Journal of Nonlinear and Convex Analysis
Volume 26, Number 7, 2025, 1971–1985

Yokohama Publishers
ISSN 1880-5221 ONLINE JOURNAL

PRUNING-AND-COMPRESSION ENHANCED VIT-S/16 MODEL FOR EFFICIENT COLLOIDAL GOLD DETECTION

YAOZHONG PENG, YUJIA GAO, FEI XIE, ZHIYUAN GUO, AND QIJUAN GAO*

Abstract. Vision Transformers (ViT) demonstrate superior performance in computer vision tasks but are limited for deployment on edge devices due to their large parameter size and computational demands. This paper proposes a ViT-S/16 model enhancement technique, which combines unstructured pruning and modifications to the MLP layer to reduce computational load while maintaining high accuracy. The enhanced model was applied to colloidal gold test kit classification through transfer learning and data augmentation, achieving a 16% accuracy improvement in the original ViT-S/16 model. Further analysis revealed that a pruning rate of 35.18% in the qkv layers substantially reduced model size and computational requirements, while the accuracy reached up to 97.13% in colloidal gold detection. The proposed enhancement technique improved the ViT model's feasibility for edge deployment, with significant potential in agricultural and food safety image detection applications.

1. Introduction

To advance agricultural supply flow and guarantee high quality of agricultural products and minimize food safety risks, pesticide residue detection of agricultural products is required. Among chromatographic, immunoassay, capillary electrophoresis, and nanobiosensor detection methods [1], the chromatographic detection methods are the most widely used, including high-performance liquid chromatography (HPLC) and gas chromatography (GC) [4, 10]. These methods are highly accurate and sensitive but are complex and require expensive equipment. With the rapid development of nanotechnology, colloidal gold has been used in the biomedical field since the 1970s. It has gradually expanded its applications in agriculture, medicine, and food safety, among other fields. Colloidal gold immunochromatographic assay (CGIA) efficiently identifies target substances by attaching antibodies or antigens to the surface of gold nanoparticles and utilizing

2020 *Mathematics Subject Classification.* 62H30, 97M50.

Key words and phrases. Vision transformer, model pruning, data augmentation, colloidal gold detection, model compression.

This work was supported by the Scientific Research Foundation of the Education Department of the Province Anhui (2023AH051303, 2023AH040132) and by the Opening Fund of State Key Laboratory of Tea Plant Biology and Utilization (SKLTOF20230127) and by Anhui Post-doctoral Science Foundation (2023B677), and by Natural science Research Project for Advanced Scholars of Hefei Normal University (2023rcjj13, 2023rcjj14) and Social Science General Program sponsored by the Ministry of Education of the People's Republic of China (23YJCZH067), and by Hefei Municipal Natural Science Foundation (202322),and by Anhui Province Postdoctoral Researcher Scientific Research Activity Funding Project (2023B677), and by Yunnan Province Key Laboratory of Modern Genomics for Wild Relatives of Crops Fund Project (CWR-2024-03).

*Corresponding author.

the specificity of immunoreactions. Through colorimetric reactions with specific reagents prepared in advance and agricultural product samples extracted, this technique allows the observation of color changes on test cards for detection. Colloidal gold technology, requiring no expensive or complex equipment and highly skilled operators, has made pesticide residue testing more straightforward and economical due to its convenience.

In recent years, the rapid development of big data and artificial intelligence has significantly advanced the application of deep learning models in various fields, particularly image classification and object recognition. Convolutional neural networks (CNNs), which capture local information in images through convolutional kernels, have been widely adopted in tasks such as agricultural production [8, 11]. Notable CNNs models, such as VGGNet and ResNet, are widely employed. However, while CNNs excel at capturing local features via their local receptive fields, they fall short in fully modeling global dependencies. The Vision Transformer (ViT), which utilized a self-attention mechanism, addressed this limitation [12]. The Transformer model was later adapted for image classification, achieving performance comparable to or surpassing CNN models in various visual tasks [2]. ViT models are gaining widespread adaptation in agricultural applications, including pest detection, crop monitoring, and residue detection [3, 6, 8].

However, ViT models face challenges due to their high parameter count and significant computational requirements, limiting their deployment in resource-constrained environments. To mitigate these issues, model compression techniques, such as pruning, have been proposed to reduce model size without significantly compromising performance. Some researchers introduced a combined approach of pruning, quantization, and Huffman coding, reducing the storage requirements of deep neural networks by over 90% with only minimal performance loss [9]. Moreover, Voita et al. (2019) [13] demonstrated that removing certain attention heads in Transformer models does not substantially affect performance in natural language processing (NLP) tasks. Additionally, some studies have optimized Transformer architectures, such as Spatten, by pruning redundant tokens and attention heads [14].

In agricultural residue detection, particularly with colloidal gold assays, ViT models have shown promise in overcoming challenges, such as identifying trace residues and differentiating between residue types. Thus, this paper utilizes different scales of Vision Transformer models, including ViT-Base/16, ViT-Base/32, ViT-Small/16, and ViT-Tiny/16, and applies transfer learning methods for image classification of colloidal gold test kits. Additionally, the models are optimized through pruning. The study explores the impact of data augmentation on the performance of these ViT models. It compares the effects before and after pruning the qkv layer in classifying small-batch colloidal gold test kit datasets. The findings provide a theoretical basis for further research on ViT model image classification tasks.

