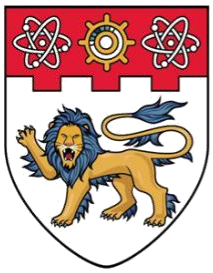

NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

SC4001: Neural Networks and Deep Learning

Group Report

No.	Name	Matriculation Number
1	Banerjee Mohor	U2222858E
2	Leong Kit Ye	U2121111K
3	William Tian	N2400769K

1. Introduction

Our team selected Task F - Flowers Recognition, using the Flowers102 dataset, which contains 102 flower species with high visual similarity and variability in scale and orientation, making it a challenging fine-grained classification task. We experimented with ResNet, VGG16, and Vision Transformers, applying enhancements like transfer learning, deformable convolutions, triplet loss, MixUp and CutMix, few-shot learning, focal margin loss and visual prompt tuning. Through these varied approaches, we aimed to identify the best model strategies for achieving high accuracy on this complex dataset.

2. Review of Existing Techniques

For Flowers102 classification, early methods used hand-crafted features and simple classifiers like SVMs but didn't capture enough detail. Deep learning models like VGG16, ResNet, and EfficientNet improved accuracy significantly, especially with transfer learning. Recently, approaches like Vision Transformers and advanced data augmentation have achieved state-of-the-art performance, making flower classification more accurate and reliable.

3.Experiments and Results

3.1. Transfer Learning on ResNet-18 and ResNet-34

The ResNet (Residual Networks) architecture is a deep convolutional neural network architecture that introduced residual connections, mitigating the vanishing gradient problem in deep networks by allowing gradients to flow directly through shortcut connections. ResNet-18 is one of the more computationally efficient models in the ResNet family, with 18 layers containing 4 stages consisting of two residual blocks each. As the Flowers dataset is relatively small, we utilize transfer learning to take advantage of the pre-trained features learned from ImageNet and fine-tune the model to adapt to the flower classification task.

3.1.1 Methodology

First, we train a ResNet-18 model scratch on the Flowers dataset without pretrained weights to establish a baseline. We then progressively froze 1 to 4 stages in ResNet-18 pretrained on ImageNet, fine-tuning the rest to find the best balance between knowledge retention and dataset-specific tuning.We trained the model using Adam optimizer, with learning rate of 0.001 and batch size of 32. Early stopping was implemented with a patience of 5 epochs. The same progressive freezing tests were conducted on ResNet-34 to evaluate the impact of model depth on performance.

3.1.2 Results

The optimal model was ResNet-18 with 3 frozen stages, achieving a test accuracy of 88.16%. The baseline ResNet-18 with no frozen layers achieved 84.76% accuracy, while freezing all 4 stages reduced accuracy to 81.54%. The optimal model for ResNet-34 was similarly the one with 3 frozen stages,

achieving a test accuracy of 81.62%. All ResNet-34 models achieved lower validation accuracy compared to their corresponding ResNet-18 counterparts.

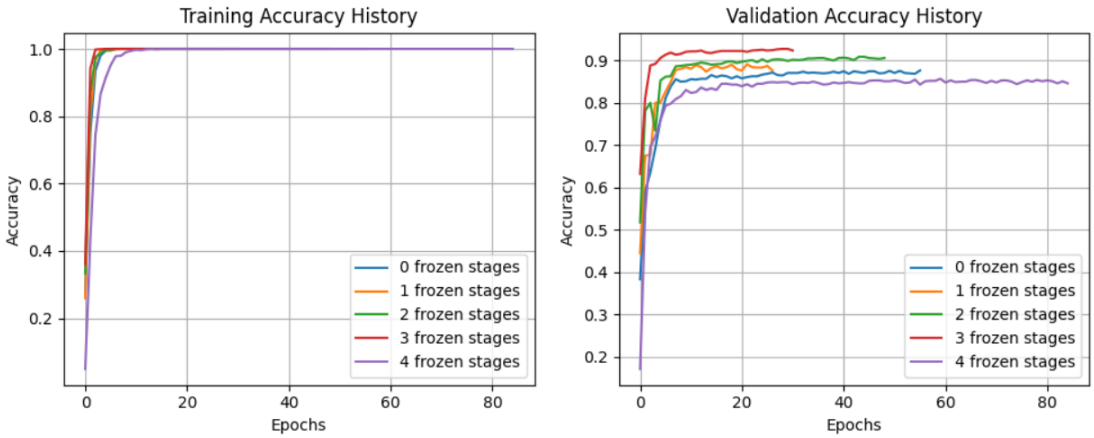

Figure 1. Histories of Training and Validations Accuracies for ResNet-18 Models with Frozen Stages

