Transfer Learning-Enhanced Optimization Framework for Handwritten Digit Recognition: A Study of Feature Reusability and Model Fine-Tuning

Shihong Zhou

Jincheng College, School of Computer and Software, Sichuan University, Chengdu, Sichuan 611731

Abstract: ***In an era characterized by rapidly increasing data types and volumes, there is a growing demand for machine learning models that can be constructed efficiently while possessing strong generalization capabilities. Transfer learning is particularly well-suited to address these requirements. It is well established that the MNIST dataset consists of handwritten digits collected from foreign populations, and writing styles vary considerably across different countries. Consequently, applying a model trained solely on MNIST to predict handwritten digits from other regions—such as China—often results in suboptimal accuracy in practical applications. To address this limitation, this paper incorporates transfer learning to enhance model performance. First, a custom dataset of the author's own handwritten digit images was collected. Subsequently, transfer learning was applied to a convolutional neural network (CNN) model pre-trained on the MNIST dataset, using this self-collected dataset. Experimental results demonstrate that following transfer learning, model accuracy improved by 7% on the dataset collected by the experimenter.***

Keywords: Convolutional Divine Network (CNN); Handwritten numbers; Transfer learning.

1. INTRODUCTION

Artificial intelligence is widely noticed in today's era, and character recognition technology is widely used in real life. Among them, handwriting digital recognition technology is a relatively basic but important part of the problem of handwriting digital notes in everyday life. For example, wanting to modify the recorded digital information, querying for a certain part of a recorded digital segment, or copying a part of it, these cannot be very effectively implemented on paper, and having handwritten digital identification and storing the handwritten input numbers in a computer can solve these problems well. Due to the different writing habits of different countries, we found that the handwritten digits in the MNIST dataset are different from the handwritten digits collected by the examiners themselves in shape. Therefore, the model trained by the MNIST dataset is not as good as expected when it is used on the numbers with different writing methods, and the model optimization using transfer learning can be well handled. Zheng, Zhou, and Lu [1] proposed an improved YOLOv5s algorithm for rebar cross-section detection in 2023. Shen et al. [2] researched the application of the whale optimization algorithm in financial payment fraud detection in 2025. Ge and Wu [3] conducted an empirical study on the adoption of ChatGPT for bug fixing among professional developers in 2023. Hu, Zhang, and Sun [4] introduced multiscale deep neural networks for text sentiment analysis in 2024. Meng [5] developed a neural network-based evaluation system for green cabling of cables in 2023. Junxi, Wang, and Chen [6] proposed a graph convolutional network based on matrix factorization (GCN-MF) for recommendation in 2024. Zhao, Zhang, and Hu [7] designed a smart warehouse track identification method using Res2Net-YOLACT+HSV in 2023. Sun [8] addressed accessibility challenges and solutions in designing inclusive interfaces for digital products in 2025. Tang et al. [9] designed and optimized a shallow-angle grating coupler for vertical emission from indium phosphide devices in 2020. Yang, Zheng, and Lu [10] constructed a multi-dimensional network credit-related transaction risk map with early warning by integrating graph neural networks in 2025. Gong et al. [11] provided a review of neural network lightweighting techniques in 2023. Zhou [12] proposed a digital precision distribution strategy for social media content on private domain platforms in the automotive industry using a collaborative filtering model based on user behavior in 2025. Yang et al. [13] developed a recursive multi-agent trading system for iterative optimized portfolio strategy under geopolitical uncertainty in 2026. Li et al. [14] introduced a memory-aware fine-tuning of SAM2 (MFT) for efficient long-sequence video object segmentation in 2026. Wu et al. [15] presented Tiny-Critic RAG, which empowers agentic fallback with parameter-efficient small language models, in 2026. Finally, Yuan et al. [16] proposed TA-Mem, a tool-augmented autonomous memory retrieval method for large language models in long-term conversational question answering, in 2026.

2. KEY TECHNOLOGIES

2.1 Convolutional Neural Networks

Convolutional Neural Networks. It was already used by LeCun in multi-layer neural networks in 1989, but in the following decade hardware computing power was completely inadequate. It was at a low point, and then as the hardware developed and the amount of computation became larger, convolutional neural networks came back into view again. Convolutional neural networks use convolutional nuclei to reduce the parameter directory so that there are still too many parameters, and use rights sharing to further reduce the parameter directories, reducing the number of rights and reducing the risk of overfitting. Convolutional neural networks consist of a series of input layers, convolutional layers, pooling layers, and fully connected layers. Convolutional neural networks have higher performance relative to other deep, feeding neural networks but use relatively fewer parameters, which has led to their rapid development in recent years and are highly valued as an efficient identification algorithm.

