Exploring Enhanced Object Detection and Classification Methods for Alstroemeria Genus Morado

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

***Abstract*—As an important ornamental plant, the automatic detection and classification of the maturity of Alstroemeria Genus Morado flowers hold significant importance in precision agriculture. However, this task faces numerous challenges due to the diversity of morphological characteristics, complex growth environments, and factors such as occlusion and lighting variations. Currently, this field is relatively unexplored, necessitating innovative methods to overcome existing difficulties. To fill this research gap, this study developed a deep learning-based object detection framework, the Alstroemeria Genus Morado Network (AGMNet), specifically optimized for the detection and classification of Alstroemeria Genus Morado flowers. This convolutional neural network utilizes multi-scale feature fusion techniques and spatial attention mechanisms, along with a dual-path detection structure, significantly enhancing its capability for automatic maturity classification and detection of flowers. Notably, AGMNet addresses the issue of class imbalance in its design and employs advanced data augmentation techniques to enhance the model's generalization ability. In comparative experiments on the morado_5may dataset, AGMNet demonstrated superior performance in Precision, Recall, and F1-score, with a 3.8% improvement in the mAP metric over the latest YOLOv9 model, showcasing stronger generalization capabilities. AGMNet is expected to play a more significant role in enhancing agricultural production efficiency and automation levels.**

Keywords*—Alstroemeria; object detection; maturity classification; multi-scale feature fusion; Convolutional Neural Network (CNN)

I. INTRODUCTION

In modern agricultural production, the importance of precision agriculture technology is increasingly highlighted, with object detection and classification becoming one of the key technologies. Alstroemeria Genus Morado, as a flower with unique morphological characteristics and ornamental value, is crucial for determining the optimal harvest time based on flower maturity. In South America, particularly in Chile and Brazil, there is a high diversity of species [1] [2]. Despite the economic and ecological value of Alstroemeria Genus Morado, research on the automated detection and classification of its flowers is still insufficient. Traditional manual detection methods are not only inefficient but also costly, with accuracy and consistency of classification results being difficult to guarantee, making them unsuitable for large-scale production needs. Fortunately, with the development of computer vision and deep learning technologies, automated object detection and classification offer new possibilities for addressing this issue [3]. Through object detection and classification technology, these species can be more accurately identified and assessed, providing support for the study and conservation of biodiversity.

In recent years, the rise of deep learning technology has brought new breakthroughs in the field of object detection and classification [4] [5], capable of automatically learning feature representations from a large amount of data, thereby reducing the reliance on manual feature extraction. By constructing deep neural network models and training them with large-scale annotated data, deep learning models can automatically learn and extract feature representations of objects, achieving efficient object detection and classification. In the field of deep learning object detection, there are mainly two types of methods. The first category is two-stage object detection algorithms, such as Region-based Convolutional Neural Network (R-CNN) [6], Faster R-CNN [7], and Spatial Pyramid Pooling Network (SPP-Net) [8]. These algorithms typically have higher detection accuracy but are slower in detection speed due to their two-stage nature. In contrast, the second category is single-stage object detection algorithms, which have faster detection speeds, such as the You Only Look Once (YOLO) series [9] and CenterNet [10], although they may make slight sacrifices in accuracy. Since detection speed is highly required in most tasks, single-stage algorithms have more advantages in practical applications.

However, despite the significant achievements of deep learning in general object detection, there are still many challenges when dealing with flower varieties with specific morphological characteristics and growth environments. Especially for flower varieties with unique shapes and growth characteristics, such as Alstroemeria Genus Morado, the diversity of morphological characteristics, complex growth environments, and potential interference factors such as occlusion and lighting changes still pose generalization challenges in detecting Alstroemeria flowers, making existing research insufficient. The study by Stan Zwinkels & Ted de Vries Lentsch on the detection of mature Alstroemeria Genus Morado flowers [11] demonstrated the feasibility of this detection method by creating an experimental dataset and designing a detection algorithm, achieving an F1-score of over 0.75 in experiments. In addition, the study of Alstroemeria pollen morphology [12] provided a foundation for later researchers to understand its morphological characteristics,

*Corresponding Author

which is helpful in developing more accurate detection algorithms. Aros et al. [13] discussed the seed characteristics and evaluation of pre-germination treatment of Alstroemeria.