3.1.3 Discussion

This outcome highlights the effectiveness of transfer learning for small datasets like Flowers102. Freezing the early stages of a pretrained model, which capture general features like edges and textures, allows the model to retain useful ImageNet knowledge, while fine-tuning the later stages enables adaptation to the specific, fine-grained characteristics of flower species. We found that freezing the first 3 stages of ResNet-18 struck the best balance, preventing overfitting yet allowing the model to learn flower-specific details. ResNet-34, despite its greater depth and higher parameter count, underperformed due to overfitting, demonstrating that deeper models are not always advantageous for small datasets.

3.2 Deformable Convolutions on ResNet-18

We enhanced traditional convolutions by introducing deformable convolutions, which allow adaptive offsets in the sampling locations, enabling the receptive field to adjust dynamically rather than following a fixed grid. This approach learns spatial offsets at each position, helping the network better capture local features with complex shapes, rotations, and deformations—particularly applicable to the Flowers102 dataset, where flowers vary widely in scale and orientation. In ResNet-18, we integrated deformable convolutions by replacing standard convolution layers within the residual blocks and trained the model both with and without to evaluate its effectiveness in capturing low-level and high-level features.

3.2.1 Methodology

We tested replacing all convolutional layers in ResNet-18's first stage (Stage 1), middle stage (Stage 3), and last stage (Stage 4) with deformable convolutions. We then trained the model in both pretraining and non-pretraining conditions. For the pretrained model, the stages before the deformed convolutional stages were frozen. This was to find the best balance between retaining pretrained knowledge and fine-

tuning for Flowers102 with deformable convolutions, and to investigate whether deformed convolutions can outperform pretrained regular convolutions.

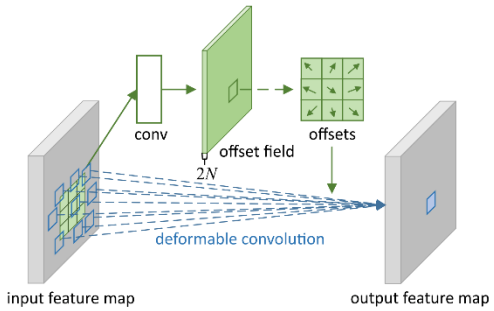

Figure 2. Visualisation of Deformable Convolutions

3.2.2 Results

The optimum model was pretrained ResNet-18 with deformable convolutions in the middle stage, achieving a test accuracy of 64.45%. However, all models underperformed compared to the baseline model of ResNet-18 trained from scratch with regular convolutions.

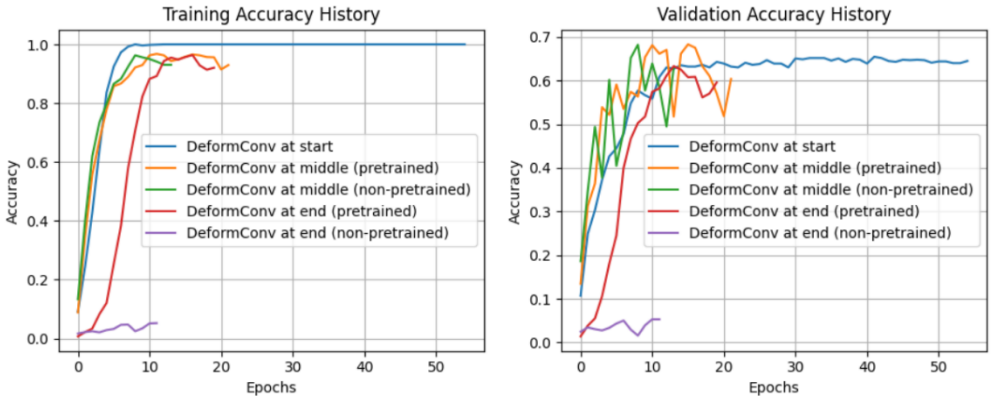

Figure 3. ResNet-18 with Transfer Learning and Frozen Stages: Training and Validation Metrics Plots

3.2.3 Discussion

The results were surprising, as we initially hypothesized that deformable convolutions would improve the model's performance by allowing it to better handle spatial variations in the flower images. Despite the adaptive nature of deformable convolutions, the results suggest that the added complexity of deformable convolutions might have hindered the model's ability to generalize effectively on the Flowers102 dataset, leading to overfitting. This can be seen from the fact that a deformable convolution model trained from scratch always achieves a lower validation accuracy than the same regular convolutional model trained from scratch.