2. Materials and methods

2.1. **Vision Transformer Model.** The Vision Transformer (ViT) model consists of three main components: Linear Projection of Flattened Patches, Transformer Encoder, and MLP Head. It also contains image patching (Patch Embedding), linear mapping, class identifiers, positional encoding, encoder layers, layer normalization,

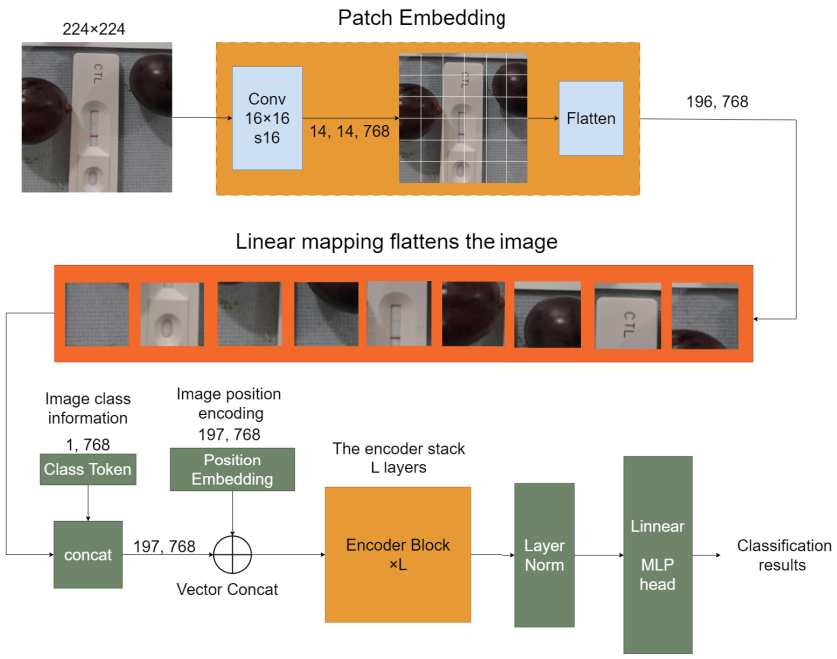

FIGURE 1. The Processing Flow of the Vision Transformer (ViT) Model

and an MLP output layer. The model effectively learns global features in images through patching, positional encoding, and the self-attention mechanism.

As shown in Figure 1, an input image size of 224 × 224 is segmented into multiple 14 × 14 patches using a kernel size of 16 × 16 and a stride of 16, resulting in patches with a feature dimension of 768. These patches are then flattened to form a matrix of size (196, 768), which is linearly mapped and processed into a sequence before inputting into the Transformer. A class identifier is appended at the beginning of the input sequence to aggregate information from the entire image for classification tasks. Positional encoding is applied to each patch, including the Class Token, to retain spatial information and help the model grasp the relative positions of different patches. The encoder consists of multiple stacked layers containing a self-attention mechanism and a feed-forward network (FFN), which extracts global image features. Following the encoder layers, the output undergoes layer normalization before entering the multilayer perceptron (MLP) head to generate the final classification results.

The Vision Transformer (ViT) model modifies the Transformer architecture for visual data. Standard Transformer modules require a two-dimensional matrix of tokens matrix as input. However, image data is typically represented as a three-dimensional tensor of shape [H, W, C]. An embedding layer utilized convolutional layers to convert this into a two-dimensional token matrix. Each transformed token vector is embedded with positional information using a specific positional encoding function before being fed into the Transformer module, comprising a Transformer Encoder and an MLP (Multilayer Perceptron).

The Vision Transformer (ViT) model modifies the Transformer architecture for visual data. Standard Transformer modules require a two-dimensional matrix of tokens matrix as input. However, image data is typically represented as a three-dimensional tensor of shape [H, W, C]. An embedding layer utilized convolutional layers to convert this into a two-dimensional token matrix. Each transformed token vector is embedded with positional information using a specific positional encoding function before being fed into the Transformer module, comprising a Transformer Encoder and an MLP (Multilayer Perceptron).

The input tokens undergo Layer Normalization in the Transformer Encoder before entering the multi-head attention layer. After a residual connection, the tokens are normalized again, passed through an MLP, and subjected to another residual connection to produce the output. This process is repeated L times, yielding feature representations that capture global information, in contrast to CNNs, which focus on extracting local information.

The key advantage of the Transformer Encoder is its ability to capture long-distance dependencies between different regions of an image, thereby improving performance in tasks such as image classification. Multi-head attention divides the input embedding into multiple subspaces (heads), computing self-attention in parallel across these subspaces. The outputs of these heads are concatenated to form the final attention output.

$$\text{Attention}(q, k, v) = \text{softmax}\left(\frac{qk^T}{\sqrt{d_k}}\right) v \tag{2.1}$$

$$\text{MultiHead}(q, k, v) = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_h\right) W_O \tag{2.2}$$

In equations (2.1) and (2.2), the input tokens are mapped to query (q), key (k), and value (v) vectors through the learnable matrices Wq, Wk, and Wv, where dk is the dimension of the query and key vectors. Each token performs a dot product with all the query and key vectors. After scaling and applying softmax normalization, the corresponding attention scores are computed. These attention scores reflect the relationship between tokens; the higher the score, the stronger the relationship between the tokens. This mechanism allows the model to establish long-range dependencies across different regions in the image.

