horticulturae

MDPI

Article

Research on a Flower Recognition Method Based on Masked Autoencoders

Yin Li [1,2,†], Yang Lv [3,†], Yuhang Ding [3], Haotian Zhu [1], Hua Gao [4,*] and Lifei Zheng [5,*]

[1] College of Information Engineering, Northwest A&F University, Xianyang 712100, China; ly@nwsuaf.edu.cn (Y.L.); zhtttkx@nwafu.edu.cn (H.Z.)
[2] Shaanxi Engineering Research Center of Agriculture Information Intelligent Perception and Analysis, Xianyang 712100, China
[3] College of Mechanical and Electronic Engineering, Northwest A&F University, Xianyang 712100, China; lvyang@nwafu.edu.cn (Y.L.); 3141790082@nwafu.edu.cn (Y.D.)
[4] College of Horticulture, Northwest A&F University, Xianyang 712100, China
[5] College of Science, Northwest A&F University, Xianyang 712100, China
* Correspondence: gaohua2378@163.com (H.G.); zhenglifei@nwsuaf.edu.cn (L.Z.)
† These authors contributed equally to this work.

Abstract: Accurate and efficient flower identification holds significant importance not only for the general public—who may use this information for educational, recreational, or conservation purposes—but also for professionals in fields such as botany, agriculture, and environmental science, where precise flower recognition can assist in biodiversity assessments, crop management, and ecological monitoring. In this study, we propose a novel flower recognition method utilizing a masked autoencoder, which leverages the power of self-supervised learning to enhance the model's feature extraction capabilities, resulting in improved classification performance with an accuracy of 99.6% on the Oxford 102 Flowers dataset. Consequently, we have developed a large-scale masked autoencoder pre-training model specifically tailored for flower identification. This approach allows the model to learn robust and discriminative features from a vast amount of unlabeled flower images, thereby enhancing its generalization ability for flower classification tasks. Our method has been applied successfully to flower target detection, achieving a Mean Average Precision (mAP) of 71.3%. This result underscores the versatility and effectiveness of our approach across various flower-related tasks, including both detection and recognition. Simultaneously, we have developed a straightforward, user-friendly flower recognition and classification software application, which offers convenient and reliable references for flower education, teaching, dataset annotation, and other uses.

Keywords: flower recognition; transfer learning; vision transformer; masked autoencoders; pre-training model

Citation: Li, Y.; Lv, Y.; Ding, Y.; Zhu, H.; Gao, H.; Zheng, L. Research on a Flower Recognition Method Based on Masked Autoencoders. *Horticulturae* **2024**, *10*, 517. https://doi.org/10.3390/horticulturae10050517

Academic Editor: Arturo Alvino

Received: 21 March 2024
Revised: 7 May 2024
Accepted: 10 May 2024
Published: 16 May 2024

1. Introduction

The rich and extensive history of flower culture in China underscores the profound significance of flowers in Chinese culture, symbolizing beauty, virtue, and the complex relationship between nature and humanity [1]. Aesthetic and Emotional Value: Flowers play a pivotal role in enhancing the environment and enriching human emotions, serving as a source of inspiration and joy in daily life and artistic expression. Many flowers are valued medicinally, as evidenced by clinical data and medical research, contributing significantly to the development of natural remedies and healthcare practices. Accurate identification of flowers is crucial both in plant science, for biodiversity conservation and ecological studies, and in the flower industry, for commercial cultivation and marketing. Accurate and efficient flower identification demands deep botanical knowledge and experience, posing significant challenges for both amateurs and professionals in distinguishing similar species. The accuracy of human identification based solely on experience and memory often

exhibits significant deviations, highlighting the need for more reliable and standardized methods in flower identification.

Automated technology for flower identification is highly accurate, rapid, straightforward, and user-friendly. Automated technology significantly enhances the efficiency of flower identification compared to traditional manual methods. These technologies serve as powerful tools for solving complex problems and boast wide applications across various fields. Computer vision and deep learning hold significant potential for development in areas such as face recognition, speech recognition, and image recognition. The advent of convolutional neural networks (CNNs) has significantly enhanced the efficiency and accuracy of flower identification in the realm of flower classification and recognition [2].

The models and technologies for flower identification have reached a mature stage, offering a variety of software solutions in the market that cater to diverse needs and applications. Despite advancements in flower identification, a significant research gap remains in the domain of flower target detection and recognition, underscoring the need for further exploration and development in this area. While CNNs have been instrumental in advancing image recognition tasks, they exhibit limitations in capturing long-range dependencies and integrating holistic features of the information, potentially affecting their performance in specific applications. Addressing the limitations of existing flower identification methods and enhancing their accuracy through the application of deep learning technology are critical research objectives for many scholars, aiming to boost the reliability and effectiveness of these systems. This paper introduces a novel flower identification method that leverages the capabilities of Masked Autoencoders (MAEs). This method is specifically engineered to address the challenges of flower classification and target detection, providing a more accurate and efficient approach to identifying various flower species. In recent years, there has been an increased focus on enhancing quality of life and promoting aesthetic appreciation. This shift in priorities underscores the importance of integrating nature and beauty into everyday life, with flowers playing a key role in this endeavor. The accurate classification and identification of flower images provided by this method can significantly benefit non-expert groups such as students and children, enabling them to understand and recognize different flower species with greater ease. Additionally, it can streamline the research process for plant experts, reducing the time and effort required for studying flower varieties and thereby facilitating their efforts in botanical research and conservation [3].

Currently, the field of flower target detection has received limited attention relative to other aspects of flower identification, underscoring the need for more focused studies and advancements in this domain. Practical Benefits: The accurate detection and identification of flowers, facilitated by advanced technologies, can significantly enhance the development of automated management and picking technology in the gardening sector. This can lead to improved efficiency and substantial economic benefits, contributing to the growth and sustainability of the industry.

The advancement of computer technology has propelled in-depth research in the field of plant species recognition, as the capabilities of computational tools and algorithms continue to evolve, enabling more sophisticated analysis and identification techniques. Despite growing interest, flower recognition and broader plant recognition remain niche research fields, with limited related literature available. This indicates a need for further exploration and publication in this area to expand the scope of knowledge. A significant barrier to the development of this research field is the scarcity of publicly available datasets, which are also limited in scale. The availability of comprehensive and diverse datasets is crucial for training and testing recognition models, and this scarcity impedes progress in the field. The process of collecting and organizing large-scale plant recognition and detection datasets is time-consuming, arduous, and demands professional expertise in plant identification. These challenges hinder the construction of robust datasets that can support the development of accurate and reliable recognition models [4]. There are two primary categories of methods for flower recognition: those based on manual features, involving

handcrafted descriptors and traditional machine learning techniques, and those based on deep learning, leveraging the power of neural networks to automatically learn features from data. Each approach has its own strengths and limitations, with the choice between them depending on the specific requirements of the recognition task.

1.1. Flower Recognition Methods Based on Manual Features

This approach employs artificially designed feature operators to extract specific features from image data for classification and recognition. These features, typically based on color, texture, shape, or a combination thereof, are carefully selected to capture the distinctive characteristics of the flowers. The process of extracting manual features is crucial in image classification and demands extensive professional knowledge in both the domain of the subject (e.g., botany) and image processing techniques. The quality of the extracted features directly impacts the accuracy of the classification and recognition results. In the context of flower recognition, traditional machine learning methods like support vector machines (SVM) and K-nearest neighbors (KNN) are commonly employed. SVM effectively finds the optimal separating hyperplane for different classes, while KNN, a straightforward yet potent algorithm, classifies objects based on the majority vote of their neighbors [5].

In their research, Das et al. implemented flower image classification utilizing color features and spatial domains [6]. Utilizing these features, they captured the unique color patterns and spatial arrangements of the flowers, enhancing the accuracy of their classification model. Nilsback and Zisserman [7] proposed a visual vocabulary for flower classification and developed an optimized nearest neighbor classifier. Their method achieved an accuracy of 71.76% for 17 flower types, demonstrating the effectiveness of their approach in distinguishing between different species. The Oxford-17 and Oxford-102 Flowers datasets, developed by Nilsback and Zisserman, have become widely used benchmarks in flower image recognition research. These datasets provide a diverse collection of flower images, enabling researchers to assess and compare the performance of various recognition methods [8]. In a recent study, Ke Xiao [9] introduced a novel method for flower recognition that combines morphological and texture features derived from the HSV color model. This approach aims to capture both the shape and texture characteristics of the flowers, thereby enhancing the discriminative power of the recognition system.

1.2. Deep Learning-Based Flower Recognition Methods

The rapid improvement in hardware performance, especially in GPUs (Graphics Processing Units), has significantly contributed to the widespread application of deep learning across various fields. These advancements have enabled the processing and analysis of large data volumes at unprecedented speeds, facilitating the development of more complex and sophisticated neural network architectures. Recently, numerous scholars have initiated explorations into the application of deep learning technology in the field of flower recognition [10]. This shift towards deep learning approaches has unlocked new possibilities for more accurate and efficient identification of flower species. Deep learning models such as AlexNet, VGGNet, GoogleNet, and ResNet have demonstrated remarkable success in image recognition challenges, including flower recognition tasks. These models have established new benchmarks for accuracy and have become foundational architectures in the field of computer vision [11].

In the context of flower recognition, Liu Shangwang and Gao Xiang [12] proposed a method utilizing deep model transfer learning for fine-grained image classification. This approach leverages the knowledge from pre-trained models to achieve more precise classification of flower species with subtle differences [13]. Wang Shuang [14] employed transfer learning to extract flower features from the AlexNet network trained on the ImageNet dataset. By adapting the pre-trained model to the specific task of flower recognition, this method achieves efficient feature extraction without extensive training from scratch [15]. Qin Min [3] combined attention mechanisms with CNNs and introduced Linear Discrim-

inant Analysis (LDA) to establish a classification model based on LD-Loss. This model aims to enhance the network's discriminative power by focusing on relevant features and reducing intraclass variability.

Despite advancements in deep learning for flower recognition, a notable gap remains in research on target detection and recognition of flowers. This area remains relatively unexplored, presenting opportunities for further investigation and development. Xie Zhouyi and Hu Yanrong [16] proposed a system utilizing the YOLOv4 architecture for multi-target flower recognition. This system is designed to simultaneously detect and classify multiple flowers within an image, showcasing the potential of deep learning for complex recognition tasks in natural settings.

In 2020, a team from Google introduced the Vision Transformer (ViT) model, which applies the Transformer architecture to image classification tasks. This groundbreaking effort was the first to introduce the Transformer architecture [17,18] into the realm of computer vision. Through a series of experiments, the authors demonstrated that the ViT model achieves higher accuracy in image classification than existing top CNN architectures after training on large datasets [19]. Unlike CNNs, the ViT model maintains greater similarity between representations obtained at shallow and deep layers, enabling it to leverage the advantages of Transformer architecture to learn deeper feature information on a global scale.

The introduction of the ViT model has enabled achievements in computer vision that are comparable to or even surpass those of CNNs, utilizing only Transformer architecture without convolutional layers. Even with small datasets, the ViT model can still yield strong experimental results by either loading pre-trained weights or utilizing a hybrid network structure that combines ResNet and ViT. This versatility demonstrates that the ViT model can achieve a performance level comparable to the top-performing CNNs while requiring fewer computing resources during the training process.