3.3 Using ResNet-18 with Triplet Loss

We used ResNet-18 with a custom adaptation for triplet loss, which structures training samples into three images: an anchor, a positive image from the same class, and a negative image from a different class. This setup encourages the network to reduce the distance between the anchor and positive images while

increasing the distance between the anchor and negative image by a specified margin, helping the model create a feature space with better separation between similar and different classes.

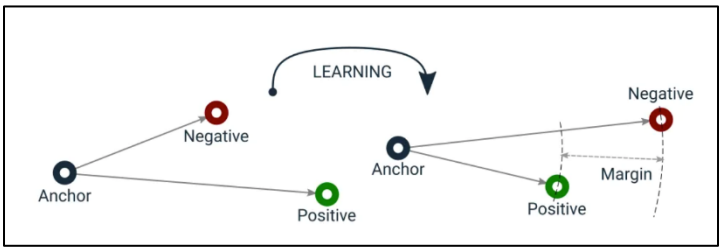

Figure 4. Triplet Loss

3.3.1 Methodology

We trained a ResNet-18 model on Flowers102 with triplet loss, using anchor-positive-negative triplets to enhance class separability. The pre-trained model was fine-tuned with a 128-dimensional embedding layer, using a batch size of 32 and early stopping on validation accuracy.

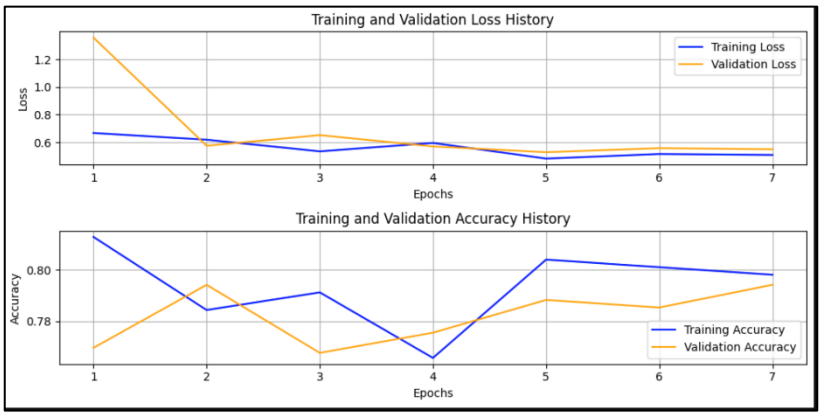

Figure 5. ResNet with Triplet Loss: Training and Validation Metrics Plots

3.3.2 Results

The model's initial training accuracy was 81.27%, and validation accuracy was 76.96%. Validation accuracy peaked at 79.41% in epoch 2, with training and validation losses plateauing early, indicating quick convergence. The final test accuracy achieved was 77.54%.

3.3.3 Discussion

The use of triplet loss enabled ResNet-18 to quickly learn a discriminative feature space for fine-grained flower classification, separating classes effectively and achieving high accuracy early in training. While a plateau in validation accuracy suggested some overfitting due to the dataset's small size, the model still achieved a strong test accuracy of 77.54%, showing good generalization. Overall, the results demonstrate that triplet loss effectively created a compact, class-separable embedding space, making it suitable for tasks with subtle inter-class differences, even with limited data.

3.4 MixUp & CutMix

MixUp is a technique referring to combining two images by producing a linear interpolation between them. Whereas CutMix is combining pairs of images by replacing a portion of an image with another one. These techniques allow models to better generalize by introducing diverse image variations in each batch, making them less prone to overfitting on specific features present in the training set.

Figure 6. Example images of MixUp, where two flower images are blended

Figure 7. Example images of CutMix, where a section of one flower image is overlaid onto another

3.4.1 Methodology

We finetuned a pre-trained ResNet-18 model with early stopping to prevent overfitting. Each batch used either CutMix or MixUp augmentation, chosen randomly, to expose the model to both methods.