To fill this research gap, there is an urgent need to develop an efficient and accurate object detection and classification method suitable for Alstroemeria Genus Morado. This study proposes a new deep learning-based object detection framework, specifically optimized for the detection of Alstroemeria Genus Morado flowers. A series of innovative technologies have been used to enhance detection performance and accuracy. The key design of the Encoder strengthens the representation of image features, and the spatial attention mechanism enhances the focus on important areas of the image. At the same time, a dual-path detection structure, combined with the main detection neck and auxiliary branch, enhances the detection capability for targets of different sizes through multi-scale feature fusion technology. In particular, the introduction of the SPPELAN module [14] and the DySample layer [15] allows AGMNet to expand the size of the feature map and fuse it with the feature maps in the Encoder, capturing context information at different levels and achieving deep, multi-scale feature extraction of the image. Finally, the Detect layer synthesizes these advanced features to output accurate detection results, making AGMNet perform well in object detection tasks in agricultural scenarios. At the same time, this comprehensive classification method enables more accurate judgment of flower maturity. To fully evaluate the performance of the model, this study selected the morado_5may dataset [16] for experiments, verified the effectiveness of the proposed method, and compared it with existing technologies, successfully overcoming the challenges brought about by the diverse morphological characteristics, complex growth environments, and potential interference factors such as occlusion and lighting changes of Alstroemeria Genus Morado. The experimental results show that the proposed AGMNet performs excellently in both performance and efficiency, superior to other computer vision methods, and has sufficient generalization.

This paper aims to address some key issues and make the following contributions as follows:

- Proposing an efficient and accurate deep learning framework for object detection and classification methods suitable for Alstroemeria Genus Morado.
- Developing a comprehensive classification method capable of accurately judging the maturity of flowers.
- Validating the effectiveness of the proposed method through a series of experiments and comparing it with existing technologies. At the same time, providing a reference for the detection and classification of other plant species.

The rest of this paper is organized as follows: Section II introduces the model design in detail. Section III provides experimental details and results. Section IV discusses and analyzes the research results in depth. Section V summarizes the paper and proposes future research directions.

II. Materials and Methods

This section provides a detailed description of the dataset utilized in the study and an explanation of the AGMNet model's design principles, structural features, and optimization techniques, while emphasizing the innovative elements of the design.

A. Datasets

To verify the proposed method, the study conducted validation on the publicly available morado_5may dataset [16], which is a dataset for object detection tasks. It was photographed and released by Delft University of Technology and Hoogenboom Alstroemeria in the greenhouse of Hoogenboom Alstroemeria company around 12 PM on May 5, 2021. The images of the dataset were taken with an iPhone 8, using a 12-megapixel camera, with a pixel resolution of 4,032 × 3,024, taken from an overhead perspective about 1.5 meters above the flower bed. The entire dataset consists of 414 images and 5,439 labeled objects, belonging to two different categories, including raw and ripe, with all images in the dataset having bounding box annotation labels. It should be noted that there is no predefined training and testing split within the dataset. A random selection method was used to divide the 414 images into training and testing sets in an approximate 8:2 ratio, which were then stored in corresponding folders, constituting the morado_5may dataset used in this research. Detailed information about the dataset is shown in Table I.

TABLE I. Detailed Information of the Dataset

Dataset	Image	Label		
		Total	***Raw***	***Ripe***
Total	414	5,439	4,679	760
Training	324	4,191	3,655	536
Test	90	1,248	1,024	224

Further, to objectively assess the model's classification capabilities, it is essential to understand the rules for category division within the dataset. The maturity classification of each flower is based on factors such as color, color uniformity, size, and the number of buds. If a flower has several buds that have begun to open, the buds are relatively large, and the color is bright purple, then it is considered ripe. These guidelines are established to help others identify incorrectly classified flowers. The complete buds are bright purple, with no yellow parts in the middle. A flower contains multiple buds that have begun to open. The buds of this flower are larger than those of other flowers. Classification example images are shown in Fig. 1.

It is worth noting that this dataset is challenging. Firstly, the issue of class imbalance is prominent because the number of immature raw flowers in the images far exceeds that of ripe flowers, leading to a model that may be biased towards predicting the more common category. Secondly, the stems and leaves of the flowers have a high color similarity, and the flowers at the lower positions are easily occluded by leaves, making the recognition and classification of the flowers more difficult. In addition, the imaging morphology of Alstroemeria

flowers is highly variable, and the uncommon flowering forms bring a test to the model's generalization capabilities. In Fig. 2, these challenges are depicted.

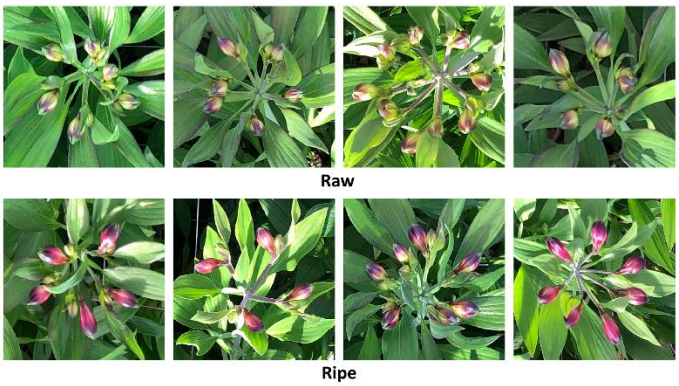

Fig. 1. Classification example images of the morado_5may dataset.