Due to its simplicity, effectiveness, and robust scalability, the ViT model is considered a milestone in the field of computer vision. It has garnered widespread attention and inspired numerous subsequent studies. For instance, Beal et al. combined the ViT and Faster RCNN models as the backbone network for feature extraction, achieving impressive results in classification and localization tasks for targeted regions [20]. Among these achievements, Alexey Dosovitskiy et al. [19] experimentally verified that the ViT-Large-I21k model achieved a classification accuracy of 99.60% on the Oxford-102 Flowers dataset, underscoring the model's potential for high-accuracy image classification tasks.

1.3. Current Research on MAEs

MAEs have emerged as a powerful technique in deep learning for understanding and analyzing data, enabling the extraction of effective features to solve complex problems. A key advantage of MAEs is their capability to train models on unlabeled data, making them highly adaptable for various downstream tasks. This flexibility is especially valuable in domains where labeled data are scarce or costly to obtain [21]. In the realm of scalable pre-training models, He, K. et al. introduced a groundbreaking MAE pre-training model based on the ViT architecture [22]. This model extends the success of BERT, a prominent language model, into the visual domain, demonstrating MAEs' potential to leverage Transformer-based architectures for visual tasks.

The versatility of MAEs is evident in their applicability to a diverse array of visual tasks across various fields, including image analysis, video processing, and audio recognition. This adaptability renders them a valuable tool for researchers and practitioners across various domains of computer vision and multimedia. A notable advancement in the application of MAEs is the dense retrieval pre-training model developed by Huawei Noah's Ark Lab. This model employs MAEs to achieve high precision in information retrieval tasks, showcasing the effectiveness of MAEs in enhancing search and retrieval capabilities. In the field of industrial inspection, Sun Jieguang [2] proposed a two-stage deep learning surface defect detection network utilizing MAE pre-training. This approach

underscores the practical utility of MAEs in addressing real-world challenges, such as identifying defects in manufacturing processes.

Overall, MAEs are distinguished by their broad applicability, simplicity, and versatility, encompassing various research fields and applications. Their use of Transformer-based models for pre-training enables them to achieve high-level data representations and learning capabilities, further solidifying their role as a cornerstone in the advancement of deep learning and artificial intelligence.

1.4. Main Contribution of Our Study

The overarching goal of this research is to advance the field of flower recognition by developing an autoencoder-based, pre-trained model that demonstrates enhanced precision in flower target detection and identification. In this study, we introduce an innovative approach to flower recognition by constructing a MAE pre-trained model, specifically developed for this purpose. Utilizing the Oxford-102 Flowers dataset, our model is distinguished by its adoption of parameters from the ViT, originally trained on the ImageNet-1K dataset. This strategic choice ensures a robust foundation for feature extraction and superior representation learning. Our methodology diverges from traditional CNN modifications, instead utilizing a novel integration of MAEs and the YOLOv5 algorithm for precise flower target detection.

The principal contributions of our study are twofold:

1. **Enhanced Target Detection:** Our approach focuses on flower target detection, not only identifying but also precisely localizing specific flowers within images. This capability is crucial for applications requiring high precision, such as botanical research and agricultural monitoring.
2. **Methodological Innovation:** We propose a combination of self-supervised pre-training using MAEs with YOLOv5-based detection, fostering a powerful synergy that enhances the accuracy and efficiency of flower detection.

These advancements represent a significant step forward in the field of flower recognition and establish a new benchmark for future research. Our findings offer valuable insights and a scalable model that can be adapted by other researchers and practitioners facing similar challenges in target detection and image recognition.

1.5. Chapter Structure of Our Study

This paper is structured to methodically explore the integration of MAEs and object detection models in flower recognition, detailing both theoretical foundations and practical applications. Section 1 reviews related work, dissecting methodologies ranging from manual feature-based to advanced deep learning-based flower recognition techniques, and provides an update on current research involving MAEs and outlines our approach, followed by Section 2, which delves into the materials and methods employed in pre-training the model. This section discusses the study framework, benchmark datasets, data preprocessing techniques, the architecture of the pre-trained model including MAE design and transfer learning, flower image reconstruction, and specific computational considerations such as the training environment and hyperparameter settings. Section 3 presents experimental results derived from the pre-training model.

Continuing, Section 4 describes the materials and methods used in developing the object detection model, highlighting data acquisition, processing, and the construction of the YOLOv5 model, including its design principles, loss function, and improvement strategies. The experimental setup and analysis are detailed in Section 4.3. Section 5 reports on the experimental results obtained from the object detection model. In Section 6, we discuss the implications of our findings, and Section 7 concludes this paper with a summary of our results and potential directions for future research.

2. Materials and Methods on the Pre-Training Model

2.1. Study Framework

This study introduces an advanced flower recognition model that employs the MAE approach to significantly enhance recognition accuracy. Our extensive review of existing literature revealed a critical need for effective flower image reconstruction to enhance recognition capabilities. The proposed model is distinguished by its dual-stage training regimen designed to maximize effectiveness. Initially, we employed the ViT-Large model, which was pre-trained on the ImageNet-1K dataset. This step leverages the robust feature-extraction capabilities of the Transformer architecture, establishing a solid foundation for advanced feature comprehension. Subsequently, in the second stage, sophisticated data augmentation techniques were applied to the Oxford-102 flower dataset. This strategy is crucial for enhancing the model's capacity to effectively generalize across a diverse range of floral images. The architectural framework of our model has been meticulously designed to facilitate precise flower image reconstruction and robust classification. It incorporates an encoder for detailed feature extraction and a decoder for reconstructing the flower images, ensuring that essential details are preserved. A critical component of the architecture is a fully connected layer, dedicated specifically to classifying the reconstructed images into their respective flower categories.

The interplay between the encoder and decoder enables the model to learn and encode meaningful representations of flower images, which are critical for accurate classification. Following pre-training, the model was fine-tuned to optimize its classification performance, focusing specifically on flower-specific features. Evaluation on the Oxford-102 flower dataset underscored the model's exceptional capabilities, demonstrating remarkably high classification accuracy. This model not only establishes a new benchmark in flower recognition but also highlights the potential of MAE approaches in complex image recognition tasks. The model's robust performance is confirmed across a spectrum of standard flower recognition tasks, demonstrating its broad applicability and effectiveness in diverse scenarios.

Figure 1 presents the detailed architecture of our approach, integrating an MAE specifically tuned for flower recognition. This architecture diagram effectively illustrates the progression from an initial comprehensive literature review to the strategic construction of a pre-trained network that leverages the capabilities of MAE, culminating in its application in advanced YOLOv5-based detection. It underscores the transformative potential of MAE to enhance recognition accuracy within the field. Further refinement of the model was achieved through the creation and utilization of a self-curated dataset, enriched by web crawling and comprehensive data augmentation techniques. This enriched dataset plays a vital role in advancing the performance of the object detection models, particularly YOLOv5, by providing a diverse range of image examples that improve the model's ability to generalize and detect objects accurately under varied conditions. The integration of these techniques establishes a solid foundation for the model's superior performance, demonstrating its effectiveness and versatility across various flower recognition tasks.

2.2. Benchmark Datasets

Computing power, algorithms, and data are indispensable elements underpinning the successful implementation of artificial intelligence tasks, including deep learning. Computing power, largely reliant on hardware such as GPUs and Tensor Processing Units (TPUs), is crucial for managing the computational demands of complex AI models. The quality of algorithms, however, is tied to the structure of the network model, which determines the efficiency and effectiveness of the learning process. The quality of data is crucial for the success of AI implementations. High-quality, well-labeled, and diverse datasets are essential for training robust and precise AI models. In the domain of flower recognition, several datasets are routinely used to evaluate the performance of AI models. The details of these datasets are summarized in the Table 1:

Table 1. Detailed Information on Three Flower Image Classification Datasets.

Dataset	Provider	Number of Images	Number of Species	Characteristics	Data Availability
Five Flower	Kaggle	3670	5	Contains 600–800 images per species: daisies, dandelions, roses, sunflowers, and tulips. Balanced for initial training.	Available at https://www.kaggle.com/datasets/kausthubkannan/5-flower-types-classification-dataset (accessed on 9 May 2024)
Oxford-17 Flowers	Visual Geometry Group, University of Oxford	1360	17	Approximately 80 photos per species. Useful for fine-grained recognition studies in natural settings.	Available at https://www.robots.ox.ac.uk/~vgg/data/flowers/17/ (accessed on 9 May 2024)
Oxford-102 Flowers	Visual Geometry Group, University of Oxford	8189	102	Comprehensive collection depicting a variety of postures and lighting conditions to test AI generalization in diverse species recognition.	Available at https://www.robots.ox.ac.uk/~vgg/data/flowers/102/ (accessed on 9 May 2024)

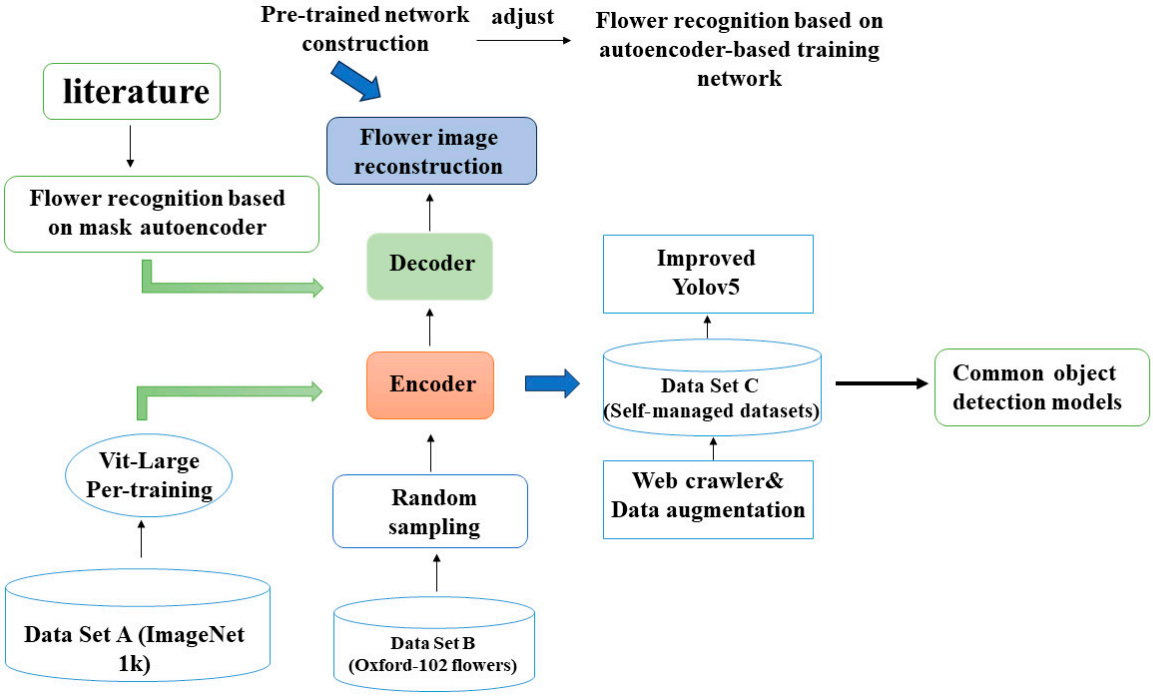

Figure 1. Technical roadmap of this article.