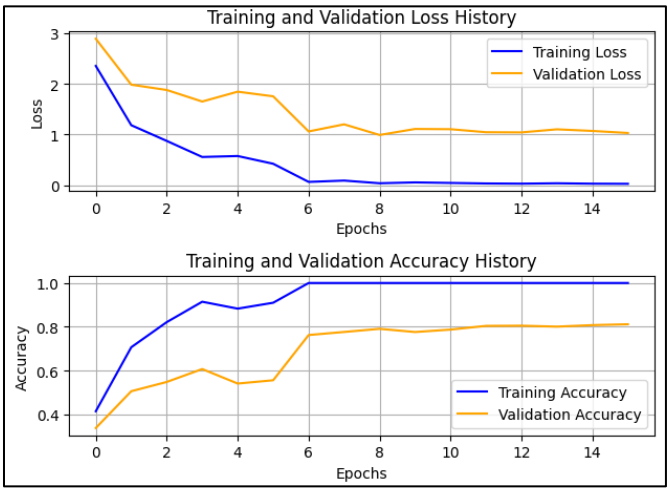

Figure 8. MixUp & CutMix: Training and Validation Metrics Plots

3.4.2 Results

The highest validation accuracy achieved was 81.17% early stopping after epoch 15, and the model achieved test accuracy of 76.84%.

3.4.3 Discussion

Our results indicate that CutMix and MixUp are effective for classification but could be further improved. These augmentations create mixed labels with weighted class probabilities (e.g., 70% rose, 30% tulip), rather than single-label outputs which increases complexity. Tracking the top predicted classes could

yield a more accurate evaluation for these mixed-label inputs. Additionally, applying both MixUp and CutMix simultaneously might enhance feature capture, adding complexity to the training process.

3.5 Few-Shot Learning using Prototypical Networks

We applied a Few-Shot Learning approach using Prototypical Networks (PN) on the dataset to handle fine-grained classification. This method builds “class prototypes” by averaging feature embeddings of support images from each class, enabling classification by measuring distances to these prototypes.

3.5.1 Methodology

We used a modified ResNet-18 model, termed EmbeddingNet, with L2 normalization for stability. In each episode, we selected a subset of classes (n-way), with a limited number of labelled examples (k-shot) as support and additional examples for queries. Prototypes were computed for each class from the support set, and query images were classified based on their closest prototype. An early stopping mechanism was applied to prevent overfitting.

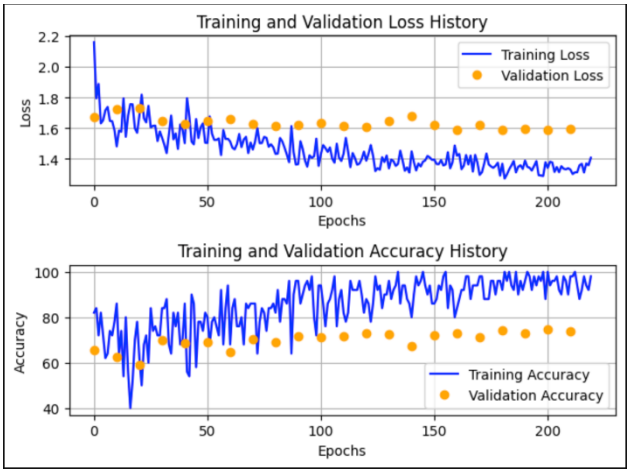

Figure 9. Few-Shot Learning: Training and Validation Metrics Plots

3.5.2 Results

The best validation accuracy achieved was 74.8%, and the final test accuracy was 81.16%. The model showed steady improvements in training accuracy, though validation accuracy plateaued, suggesting a limit on the model’s generalization with Few-Shot Learning.

3.5.3 Discussion

Prototypical Networks effectively utilized limited samples, achieving reasonable accuracy by focusing on prototype-based classification. However, validation results indicated some overfitting, likely due to the complexity of the Flowers102 dataset and limited labelled data. The use of L2-normalized embeddings provided stability, while early stopping helped to maintain generalizability.

