Sigmoid Linear Unit (SiLU) [29], Rectified Linear Unit (ReLU) [30], and Leaky ReLU [31]. In constructing the GEFDNet model, this paper specifically selects SiLU as the activation function due to its combination of linear and nonlinear characteristics, which effectively enhances the network's nonlinear expressive power and learning efficiency. The definition of the SiLU activation function is presented in Eq. (1):

$$SiLU(x) = x \cdot \sigma(x) = x \cdot \frac{1}{1+e^{-x}} \quad (1)$$

The primary features of SiLU include its monotonicity, ensuring that as the input x increases, the output also increases, aiding in mitigating the vanishing gradient problem in deep networks. Its linearity for positive input values simplifies the nonlinear complexity in the positive range. SiLU's zero-centering characteristic, which outputs zero when $x = 0$, helps in centering the data, and when combined with batch normalization techniques, further improves the efficiency of model training. Additionally, SiLU boasts high computational efficiency as it involves only basic exponential and division operations, making it suitable for rapid execution on limited computational resources.

The Rectified Linear Unit (ReLU) activation function is one of the most popular nonlinear activation functions in deep learning. Its definition is straightforward and intuitive, expressed in Eq. (2):

$$ReLU(x) = max(0, x) \quad (2)$$

This function introduces nonlinearity by setting all negative values to zero, allowing only positive values to pass through, while maintaining computational efficiency. The main advantage of ReLU is its acceleration of the neural network training process; however, it also has some drawbacks, the most notable being the "dead ReLU" problem, where neurons corresponding to negative inputs may never activate, causing their weights to no longer update during training. In addition, ReLU's output is not zero-centered, which may affect the stability and convergence speed of the model during training.

Leaky ReLU is an improved version of ReLU, aiming to address the dead ReLU problem. Its formula is given in Eq. (3):

$$Leaky\ ReLU(x) = max(\alpha \cdot x, x) \quad (3)$$

Where α is a small positive number, typically taken as 0.01. Leaky ReLU introduces a small linear term when the input value is negative, ensuring that neurons with negative inputs still have a non-zero gradient, thus alleviating the problem of neuron death. This slight linear operation allows neurons with negative input values to still update their weights during the training process. However, Leaky ReLU introduces an additional hyperparameter α, which, if not chosen properly, may affect the network's convergence speed or lead to suboptimal model performance.

In deep learning, the choice of the appropriate activation function is crucial for model performance. Compared to ReLU and Leaky ReLU, SiLU offers several significant advantages, making it an ideal choice for grape detection models. SiLU's self-normalizing characteristic makes its output a linear transformation of the input in the positive range and approaches zero in the negative range, which helps stabilize network output and enhance generalization ability. Moreover, SiLU does not require additional parameters like Leaky ReLU, simplifying model training and hyperparameter adjustment. At the same time, the biological plausibility of SiLU further ensures the naturalness and efficiency of the activation pattern.

IV. EXPERIENCE

This section provides an overview of the evaluation criteria and experimental design, followed by a presentation of the GEFDNet model's performance and a comparison with existing technologies. In addition to a comprehensive performance assessment using metrics such as Precision, Recall, and F1-score (F1), visual attention contrast experiments are introduced to further analyze the model's detection mechanisms. Utilizing Grad-CAM technology, the areas of focus when processing different grape samples are visualized, revealing GEFDNet's advantages in target recognition. Finally, the model's lightweight effect is evaluated, emphasizing its potential for efficient deployment in resource-constrained environments.

A. Experimental Conditions and Details

The study was conducted on a PC equipped with an AMD Ryzen 7 5800H 8-core processor (3.20 GHz) CPU and an NVIDIA GeForce GTX 3090 GPU. The software tools included the PyTorch 2.0.0 deep learning framework [32], CUDA version 11.8 parallel computing framework, and CUDNN version 8.9.5 deep neural network acceleration library. Standard data preprocessing methods were employed to fully leverage the dataset's information, including image scaling, cropping, and normalization, along with data augmentation techniques such as random flipping and rotation to enhance the model's generalization capability. Stochastic Gradient Descent (SGD) was used as the optimizer, with network training parameters set to an input size of 640×640 pixels, a batch size of 16, an initial learning rate of 0.01, a decay rate of 0.001, and a momentum parameter of 0.937. Considering convergence, 300 epochs of training were deemed sufficient for the model to reach a state of convergence.