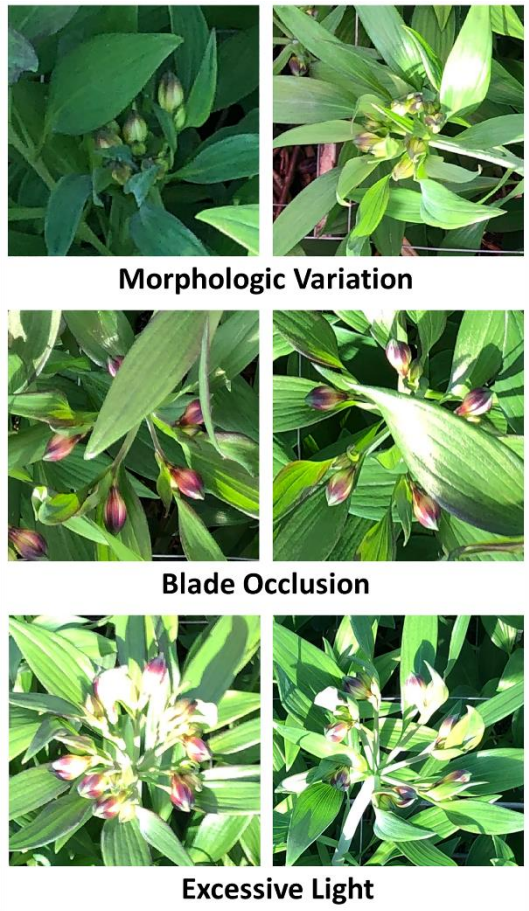

Fig. 2. The six main challenges of the morado_5may dataset.

In summary, by validating on the morado_5may dataset, the universality and effectiveness of the proposed method can be comprehensively evaluated.

B. Model Construction

When applying neural networks in the agricultural field, there are many factors to consider, mainly from external factors in the field. To address these challenges, an innovative deep learning-based object detection model, the Alstroemeria Genus Morado Network (AGMNet), was proposed in this study. The overall network structure is primarily composed of three parts: the Encoder, the Decoder, and the Head network. It also employs a dual-path detection structure [14] to mitigate the issue of information loss due to network depth. The model structure is depicted in Fig. 3, and the subsequent sections will detail their configuration.

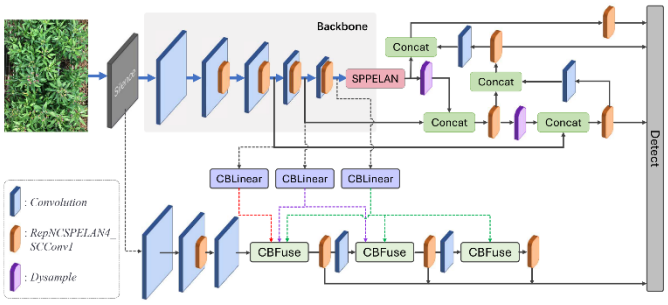

Fig. 3. Architecture of AGMNet.

1) Encoder Design Details: In the architecture of AGMNet, the Encoder serves as the main body of the model, responsible for extracting deep features of the image. Given an input image $I \in R^{H\times W\times 3}$, the Silence module, located at the forefront of the network structure, does not perform substantial operations. It is designed to retain the original image features to provide them to the main and auxiliary detection Decoders in the Neck for object detection. Subsequently, to reduce the spatial dimensions of the feature maps and increase the number of channels, multiple convolutional layers with a kernel size of 3×3 and a stride of 2 are used, halving the spatial dimensions of the image while increasing the feature depth. This convolution operation can be defined as Eq. (1):

$$X_{out} = \sigma(\sum_{i=1}^{N}(X_{in} * W_i + b_i)) \quad (1)$$

where, σ represents the activation function, $*$ denotes the convolution operation, W is the convolution kernel, b is the bias term, and X_{out} is the output feature map. When the input image size is H_{in}×W_{in}, the convolution kernel size is F×F, the padding is P, and the stride is S, the output feature map size can be calculated using the following Eq. (2) and Eq. (3):

$$H_{out} = \left\lfloor \frac{H_{in} + 2P - F}{S} \right\rfloor + 1 \quad (2)$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2P - F}{S} \right\rfloor + 1 \quad (3)$$