3.6 VGG16 with Multi-Scale and Squeeze-and-Excitation Blocks

The original VGG16, developed by the Visual Geometry Group, consists of five blocks of 3x3 convolutional layers followed by max-pooling, with three fully connected layers for classification. However, it lacks features like multi-scale extraction and attention, which improve pattern capture and feature focus. To adapt VGG16 for fine-grained flower classification, we added a Multi-Scale Convolutional Residual Block and a Squeeze-and-Excitation (SE) Block. The multi-scale block uses parallel 1x1, 3x3, and 5x5 convolutions, for dimensionality reduction, capturing finer textures and capturing broader patterns respectively. The SE block adds channel-wise attention: a “squeeze” step compresses each channel to a single value via global average pooling, while an “excitation” step uses two fully connected layers to generate weights, highlighting informative channels. The modified VGG16 integrates these blocks into the later layers for complex feature extraction, with an updated classification head that includes batch normalization and dropout to improve generalization.

3.6.1 Methodology

The modified VGG16 was trained using cross-entropy loss, Adam optimizer (0.0001 learning rate), batch size of 32, and early stopping after 5 epochs. We tested two baselines—one with VGG16’s convolutional layers frozen, fine-tuning only the classifier, and another with full fine-tuning of all layers.

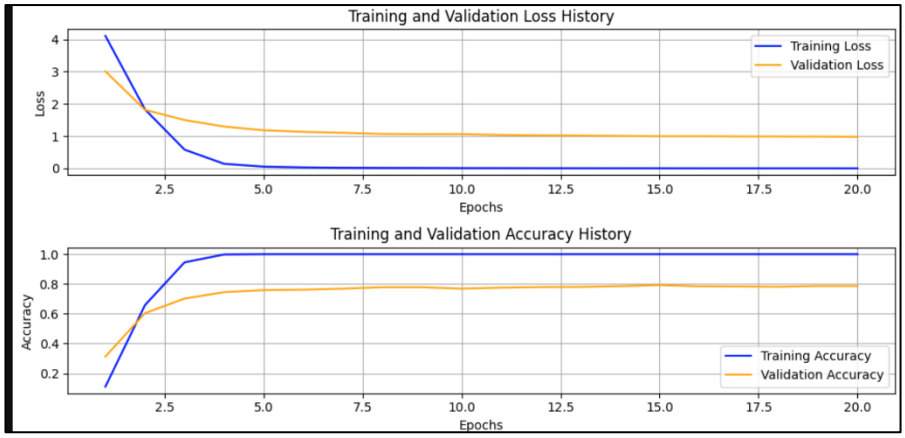

Figure 10. VGG16 with Multi-Scale and SE Blocks: Training and Validation Metrics Plots

3.6.2 Results

The modified VGG16 achieved a validation accuracy of 79.22% and a test accuracy of 77.79%. The fully frozen VGG16 baseline, where only the classifier was trained, reached a test accuracy of 72.22%. The fully fine-tuned VGG16 baseline achieved a test accuracy of 72.08%.

3.6.3 Discussion

The modified VGG16 outperformed the baselines, demonstrating that multi-scale feature extraction and channel attention effectively captured fine details in the dataset. Multi-scale convolutions handled diverse feature sizes, while SE blocks emphasized relevant channels, boosting accuracy. In comparison, the fully

frozen VGG16 lacked adaptability, and the fully fine-tuned version showed slight overfitting. The modified VGG16 balanced stability and flexibility, enhancing generalization to Flowers102's intricate patterns.

3.7 Vision Transformer with Focal Margin Loss

The Vision Transformer (ViT) is a deep learning model that applies transformer architectures, traditionally used in NLP, to image classification by treating images as sequences of patches. To enhance its performance on fine-grained tasks, we implemented a custom Focal Margin Loss. This loss combines focal loss, which emphasizes challenging examples, with a margin component that enforces clear boundaries between classes, improving generalization on datasets with subtle inter-class differences.

3.7.1 Methodology

We trained the ViT-Base model (with a 16x16 patch size) with Focal Margin Loss, applying early stopping to prevent overfitting. Two baselines were established: a fully fine-tuned ViT and a fully frozen ViT (training only the classifier head).

