Applied Deep Learning Assignment 3

Student: Aya Wahbi 01427598

Project Title: Digit Recognition with CNN and Data Augmentation

Introduction and Problem Statement

In this project, we aimed to develop a robust solution for handwritten digit recognition using the well-known MNIST dataset. Handwritten digit recognition serves as a benchmark in computer vision and deep learning. Although high baseline accuracies can be achieved on MNIST, further improvements remain challenging yet essential to enhance overall model performance.

Why is This a Problem?

Handwritten digit recognition poses several challenges:

- Variability in handwriting styles, strokes, and noise in the data can create difficulties in generalizing across all digit classes.
- Although MNIST is considered “solved” by many deep learning approaches, achieving near-perfect accuracy (e.g., 99.99%) requires careful handling of overfitting and model regularization.
- Incremental improvements beyond 99.8% accuracy demand advanced strategies, making it an excellent test case for exploring data augmentation, architecture refinements, and learning rate strategies.

Our Solution

Our solution evolved over several iterations:

1.Baseline Model

Initially, a basic Convolutional Neural Network (CNN) was implemented. Without additional techniques, it achieved a validation accuracy of about 99.225%.

2.Data Augmentation

By incorporating data augmentation (applying rotations, translations, and zooming on the training images) the baseline accuracy improved to approximately 99.375%. Augmentation provided a more diverse training set, reducing overfitting.

3.Model Refinement

Further improvements were realized by enhancing the CNN with:

- Batch Normalization Layers: added after convolutional and dense layers to stabilize the learning process.
- Learning Rate Scheduling: using the ReduceLROnPlateau callback to ease the optimization process when plateauing occurs.
  This refined model reached a validation accuracy of around 99.575%.

Deep learning, and in particular convolutional neural networks, are well suited for this task. Their ability to learn hierarchical features makes them a strong fit for image recognition. Nonetheless, without proper regularization techniques (data augmentation, dropout, and batch normalization), deep learning models can overfit, which is why our solution combines the strengths of CNNs with robust training strategies.

Main Take-Aways and Insights

- Data Augmentation: Enlarging the effective training dataset through transformations was crucial in improving generalization.
- Batch Normalization: Adding these layers smoothed the optimization landscape and allowed the network to converge faster and more reliably.
- Learning Rate Scheduling: Dynamically reducing the learning rate when validation accuracy stalled helped the model avoid local minima and refine performance.
- Incremental Improvements: Even in a “solved” dataset like MNIST, each refinement produced important gains, although with diminishing returns.

- Trade-Off: While the refined model achieved better accuracy, it required substantially more training time (approximately 1047 seconds versus 165 seconds for the augmented baseline).

Reflections and Future Work

If undertaking a similar project again, the following considerations would be taken into account:

- Experiment with deeper architectures or ensemble methods (e.g., combining predictions from several refined models) to push accuracy even higher.
- Further tune hyperparameters and explore optimizers beyond Adam, such as RMSProp or AdamW, which might offer additional performance gains.
- Allocate more time initially for the pre-processing and augmentation pipeline, a step that turned out to be more time-consuming than anticipated.

In hindsight, while our initial estimates were optimistic regarding training times, the iterative process of refining the model (adding batch normalization and implementing adaptive learning rate schedules) required additional efforts. Overall, the project expanded from an initial 200–300 seconds training cycle to a more nuanced approach that demanded closer to 1000 seconds, a trade-off that proved worthwhile for the accuracy improvements achieved.

Conclusion

This project has demonstrated how incremental improvements through data augmentation, architectural refinements (batch normalization), and adaptive learning rate strategies can significantly enhance model performance on the MNIST digit recognition task. Although achieving 99.99% accuracy remains extremely challenging, our refined CNN achieved nearly 99.6% validation accuracy, representing a solid step forward. The insights and techniques gained here are not only applicable to MNIST but can also inform strategies for more complex vision problems where deep learning plays a pivotal role.

Time Spent

The project took approximately 40–45 hours overall. Initial model development and baseline testing were completed relatively quickly, but the process of tuning the data augmentation process, integrating batch normalization, and experimenting with learning rate scheduling required additional iterations and time compared to our initial estimates. The underestimation was primarily due to the complexity of tuning the hyperparameters and managing the trade-offs between model complexity and training time.

Final Remarks

The iterative improvements in this project highlight the importance of not only selecting an appropriate deep learning architecture but also employing robust training strategies. While deep learning provides a powerful solution for image recognition tasks, the refined techniques and careful engineering are essential to approach the upper performance limits of even well-studied datasets.