B. Assessment of Model Performance

For the assessment of the proposed model within this study, the metrics of mAP@0.5, mAP@0.5:0.95, and F1-score were selected. The performance was compared against seven benchmark models, namely CenterNet [18], Faster R-CNN [13], SSD [33], FCOS [34], EfficientDet [35], YOLOv7-tiny [36], and YOLOv9 [4], utilizing the same dataset. The GEFDNet model underwent training and testing under identical conditions as the benchmark models, with evaluation based on Precision (P), Recall (R), F1-score (F1), and mean Average Precision (mAP).

Understanding the significance of these metrics requires clarity on the concepts of true positives (TP), false positives (FP), and false negatives (FN). TP represents the count of correctly identified samples, while FP denotes the instances of incorrect identifications. FN corresponds to the number of missed detections. The sum of "TP + FP" indicates the total inferred grape fruits by the model, and "TP + FN" accounts for the actual total count of fruits in the image.

Precision (P), which measures the accuracy of the model's positive predictions, is calculated as the ratio of true positive predictions to the total predicted positives, as illustrated in Eq. (4). This metric reflects the model's proficiency in accurately predicting positive outcomes.

$$Precision = \frac{TP}{TP+FP} \tag{4}$$

Recall (R), depicted in Eq. (5), is the ratio of true positive predictions to the total actual positives, quantifying the model's effectiveness in capturing all actual positive instances.

$$Recall = \frac{TP}{TP+FN} \tag{5}$$

The F1-score, represented by Eq. (6), is the harmonic mean of Precision and Recall, with a higher F1-score indicating a better balance between Precision and Recall.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \tag{6}$$

Average Precision (AP) signifies the area under the Precision-Recall (P-R) curve, calculated using an integral as shown in Eq. (7), a comprehensive indicator that takes into account both Precision and Recall.

$$mAP = \frac{1}{n}\sum_{1}^{n} \int P(dR) \tag{7}$$

The mAP@0.5 variant computes the average AP value at an Intersection over Union (IoU) threshold of 0.5 for all object categories. Furthermore, mAP@0.5:0.95 is determined to evaluate the model's performance across a spectrum of IoU thresholds, offering a stringent assessment of performance by representing the average mAP at various IoU thresholds ranging from 0.5 to 0.95 with increments of 0.05. The F1-score evaluates the methodology's performance by balancing the importance of accuracy and recall.

The results of the GEFDNet experiments are detailed in Table I, showcasing the performance of each model within the test dataset.

TABLE I. QUANTITATIVE RESULTS ON THE TEST DATASET

Model	P	R	F1	mAP@0.5	mAP@0.5:0.95
CenterNet	0.79	**0.85**	0.82	0.751	0.330
Faster R-CNN	0.79	0.82	0.81	0.815	0.398
SSD	0.26	0.59	0.36	0.239	0.095
FCOS	0.82	**0.85**	**0.84**	0.843	0.508
EfficientDet	0.07	0.57	0.12	0.095	0.018
YOLOv7-tiny	0.44	0.45	0.44	0.423	0.111
YOLOv9	**0.88**[a]	0.78	0.83	0.864	**0.601**
GEFDNet	**0.88**	0.81	**0.84**	**0.894**	0.596

[a]-The best performance is indicated in bold.

The experimental results demonstrate GEFDNet's significant advantage in target detection performance compared to other advanced methods. GEFDNet achieved an F1-score of 0.84, tying with YOLOv9 for the highest score, indicating an excellent balance between precision and recall. Particularly, in the key metric of mAP@0.5, GEFDNet surpassed all other models with a value of 0.894, including YOLOv9's 0.864, highlighting its superior detection accuracy at medium IoU thresholds. Furthermore, GEFDNet's comprehensive evaluation of model performance across different IoU thresholds from 0.5 to 0.95, mAP@0.5:0.95, achieved a value of 0.596, slightly trailing YOLOv9's 0.601, but this performance still ranks second among all compared models, showing its consistency and robustness across different IoU threshold ranges.