2.3. Benchmark Datasets Preprocessing

The impact of data on deep learning tasks cannot be overstated, as data quantity and quality are critical factors in determining the success of these tasks. Among the various datasets used for studying fine-grained flower images, the Oxford-102 Flowers dataset is distinguished as one of the largest and most diverse. It contains 102 species and a total of 8189 color images, offering a rich resource for researchers and practitioners in the field of computer vision and machine learning. The composition of the dataset is well-structured, with the training and validation sets each containing 1020 images, equating to 10 images per class. The test set is more extensive, comprising 6149 images, which facilitates a comprehensive evaluation of the performance of deep learning models trained on this dataset. An example image is depicted in Figure 2. Given the varying resolution and size of images in the dataset, resizing is an essential preprocessing step. This step is necessary to avoid issues during random cropping and to maintain noticeable feature differences

between samples. Uniform scaling of image size is crucial to ensure fairness in network comparison. Consequently, the image size is uniformly scaled to 224 × 224, aligned with sample sizes from various experimental datasets in image recognition. This standardization facilitates the comparison of different network architectures and their performance in the task of flower image classification.

Figure 2. Example images from the Oxford-102 flowers dataset.

2.4. Structure of the Pre-Train Model

2.4.1. MAE Design

Figure 3 depicts the specialized architecture of the MAE, highlighting its unique asymmetric encoder–decoder structure tailored for our application in flower image reconstruction. Unlike traditional autoencoders that often employ symmetrical dimensions for encoding and decoding, MAEs leverage an asymmetric architecture, where the encoder is designed to process a greater volume of data than the decoder. In this figure, the input image is divided into patches and partially masked, illustrating the model's capacity to handle incomplete data [18]. The encoder, a larger and more complex network component, compresses the visible patches into a compact latent representation, capturing the critical information while disregarding the masked sections. These encoded data are then processed by a relatively smaller decoder, tasked with reconstructing the full image, including the previously masked portions. The result is a reconstructed target image that is then evaluated against the original for fidelity, demonstrating the MAE's effectiveness in feature preservation and image recovery [23]. This asymmetric design is crucial for the MAE's efficiency and accuracy in tasks that require robust feature extraction from incomplete datasets, as exemplified by our flower recognition model.

2.4.2. Transfer Learning

Transfer learning is defined as the process of leveraging knowledge acquired from one task and applying it to a different, yet related task, after making necessary adjustments. This methodology encompasses fundamental concepts such as the domain, which includes the source domain (where the model is initially trained) and the target domain (where the model is subsequently applied), and the task, which refers to the specific problem being addressed. A notable challenge with the Oxford-102 Flowers dataset is the risk

of overfitting due to its limited number of labeled images. Directly training the MAE pre-training model on this dataset can result in a model that may not effectively generalize to new data. To mitigate this risk, transfer learning is employed to augment the dataset by leveraging knowledge from additional datasets, such as ImageNet-1K. Transfer learning is most effective when the data in the target domain resemble the original dataset.

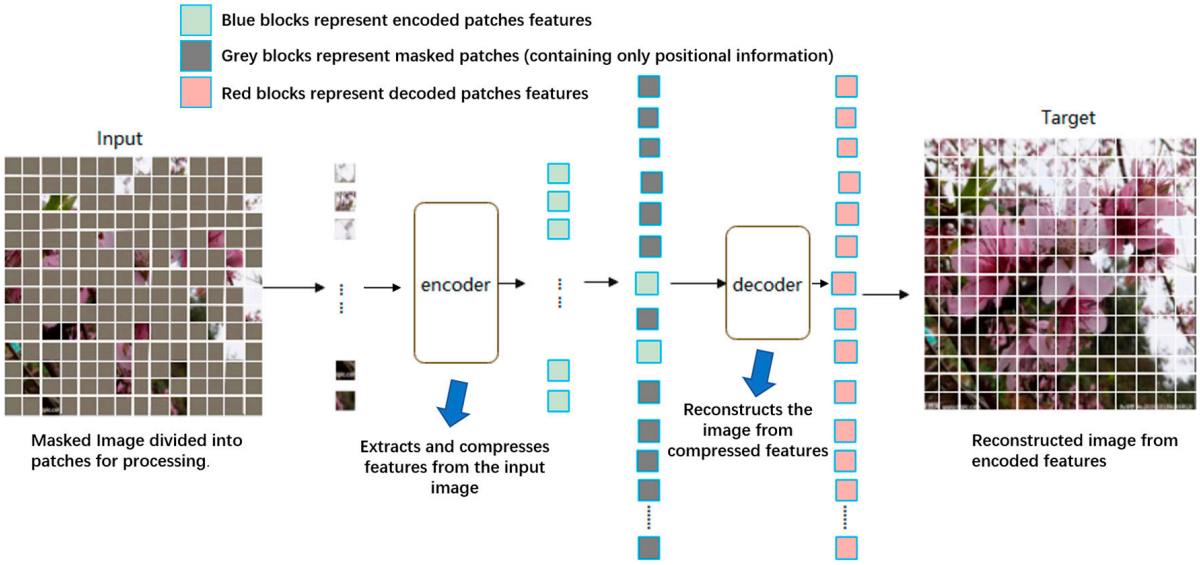

Figure 3. Structure of the MAE.

From the previous discussion of ViT, it is evident that ViT-Large has multiple advantages; therefore, we selected it as the base model for our pre-trained model. Figure 4 presents a visual representation of the architecture that utilizes ViT-Large as the foundational pre-trained model for our proposed system [24]. The ViT-Large model is integral to our framework, providing a high-capacity neural network pre-trained on the ImageNet-1K dataset, comprising 1000 classes and over a million images. This pre-training endows the model with a sophisticated understanding of visual representations, which is further adapted to our specific context of flower classification.

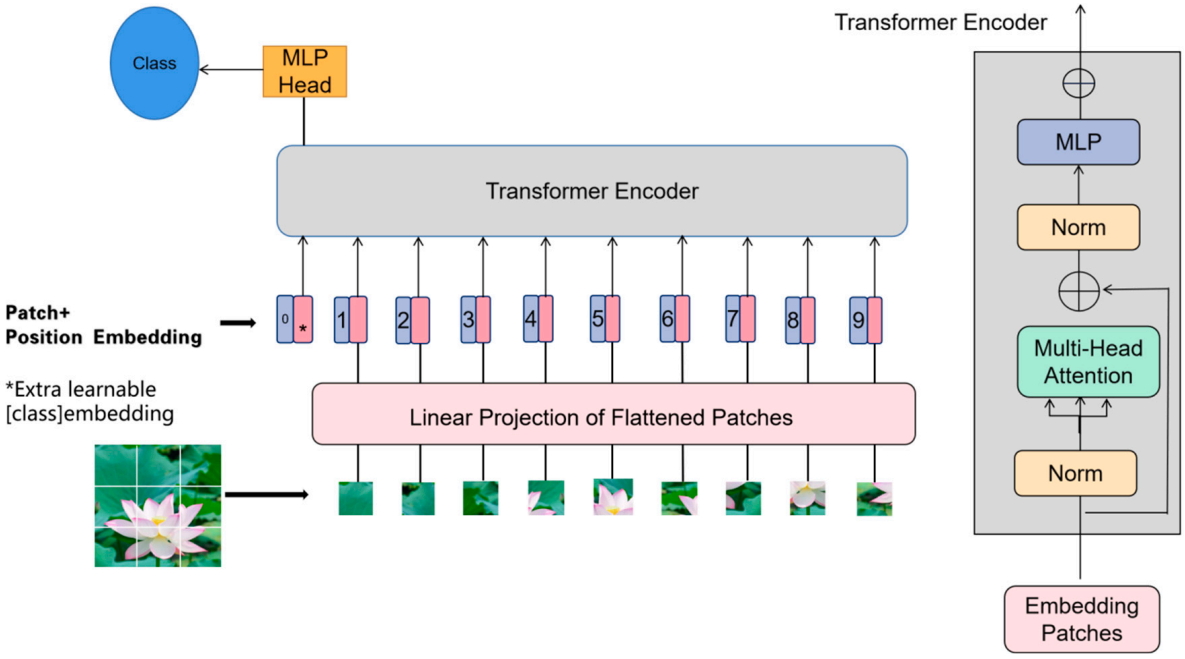

Figure 4. Schematic diagram of ViT model structure.

Within the ViT-Large model, an image is divided into a sequence of patches. These patches are then flattened and linearly projected to produce a series of embedding vectors. An additional learnable [class] embedding is appended to the sequence for classification purposes. This series, augmented with position embeddings to preserve locational information, is fed into the transformer encoder.

The transformer encoder comprises alternating layers of multi-headed self-attention and multilayer perceptrons (MLPs), with normalization layers interspersed between them [25]. This design allows the model to capture a rich hierarchy of features at various levels of abstraction. The output of the transformer encoder passes through an MLP head, which serves as a classifier, utilizing the [class] embedding to generate the final class predictions. The use of ViT-Large, with its extensive pre-training and high capacity for feature extraction, signifies a significant evolution from conventional convolutional approaches.

In the depicted process (Figure 5), parameter weights from the ViT-Large model, initially pre-trained on the extensive ImageNet-1K dataset, are extracted and strategically transferred to the Oxford-102 Flowers dataset. This transfer involves adapting the model parameters through an MAE-based pre-training approach, tailored specifically to enhance its relevance and performance in the floral domain. This adaptation not only preserves the robust features learned from a diverse set of generic images but also fine-tunes the model to recognize and classify various flower species with enhanced accuracy. Figure 5 illustrates this transfer and adaptation process, highlighting the flow from initial training on ImageNet-1K to its application on the specialized floral dataset, thereby significantly enhancing the model's discriminatory power in flower recognition tasks.

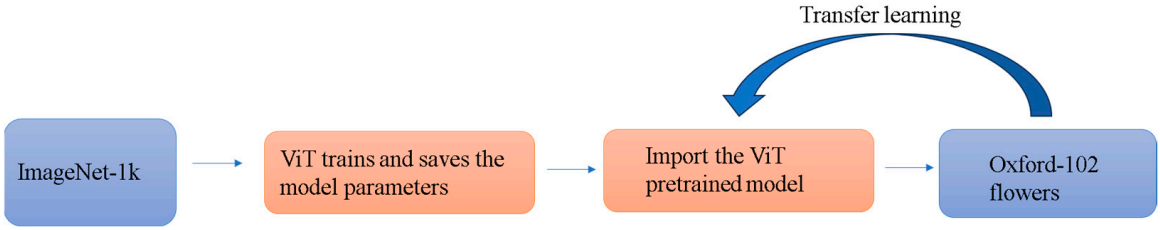

Figure 5. Model training process.