Further, AGMNet designed a RepNCSPELAN4SCConv1 module to extract features and enhance the feature representation capability. The specific module design is as follows: first, a 1×1 convolutional layer Conv reduces the number of channels in the input feature map from c1 to c3. Then, the feature map goes through two consecutive RepNCSP_SCConv modules, each containing a standard convolutional layer and an SCConv attention layer [17], along with a residual connection. These modules further transform and refine the channel number from c3 to c4. Finally, the original c3 output is concatenated with the outputs of the two

RepNCSP_SCConv modules (a total of 2×c4) on the channel dimension to form a richer feature representation. The concatenated feature map is then passed through a final 1×1 convolutional layer Conv4, converting the number of channels to the final output c2. Throughout this process, the SCConv layer uses a combination of average pooling and convolutional operations to implement a spatial attention mechanism, which helps the model to focus more on important areas of the image, thereby enhancing detection performance.

In summary, by stacking multiple convolutional layers, activation functions, and downsampling layers to gradually extract multi-scale features of the image, the entire Encoder consists of five convolutional layers and four feature extraction layers, specifically defined as C3(2)-C3(2)-R-C3(2)-R-C3(2)-R-C3(2)-R, where Ck(m) represents a two-dimensional convolutional layer with a k×k kernel size and a stride of m, and R is a feature extraction layer. After the transformation by the Encoder, the input image will complete a 32x downsampling, and the feature map is reduced to 1/32 of the original image size, outputting multiple feature maps of different depths that are used in the Decoder. Such Encoder design helps to improve object detection performance, especially when dealing with occlusions and small targets.

2) Decoder design details: In the design of AGMNet, the Decoder part employs multi-scale feature fusion technology and is designed with a Main Branch and an Auxiliary Branch to perform object detection simultaneously. This design enhances the model's ability to detect targets of different scales, improves feature expression capabilities, and allows for more effective information flow between different network layers.

In the Main Branch, the SPPELAN module [14] is first used to receive the high-dimensional feature maps output by the Encoder to enhance the receptive field and extract multi-scale features. Then, a DySample layer is connected to perform dynamic upsampling of the feature map, expanding the size to facilitate fusion with larger feature maps. This fusion operation is implemented through a Concat module, which concatenates the upsampled feature map with a feature map of the same size from the Encoder in the depth direction, forming a new feature map that integrates information from different levels. This fusion strategy helps the model capture context information at different levels and improves the model's ability to detect multi-scale targets. Similarly, the same convolutional layers as in the Encoder are used to perform downsampling operations again. Next, the RepNCSPELAN4SCConv1 layer is used again to perform convolutional operations on the fused features, reducing the number of convolutional kernel parameters.

In the Auxiliary Branch, the CBLinear layer is first used to extract features from the 8x, 16x, and 32x downsampling layers of the Encoder, transforming these features from different levels to match the required number of channels. The transformation operation can be expressed as Eq. (4) to Eq. (6):

$$F_{cb}^{8x} = \text{XBΛινεαρ}(F_{enc}^{8x}; C_{out}) \quad (4)$$

$$F_{cb}^{16x} = \text{XBΛινεαρ}(F_{enc}^{16x}; C_{out}) \quad (5)$$

$$F_{cb}^{32x} = \text{XBΛινεαρ}(F_{enc}^{32x}; C_{out}) \quad (6)$$

where, $F_{enc}^{8x}, F_{enc}^{16x}, F_{enc}^{32x}$ represent feature maps at different scales, and C_{out} is the target channel number. Then, feature fusion technology is used, and after each downsampling operation, the CBFuse layer fuses the output of the auxiliary branch with the feature map of the main branch. This fusion operation combines feature information from different levels, further enhancing the feature expression capabilities. Specifically, AGMNet adopts a specific fusion strategy based on the level and channel number of the feature map to ensure that the fused feature map retains the key information of the original features and introduces new contextual information. This further improves the model's detection performance, especially when dealing with small and blurred targets. The auxiliary branch, through parallel processing of additional feature maps, can capture information that the main detection branch may miss.

The design of the Decoder effectively utilizes the multi-scale features extracted by the Encoder and further enhances the feature expression capabilities through feature fusion and convolutional operations. This design allows the model to more accurately detect targets of different sizes, giving AGMNet an advantage in the object detection and classification tasks of Alstroemeria. Finally, the Detect layer receives feature maps of different scales and generates the final detection results.

C. Activation Function

In deep learning, the role of activation functions in neural networks is to introduce nonlinearity, allowing the network to model complex functions. Different activation functions have different mathematical properties and computational efficiencies. Commonly used activation functions include Sigmoid-weighted Linear Unit (SiLU) [18], Rectified Linear Unit (ReLU) [19], and Exponential Linear Unit (ELU) [20], etc.