The MLP Head, used for classification, consists of fully connected layers and activation functions. It further extracts and combines features through nonlinear transformations, enhancing the model's representational power and generalization ability. The classification results are output by the final MLP Head.

2.2. **Model pruning.** Model pruning targets trained models by removing redundant parameters based on criteria that evaluate the significance of specific parameters to the target task. This process compresses the model, reducing both the number of parameters and computational load while maintaining accuracy.

Common pruning algorithms include structured and unstructured pruning [7]. Structured pruning achieves compression by removing entire channels, layers, convolutional kernels, and other structural components of the network. This can include channel, filter, neuron, and layer pruning, as shown in Figure 2. Some neurons in

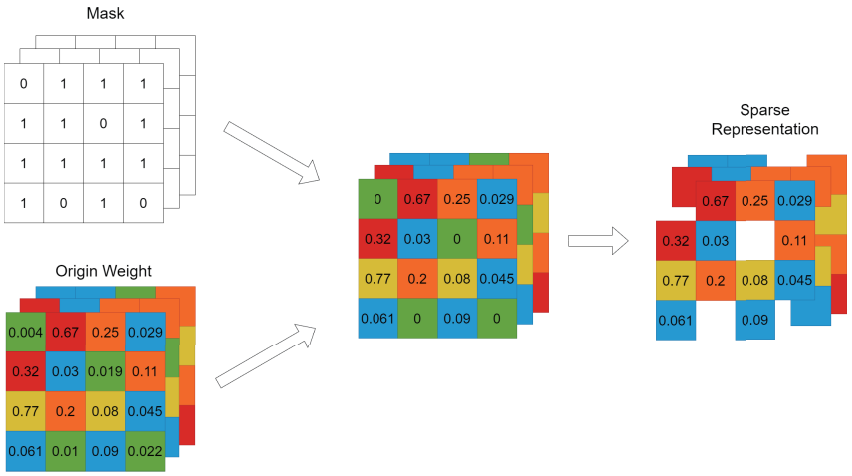

FIGURE 2. The application process of unstructured pruning in the ViT model.

neural networks do not capture useful information, and removing these neurons does not significantly affect network performance. Research has shown that smaller weights typically have a negligible impact on model accuracy.

Based on this approach, this study primarily prunes the multi-head self-attention layers of the ViT model, with a specific focus on the qkv linear layer. The pruning strategy is as follows: First, the original model is trained on the target dataset. Next, a pruning threshold is defined, and a unified mask is applied to all qkv layers. The pruning rate of the qkv layers is then computed, and the threshold is iteratively adjusted until it converges within a predefined range. Finally, this optimized threshold is applied for one-shot pruning, followed by fine-tuning to restore model accuracy.

2.3. **Model Evaluation Metrics.** In the experiments conducted in this paper, the hyper-parameters adjusted include batch size, learning rate, and the number of epochs. The batch size determines the number of samples used in a single training iteration. Larger batch sizes consume more memory and can speed up the training process, but may also lead to instability and convergence issues. Smaller batch sizes use less memory and train more slowly, but typically result in more stable converge. The learning rate controls the size of the steps taken during parameter updates. Larger learning may accelerate convergence but can cause oscillations around the optimal value or even lead to divergence, whereas a lower learning rate ensures more stable training but slower convergence. The number of epochs controls how often the model iterates over the entire training dataset. More epochs can help the model learn more from the data, but excessive epochs can lead to overfitting, decreasing the model's ability to generalize to new data.

The performance of the trained model is evaluated using a confusion matrix, which visualizes the classification results. Each column of the confusion matrix represents a predicted category, while each row represents the true category of the data. True Positive (TP) defines the number of instances the model accurately

identified as positive; True Negative (TN) defines the number of negative samples correctly classified as negative; False Positive (FP) defines the number of negative samples incorrectly classified as positive; and False Negative (FN) defines the number of positive samples incorrectly classified as negative. Equation (2.3) computes accuracy, defined as a ratio of correctly predicted samples to the total number of samples. Equation (2.4) calculates Precision, the proportion of actual positive cases among those predicted as positive. Equation (2.5) computes Recall, the proportion of true positive cases correctly identified by the model.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.4}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.5}$$

3. Results

3.1. **Experimental Environment.** The experimental setup was on a Windows 10 operating system platform, with an NVIDIA RTX3050 Laptop GPU with 4GB VRAM and an AMD R5 5600 CPU. The deep learning framework used was Pytorch. The input image resolution was set to 224×224, and images were normalized to [0, 1] before being fed into the model for training. All images were divided into training and validation sets in an 8:2 ratio. The training process of the neural network model was accelerated using the most commonly used Stochastic Gradient Descent (SGD) optimizer, which has the advantages of low computational cost and fast model convergence. All experiments were conducted in the aforementioned experimental environment.