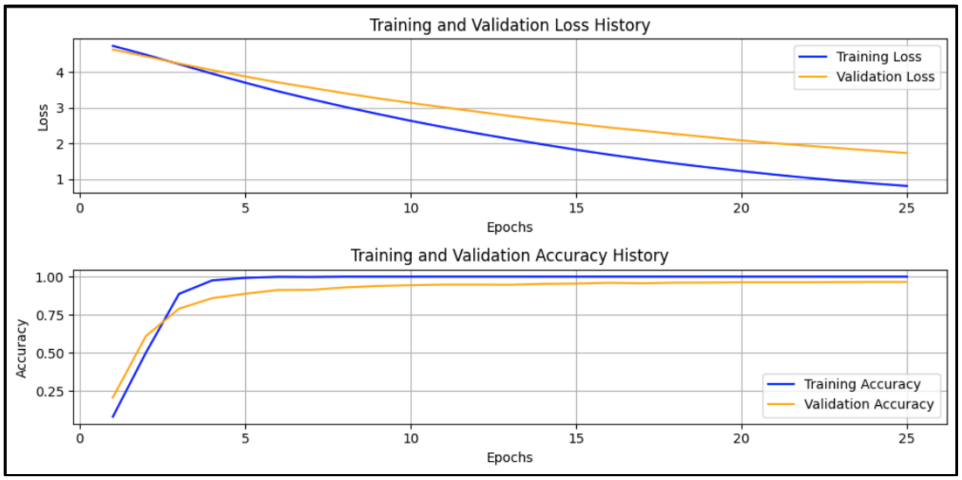

Figure 11. ViT with Focal Margin Loss: Training and Validation Metrics Plots

3.7.2 Results

ViT with Focal Margin Loss achieved a test accuracy of 95.10%, showing superior performance. The fully fine-tuned ViT baseline reached a test accuracy of 94.29%, demonstrating adaptability with all layers trainable. The fully frozen ViT baseline achieved a test accuracy of 91.19%, reflecting limited adaptability.

3.7.3 Discussion

Focal Margin Loss effectively improved accuracy by helping the model focus on hard-to-classify samples while maintaining clear class boundaries. The focal component prioritized challenging examples, essential for datasets like Flowers102 with similar classes, while the margin component reduced class overlap in the feature space, enhancing generalization. This balanced approach outperformed both fully frozen and fully fine-tuned baselines in fine-grained classification.

3.8 Visual Prompt Tuning in Visual Transformer

We used visual prompt tuning as an efficient alternative to full fine-tuning for the ViT , introducing learnable prompt tokens that act as task-specific guides by being prepended to each image's input sequence. During training, only these prompt tokens are updated, while all other ViT parameters remain frozen, leveraging the model's pre-trained knowledge with minimal adjustments. These tokens gradually encode task-specific information, enabling targeted adaptation without altering the main weights.

3.8.1 Methodology

We fine-tuned a ViT-Base model (with a 16x16 patch size) on the Flowers102 dataset, testing prompt lengths of 5, 10, 15, 25, 50, 75, 100, 500, and 1000 tokens to enhance performance. Training and validation losses were monitored with early stopping.