The Sigmoid function maps any real number to the interval (0, 1), defined by the Eq. (7):

$$\text{Σιγμοιδ}(x) = \frac{1}{1+e^{-x}} \quad (7)$$

SiLU dynamically adjusts the scaling of the input x through the output of the sigmoid function, retaining the linear part of the input information while introducing nonlinearity. Its output range is limited, which helps to avoid the vanishing gradient problem and, to some extent, prevents the "dead neuron" issue. The ReLU activation function is a more concise nonlinear function, defined by the Eq. (8):

$$ReLU(x) = max(0, x) \quad (8)$$

ReLU has a gradient of 1 when the input is positive, effectively alleviating the vanishing gradient problem and has high computational efficiency. However, ReLU has a gradient of 0 when the input is negative, which can lead to some neurons never being activated during the training process, also known as the "dead" phenomenon. Furthermore, the ELU function combines the characteristics of ReLU and Sigmoid, defined by the Eq. (9):

$$ELU(x) = \begin{cases} x & ,x > 0 \\ \alpha(e^x - 1), & x \leq 0 \end{cases} \quad (9)$$

where, α is a parameter that adjusts the gradient of negative input values, typically set to a small positive constant (e.g., 0.1 or 1.0). ELU has soft saturation for negative inputs, which can reduce the problem of dead neurons, but the computation is relatively complex.

Considering the specific needs of the AGMNet model, SiLU (Sigmoid Linear Unit) was selected as the activation function for the model. SiLU not only maintains the non-zero gradient characteristic of ELU in the negative value area but also avoids additional exponential operations and effectively prevents the dead ReLU issue, maintaining the continuity of the gradient. This makes SiLU more effective in dealing with complex nonlinear relationships, helping the model capture more refined feature representations and thereby enhancing the performance of object detection.

III. EXPERIMENTS

In this section, the evaluation metrics and experimental details are first introduced. Subsequently, the performance will be reported, and the proposed AGMNet model will be compared with existing methods. The annotated data was statistically analyzed, and the model's performance was comprehensively assessed using common evaluation metrics, with visualization techniques employed to display and analyze the model's detection results.

A. Experimental Conditions and Details

In this study, the publicly available morado_5may dataset was selected for experimental validation. To ensure the accuracy and reliability of the experiments, the experimental conditions were meticulously set, and the details were refined. During the model training process, special attention was given to the selection of the loss function, the configuration of the optimization algorithm, and the adjustment of hyperparameters. To enhance the model's generalization capability, data augmentation techniques such as random scaling, rotation, and color transformation were implemented. Mini-batch stochastic gradient descent (SGD) was used as the optimizer to avoid the computational resource waste caused by calculating the gradients of the entire dataset. The initial learning rate was set to 0.01, with a batch size of 4 and a momentum factor of 0.937. Considering the convergence, the model was trained for 300 epochs, which allowed it to reach a state of convergence.

The experiments were conducted on a machine equipped with an NVIDIA GeForce GTX 3090 GPU, using the PyTorch 2.0.0 deep learning framework [21] for model training and evaluation, with the CUDA version 11.8 parallel computing framework and the CUDNN version 8.9.5 deep neural network acceleration library to fully utilize the parallel computing capabilities of the GPU. These experimental conditions and details ensure that the model can fully learn the characteristics of the dataset and provide an objective evaluation and accurate comparison of the model's performance. Moving forward, comparative experiments will be conducted to validate the effectiveness of the proposed method, and a thorough exploration will be made regarding its potential and value in practical applications.

B. Comparison of Model Performance with Different Object Detection Methods

In this study, to comprehensively evaluate the performance of the proposed object detection model, multiple evaluation metrics were selected for comparison with benchmark models on four public datasets. These benchmark models include YOLOv5 [22], YOLOv8 [23], and YOLOv9 [14]. AGMNet was trained and tested under the same experimental conditions as these benchmark models and evaluated based on indicators such as Precision (P), Recall (R), F1-score (F1), and mean Average Precision (mAP).

Precision (P) represents the proportion of objects correctly predicted by the model out of all predicted objects, Recall (R) represents the proportion of objects correctly predicted by the model out of all actual objects, and F1-score (F1) is the harmonic mean of Precision and Recall, providing a balanced perspective of the model's accuracy and recall rate. mAP is the average of the average precision over multiple different IoU thresholds, which can more comprehensively evaluate the model's performance under different thresholds and is an important performance indicator. Specifically, mAP@0.5 and mAP@0.5:0.95 represent the mAP values at an IoU threshold of 0.5, and the average mAP value as the IoU threshold changes from 0.5 to 0.95 (with a step size of 0.05), with the latter being a more stringent assessment of performance. Their definitions are as Eq. (10) to Eq. (13):

$$P = \frac{TP}{TP+FP} \tag{10}$$

$$R = \frac{TP}{TP+FN} \tag{11}$$

$$F1 = 2 \times \frac{P \times R}{P+R} \tag{12}$$

$$mAP = \frac{1}{n}\sum_{1}^{n} P(R)d(R) \tag{13}$$

where, True Positives (TP), False Positives (FP), and False Negatives (FN) represent the number of true positives, false positives, and false negatives, respectively. "TP + FP" is the total number of objects detected by the model, and "TP + FN" is the total number of actual objects in the image. As shown in Table II, the performance of each model in the four datasets is displayed.