2.4.3. Loss Function

In this paper, we have selected the Mean Squared Error (MSE) loss function to optimize our model's performance. The choice of the MSE loss function is motivated by its smooth and continuous curve, which is differentiable at any point. This property is particularly advantageous for optimizing model parameters, as it facilitates straightforward updates using regression algorithms. Furthermore, the MSE loss function exhibits a desirable characteristic whereby the gradient decreases proportionally as the error decreases. This leads to rapid convergence of model parameters toward the minimum value, even with a fixed learning rate. This behavior benefits both the efficiency and stability of the training process, as it minimizes the likelihood of oscillations and ensures rapid adjustment of model parameters to optimal values. The calculation of the MSE loss function is straightforward and efficient, as demonstrated in Equation (1). This ease of calculation further enhances the appeal of the MSE loss function in our study, facilitating a more streamlined implementation and computation process.

$$MSE = \frac{\sum_{i=1}^{n} (f(x_i) - y_i)^2}{n} \quad (1)$$

In this equation, n represents the total number of observations. The function $f(x_i)$ is the predicted value outputted by the model for the i-th observation, where x_i denotes the input features of the i-th observation. The variable y_i refers to the actual observed value corresponding to x_i. The term $(f(x_i) - y_i)^2$ computes the squared difference between the predicted value and the actual value for each observation, quantifying the prediction error

for each point. The summation of these squared differences is then averaged over all n observations to yield the MSE, which provides a measure of the model's overall performance.

2.5. Flower Image Reconstruction

The MAE pre-training model plays a crucial role in reducing pixel redundancy in original flower images, thereby enhancing the efficiency of the learning process. A key aspect of this model is the generation of a crucial self-supervised task: reconstructing flower images, which aids in learning meaningful data representations. The random masking process is a vital component of the model's operation. It involves dividing each input flower image into 196 image blocks, each measuring 16 × 16 pixels. A subset of these blocks, typically approximately 75%, is randomly masked, leaving only 49 visible image blocks. This process ensures that the model focuses on learning from a smaller subset of data, thereby minimizing computational overhead. The encoder in the model is specifically designed to process only the remaining visible image blocks after masking. Selective processing further contributes to minimizing computational overhead.

Figure 6 provides a schematic representation of the autoencoder framework employed for flower image reconstruction. Beginning with the input image, patches are extracted and flattened before being projected linearly to form patch embeddings. A subset of these embeddings, termed 'tokens', is then masked, simulating a scenario where the model must predict missing information, thereby fostering robust feature extraction.

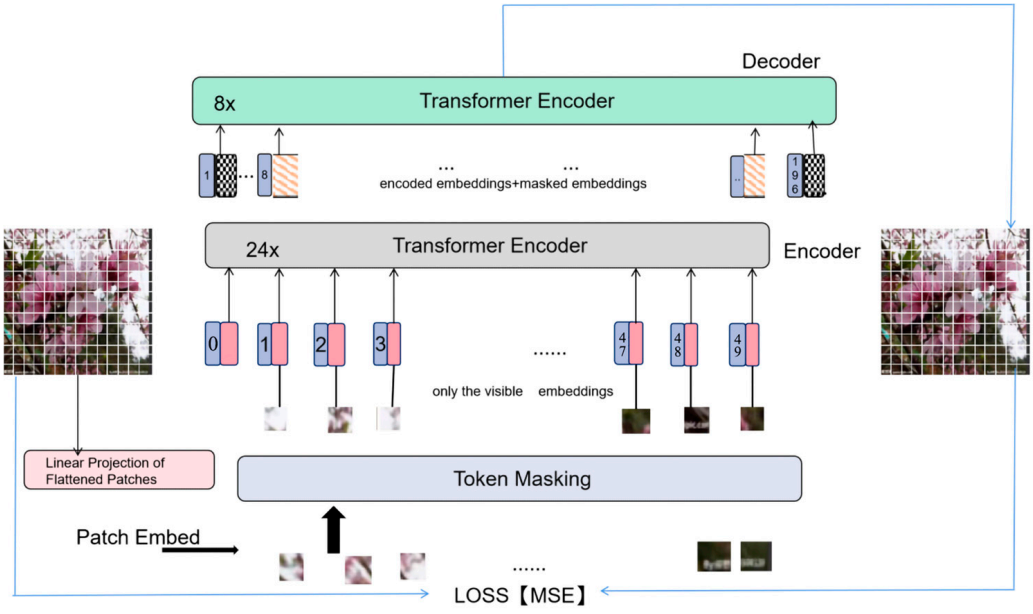

Figure 6. Reconstruction of flower images.

In the encoder stage, the visible embeddings undergo multiple transformations through a series of 24 transformer encoder layers, capturing complex patterns and dependencies. This comprehensive encoding process ensures that the information retained from the unmasked tokens is both rich and informative. The decoder module, comprising eight transformer encoder layers, is responsible for reconstructing the image. It processes both the encoded and masked embeddings, attempting to restore the original image at the pixel level. This process emphasizes the recovery of fine details and overall structure from the limited visible data.

A MSE loss is then calculated, measuring the difference between the reconstructed image blocks and the original, unaltered image. This loss quantifies the model's performance in terms of its ability to accurately reconstruct the image, providing a basis for optimizing the model's parameters. The model's ability to accurately reconstruct images with missing

patches is critical for its application in precise flower target detection. By refining the model against this loss, it becomes adept at capturing the essential features of flower images.

2.6. Training Environment

For the experiment, the Oxford-102 Flowers dataset was selected owing to its diversity and relevance in flower recognition tasks. To ensure consistency and comparability, all images in the dataset were resized to 224 × 224 pixels. This resolution is a common choice in image recognition experiments and facilitates efficient image processing by the model. The model training was conducted using Python 3.8, a programming language widely used in data science and machine learning. The PyTorch deep learning framework was chosen for its flexibility, user-friendliness, and support for dynamic computation graphs, which makes it suitable for developing and training complex models such as the one used in this experiment. To leverage GPU computational power, the experiment utilized CUDA version 11, a parallel computing platform and programming model developed by NVIDIA. Additionally, timm version 0.3.2, a library of pre-trained models for PyTorch, was used to ease the implementation and experimentation with various model architectures.

The hardware configuration and parameters used during model training play a critical role in reproducing the experiment and understanding its computational requirements. These details are provided in Table 2, including information such as the type of GPU used, batch size, learning rate, and other relevant parameters affecting the training process and the model's performance.

Table 2. Model Training Configuration and Parameters.

Configuration Name	Configuration Parameters
Operating System	Ubuntu
Programming Language	Python 3.8
Memory	140 G
Graphics Card Model	4 × A40
Deep Learning Framework	Pytorch

2.7. Hyperparameter Settings

The learning rate is a crucial parameter in the training process, as it determines the step size update in each iteration. It directly impacts the model's convergence state, with a higher learning rate potentially resulting in faster convergence but also increasing the risk of overshooting the optimal solution. Conversely, a lower learning rate may lead to slower convergence but enhanced stability [26].

The batch size, defined as the number of samples selected for training at one time, is another crucial parameter. It is typically chosen to be a power of 2 for computational efficiency. The batch size influences the model's generalization performance, with a larger batch size providing a more precise estimate of the gradient but potentially resulting in poorer generalization.

In the context of training, one epoch is defined as the process of training the model once with all the samples in the training set. The number of epochs determines the total number of times the training set is utilized during the training process.

In the final settings of this study, after several adjustments, the batch size was set to 256, and the initial learning rate was set to 0.001. These settings were chosen based on empirical evidence and the specific requirements of the task.

Furthermore, a learning rate decay strategy was implemented to dynamically adjust the learning rate based on the model's performance. The learning rate was reduced by a factor of 0.1 whenever the loss function did not show a significant decrease, assisting in fine-tuning the model's convergence. The model was trained for a total of 500 epochs, providing sufficient opportunity for the model to learn and adapt to the training data. The hyperparameter configuration is presented in Table 3.

Table 3. Hyperparameter Settings of Pre-Train model.

Configuration Name	Configuration Parameters
Initial Learning Rate	0.001
Batch size	256
Epochs	500
Optimizer	RMSprop

2.8. Evaluation Metrics

The loss function serves as a crucial metric in deep learning, evaluating the degree of discrepancy between predicted and actual values. It acts as an indicator of the model's accuracy, where a lower loss value signifies enhanced model robustness. In the context of the MAE pre-trained model, the primary task involves reconstructing the complete image using masked samples. We utilize Equation (1) as the loss function.

3. Experimental Results on the Pre-Training Model

As shown in Figure 7, after 500 training rounds, the network's convergence is carefully analyzed. The curve flattens after 300 rounds, indicating that the pre-trained model's learning ability has nearly reached its optimum. This convergence suggests that the network's image reconstruction capability is approaching its optimum, demonstrating the effectiveness of the training process. In Figure 8, we present a visual illustration of the processing steps employed by the MAE in our proposed flower recognition model. The sequence commences with the original image of a flower. This image is transformed to obscure specific portions, resulting in a 'masked' version. Subsequently, our model endeavors to reconstruct the occluded regions, using learned representations to predict the missing pixels.

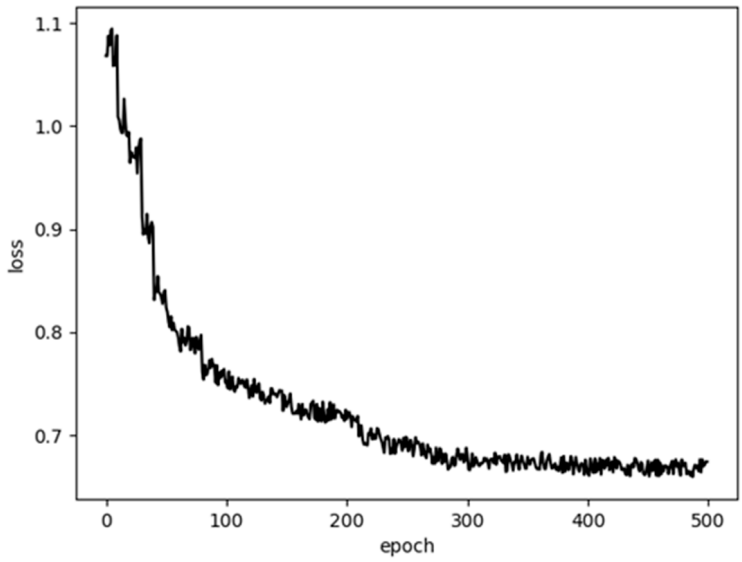

Figure 7. Decrease in *MSE* loss of the pre-trained model with training epochs.

After training the MAE on the original Oxford 102 Flowers dataset, this chapter describes the removal of the decoder component, retaining only the encoder for flower feature extraction. Subsequently, a simple fully connected layer follows the encoder to classify flower images, demonstrating the encoder's robust feature extraction capabilities. To this end, techniques such as random cropping, random flipping, and grayscale transformations were applied to augment the original Oxford 102 Flowers dataset. The weights of the encoder were frozen, and only 50 training epochs were conducted on the fully connected layers. Setting the learning rate to 0.0001 and batch size to 256, Figure 9 illustrates the change curves for the classification loss function (left) and accuracy (right)

for flower images. It was observed that after 35 training epochs, both the learning rate and accuracy converged.

