ECE371 Neural Networks and Deep Learning Assignment 1

WeiHao He 22308047
Sun Yat-sen University, Shenzhen Campus
hewh28@mail2.sysu.edu.cn

May 14, 2025

Abstract

The purpose of this document is to provide both the basic template of the assignment and submission guidelines. Abstracts should be a single paragraph, between 4–6 sentences long, ideally. Briefly describe the work you have completed and the insights you have gained. The main text should not exceed 4 pages, excluding references.

1 Introduction

The core content of this assignment is to explore the practical application effects of transfer learning in image classification tasks. Personally, I believe that the value of this experiment lies not only in implementing a classification model but more importantly in understanding the workflow and critical considerations of deep learning in practical applications. Through this experiment, we can intuitively feel the powerful capabilities of pre-trained models and the impact of techniques such as data augmentation on model performance.

2 Method

In the data augmentation section, I used five transformation methods:

- Random cropping and scaling to ensure the model adapts to different compositions.
- Horizontal flipping to simulate mirror angles.
- Color jitter to enhance lighting robustness.
- Rotation and flat shifting to simulate changes in shooting angles.

These transformations work together to effectively expand limited datasets, prevent overfitting, and improve the model's generalization ability.

In terms of model structure adjustment, the ResNet18 pre-trained feature extraction layer is retained, and only the final fully connected layer is replaced. This transfer learning strategy fully utilizes the general visual features learned by the ImageNet pre-trained model, significantly reducing training costs. The output layer dimension is set to the number of flower categories to adapt the model to the new classification task.

The choice of loss function is cross-entropy loss, which is the standard choice for multi-classification problems and can effectively handle class probability distributions. The optimizer uses SGD with momentum, an initial learning rate of 0.001, and a `StepLR` scheduler (decays by 0.1 times every 7 epochs). This combination ensures fast convergence in the initial stage and fine-tuning of parameters in the later stage. A momentum term of 0.9 helps accelerate convergence and escape from local optima.

The backpropagation implementation adopts the PyTorch automatic differentiation system, which automatically calculates gradients and performs parameter updates. Choose `state_dict` instead of the complete model for model saving, which not only saves storage space but also ensures loading flexibility. The best model is selected based on the accuracy of the validation set, ensuring the use of the version with the strongest generalization ability during deployment.

These choices together constitute an efficient transfer learning process, achieving good classification performance with limited computing resources. The experimental results show that the method can achieve a validation accuracy of about 97.5% on the flower dataset, demonstrating the effectiveness of the proposed approach.

3 Conclusion

Through this experiment, I have deeply recognized the key role of data augmentation and transfer learning in improving model performance. Data augmentation methods can effectively enhance the robustness of the model, while the transfer learning strategy based on ResNet18 significantly reduces training costs. During the optimization process, properly setting learning rate decay and momentum parameters is crucial for model convergence. This practice has given me a deeper understanding of the implementation of deep learning engineering, laying a solid foundation for my subsequent learning.