2.2 Convolutional layer

Enhance feature information, filter irrelevant information. It is the most important role of convolutional layers, thereby having convolutional layer. The most important part of a convolutional neural network is its convolution layer, and the core of the convolution layer is convolution. Convolution is an operation of two functions. This operation is called convolution. In dealing with image recognition problems, the convolutional nucleus does convolution with the two-dimensional image, that is, the convoluted nucleus accumulates with each position of the two-dimension image. In the field of image processing, different convolution kernels can be used to extract different characteristics from images. Convolutions have mechanisms and power values of local perception to share.

2.3 Pooling layer

Since direct use of these convolutional layers to obtain image characteristics to train classifiers can be very heavy, and it is easy to generate fits, pooling layers are introduced to solve this problem. Pooling Also known as lower sampling, the main role of pooling is to reduce computational effort by retaining key features while reducing network parameters, and can reduce overfitting to a certain extent and improve model generalization capabilities. The commonly used pooling methods are maximum pooling, mean pooling, global maximum pooling. These are three kinds.

2.4 Fully connected layer

The full-connectivity layer is at the tail end of a convolutional neural network, which consolidates some of the information that is classically differentiated in the previous convolutional or pooled layer. The function of the preceding layers of the full connection layer is to extract local features, and the function of the full connect layer is to reconstruct the local features extracted by the preceding layer through the power value matrix into a complete diagram. Its main function is to reduce the impact of features from different locations on the overall classification results.

2.5 Transfer learning

Using transfer learning to solve the problem of difficult access to convolutional labels is a good method [7]. Computer use of migration learning in processing visual tasks and natural language is common, and models of these problems often require large amounts of data and complex network structures. Simply put, using similarities between data, tasks, and models to simply apply a trained content to a new task is transfer learning. The object that is migrated is called the source domain, and the domain that is given "experience" is called a target domain. Transfer learning is not possible between any source and destination domains, and if the source and destination are completely unlike each other, it may cause "negative migration," i.e. what is learned in the source domain will play a negative role in the destination domain. Compared to traditional machine learning, training and test data for migration learning are subject to a different distribution, And it doesn't require data annotations as large as traditional machine learning, which builds a different model for each task, but with migration learning, a model can be migrated between multiple different tasks.

3. APPLICATION IMPLEMENTATION

3.1 Data set comparison

The MNIST dataset collected 60,000 training images and 10,000 test images. Each picture displays a number from 0 to 9, and the sample also contains the corresponding label of each training data, and the label set contains the 10 classification data of 0 to 9 [8]. The data in this dataset has been normalized, using equal-sized grayscale images, and some handwritten characters in the MNIST dataset are shown as Figure 1.

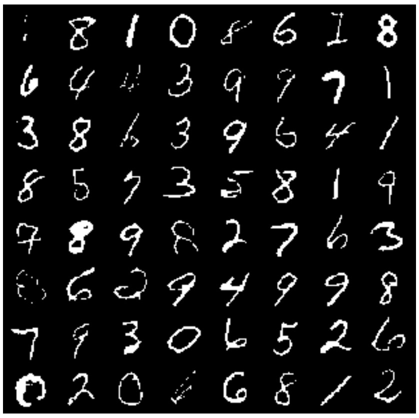

Figure 1: MNIST Dataset Photo Gallery

Through handwritten input, the experimenters collected 10 images in the training set - 0 - 9 and 5 images in a test set - 0.9. Some of the handwritten characters are shown in Figure 2. Create a 512 * 512 canvas, set its background color to black, leave a trail of drag on the canvas white by clicking and dragging the mouse, In turn, you can implement the input of handwritten numbers, unwind the mouse to stop the canvas, and save the entire canvas as a picture in the local folder. Put the pictures as training sets and test sets in two folders, each of which has folders corresponding to the ten numbers 0-9. Put the pictures corresponding to handwritten numbers in the corresponding folders, as training sets for experimenters' handwritten numeric numbers and test sets

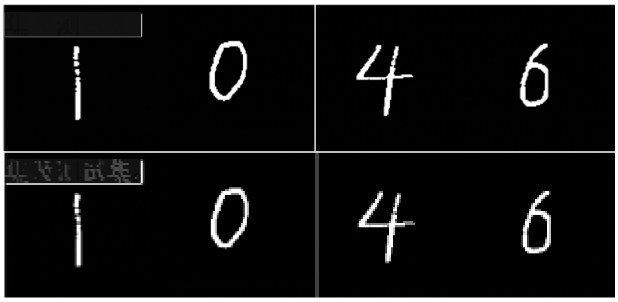

Figure 2: The experimenter's handwritten dataset is illustrated

Observational comparisons between MNIST datasets and experiments Images from this handwritten dataset reveal significant differences in the numbers in both datasets. People in other countries write the same numbers very differently than people in our country. Thus, models trained with the MNIST training set are relatively less accurate with the numbers we hand-typed, which leads to transfer learning using the trained model as a starting point for new models, thus increasing the accuracy.