TABLE II. PERFORMANCE EVALUATION RESULTS OF DIFFERENT MODELS

Model	P	R	F1	mAP@0.5	mAP@0.5:0.95
YOLOv5	0.725	0.722	0.723	0.754	0.530
YOLOv8	0.715	0.755	0.734	0.762	0.564
YOLOv9	0.704	0.802	0.750	0.788	0.630
AGMNet	**0.737**[a]	**0.807**	**0.770**	**0.826**	**0.637**

[a] Optimal performance is indicated in bold.

Through experimentation, the performance of these models was compared and analyzed on evaluation metrics such as Precision (P), Recall (R), F1-score (F1), and mean Average Precision (mAP). It is evident that the AGMNet model achieved the best performance across all assessment metrics. AGMNet reached an F1-score of 0.770, indicating that

AGMNet can effectively detect most real targets while maintaining high precision and recall rates. In addition, although YOLOv9 achieved a recall rate of 0.802, its precision was slightly lower, resulting in an F1-score slightly lower than AGMNet. The performance of the YOLO series models in mAP@0.5 and mAP@0.5:0.95 also did not surpass AGMNet, which means that AGMNet has stronger generalization capabilities under different IoU thresholds and can maintain high detection accuracy, especially within the more stringent IoU threshold range. It is worth noting that in the article by the author of the morado_5may dataset [11], the experimental model achieved an F1 result of 0.755, which shows that the performance of the AGMNet model has indeed been improved.

In the experiments, models such as CenterNet [10], Faster R-CNN [7], FCOS [24], and EfficientDet [25] were also tested. Their performance on the test dataset was notably poor, with an mAP@0.5 value not exceeding 0.3, significantly lower than the over 0.8 they could achieve on the training dataset. Although they showed a decreasing trend in loss values during training and ultimately reached a loss value of less than 1, they almost failed to successfully detect targets on the unseen test dataset. This phenomenon reveals their lack of generalization capabilities when dealing with datasets with complex backgrounds and more occlusions. Their performance dropped sharply when facing unseen target poses, occlusion situations, small targets, or background interference.

C. Comparison of Classification Performance with Different Object Detection Methods

After evaluating the performance of different models, further attention was given to their classification performance in the morado_5may dataset. This dataset contains two labels, raw and ripe, which represent unripe and ripe fruits, respectively. Similarly, metrics such as Precision (P), Recall (R), F1, and mean Average Precision (mAP) were used to comparatively assess them. The experimental results are shown in Table III.

TABLE III. CLASSIFICATION PERFORMANCE EVALUATION RESULTS OF DIFFERENT MODELS

Model	Class	P	R	F1	mAP@0.5	mAP@0.5:0.95
YOLOv5	Raw	0.733	0.784	0.758	0.778	0.522
	Ripe	**0.716**	0.661	0.687	0.729	0.538
YOLOv8	Raw	0.719	0.799	0.734	0.801	0.555
	Ripe	0.710	0.711	**0.757**	0.724	0.574
YOLOv9	Raw	0.733	**0.849**	0.787	0.829	0.627
	Ripe	0.675	0.754	0.712	0.748	0.632
AGMNet	Raw	**0.794**[a]	0.806	**0.800**	**0.857**	**0.630**
	Ripe	0.681	**0.809**	0.740	**0.795**	**0.645**

[a] The best performance for each category is indicated in bold.

For the raw category, the AGMNet model achieved the highest scores in Precision, Recall, and F1, indicating that AGMNet has higher accuracy and fewer missed detections when identifying unripe fruits. For the classification task of the ripe category, although AGMNet is slightly lower than YOLOv9 in Precision, it leads in Recall and F1, especially with a Recall of 0.809, showing AGMNet's higher recall rate when identifying ripe fruits. In addition, AGMNet also performed well in the mAP indicators, proving its overall performance superiority.

It is worth noting that AGMNet's performance in detecting unripe category flowers is particularly outstanding. Unripe flowers have greater difficulty in recognition because their characteristics are not as obvious as those of ripe flowers, and they are also smaller in size. These results further confirm the effectiveness of AGMNet in the tasks of object detection and classification of Alstroemeria Genus Morado. AGMNet, through its advanced network structure and optimization algorithms, can effectively handle the issue of class imbalance and achieve accurate classification in complex backgrounds, demonstrating stronger robustness.