C. Visual Attention Contrast Experiment

To further examine the differences in detection effects between the proposed GEFDNet model and the existing YOLOv9 model, the Grad-CAM algorithm [37] was employed to visualize and compare the activation heat maps of the two models at different layers. Grad-CAM generates visual heat maps by combining the model's gradients and feature maps, revealing the visual areas the model focuses on when making specific predictions. This intuitive approach allows for a deeper understanding of the model's performance advantages and potential limitations, as shown in Fig. 4, which provides two sets of diagrams.

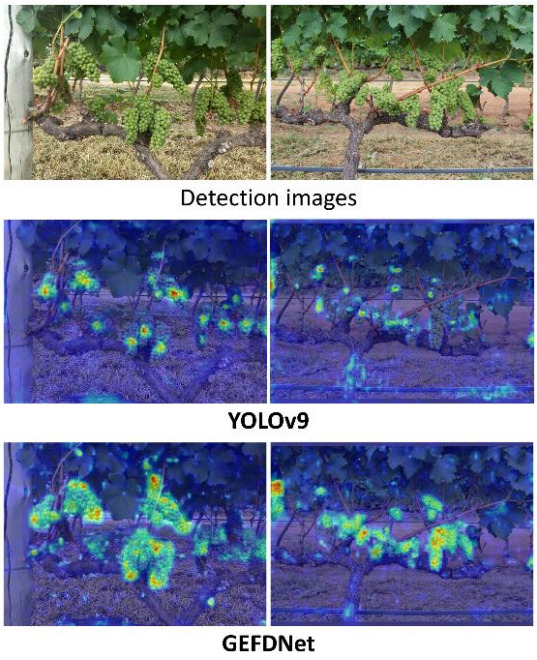

Fig. 4. Grad-CAM Visualization results.

When evaluating heat maps, focus on the following three key features to measure model performance:

- Clear boundary identification: The heat map should clearly depict the outline of the target object, demonstrating the model's high precision in spatial positioning.
- Noise suppression ability: The ideal heat map should not show excessive activation on image noise or irrelevant details, indicating that the model can effectively filter out unimportant information.
- Coverage of important features: The heat map should cover the key features of the target object, which are crucial for the object's recognition and classification.

Through the visualization results of Grad-CAM, it can be observed that GEFDNet has advantages in the above three aspects. Firstly, when localizing fruit targets, GEFDNet shows clearer boundaries and more focused attention, while YOLOv9, although able to recognize targets, also pays attention to the background, leading to scattered attention. Secondly, the heat map of GEFDNet performs better in suppressing image noise, indicating that it has stronger robustness when dealing with complex backgrounds and occlusions. Thirdly, the heat map of GEFDNet better covers the key features of grapes, aiding the model in more accurate recognition and classification of target objects.

D. Validation of Comprehensive Detection Capability

To verify the comprehensive detection capability of the improved GEFDNet model in different environments, we selected samples with varying lighting and density and conducted comparative experiments focusing on the model with similar performance to YOLOv9. We also conducted a detailed analysis of the visualization results of both models. This process revealed potential errors when detecting specific types of targets, thereby providing targeted guidance for subsequent model optimizations. Building on this, we further investigated specific conditions where the model might encounter difficulties. Fig. 5 illustrates the three most severe errors in the test dataset. The blue frames indicate the magnified portions of the images, while the yellow areas denote the regions of the grape clusters that were missed by the detection model.

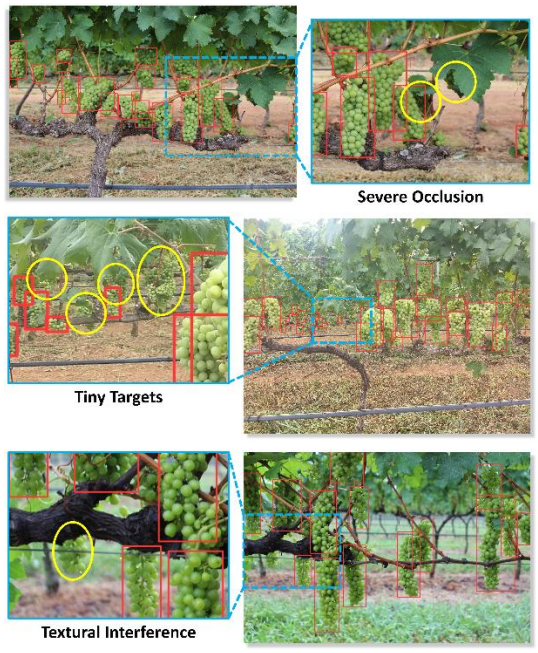

Fig. 5. The three most severe errors.