The experiment employed unstructured pruning. A pretrained base ViT model was first loaded and trained on the target dataset. The trained model was then subjected to pruning in the q, k, and v layers of the Multihead Attention module, with a threshold set at 0.025. Weights in the qkv layers were compared against this threshold: weights exceeding the threshold were assigned a pruning mask of 0 (False), while those below the threshold were assigned a mask of 1 (True). Weights with a mask of 0 were pruned, whereas those with a mask of 1 remained unchanged, resulting in a pruned model. Finally, the model was fine-tuned to restore accuracy. To evaluate the sensitivity of the qkv layer to pruning, different pruning rates were tested on the ViT-Ti/16 model, specifically 15%, 35%, 75%, and 85%. For other models, pruning rates of approximately 35% and 70% were selected for the qkv layers, with an overall pruning rate fluctuating within $\pm$ 5%.

3.2. **Model Comparison Before and After Data Augmentation.** Determining the negativity or positivity of test reagents typically involves manually inspecting the test strips. If the upper band is lighter and the lower band is darker, or if both bands are of similar intensity, the result is considered negative; if the upper band is

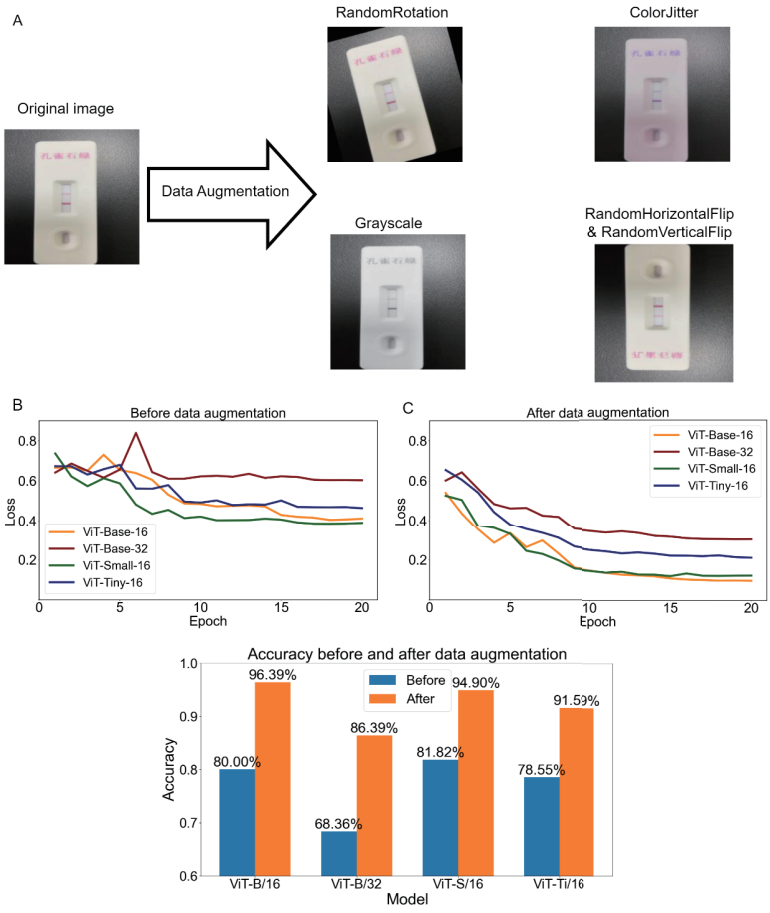

FIGURE 3. The application process of unstructured pruning in ViT models.

darker than the lower band, it is positive. Following this approach, training images were manually categorized into negative and positive classes.

The accuracy of each ViT model was compared before and after data augmentation. As shown in Figure 3D, blue bars represent accuracy before data augmentation, and orange bars represent accuracy after. All four ViT models exhibited significant improvements in accuracy following dataset enhancement, with increases of approximately 16%, 18%, 13%, and 13%, respectively. The training loss curve for the validation set in Figure 3C demonstrated faster converging with reduced fluctuation than the loss curve without data augmentation shown in Figure 3B. This indicates that data augmentation and dataset expansion improved the accuracy, generalization, and stability of the ViT models, making them more suited for image classification tasks.

When trained on the target dataset, the ViT-S/16 model performed slightly better than the ViT-B/16 without data augmentation. However, after augmentation, the ViT-B/16 emerged as the best-performing model, achieving a validation set accuracy of 96.39% with a loss close to 0.1. In contrast, the ViT-S/16 model had a

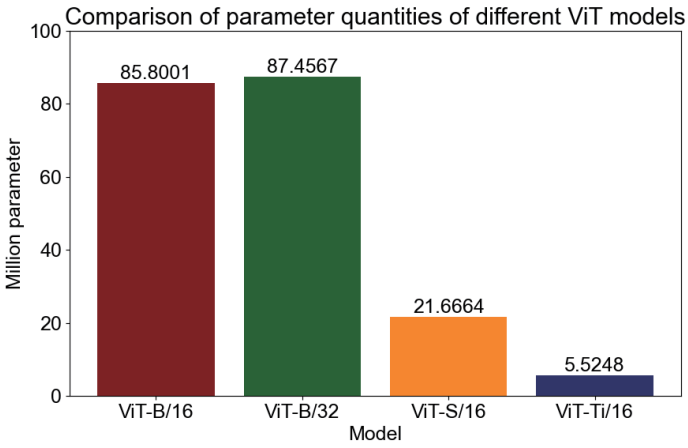

FIGURE 4. Comparison of parameter counts across different Vision Transformer (ViT) models.

slightly lower validation accuracy of 94.9% but a loss value similar to that of the ViT-B/16 model.