D. Visualization of Typical Errors

In the task of object detection, missed detections and false detections are the two major issues affecting the model's performance. To delve into the causes of these errors, a visual investigation was conducted on the detection results of AGMNet and other benchmark models. During the evaluation process, a confidence threshold was carefully set to ensure optimal counting metrics on the dataset. This strategy helped to filter out the model's most confidentw detection results while excluding errors that might be brought by low-confidence predictions.

For missed detections, it was observed that these often occur when the target features are not distinct, the background is complex, or the target is occluded. In the visual results, blue arrows were used to point to these targets that were not detected. These targets may be due to their small size, high degree of integration with the background, or severe occlusion, making it difficult for the model to accurately capture their features. Differences in feature extraction and contextual understanding among different models also further affect the situation of missed detections. As for false detections, they usually occur when the model incorrectly identifies non-target objects as target categories. In the visualization images, yellow arrows point to these falsely detected targets. These errors may stem from the model's vague understanding of category boundaries or the issue of class imbalance in the dataset. When the model fails to fully learn the subtle differences between different categories during the training process, misclassification is likely to occur. In the detection of Alstroemeria Genus Morado, false detections may occur when plant structures that are similar in shape but not part of the target category are incorrectly classified as ripe or unripe flowers. To provide a clear illustration of these errors, Fig. 4 presents visual examples of typical missed (blue arrow) and false detected (yellow arrow) cases.

Through careful review of the object detection results, several typical error types and their potential causes were identified:

Fig. 4. Visualizing typical missed (blue arrow) and false detected (yellow arrow) cases.

Missed Detections: YOLO series models exhibit significant missed detection issues when detecting small, occluded, or targets with colors similar to the background. This problem can be attributed to the model's inability to fully capture the detailed information of these targets during the feature extraction phase, leading to their neglect in subsequent detection stages.

False Positives: On the other hand, false positives often occur when flowers are in the transitional phase between maturity and immaturity, making it difficult for the model to classify accurately. Additionally, imaging issues under strong light conditions can also cause the color features of mature flowers to distort, leading them to be misjudged as immature. These situations indicate that the model has limitations in dealing with detailed variations in color and shape.

Boundary Box Issues: In some detection results, it was noticed that targets with incomplete edges are easily ignored by the model. This may be due to the model's failure to fully consider the information in the edge areas when processing images, or because targets in these areas suffer loss during the feature extraction process. For example, in the case images of the YOLOv5 and YOLOv8 detection results, the target in the lower left corner was not detected. In contrast, YOLOv9 and AGMNet successfully addressed such issues.

In summary, the error analysis of AGMNet in object detection tasks indicates that the model has significant advantages in detecting small targets, occluded targets, and targets at the image edges, thanks to its innovative structure and algorithmic optimizations. These features of AGMNet give it important practical value in application scenarios such as precision agriculture, especially in object detection tasks that require high accuracy and robustness.

IV. Discussion

This paper introduces the AGMNet model for the object detection and classification task of Alstroemeria Genus Morado flowers, showcasing its superior performance. Comparative analysis has validated the model's advantages in object detection and classification. AGMNet's dual-path detection structure, featuring a main detection trunk and auxiliary branches, offers robust support for dealing with occlusions and multi-scale targets. This architecture not only bolsters the model's robustness but also demonstrates AGMNet's enhanced generalization across different IoU thresholds, particularly within stricter IoU ranges where its performance benefits are more evident. When compared to the YOLO series models, AGMNet has highlighted its potential and value in object detection tasks. The outcomes confirm AGMNet's practical application potential in precision agriculture, especially in scenarios demanding high accuracy and robustness. The introduction of AGMNet substantiates the efficacy of deep learning technology in precision agriculture and sets a foundation for subsequent research. Nevertheless, despite AGMNet's commendable performance in numerous instances, issues persist, such as missed detections when targets are heavily occluded or closely resemble the background in color. Additionally, false detections are prevalent during the transitional phase of flower maturation, suggesting that the model can improve in capturing nuanced variations in color and shape.

To counter these limitations, future efforts should concentrate on several fronts: the model requires further refinement to more adeptly manage occlusions and background interference. Constructing a more extensive dataset of Alstroemeria flowers, replete with detailed annotations, is essential. Developing a more lightweight model to meet real-time detection requirements will enhance the object detection model, improving its adaptability to targets across diverse environmental conditions. Future studies will also address more tangible needs in agricultural applications, offering effective technical support for plant disease and pest monitoring, plant population statistics, and ecological conservation.