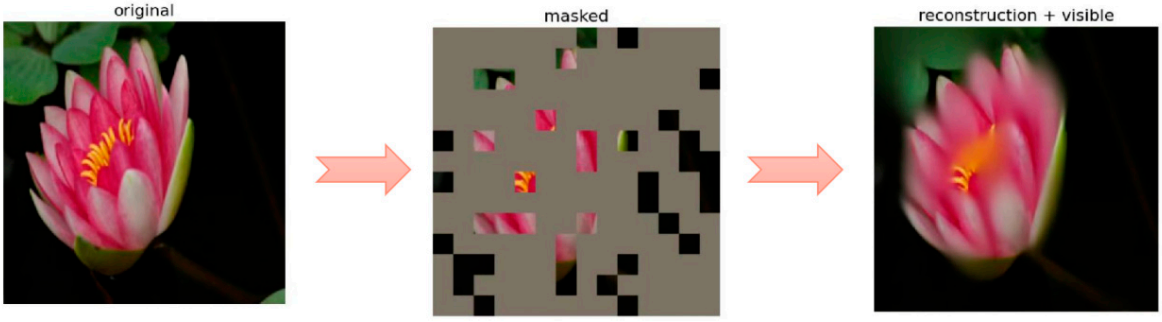

Figure 8. Visualization of flower image reconstruction.

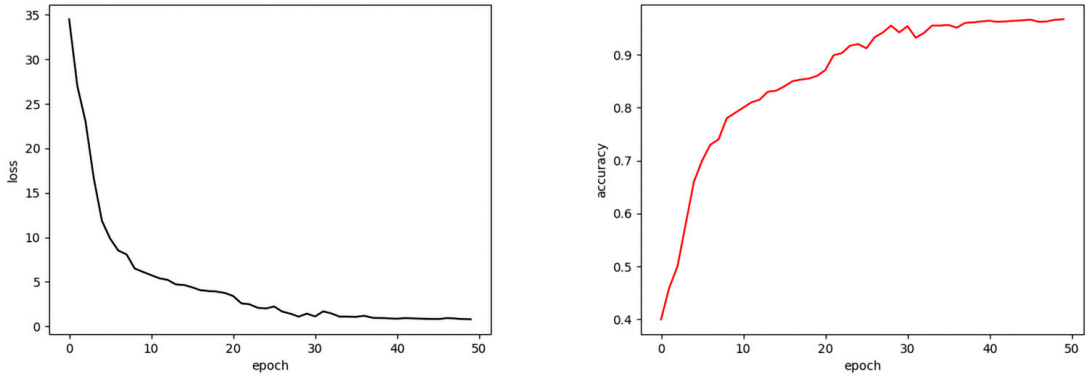

Figure 9. Changes in loss function value and classification accuracy with training rounds.

Meanwhile, we calculated accuracy at 1 (acc @ 1), accuracy at 3 (acc @ 3), and accuracy at 5 (acc @ 5) for the flower classification model; the training results are presented in Figure 10. Among these, accuracy at 1 (acc @ 1) refers to the accuracy of the first prediction in each batch, indicating the model's ability to correctly identify the primary category. Accuracy at 3 (acc @ 3) and accuracy at 5 (acc @ 5) denote the model's accuracy for the top three and top five predictions in each batch, respectively. These metrics are particularly useful for evaluating the accuracy of classification models with numerous categories, such as this dataset, which contains 102 categories.

It is important to note that all images in the dataset were used in the pre-training without corresponding labels, thereby avoiding any issues of high accuracy due to label leakage in the supplementary classification experiments. To demonstrate the superiority of the pre-trained flower classification model based on the MAE, this section compares it with traditional manual feature-based, CNN-based, and ViT model-based methods using the publicly available Oxford 102 Flowers dataset, as detailed in Table 4.

Table 4. Comparison of Classification Accuracy with Other Models.

Method	Accuracy
Automated flower classification over a large number of classes [27]	72.80%
Image segmentation for large-scale subcategory flower recognition [28]	80.70%
ResNet101 [29]	95.95%
VGG-16 [30]	80.91%
ViT-L-I1k-MAE (our)	**99.60%**

From the first and second rows of the table, it is evident that traditional flower classification methods based on manual features exhibit relatively low accuracy. The primary

reason is that historically, computer technology was not sufficiently advanced, and manually designed features, based on experience, were error-prone, failing to effectively address the issues of inter-class similarity and intraclass differences among flowers [3]. From the third and fourth rows, it is clear that flower recognition methods based on CNNs represent significant advancements over manual feature methods. Notably, our model, ViT-L-I1k-MAE, has achieved an impressive 99.60% accuracy, underscoring the substantial benefits of pre-trained models for feature extraction.

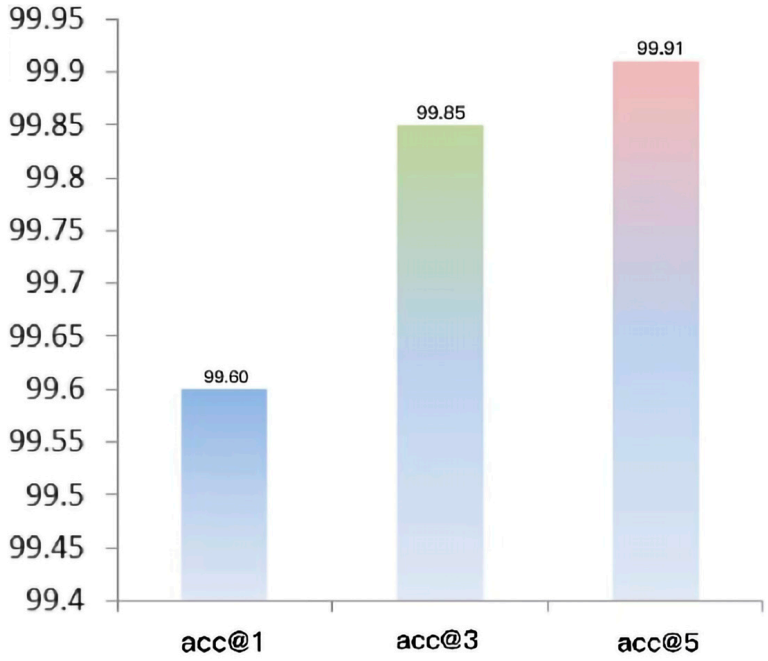

Figure 10. Training results of flower image classification.

To broaden the model's applications, the pre-trained MAE is adapted for flower object detection tasks. This adaptation involves integrating the encoder component of the MAE into the YOLOv5 backbone network, thus creating a specialized flower object detection model. This modification enables the model to utilize its learned representations to detect floral objects in images, emphasizing the versatility and transferability of the MAE approach.

4. Materials and Methods on Object Detection Model

4.1. Data Acquisition and Processing

Current research in flower image recognition focuses primarily on classification because of the limited availability of public datasets for flower object detection. In this section, using the Selenium tool in Python, we use web crawling technology to collect flower images from Baidu (https://image.baidu.com/ accessed on 9 May 2024), where the image source region is limited to China, to create a custom private dataset for this study.

Ultimately, we obtained 13 classes of flower images characterized by high inter-class similarity and significant intra-class variation. These classes include *Lysimachia foemina*, *Arctium lappa*, *Allium macrostemon*, *Allium tuberosum*, *Taraxacum*, *Gynura aurantiaca*, *Tagetes*, *Cichorium intybus*, *Hibiscus cannabinus*, *Achillea millefolium*, *Bellis perennis*, *Apium graveolens*, *and Gardenia jasminoides*. Each class consists of images depicting various colors and growth stages, totaling 4566 images. An example is shown in Figure 11.

Utilizing the LabelImg tool, flower images were annotated in YOLO format, resulting in .txt files. To prevent overfitting due to the small dataset size, each flower image was augmented with random flips and random crops, yielding a total of 10,513 images for data augmentation. Subsequently, the dataset was divided into training, validation, and testing sets at an 8:1:1 ratio.

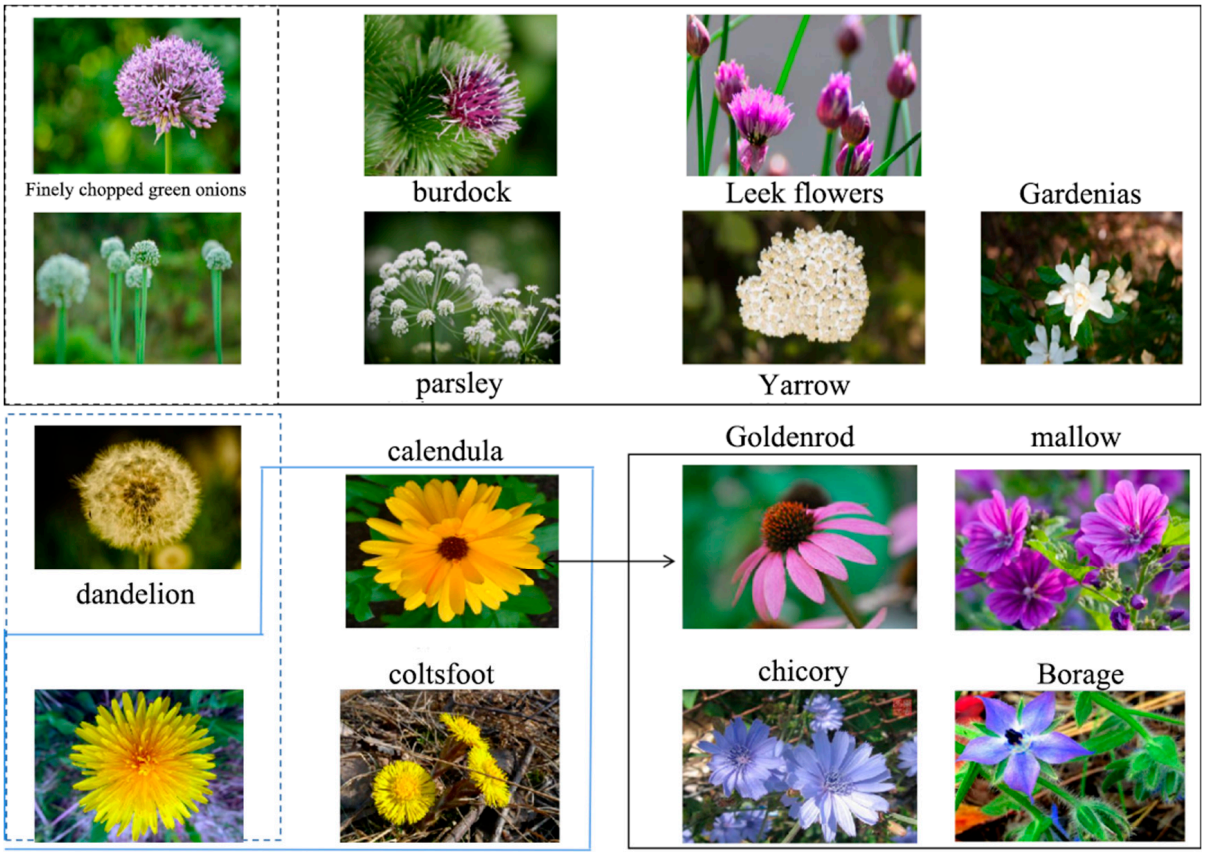

Figure 11. Example images from the flower object detection dataset.