Firstly, there is the situation of extreme occlusion, where grapes are almost completely obscured by a large amount of foliage, with very little exposed. Under such extreme conditions, although GEFDNet performs better than YOLOv9 overall, there is a decline in detection accuracy. This is mainly because in the extreme occlusion environment, the information available for extracting effective features is greatly reduced. Despite the model's dual-path detection structure and EFFM module trying their best to capture features, it is still difficult to overcome the severe lack of information.

The second scenario is when facing extremely small grapes, the detection accuracy of GEFDNet decreases. This is because during the downsampling process of the model's Backbone network, the detailed features of very small grapes may be lost, making it difficult for the model to identify them accurately. The third scenario is in highly complex backgrounds, which include a large number of distractors similar in color and texture to grapes, as well as scenes with complex lighting and shadow variations. In such cases, although GEFDNet can filter out some irrelevant information, it is still interfered with by similar objects, resulting in a certain degree of false positives and false negatives. This indicates that the model's ability to resist interference needs to be further improved when dealing with highly complex backgrounds.

During the evaluation process, particular attention was given to the confidence threshold setting that yields the optimal mean Average Precision (mAP@0.5) across the entire test dataset. This strategy ensures the objectivity of the assessment while filtering for detections that the model is more confident in, effectively avoiding the impact of low-confidence predictions on the fairness of the evaluation. After the detection process, representative cases of false positives and false negatives were selected and visually presented, as shown in Fig. 6, where the blue areas indicate targets that were either missed or misidentified by the model.

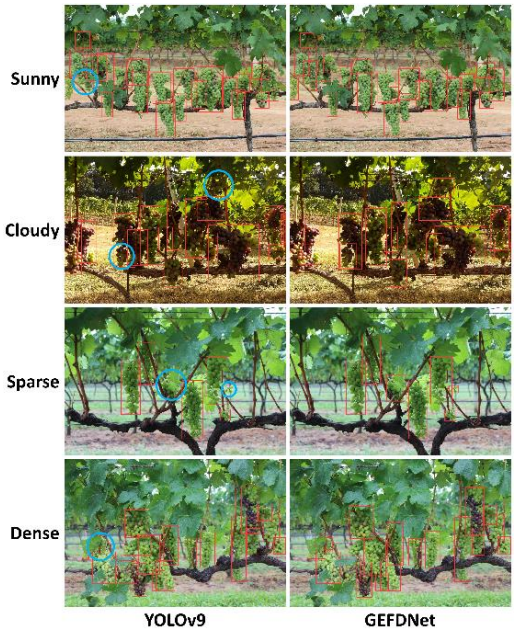

Fig. 6. Comparison of detection effects between YOLOv9 and GEFDNet models.

By carefully examining these results, the following typical errors and their causes can be identified: Under sunny conditions, both YOLOv9 and GEFDNet demonstrated good detection performance. However, YOLOv9 exhibited missed detections for small targets, likely due to insufficient feature extraction capabilities. Under overcast and uneven lighting conditions, YOLOv9 missed detections for grapes obscured by leaves and for small grapes beneath larger grapes. Furthermore, under varying densities, YOLOv9 consistently missed detections for grapes obscured by leaves, whether in sparse or densely clustered distributions. In contrast, GEFDNet effectively addressed these issues, particularly in detecting occluded and densely clustered grape clusters. These analyses not only reveal the limitations of YOLOv9 but also point the way for further optimization and development of GEFDNet.