The parameter counts for ViT-B/16 and ViT-B/32 are the largest, at 85.8001M and 87.4567M, respectively, requiring greater computational resources. In contrast, the earlier models, ViT-S/16 and ViT-Ti/16, have significantly fewer parameters, at 21.6664M and 5.5248M, respectively. Despite ViT-Ti/16 having the smallest parameter count, ViT-S/16 delivers performance comparable to ViT-B/16 on a dataset of 10,000 images of colloidal gold test reagents, demonstrating its ability to maintain high performance with fewer parameters.

3.3. **ViT Model Training Optimization.** Building on the strengths of the aforementioned ViT-S/16 model, this study further optimized the training parameters of the ViT-S/16 model, using the validation set loss curve for assessment. By adjusting the Batch size, Learning rate, Epoch, and Optimizer, the study investigated the impact of these hyperparameters on the Validation Loss of the ViT model, as shown in Figure 5A, comparing SGD and Adam optimizers. Under the same number of training iterations, SGD demonstrated a faster reduction in loss, ultimately achieving a lower validation loss. In contrast, Adam showed greater fluctuations in loss during training and less effective convergence than SGD.

The results in Figure 5B indicate that the loss curves for batch sizes of 16 and 32 were nearly identical, both converging quickly to a lower loss value. The results in Figure 5C indicate that the model with a learning rate of 0.001 experienced a quicker decline in loss, ultimately reaching a lower loss value. In contrast, the model with a learning rate of 0.0003 declined more slowly.

Finally, adjustments were made to the Learning Rate Scheduling in the optimizer. As depicted in Figure 5D, training was conducted using Step Decay and Cosine Annealing strategies. Step Decay involved reducing the learning rate to 10% of its original value every eight iterations. The loss curve from Cosine Annealing was smoother, and the final results were slightly better than with step decay. Based

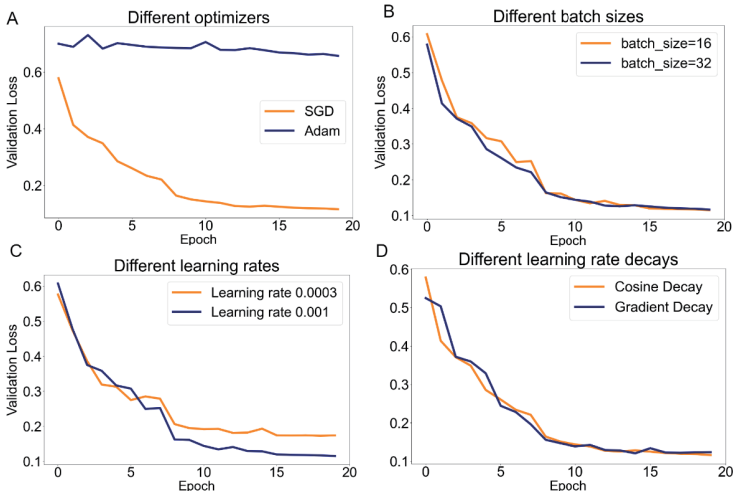

FIGURE 5. Comparison of validation set loss for ViT models based on different hyper-parameters.

on these experimental results, the hyperparameters were set to a Batch Size of 32, Learning Rate of 0.001, Epoch of 20, and using Cosine Annealing for learning rate scheduling.

3.4. **Model Pruning Comparison Results.** Based on the image dataset expanded to 10,000 samples, we conducted pruning experiments on four common ViT-Base models. The hyperparameters for model training were set as follows: the learning rate, batch size, and number of epochs were set to 0.001, 32, and 15, respectively. Consistent with previous experiments, the SGD optimizer and the cosine annealing learning rate scheduler were used. The experimental results are listed in Table 1.

TABLE 1. **Comparison of performance model accuracy in different pruning rate**

Model	Baseline Accuracy	Low pruning rate Accuracy	High pruning rate Accuracy
ViT-B/16	96.49%	97.13%	96.39%
ViT-B/32	86.39%	92.53%	92.28%
ViT-S/16	94.90%	97.08%	96.49%
ViT-Ti/16	91.59%	94.56%	94.01%

The accuracy of the ViT-B/16 and ViT-B/32 models was 96.49% and 86.39%, respectively, with ViT-B/16 exhibiting superior performance and higher accuracy. Both models have similar parameter counts of 85.80M and 87.45M, respectively. In comparison, the ViT-S/16 and ViT-Ti/16 models, which have fewer parameters (21.6M and 5.52M, respectively), still achieved relatively high accuracies of 94.90% and 91.59%, respectively, making them particularly well-suited for resource-constrained environments.

We first performed unstructured pruning on the qkv layers of the ViT-Ti/16 model, which has the smallest parameter count among ViT models, followed by accuracy recovery training to investigate the impact of different pruning rates on model accuracy. We applied four pruning rates for the ViT-Ti/16 model: 10%, 35%, 70%, and 85%. The results are summarized in Table 2.