4.2. Object Detection Model Construction

4.2.1. YOLOv5 Model Principles

YOLOv5 improves on YOLOv4 by enhancing the Backbone, Neck, Head, and output layers, thus further boosting the algorithm's performance. In this section, the MAE pre-trained model is integrated with and optimized using the YOLOv5 model. The flower object detection model based on MAEs has been developed. The structure of YOLOv5 is shown in Figure 12:

- Input

YOLOv5 utilizes Mosaic data augmentation to enlarge the dataset. Additionally, it integrates anchor box size calculation into the model training process, automatically determining the optimal anchor framework. Moreover, it uniformly resizes images to a standard size using adaptive scaling, minimizing redundant information and enhancing network inference speed.

- Backbone

The Backbone network is responsible for extracting features from the input images using CNNs.

- Neck

During the feature extraction process from images, some local information may be lost. The Neck network combines feature maps from different levels of the network to capture richer feature information from the image, which is then fed into the Head layer.

- Head

The Head layer conducts the final regression prediction, enhancing classification and localization.

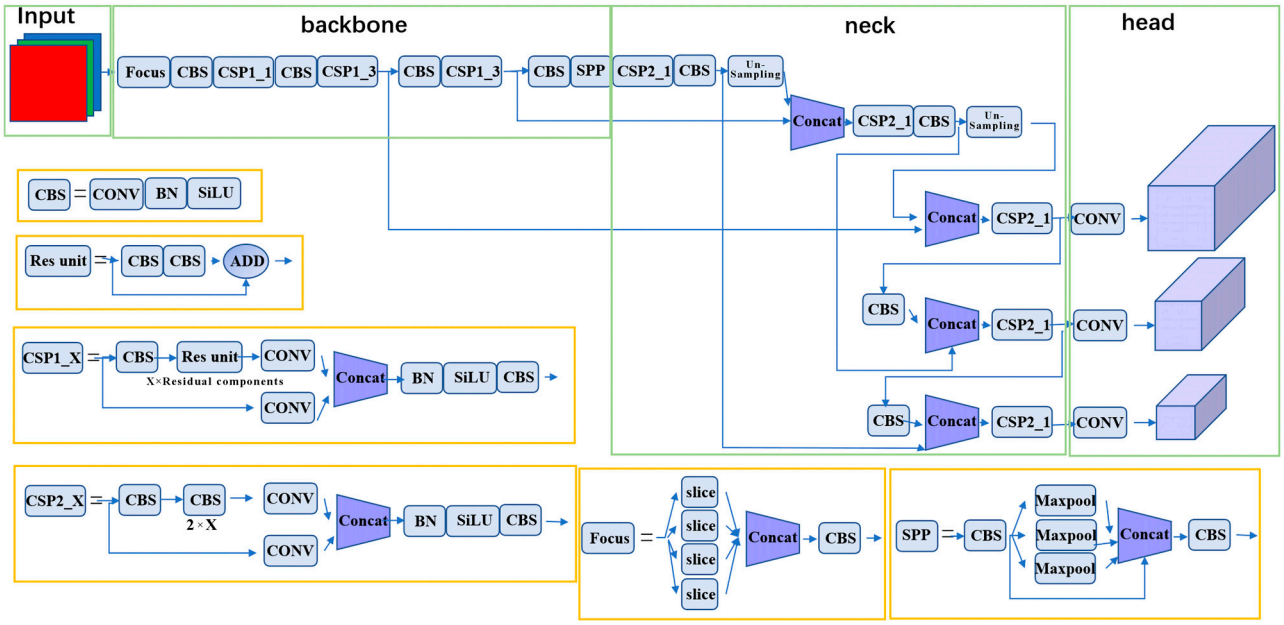

Figure 12. YOLOv5 model structure.

4.2.2. YOLOv5 Loss Function

The localization loss employs the Complete Intersection over Union ($CIoU$) loss to determine the error between the predicted box and the true box. $CIoU$ considers the overlap area, center point distance, and aspect ratio, enhancing the comprehensiveness of box regression. The IoU and $CIoU$ loss formulas are as follows [31]:

$$IoU = \frac{|A \cap B|}{|A \bigcup B|} \tag{2}$$

Here, $|A \cap B|$ represents the area of intersection between the bounding boxes A and B, and $|A \bigcup B|$ denotes the area of their union.

$$Loss_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \tag{3}$$

In this formulation:

$\rho^2(b, b^{gt})$ measures the squared Euclidean distance between the central points of the predicted bounding box b and the ground truth bounding box b^{gt}, providing a scale-invariant metric of central point differences.

c represents the diagonal length of the smallest enclosing box covering both b and b^{gt}, used to normalize the central point distance.

α is a weighting factor that scales the importance of the consistency term v, which quantifies the aspect ratio consistency between the predicted and ground truth bounding boxes. This term is particularly important in object detection to maintain the proportional dimensions of the bounding boxes.

4.2.3. YOLOv5 Improvement Strategy

In this section, the encoder obtained from the MAE pre-trained model replaces the Backbone component of YOLOv5 for feature extraction from flower images. Figure 13 illustrates the integration of the MAE and YOLOv5 in our proposed model, enhancing feature extraction in flower image detection. In this innovative approach, the backbone of YOLOv5, traditionally responsible for initial feature extraction, is substituted with the encoder from a pre-trained MAE. This encoder, pre-conditioned on a flower-specific dataset, harnesses the powerful representational capabilities of the ViT-Large architecture. The

extracted features are then propagated through the Neck, comprising a series of processing layers designed to spatially refine these features before prediction. Subsequently, the refined features enter the Head of the model, which utilizes Non-Maximum Suppression (NMS) to discern the most probable bounding boxes from an array of candidates, outputting a matrix with dimensions $7 \times 7 \times 23$. This matrix includes class probabilities, bounding box coordinates, and objectness scores. Adjustments to the position offsets within the candidate boxes are made to enhance the precision of the final object detection.

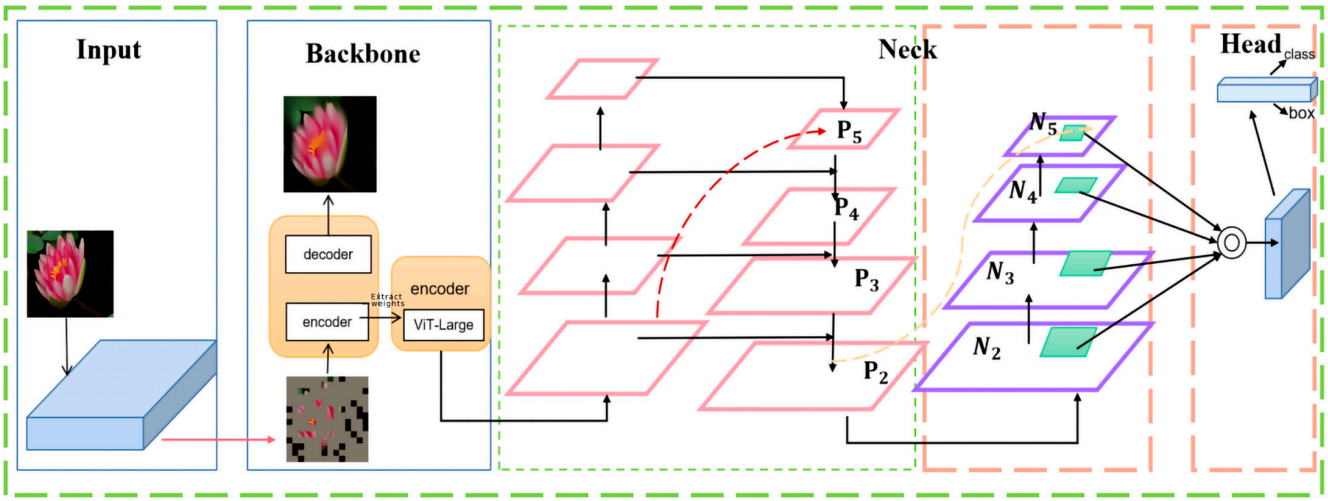

Figure 13. Improved YOLOv5 object detection model structure.

4.3. Experiment and Analysis

4.3.1. Experimental Environment

The experiments in this chapter were conducted on the Ubuntu operating system, utilizing the PyTorch deep learning framework, CUDA version 11, and timm (PyTorch Image Models) version 0.3.2, and were programmed in Python 3.8. The hardware configuration used was the same as that in Table 2 of Section 2.

4.3.2. Hyperparameter Settings

In this chapter's research on flower image classification using the MAE pre-trained model, the batch size was set to 32, and the initial learning rate was set to 0.001. The model was trained for 300 epochs. The hyperparameter configuration is presented in Table 5.

Table 5. Hyperparameter Settings of Object Detection Model.

Configuration Name	Configuration Parameter
Learning rate	0.001
Batch size	32
Epochs	300
Optimizer	RMSprop

4.3.3. Evaluation Metrics

(1) *Precision*: It is the proportion of correctly predicted positive instances among the instances predicted as positive. It is calculated as:

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

where TP (True Positives) are the correctly predicted positive instances, and FP (False Positives) are the instances incorrectly predicted as positive.

(2) *Recall*: It is the proportion of correctly predicted positive instances among all actual positive instances. It is calculated as:

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

where FN (False Negatives) are the actual positive instances that were incorrectly predicted as negative.

(3) Mean Average Precision (mAP): It is the average of AP (Average Precision) and is a primary evaluation metric for object detection algorithms. A higher mAP indicates better detection performance of the object detection model on the given dataset. It is calculated as follows:

$$mAP = \frac{\sum_{i=1}^{k} AP_i}{K} \tag{6}$$

where k represents the number of classes. When $k = 1$, $mAP = AP$. When $k > 1$, mAP is the mean of AP. AP is the area under the precision-recall curve interpolated to have smooth curves. The threshold is generally set to 0.5, meaning that predicted boxes with an IoU greater than 0.5 are considered valid, denoted as mAP@0.5.

$$AP = \sum_{i=1}^{n-1} (r_{i+1} - r_i) p_{interp}(r_{i+1}) \tag{7}$$

In this equation:

r_i and r_{i+1} represent the recall values at the i-th and $(i + 1)$-th thresholds, respectively. These values are part of the sorted list of all recall values obtained by varying the decision threshold on the detection confidence.

p_{interp} is the interpolated precision at recall level r_{i+1}. This interpolation ensures that the precision is adjusted to reflect the maximum precision observed for all recall levels greater than or equal to r_{i+1}, which helps to handle the variations in precision at different recall thresholds.

The term $(r_{i+1} - r_i)$ calculates the increment in recall from one threshold to the next, and the product of this increment with the interpolated precision gives the contribution of each segment to the overall AP.

5. Experiment Results on Object Detection Model

The training results of the flower detection and recognition model based on the MAE are shown in Figure 14.