E. Model Lightweighting

In the development of deep learning models, lightweighting is a critical optimization direction, particularly for application scenarios with constrained resources, such as edge device deployment in the agricultural sector. Lightweight models maintain sufficient detection accuracy while reducing computational load and storage requirements, thereby enhancing model operational efficiency and lowering deployment costs. In the experiments focused on model lightweighting, the recognition performance parameters of the YOLOv9 and GEFDNet models on the test dataset were compared, with the results presented in Table II. Key performance indicators such as Frames Per Second (FPS) were utilized, supplemented by the count of parameters and the size of the weight files to evaluate the models.

TABLE II. LIGHTWEIGHTING COMPARISON BETWEEN YOLOV9 AND GEFDNET MODELS

Model	F1	mAP@0.5	FPS	Parameters	Weights
YOLOv9	0.83	0.864	42.01	48.60 M	98.00 M
GEFDNet	**0.84**	**0.894**	**52.36**	**44.11 M**	**88.90 M**

The comparative experimental data clearly demonstrate the advantages of GEFDNet across multiple key indicators. Specifically, in terms of mAP@0.5, GEFDNet outperformed YOLOv9 with a score of 0.894 versus 0.864, marking a 3.5% improvement. This enhancement indicates that GEFDNet has achieved higher detection accuracy. Moreover, alongside the increase in precision, GEFDNet has also realized optimizations in lightweighting. The model's parameter volume has been reduced from 48.60M in YOLOv9 to 44.11M, and the weight file size has also been minimized from 98.00M to 88.90M. Furthermore, the detection frame rate (FPS) has been increased from 42.01 FPS of YOLOv9 to 52.36 FPS. These enhancements not only alleviate the storage burden but also imply that in resource-constrained environments, such as edge devices in the agricultural sector, GEFDNet can be deployed at a reduced cost. For application scenarios demanding high real-time performance, such as harvesting robots, these improvements are crucial for ensuring the system's response speed and processing capabilities.

V. Discussion and Future Work

The GEFDNet model introduced in this study offers a range of significant advantages in the field of grape detection in orchards. Firstly, the model integrates an innovative and efficient feature fusion module, the Enhanced Feature Fusion Module (EFFM), with a 16x downsampling Backbone network. This integration effectively balances detection accuracy and computational efficiency, reducing the model's parameter volume while increasing the frame rate, which is crucial for applications with high real-time requirements. Secondly, the introduction of the EFFM module enhances the model's ability to detect grapes against complex backgrounds and dense targets. Moreover, the high mean Average Precision (mAP) values demonstrated on the Embrapa WGISD dataset substantiate the model's excellent generalization and robustness.

Despite the positive outcomes of this study, there are certain limitations. It should be noted that the dataset used in this study is derived from a single crop species, and therefore, future testing and validation on more diverse datasets are required. Particularly, testing under poor lighting conditions and for extremely dense or very small-sized grapes should be conducted. Additionally, future research plans should expand and diversify the training datasets. Although the Embrapa WGISD dataset provides valuable resources for grape detection research, it has limitations, such as insufficient images of certain grape varieties, ripeness levels, and environmental conditions [38]. Moreover, to fully assess the potential of GEFDNet in real-world applications, future work will include real-time deployment assessments on actual hardware platforms like edge devices, drones, and agricultural robots. This aligns with the current trend in the field of agricultural automation towards evaluating practical application of models [39, 40]. This will help reveal the model's performance in resource-constrained environments and provide key insights for practical applications.

VI. Summary

Efficient and accurate detection of grapes in orchards has always been a challenging task. In this study, a high-precision, low-complexity deep learning model for grape detection in orchard environments, GEFDNet, was proposed, along with the innovative EFFM module integrated into the 16x downsampling Backbone network and optimized Neck structure. GEFDNet achieves model lightweighting while maintaining high accuracy, significantly enhancing the model's operational efficiency and practicality. The main achievements include a minimum 3.5% increase in mean Average Precision (mAP@0.5) on the test dataset, a reduction of about 9.24% in model parameter volume, and a 10.35 FPS increase in frame rate, validating the effectiveness of model lightweighting. Through Grad-CAM visualization analysis, GEFDNet's superior detection capabilities and precision in target recognition in complex scenarios have been demonstrated.

In summary, the development of the GEFDNet model not only promotes the advancement of agricultural automation technology but also provides a new perspective for the application of deep learning in complex scenarios. With the continuous deepening of future work, it is anticipated that GEFDNet will unleash greater potential in practical applications and make a substantial contribution to agricultural modernization.