V. Conclusion

This study aimed to address the insufficient object detection and classification performance of Alstroemeria Genus Morado flowers, filling a gap in this line of research. Innovatively, this study proposed the AGMNet model, which incorporates Encoder and Decoder structures. By applying a range of innovative technologies, including multi-scale feature fusion, spatial attention mechanisms, and dual-path detection structures, AGMNet has surpassed existing YOLO series models in key performance indicators, demonstrating

exceptional performance. Comprehensive experimental evaluations were conducted using the morado_5may dataset, and the results showed that compared to other benchmark models, AGMNet achieved higher levels in terms of precision, recall, and mAP metrics. However, despite the positive outcomes, there are still some issues that need to be further explored and resolved in future work. Specifically, addressing class imbalance, enhancing model generalization, improving computational efficiency, adapting to environmental changes, and creating larger-scale datasets are all key directions for the next phase of research. It is anticipated that through continued research, AGMNet can play a greater role in the field of precision agriculture, making a more significant contribution to the improvement of agricultural production efficiency and automation levels.

REFERENCES

[1] M. P. Bridgen, "Alstroemeria," in Ornamental Crops, J. Van Huylenbroeck Ed. Cham: Springer International Publishing, 2018, pp. 231-236.

[2] M. R. Dhiman and B. Kashyap, "Alstroemeria: conservation, characterization, and evaluation," in Floriculture and Ornamental Plants: Springer, 2022, pp. 117-151.

[3] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936-944.

[4] V. Finot, C. Baeza, E. Ruiz, O. Toro, and P. Carrasco, "Towards an integrative taxonomy of the genus Alstroemeria (Alstroemeriaceae) in Chile: a comprehensive review," Studies in Biodiversity, 2018, pp. 229-265.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, 2015, pp. 436-444.

[6] G. Gkioxari, B. Hariharan, R. B. Girshick, and J. Malik, "R-CNNs for Pose Estimation and Action Detection," arXiv preprint arXiv:1406.5212, 2014.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "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.

[8] P. Purkait, C. Zhao, and C. Zach, "SPP-Net: Deep Absolute Pose Regression with Synthetic Views," arXiv preprint arXiv:1712.03452, 2017.

[9] J. Terven and D. Cordova-Esparza, "A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond," arXiv preprint arXiv:2304.00501, 2023.

[10] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as Points," arXiv preprint arXiv:1904.07850, 2019.

[11] StanZwinkels. 2021. Detection of ripe flowers of the Alstroemeria genus Morado. Available at: https://stanzwinkels.medium.com/detection-of-ripe-flowers-of-the-alstroemeria-genus-morado-2028186f50af, accessed on 10 March 2023.

[12] A. K. M. G. Sarwar, Y. Hoshino, and H. Araki, "Pollen morphology and infrageneric classification of Alstroemeria L. (Alstroemeriaceae)," Grana, vol. 49, 2010, pp. 227-242.

[13] D. Aros, P. Barraza, Á. Peña-Neira, C. Mitsi, and R. Pertuzé, "Seed Characterization and Evaluation of Pre-Germinative Barriers in the Genus Alstroemeria (Alstroemeriaceae)," Seeds, vol. 2, no. 4, 2023, pp. 474-495.

[14] C.-Y. Wang, I.-H. Yeh, and H. Liao, "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information," arXiv preprint arXiv:2402.13616, 2024.

[15] W. Liu, H. Lu, H. Fu, and Z. Cao, "Learning to Upsample by Learning to Sample," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 6027-6037.

[16] Ted Lentsch. 2021. Available at: https://www.kaggle.com/datasets /teddevrieslentsch/morado-5may, accessed on 10 March 2024.

[17] J. Li, Y. Wen, and L. He, "Scconv: spatial and channel reconstruction convolution for feature redundancy," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 6153-6162.

[18] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," Neural networks, vol. 107, 2018, pp. 3-11.

[19] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in Proceedings of the fourteenth international conference on artificial intelligence and statistics, 2011: JMLR Workshop and Conference Proceedings, pp. 315-323.

[20] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," arXiv preprint arXiv:1511.07289, 2015.

[21] A. Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library," arXiv preprint arXiv:1904.07850, 2019.

[22] Jocher, G. 2020. YOLOv5 by Ultralytics (Version 7.0) [Computer software]. https://doi.org/10.5281/zenodo.3908559.

[23] Jocher G, Chaurasia A, and Qiu J. 2023. Ultralytics YOLO (Version 8.0.0) [Computer software]. Available at: https://github.com/ultralytics/ultralytics. accessed on 7 March 2023.

[24] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-Stage Object Detection," in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 27 Oct.-2 Nov. 2019, pp. 9626-9635.

[25] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10781-10790.