TABLE 2. **Comparison of performance and number of parameters for different ViT-Ti/16 model pruning rates.**

qkv Layer Pruning Rate	Threshold	Accuracy(%)	Parameter Count
12.63%	0.009	94.56%	5.35M
33.67%	0.025	94.56%	5.08M
68.80%	0.060	94.01%	4.61M
86.03%	0.090	90.40%	4.38M

The model's accuracy remained almost unchanged at low pruning rates of 12.63% and 33.67%, with the 33.67% pruning rate providing a better compression result. At high pruning rates of 68.80% and 86.03%, the model with 68.80% pruning achieved approximately 3.6% higher accuracy than the one with 86.03% pruning. We applied 35% and 70% pruning rates to the qkv layers of various ViT models and conducted accuracy recovery training. The results after accuracy recovery are presented in Table 3.

TABLE 3. **Comparison of performance and parameter counts for low pruning rate.**

Model	Threshold	qkv Layer Pruning Rate	Accuracy(%)	Parameter Count
ViT-B/16	0.009	34.95%	97.13%	78.37M
ViT-B/32	0.025	39.19%	92.53%	79.13M
ViT-S/16	0.025	35.18%	97.08%	19.7M
ViT-Ti/16	0.025	33.67%	94.56%	5.08M
ViT-B/16	0.025	74.01%	96.39%	70.08M
ViT-B/32	0.060	75.93%	92.28%	71.33M
ViT-S/16	0.060	71.07%	96.49%	17.79M
ViT-Ti/16	0.060	68.80%	94.01%	4.61M

After applying low pruning rates to the qkv layers, the models exhibited notable improvements in accuracy compared to their original versions. For instance, the accuracy of ViT-B/16 increased from 96.49% to 97.53%, with a pruning rate of 20.64%, leading to a parameter reduction to 81.45M. Similarly, the accuracy of ViT-B/32 improved from 86.39% to 92.53%, with a pruning rate of 39.19% and a parameter reduction to 79.13M. Likewise, the accuracy of ViT-S/16 increased from 94.90% to 97.08%, with a pruning rate of 35.18%, reducing the model size to 19.7M. Finally, the accuracy of ViT-Ti/16 improved from 91.59% to 94.56%, with a pruning rate of 33.67% and a parameter reduction to 5.08M.

After applying high pruning rates to the qkv layer, the performance changes compared to the low pruning rate models were derived and summarized in Table 4.

TABLE 4. **Comparison of performance and parameter counts for high pruning rates**

Model	Threshold	qkv Layer Pruning Rate	Accuracy(%)	Parameter Count
ViT-B/16	0.025	74.01%	96.39%	70.08M
ViT-B/32	0.060	75.93%	92.28%	71.33M
ViT-S/16	0.060	71.07%	96.49%	17.79M
ViT-Ti/16	0.060	68.80%	94.01%	4.61M

The accuracy of ViT-B/16 decreased from 97.53% to 96.39%, with a pruning rate of 74.01% for the qkv layer and the parameter count reduced to 70.08M; the accuracy of ViT-B/32 decreased slightly from 92.53% to 92.28%, a reduction of about 0.03%, with a pruning rate of 75.93% for the qkv layer and the parameter count reduced to 71.33M; the accuracy of ViT-S/16 decreased slightly from 97.08% to 96.49%, with a pruning rate of 71.07% for the qkv layer and the parameter count reduced to 17.79M; the accuracy of ViT-Ti/16 decreased from 94.56% to 94.01%, with a pruning rate of 68.08% for the qkv layer and the parameter count reduced to 4.61M. Although some models show a slight drop in accuracy, overall, the changes in accuracy demonstrate the effectiveness of pruning optimization in these models.

Furthermore, under lower pruning rates, the results show that in Table 5, ViT models of base size and below show improvements in accuracy while reducing the model's parameter count, thus enhancing their generalization ability.

TABLE 5. **Comparison of performance model accuracy in different pruning rate**

Model	Baseline Accuracy	Low pruning rate Accuracy	High pruning rate Accuracy
ViT-B/16	96.49%	97.13%	96.39%
ViT-B/32	86.39%	92.53%	92.28%
ViT-S/16	94.90%	97.08%	96.49%
ViT-Ti/16	91.59%	94.56%	94.01%

At higher pruning rates, the accuracy values of ViT-B/32, ViT-S/16, and ViT-Ti/16 models slightly dropped compared to those of lower pruning rate models, exceeding those of the original models. The accuracy of the ViT-B/16 model slightly decreased, but its parameter count dropped from 85.8M to 70.08M, being beneficial for deployment. ViT-S/16 and ViT-Ti/16 strike the best balance between parameter count and accuracy. In the lower pruning rate experiments, all models show improvement in accuracy. All models, except ViT-B/16, improve accuracy in the higher pruning rate experiments. Particularly, the ViT-S/16 model slightly outperforms the ViT-B/16 model, delivering the best performance overall.

The confusion matrix reflects the predictive performance of classification models across different categories, effectively demonstrating the balance of the classification

model. As shown in Figure 6, in the confusion matrices after accuracy recovery for the four types of ViT models, all four models perform well-balanced classifications, with the ViT-S/16 model showing the best performance.