Acknowledgement

This research was supported by grants from the “Guangzhou Institute of Software Engineering - Guangzhou Xinwei Internship Base Project" (No. SXJD20210101), the "Guangzhou Institute of Software Engineering - Guangdong Teddy Intelligent Technology Co., Ltd. Big Data Industry-Education Integration Practice Teaching Base Project" (No. SJJD202301), and the "Guangzhou Institute of Software Engineering - Guangdong Teddy Practice Teaching Base" for the 2024 Off-Campus Practice Teaching initiative.

References

[1] Alston, Julian M. and Olena Sambucci. "Grapes in the World Economy." The Grape Genome, Springer International Publishing, 2019, pp. 1-24.

[2] Rakhmatovich, Kholmuminov. "Fundamentals of Targeted Integrative Program Development for Rural Labor Market Growth in Surplus Regions." International Journal of Economics and Financial Issues, vol. 14, 2024, pp. 239-244.

[3] Jiqing, Chen et al. "Efficient and Lightweight Grape and Picking Point Synchronous Detection Model Based on Key Point Detection." Computers and Electronics in Agriculture, vol. 217, 2024, p. 108612.

[4] Wang, Chien-Yao et al. "Yolov9: Learning What You Want to Learn Using Programmable Gradient Information." ArXiv, vol. abs/2402.13616, 2024.

[5] LeCun, Yann et al. "Deep Learning." nature, vol. 521, no. 7553, 2015, pp. 436-444.

[6] Xu, Bo et al. "Apple Grading Method Design and Implementation for Automatic Grader Based on Improved Yolov5." Agriculture, vol. 13, no. 1, 2023, p. 124.

[7] Muhammad, Nur et al. "Evaluation of Cnn, Alexnet and Googlenet for Fruit Recognition." Indonesian Journal of Electrical Engineering and Computer Science, vol. 12, 2018, pp. 468-475.

[8] Raskar, Soham. "Enhancing Agricultural Sustainability: Automated Crop Disease Detection through Image Processing Techniques." International Journal for Research in Applied Science and Engineering Technology, vol. 12, 2024, pp. 3925-3929.

[9] Lapates, Jovelin M. "Corn Crop Disease Detection Using Convolutional Neural Network (CNN) to Support Smart Agricultural Farming," International Journal of Engineering Trends and Technology, vol. 72, no. 6, 2024, pp. 195-203.

[10] S, Deepika et al. "Advancements in Agricultural Technology: A Comprehensive Review of Machine Learning and Deep Learning Approaches for Crop Management and Disease Detection." International Journal of Advanced Research in Science, Communication and Technology, 2024, pp. 111-120.

[11] Mimenbayeva, Aigul et al. "Applying Machine Learning for Analysis and Forecasting of Agricultural Crop Yields." Scientific Journal of Astana IT University, 2024, pp. 28-42.

[12] Virani, VB et al. "Machine Learning-Based Comparative Analysis of Weather-Driven Rice and Sugarcane Yield Forecasting Models." ORYZA-An International Journal of Rice, vol. 61, no. 2, 2024, pp. 150-159.

[13] Ren, S. et al. "Faster R-Cnn: Towards Real-Time Object Detection with Region Proposal Networks." IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 6, 2017, pp. 1137-1149.

[14] Purkait, Pulak et al. "Spp-Net: Deep Absolute Pose Regression with Synthetic Views." ArXiv, vol. abs/1712.03452, 2017.

[15] Liu, Wei et al. "Ssd: Single Shot Multibox Detector." Computer Vision – ECCV 2016, Translated by Bastian Leibe et al., Springer International Publishing, 2016, pp. 21-37.

[16] Terven, Juan R. et al. "A Comprehensive Review of Yolo Architectures in Computer Vision: From Yolov1 to Yolov8 and Yolo-Nas." Mach. Learn. Knowl. Extr., vol. 5, 2023, pp. 1680-1716.

[17] Redmon, Joseph et al. "You Only Look Once: Unified, Real-Time Object Detection." 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 779-788.

[18] Zhou, Xingyi et al. "Objects as Points." ArXiv, vol. abs/1904.07850, 2019.