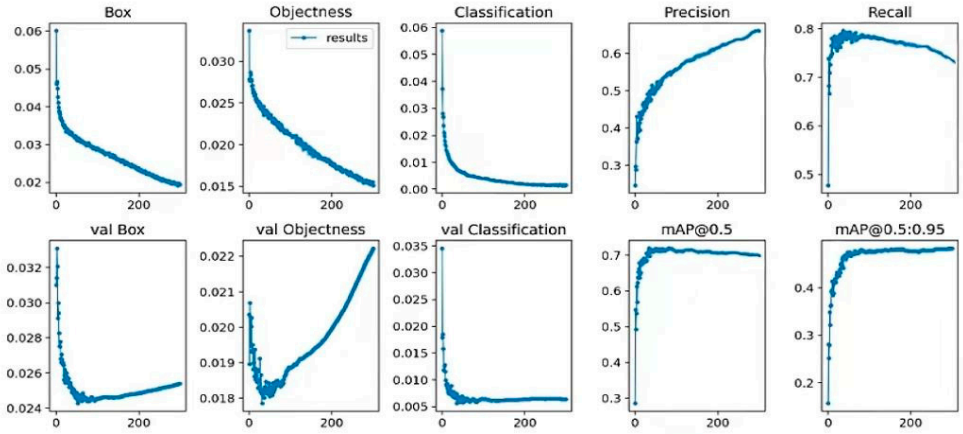

Figure 14. Evaluation metrics variation with training epochs.

Precision and recall are commonly used to evaluate a model's performance. The area under the P–R curve represents the average precision (AP), with a larger area indicating better model performance. After experimenting with the flower object detection method

based on the MAE on a self-built flower dataset, the P–R curve shown in Figure 15 was obtained, with an average mAP@0.5 of 71.3% for all classes.

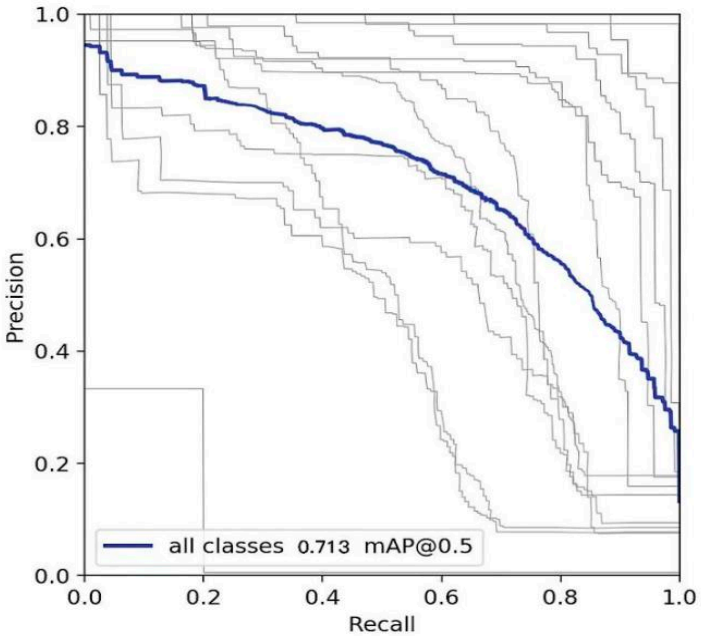

Figure 15. Precision-recall curve.

The flower object detection method proposed in this paper, which improves YOLOv5 using MAEs, was compared with YOLOv5s and YOLOv4, with the results shown in Table 6.

Table 6. Comparison of Object Detection mAP and Precision.

Method	mAP@0.5	Precision	Training Time	Inference Time (ms)
Yolov5s	69.8%	86.7%	3 h 15 m	3.1
Yolov4	67.4%	90.3%	3 h 54 m	6.8
Our	**71.3%**	**91%**	4 h 37 m	5.2

The proposed model demonstrates significant advancements in flower detection and recognition over existing models. Our method shows an increase in mAP@0.5 of 1.5 percentage points compared to YOLOv5s and 4.1 percentage points compared to YOLOv4. Additionally, our approach improves precision by 4.3 percentage points compared to YOLOv5s and 0.7 percentage points compared to YOLOv4.

Notably, although the proposed model requires a longer training time compared to both YOLOv5s and YOLOv4, this is offset by its efficiency during inference. The inference time of the proposed model is reduced by approximately 23% compared to YOLOv4, although it is slightly higher than that of YOLOv5s. This demonstrates a balanced improvement in both accuracy and operational efficiency, which is critical for real-time applications. The increased training time is attributed to the more complex MAE-based feature extraction network, which nevertheless results in higher precision and faster inference times than the earlier YOLOv4 model.

As shown in Figure 16, the use of a MAE-enhanced feature extraction framework within the YOLO architecture leverages more sophisticated pre-trained embeddings to provide improved detection capabilities, particularly in categorizing diverse and complex flower species. Subsequently, the recognition and detection results of this method were visualized using PyQt5 with Python 3.8 and OpenCV drawing techniques.

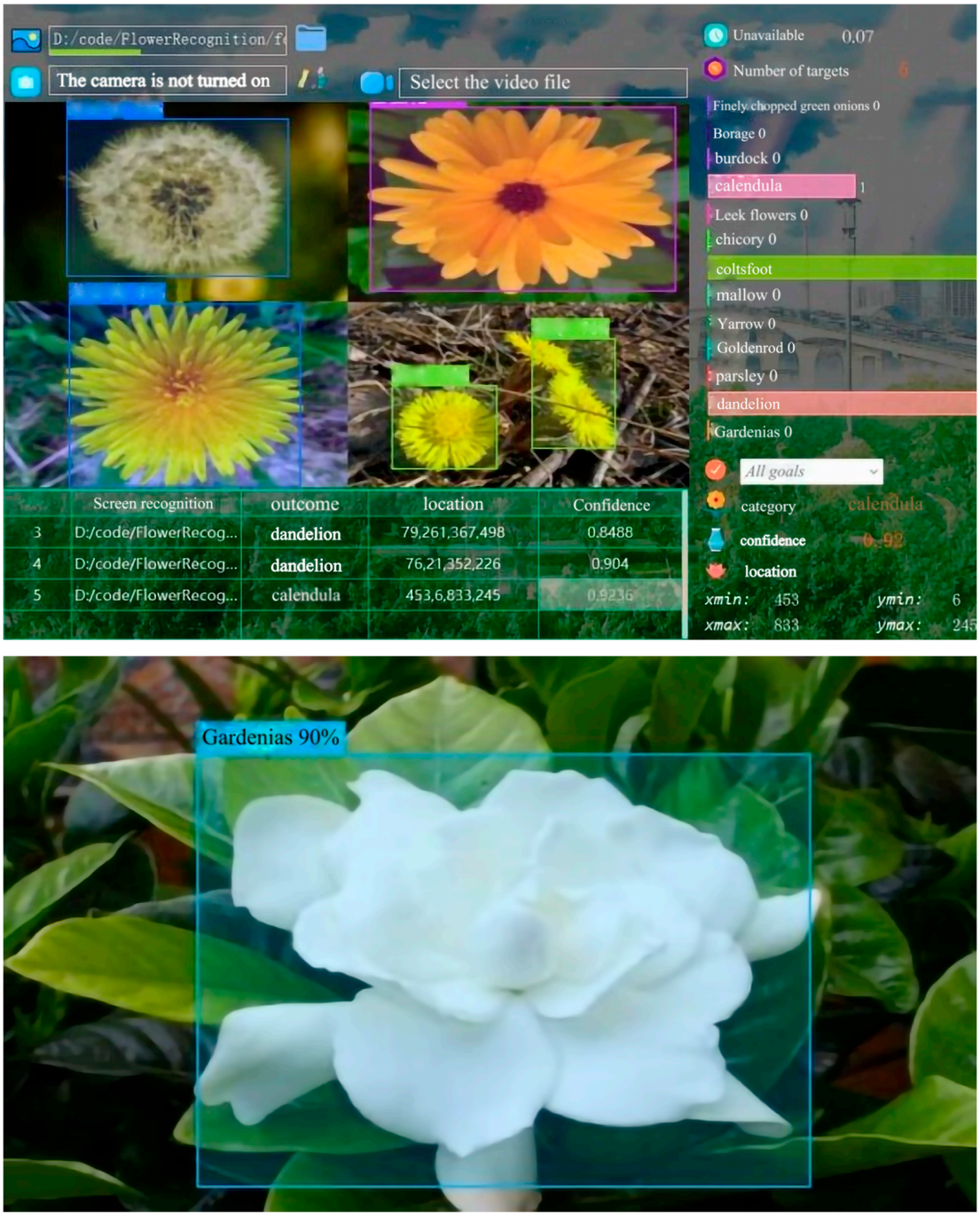

Figure 16. Visualization of flower detection and recognition using the proposed method.

6. Discussion

Figure 14 illustrates a multi-faceted evaluation of the object detection model across several metrics over 300 epochs. In the top row, the 'Bounding Box', 'Objectness', and 'Classification' loss curves reveal an overall descending trend, which indicates a progressive refinement in the model's ability to accurately predict bounding boxes, discern objects from background noise, and correctly classify detected objects. Notably, the 'Bounding Box' and 'Classification' losses demonstrate a steady decline, while the 'Objectness' loss relatively plateaus, which suggests that the model quickly learns to differentiate objects from the background, with further improvements primarily in bounding box regression and classification accuracy.

The bottom row offers further insights. The 'val Box', 'val Objectness', and 'val Classification' losses initially decrease, with a noticeable uptick in 'val Box' and 'val Objectness' around the 150th epoch. This may indicate a point of overfitting, where the model begins to learn noise from the training data, thereby reducing its generalization capability to the validation set. Upon evaluating performance metrics, the 'mAP@0.5' and 'mAP@0.5:0.95' exhibit an ascending trend that plateaus, indicating the model's increasing and then stable proficiency in accurately detecting objects with a high degree of overlap (greater than 50%) with truth bounding boxes. The difference in saturation levels between 'mAP@0.5' and 'mAP@0.5:0.95' suggests that while the model performs well at the more lenient threshold of 0.5 IoU, there is a performance drop-off at stricter thresholds, highlighting a potential area for further model refinement.

Lastly, 'Precision' and 'Recall' curves ascend towards a plateau, demonstrating the model's improved ability to correctly identify positive samples and its success rate in not missing actual positives. However, the slight dip at the end of the curves suggests an increase in the number of FN. This condition indicates that the model may be overfitting due to factors such as imbalanced datasets and background noise. Overall, Figure 13 suggests a robust model with room for enhancement, particularly in handling overfitting and improving detection at stricter IoU thresholds. Future work should consider exploring advanced regularization methods and more extensive datasets to mitigate overfitting and enhance model accuracy.

7. Conclusions

With the rapid advancement of deep learning and computer vision technologies, image recognition and detection have attracted extensive study from numerous scholars. In the field of flower recognition, the enhancement of efficiency and accuracy in flower identification has been a task of enduring research significance. Fine-grained image recognition techniques, particularly those based on CNNs, have matured significantly, with extracted features demonstrating strong expressiveness and achieving notable results in fine-grained image recognition. However, due to the excessive granularity of key points in these images, traditional CNNs struggle to extract all key point information. Given the shortcomings of CNN-based fine-grained image recognition methods, this paper explores the application of the ViT to fine-grained flower image recognition. However, the application of ViT to flower images encounters challenges such as small flower datasets and high computational resource consumption. To address these challenges, this paper conducts research on the task of flower image recognition, with the main research work summarized as follows: construction of a pre-trained model based on MAEs. This study employs the self-supervised learning capabilities of MAEs to construct a pre-trained model on the Oxford-102 flowers dataset. Specifically, ViT-Large serves as the encoder, and an 8-layer Transformer structure functions as the decoder, enabling the ViT model to acquire more robust features during pre-training on small, unlabeled datasets, and facilitating seamless transition to flower recognition-related tasks upon decoder removal.