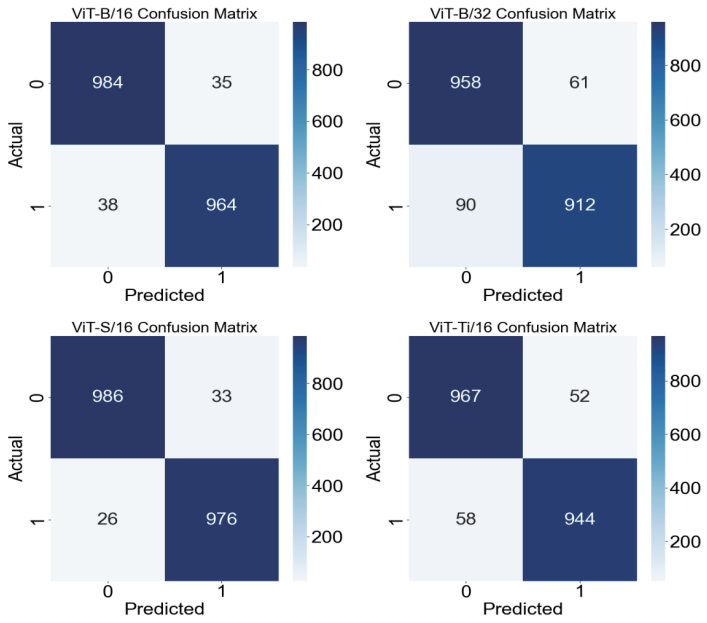

FIGURE 6. The confusion matrix reflects the predictive performance of classification models

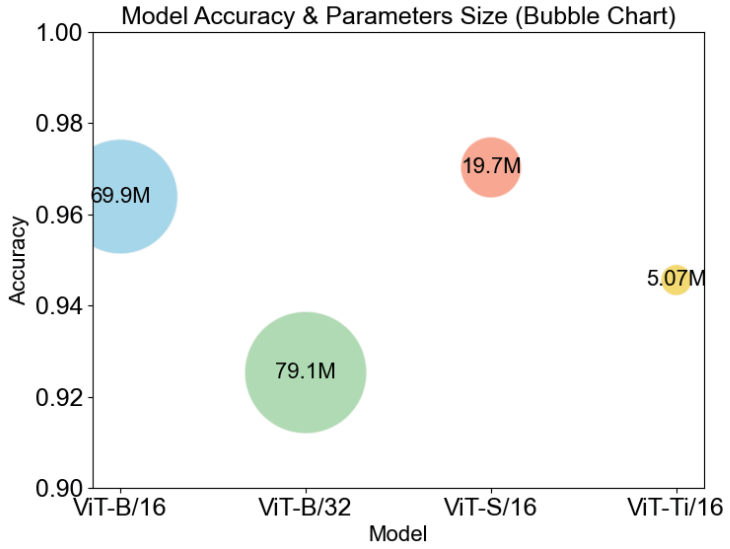

FIGURE 7. Bubble chart of accuracy and parameter count after model pruning.

Additionally, this study explores the relationship between accuracy and parameter count across different ViT models. Figure 7 shows that the ViT-B/16 and ViT-S/16 models achieve high accuracies close to 0.98 and 0.97, respectively, with fewer parameters. Although the ViT-Ti/16 has the smallest parameter count (5.07M), its accuracy is lower than the first two, at about 0.95. However, the ViT-B/32 has a larger parameter count and lower accuracy. Thus, the results suggest that the ViT-S/16 is optimal.

Finally, this study evaluated the accuracy of different pruned ViT models on the validation set, as shown in Figure 8A. The results indicate that the pruned ViT-S/16 model achieved the highest accuracy on the validation set, approaching 0.97, demonstrating excellent pruning effects. Both ViT-B/16 and ViT-Ti/16 also maintained high accuracies, each above 0.90; however, the validation accuracy of ViT-B/32 was lower, not achieving the same performance level as the other models post-pruning. Additionally, Figure 8B shows the loss curves, with ViT-S/16 having the lowest and most stable convergence, consistent with high accuracy. The loss curves for ViT-B/16 and ViT-Ti/16 were relatively low and showed little variation. The loss curve for ViT-B/32 fluctuated more and did not converge effectively.

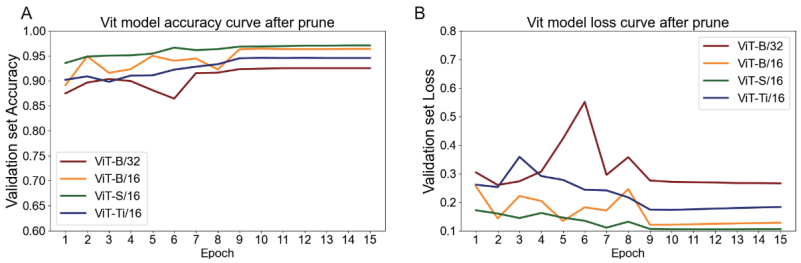

FIGURE 8. Comparison of loss curves before and after pruning.

4. CONCLUSION

This study compared the performance of Vision Transformer (ViT) models of different scales on a small-batch colloidal gold test kit dataset, utilizing data augmentation, model training optimization, and qkv layer compression techniques. The results indicate that data augmentation significantly improved the accuracy and generalization capability of the ViT models in colloidal gold test kit image classification, particularly under varying image quality conditions. Data augmentation techniques effectively enhanced model recognition accuracy. After training optimization, the ViT-S/16 model achieved optimal performance using the cosine annealing algorithm for learning rate decay adjustment, further enhancing training efficiency.