[19] Wu, Xinyu et al. "A Lightweight Grape Detection Model in Natural Environments Based on an Enhanced Yolov8 Framework." Frontiers in Plant Science, vol. 15, 2024.

[20] Behera, Santi Kumari et al. "Fruits Yield Estimation Using Faster R-Cnn with Miou." Multimedia Tools and Applications, vol. 80, no. 12, 2021, pp. 19043-19056.

[21] Aguiar, André Silva et al. "Grape Bunch Detection at Different Growth Stages Using Deep Learning Quantized Models." Agronomy, vol. 11, no. 9, 2021, p. 1890.

[22] Pereira, Carlos S. et al. "Deep Learning Techniques for Grape Plant Species Identification in Natural Images." Sensors, vol. 19, no. 22, 2019, p. 4850.

[23] Shuai Rong, Xinghai Kong Ruibo Gao Zhiwei Hu and Yang Hua. "Grape Cluster Detection Based on Spatial-to-Depth Convolution and Attention Mechanism." Systems Science \& Control Engineering, vol. 12, no. 1, 2024, p. 2295949.

[24] Marani, Roberto et al. "Deep Neural Networks for Grape Bunch Segmentation in Natural Images from a Consumer-Grade Camera." Precision Agriculture, vol. 22, 2020, pp. 387-413.

[25] Sozzi, Marco et al. "Grape Yield Spatial Variability Assessment Using Yolov4 Object Detection Algorithm." Precision Agriculture'21, Wageningen Academic Publishers, 2021, pp. 193-198.

[26] Huipeng, Li et al. "A Real-Time Table Grape Detection Method Based on Improved Yolov4-Tiny Network in Complex Background." Biosystems Engineering, vol. 212, 2021, pp. 347-359.

[27] Gebru, Timnit et al. "Datasheets for Datasets." Communications of the ACM, vol. 64, 2018, pp. 86-92.

[28] Tzutalin, D. (2022). LabelImg is a graphical image annotation tool and label object bounding boxes in images. URL https://github.com/tzutalin/labelImg.

[29] Elfwing, Stefan et al. "Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning." Neural networks, vol. 107, 2018, pp. 3-11.

[30] Glorot, Xavier et al. "Deep Sparse Rectifier Neural Networks." Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Translated by Gordon Geoffrey et al., vol. 15, PMLR, 2011, pp. 315-323.

[31] Xu, Bing et al. "Empirical Evaluation of Rectified Activations in Convolutional Network." ArXiv, vol. abs/1505.00853, 2015.

[32] Paszke, Adam et al. "Pytorch: An Imperative Style, High-Performance Deep Learning Library." Proceedings of the 33rd International Conference on Neural Information Processing Systems, Curran Associates Inc., 2019, pp. 8026-8037.

[33] Bastian Leibe et al. "Ssd: Single Shot Multibox Detector." Springer International Publishing, Computer Vision – ECCV 2016, 2016, pp. 21-37.

[34] Tian, Z. et al. "Fcos: Fully Convolutional One-Stage Object Detection." 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 9626-9635.

[35] Tan, Mingxing et al. "Efficientdet: Scalable and Efficient Object Detection." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10781-10790.

[36] Wang, Chien-Yao et al. "Yolov7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors." 2023, pp. 7464-7475.

[37] Selvaraju, Ramprasaath R. et al. "Grad-Cam: Visual Explanations from Deep Networks Via Gradient-Based Localization." 2017, pp. 618-626.

[38] Sivasubramanian, Arrun et al. "Object Detection under Low-Lighting Conditions Using Deep Learning Architectures: A Comparative Study." Advances in Data Science and Computing Technologies, 2023, pp. 269-276.

[39] Hert, Daniel et al. "Mrs Drone: A Modular Platform for Real-World Deployment of Aerial Multi-Robot Systems." Journal of Intelligent & Robotic Systems, vol. 108, no. 4, 2023, p. 64.

[40] Alibabaei, Khadijeh et al. "Real-Time Detection of Vine Trunk for Robot Localization Using Deep Learning Models Developed for Edge Tpu Devices." Future Internet, vol. 14, 2022, p. 199.