Flower Object Detection Leveraging MAE Pre-training. Addressing the scarcity of research in flower object detection and the suboptimal detection accuracy of the YOLOv5 model within this domain, this study transfers the pre-trained model to specific flower object detection tasks. The application of pre-training significantly enhances model training efficiency and mitigates issues arising from dataset imbalances. For flower feature extraction, the model substitutes the YOLOv5 backbone network with the encoder obtained from MAE-based pre-training. Ultimately, it achieves a mAP of 71.3%, enabling higher-precision flower object detection at reduced computational costs.

The flower recognition method proposed in this paper, which utilizes MAEs, demonstrates outstanding performance in flower object detection. However, there remain issues that require further improvement and resolution: The flower dataset used for designing the pre-trained model utilizing MAEs and implementing the flower image classification task was constructed by the University of Oxford laboratory and primarily includes common

flower species in the UK, which differ from those in China. Additionally, when implementing flower target detection, the self-built flower dataset contains only 13 species, which are relatively limited in number. If more native flower species in China were utilized, it would be feasible to construct a flower recognition model better suited for the Chinese populace, thereby promoting the development of botanical research in China. The MAE pre-trained model employs ViT as the baseline, and the extensive number of parameters in ViT could restrict the deployment of the model to mobile platforms such as smartphones. Therefore, determining how to eliminate redundant parameters or streamline the model while maintaining network accuracy and precision remains a future research challenge.

Author Contributions: Y.L. (Yin Li), funding acquisition, software, data curation, formal analysis, and writing—original draft. Y.L. (Yang Lv), investigation, methodology, software, visualization, formal analysis, writing—original draft, and writing—review and editing. Y.D., visualization, formal analysis, and methodology. H.Z., visualization and formal analysis. H.G. and L.Z., conceptualization, supervision, writing—review and editing, formal analysis, and funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Major Science and Technology Project of Shaanxi Province under grant No.2020zdzx03-03-02; the Seed Hatching Project of Yangling Demonstration Area under grant numbers 2022-JSCY-09 and 2022-JSCY-11; National Key Research and Development Program of China under grant numbers 2023YFD2301000 and University-Industry Collaborative Education Program, Ministry of Education, PRC under grant numbers 231106627201542.

Data Availability Statement: The raw data and source codes supporting the conclusions of this article will be made available by the authors on request.

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

References

1. Mei, S. The Medicinal Value of Flowers. *For. Hum.* **2002**, *2*, 29–31. Available online: https://chn.oversea.cnki.net/KCMS/detail/detail.aspx?dbcode=CJFD&dbname=CJFD2002&filename=SLRL200202019&uniplatform=OVERSEA&v=JpUuGHBa6kPGcwklCO9OIKzX1NYpynW43b2Xr8jfKSuqubZHCamW23KE-q6lmfjf (accessed on 9 May 2024). (In Chinese)
2. Sun, J. Research on Deep Learning Defect Detection Network Based on MAE Pretraining. *Inf. Comput.* **2022**, *24*, 161–166. Available online: https://chn.oversea.cnki.net/KCMS/detail/detail.aspx?dbcode=CJFD&dbname=CJFDLAST2023&filename=XXDL202224046&uniplatform=OVERSEA&v=8Crfb2VCl6syY6KHzDwomvSMjCgzZ9g-7eMk5gaNp1dWR6AlqgApINYsjqWakYbY (accessed on 9 May 2024). (In Chinese)
3. Qin, M. Research on Flower Image Classification and Recognition Model Based on Deep Learning. Master's Thesis, Guangxi Normal University, Guilin, China, 2020. (In Chinese) [CrossRef]
4. Qian, Y. Enhancing Automatic Emotion Recognition for Clinical Applications: A Multimodal, Personalized Approach and Quantification of Emotional Reaction Intensity with Transformers (Order No. 30688324). (2954319707). 2023. Available online: https://www.proquest.com/dissertations-theses/enhancing-automatic-emotion-recognition-clinical/docview/2954319707/se-2 (accessed on 9 May 2024).
5. Koné, A.; Es-Sabar, A.; Do, M.-T. Application of Machine Learning Models to the Analysis of Skid Resistance Data. *Lubricants* **2023**, *11*, 328. [CrossRef]
6. Das, M.; Manmatha, R.; Riseman, E.M. Indexing flowers by color names using domain knowledge-driven segmentation. In Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision. WACV'98 (Cat. No. 98EX201), Princeton, NJ, USA, 19–21 October 1998; pp. 94–99.
7. Nilsback, M.E.; Zisserman, A. A Visual Vocabulary for Flower Classification. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2006; pp. 1447–1454. [CrossRef]
8. Lyu, X.; Chen, Z.; Wu, D.; Wang, W. Natural language processing and Chinese computing. In Proceedings of the 9th CCF International Conference, NLPCC, Zhengzhou, China, 14 October 2020; pp. 710–721.
9. Ke, X.; Chen, X.; Li, S. Flower Image Retrieval Based on Multi-Feature Fusion. *Comput. Sci.* **2010**, *11*, 282–286. Available online: https://chn.oversea.cnki.net/KCMS/detail/detail.aspx?dbcode=CJFD&dbname=CJFD2010&filename=JSJA201011073&uniplatform=OVERSEA&v=-sgfJ21OBuozqz420r9TnLQss_Z7Rre16HeJWuU7HzY0qLXUIAhMq0effahMI6Bo (accessed on 9 May 2024). (In Chinese)
10. Liu, B.; Ding, Z.; Zhang, Y.; He, D.; He, J. Kiwifruit Leaf Disease Identification Using Improved Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 13–17 July 2020; pp. 1267–1272. [CrossRef]

11. Liu, Y.; Zheng, F.B. Object-oriented and multi-scale target classification and recognition based on hierarchical ensemble learning. *Comput. Electr. Eng.* **2017**, *62*, 538–554. [CrossRef]
12. Liu, S.; Gao, X. Fine-grained image classification method based on deep model transfer. *J. Comput. Appl.* **2018**, *38*, 2198–2204. [CrossRef]
13. Lv, R.; Li, Z.; Zuo, J.; Liu, J. Flower Classification and Recognition Based on Significance Test and Transfer Learning. In Proceedings of the 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE), Guangzhou, China, 15–17 January 2021; pp. 649–652. [CrossRef]
14. Wang, S. Research and Implementation of Flower Recognition Algorithm Based on Machine Learning. Master's Thesis, University of Electronic Science and Technology of China, Chengdu, China, 2018. Available online: https://chn.oversea.cnki.net/KCMS/detail/detail.aspx?dbcode=CMFD&dbname=CMFD201802&filename=1018991547.nh&uniplatform=OVERSEA&v=h_si8MuDyK-eKkyt7VPNMXi7zvkU_flV5USjpPDLqY1ANvH1REQduycwspFgPME5 (accessed on 9 May 2024). (In Chinese)
15. Samragh, M.; Farajtabar, M.; Mehta, S.; Vemulapalli, R.; Faghri, F.; Naik, D.; Tuzel, O.; Rastegari, M. Weight subcloning: Direct initialization of transformers using larger pretrained ones. *arXiv* **2023**, arXiv:2312.09299.
16. Xie, Z.; Hu, Y. A Multi-Target Flower Recognition System Based on YOLOv4. *J. Nanjing Agric. Univ.* **2022**, *45*, 818–827. Available online: https://chn.oversea.cnki.net/KCMS/detail/detail.aspx?dbcode=CJFD&dbname=CJFDLAST2022&filename=NJNY202204023&uniplatform=OVERSEA&v=T5VrxUPl9fU-K_Plk_4nfP6sYGq5m7x9fF-GTxbokYDtqmA4RAASoudygYGziRKA (accessed on 9 May 2024). (In Chinese)
17. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762. [CrossRef]
18. Zhang, C.; Huang, W.; Liang, X.; He, X.; Tian, X.; Chen, L.; Wang, Q. Slight crack identification of cottonseed using air-coupled ultrasound with sound to image encoding. *Front. Plant Sci.* **2022**, *13*, 956636. [CrossRef] [PubMed]
19. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929. [CrossRef]
20. Beal, J.; Kim, E.; Tzeng, E.; Park, D.H.; Kislyuk, D. Toward transformer-based object detection. *arXiv* **2020**. [CrossRef]
21. Roy, B.K.; Chaturvedi, A.; Tsaban, B.; Hasan, S.U. Cryptology and Network Security with Machine Learning. Available online: https://link.springer.com/book/10.1007/978-981-99-2229-1 (accessed on 9 May 2024).
22. He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; Girshick, R. Masked autoencoders are scalable vision learners. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 16000–16009.
23. Singh, M.; Duval, Q.; Alwala, K.V.; Fan, H.; Aggarwal, V.; Adcock, A.; Joulin, A.; Dollár, P.; Feichtenhofer, C.; Girshick, R.; et al. The effectiveness of MAE pre-pretraining for billion-scale pretraining. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 5484–5494.
24. Dong, G.; Li, W.; Dong, Z.; Wang, C.; Qian, S.; Zhang, T.; Ma, X.; Zou, L.; Lin, K.; Liu, Z. Enhancing Dynagraph Card Classification in Pumping Systems Using Transfer Learning and the Swin Transformer Model. *Appl. Sci.* **2024**, *14*, 1657. [CrossRef]
25. Man, X.; Zhang, C.; Feng, J.; Li, C.; Shao, J. W-mae: Pre-trained weather model with masked autoencoder for multi-variable weather forecasting. *arXiv* **2023**, arXiv:2304.08754.
26. Jin, S. Exploring the Causes of Artwork Misclassification Through Machine Learning (Order No. 30636934). (2942088587). 2023. Available online: https://www.proquest.com/dissertations-theses/exploring-causes-artwork-misclassification/docview/2942088587/se-2 (accessed on 9 May 2024).
27. Nilsback, M.E.; Zisserman, A. Automated flower classification over a large number of classes. In Proceedings of the 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing, Bhubaneswar, India, 16–19 December 2008; pp. 722–729.
28. Angelova, A.; Zhu, S.; Lin, Y. Image segmentation for large-scale subcategory flower recognition. In Proceedings of the 2013 IEEE Workshop on Applications of Computer Vision (WACV), Clearwater Beach, FL, USA, 15–17 January 2013; pp. 39–45.
29. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
30. Hou, X.; Liu, H.; Hou, W. Flower Classification Based on Improved VGG16 Network Model. *Comput. Syst. Appl.* **2022**, *31*, 172–178. Available online: http://www.c-s-a.org.cn/1003-3254/8582.html (accessed on 9 May 2024). (In Chinese)
31. Zhao, X.; Xiao, N.; Cai, Z.; Xin, S. YOLOv5-Sewer: Lightweight Sewer Defect Detection Model. *Appl. Sci.* **2024**, *14*, 1869. [CrossRef]