In the qkv layer compression experiments, accuracy improved across all models at low pruning rates, with the ViT-B/16 model achieving the highest accuracy of 97.13%. Although some models exhibited a slight decline in accuracy as pruning rates increased, ViT-S/16 outperformed ViT-B/16 under higher pruning rates while significantly reducing the parameter count. These findings confirm the effectiveness of qkv layer compression in improving model efficiency and reducing computational resource consumption. The experiments successfully classified colloidal gold test kit

images, demonstrating the potential of qkv layer compression methods for agricultural detection tasks.

The innovation of this study lies in proposing a new strategy to optimize ViT models through qkv layer compression, providing a theoretical foundation for model optimization by balancing pruning rates. The results indicate that at low pruning rates, accuracy loss is minimal, while at higher pruning rates, ViT-B/16 experiences only a limited performance decline. This suggests that the proposed optimization method effectively enhances model efficiency while maintaining high accuracy. Future research could explore compression techniques applied to different layers and extend this optimization approach to a broader range of agricultural detection tasks, further improving model accuracy and computational efficiency.

References

[1] Á. Ambrus, V. V. N. Doan, J. Szenczi-Cseh, H. Szemánné-Dobrik and A. Vásárhelyi, *Quality control of pesticide residue measurements and evaluation of their results*, Molecules **18** (2023): 94.

[2] V. G. Dhanya, A. Subeesh, N. L. Kushwaha, D. K. Vishwakarma, T. N. Kumar, G. Ritika and A. N. Singh, *Deep learning based computer vision approaches for smart agricultural application*, Artificial Intelligence in Agriculture **6** (2022), 211–229.

[3] Y. Dong, X. Yao, W. Zhang and X. Wu, *Development of simultaneous determination method of pesticide high toxic metabolite 3,4-Dichloroaniline and 3,5 Dichloroaniline in Chives using HPLC-MS/MS*, Foods **12** (2023): 2875.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby, *An image is worth* 16×16 *words: Transformers for image recognition at scale*, in: Proceedings of the International Conference on Learning Representations (2021). arXiv: `https://arxiv.org/abs/2010.119292010.11929`.

[5] S. Han, H. Mao and W. J. Dally, *Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding*, 2015, arXiv preprint arXiv:1510.00149.

[6] M. A. Haque, C. K. Deb, S. Marwaha, S. Dutta, M. U. D. Shah, A. Saikia and A. Shukla, *Rice disease identification using vision transformer (ViT) based network*, in: Proceedings of the International Conference on Deep Learning, Artificial Intelligence, and Robotics, Springer, Cham, 2024, pp. 732–741.

[7] P. Molchanov, S. Tyree, T. Karras, T. Aila and J. Kautz, *Pruning convolutional neural networks for resource efficient inference*, 2016, arXiv preprint arXiv:1611.06440.

[8] H. Pan, L. Xie and Z. Wang, *Plant and animal species recognition based on dynamic vision transformer architecture*, Remote Sensing **14** (2022): 5242.

[9] Q. Pan, M. Gao, P. Wu, J. Yan and M. A. AbdelRahman, *Image classification of wheat rust based on ensemble learning*, Sensors **22** (2022): 6047.

[10] K. Shaheed, I. Qureshi, F. Abbas, S. Jabbar, Q. Abbas, H. Ahmad and M. Z. Sajid, *EfficientRMT-Net–An efficient ResNet-50 and vision transformers approach for classifying potato plant leaf diseases*, Sensors **23** (2023): 9516.

[11] J. A. Sofi, A. A. Dar, I. Jan, G. I. Hassan, S. R. Dar, A. H. Mughal and N. A. Dar, *Development and validation of gas chromatography with electron capture detection method using QuEChERS for pesticide residue determination in cucumber*, Biomedical Chromatography **37** (2023): e5647.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez and I. Polosukhin, *Attention is all you need*, Advances in Neural Information Processing Systems **30** (2017), 5998–6008.

[13] E. Voita, D. Talbot, F. Moiseev, R. Sennrich and I. Titov, *Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned*, 2019, arXiv preprint arXiv:1905.09418.

[14] H. Wang, Z. Zhang and S. Han, *Spatten: Efficient sparse attention architecture with cascade token and head pruning*, in: 2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA), 2021, pp. 97–110.

Manuscript received October 24, 2024
revised February 21, 2025

Y. Z Peng
School of Computer and Artificial Intelligence, Hefei Normal University, Hefei, China
E-mail address: myemil2023@163.com

Y. J. Gao
School of Information and Artificial Intelligence, Anhui Agricultural University, Hefei, China
E-mail address: gaoyujia@ahau.edu.cn

F. Xie
School of Computer and Artificial Intelligence, Hefei Normal University, Hefei, China
E-mail address: xiefei@hfnu.edu.cn

Z. Y. Guo
School of Computer and Artificial Intelligence, Hefei Normal University, Hefei, China
E-mail address: zhiyuan_guo@hfnu.edu.cn

Q. J. Gao
School of Computer and Artificial Intelligence, Hefei Normal University, Hefei, China;
Xie Yuda Tea Co., Ltd. Postdoctoral Research Station, Huangshan, China
E-mail address: gaoqijuan@hotmail.com