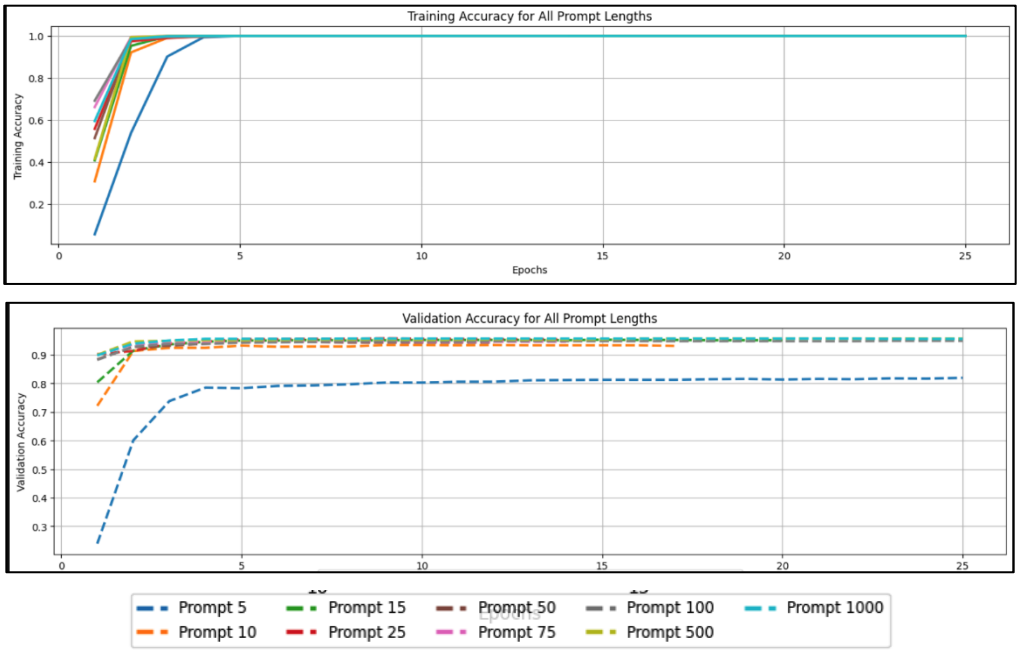

Figure 12. Variation with Prompt Length: Training and Validation Metrics Plots

3.8.2 Results

500 tokens (chosen by hyperparameter tuning) achieved the highest test accuracy of 95.3%. Shorter prompts (5 and 10 tokens) showed higher validation losses and lower accuracies. Full fine-tuning also achieved 94.29% test accuracy, showing strong adaptability with all parameters trainable. Full freezing resulted in a test accuracy of 91.19%, reflecting limited adaptability with fixed weights.

3.8.3 Discussion

Our results show that prompt tuning with 500 tokens outperforms full fine-tuning and the fully frozen baseline on Flowers102, training far fewer parameters and adapting well to dataset-specific features. A prompt length of 500 tokens provided an optimal balance, delivering sufficient task context without unnecessary complexity, as shorter prompts lacked information and longer ones offered diminishing returns. Overall, prompt tuning proves to be a practical, cost-effective approach, achieving high accuracy with reduced computational demands.

Model	**Test Accuracy (%)**
ResNet-18 trained from scratch (baseline)	84.76
Pretrained ResNet-18 with 3 frozen stages	88.16
Pretrained ResNet-34 with 3 frozen stages	81.62
Pretrained ResNet-18 with mid-stage deformable convolutions	64.45
Pretrained ResNet-18 with Triplet Loss	77.54
Pre-trained ResNet-18 MixUp & CutMix	76.84
Few-Shot Learning approach using Prototypical Networks	81.16
VGG16 with Multi-Scale and Squeeze-and-Excitation Blocks	72.08
Vision Transformer with Focal Margin Loss	95.10
Visual Prompt Tuning in Visual Transformer	**95.30 (best)**

Table 1. Compilation of Results

4. Conclusion and Future Improvements

In conclusion, Visual Prompt Tuning and Focal Margin Loss in ViT outperformed all other models on Flowers102, achieving test accuracies of 95.3% and 95.1%, respectively. These custom adaptations proved effective, leveraging ViT's capacity to focus on challenging samples and adapt with minimal parameters. ResNet-18 with targeted transfer learning also performed well, achieving 88.78%. Overall, lightweight, targeted modifications emerged as highly effective alternatives to full fine-tuning. Future improvements could explore alternative forms of prompting, such as language-based prompts or learned prompts that could provide even more flexibility in how the model attends to important features.

5. References

Chen, J. (2023). Optimized Hybrid Focal Margin Loss for Crack Segmentation. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2302.04395

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y. (2017). Deformable Convolutional Networks. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/1703.06211

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., … Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2010.11929

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/1512.03385

Jia, M., Tang, L., Chen, B.-C., Cardie, C., Belongie, S., Hariharan, B., & Lim, S.-N. (2022). Visual Prompt Tuning. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2203.12119

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/1409.1556

Snell, J., Swersky, K., & Zemel, R. S. (2017). Prototypical Networks for Few-shot Learning. arXiv [Cs.LG]. Retrieved from http://arxiv.org/abs/1703.05175

Van der Merwe, R. (2020). Triplet Entropy Loss: Improving The Generalisation of Short Speech Language Identification Systems. CoRR, abs/2012.03775. Retrieved from https://arxiv.org/abs/2012.03775

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., & Yoo, Y. (2019). CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/1905.04899

Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2018). mixup: Beyond Empirical Risk Minimization. arXiv [Cs.LG]. Retrieved from http://arxiv.org/abs/1710.09412

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., … He, Q. (2020). A Comprehensive Survey on Transfer Learning. arXiv [Cs.LG]. Retrieved from http://arxiv.org/abs/1911.02685