3.2 Model network structure

The LeNet-5 network model is used in this paper [9]Structure As shown in Figure 3, in 1998, the master of convolutional neural networks, LeCun, proposed LeNet, using LeNet to enable handwritten digital recognition.[10]. LeNet-5 adds a pooling layer on the basis of LeNet to filter the input features. The LeNet-5 model structure starts with an input layer, a convolution layer plus a pooling layer, another convolution layer and a pooled layer, then a number of fully connected layers, and finally an output layer.

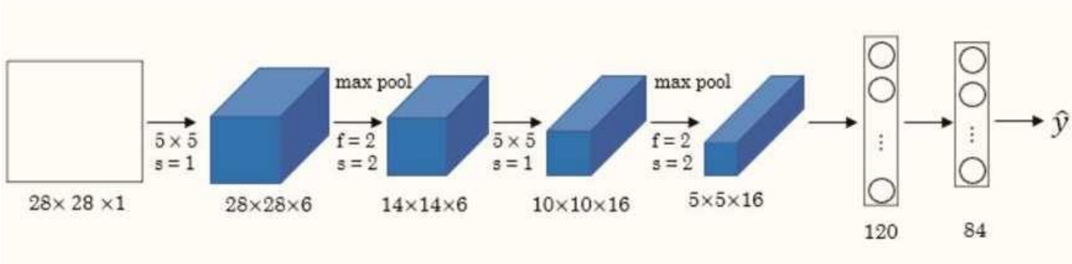

Figure 3: LeNet-5 model used in the experiment

3.3 Training steps and test results

Using the MNIST dataset, the output of the full connection layer is 10, select the SGD optimizer, set the learning rate to 0.05, and set the batchsize to 64, and the loss value curve of the 20 rounds of training is shown in Figure 3. The actual recognition was not as good as expected, as shown in Figure 4.

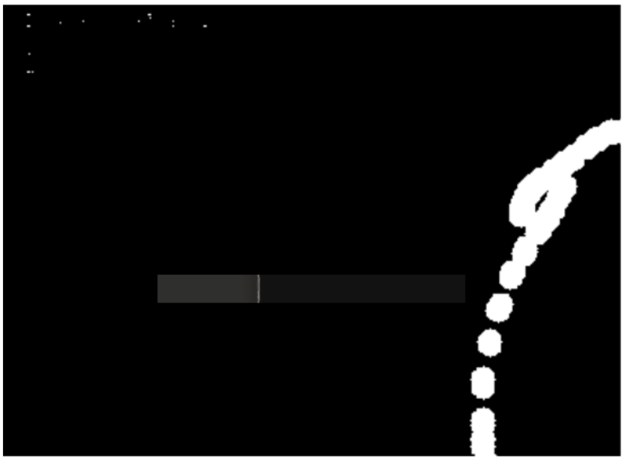

Figure 4: Actual identification results

The accuracy of the model trained by the MNIST dataset is tested by the actual examiner, and the actual test accuracy is shown in Table 1.

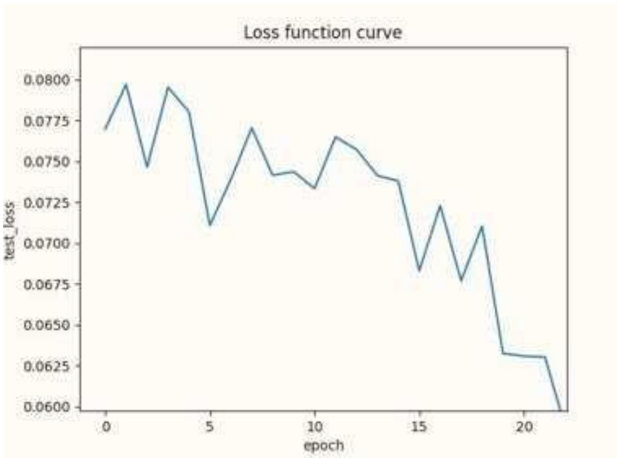

Figure 5: Curve Plot of Experimental Loss Values

Table 1: Experimental accuracy

Epoch	1	2	3	4	5	6	7	8	9	10
Acc	0.60	0.64	0.64	0.62	0.68	0.66	0.68	0.68	0.70	0.66

3.4 Handwritten digit data set for transfer learning experiment

The accuracy rate was found to be unsatisfactory after multiple training rounds, so a migration learning approach was used to improve recognition accuracy on the basis of a previously established model training network. Models trained using MNIST datasets were migrated to datasets collected by the experimenters themselves. Freeze all networks outside the full connection layer, set the full connection level output to 10, select a SGD optimizer, set the learning rate to 0.0002, set the batchsize to 6, and get a loss curve from 1000 cycles of training as shown in Figure 6.

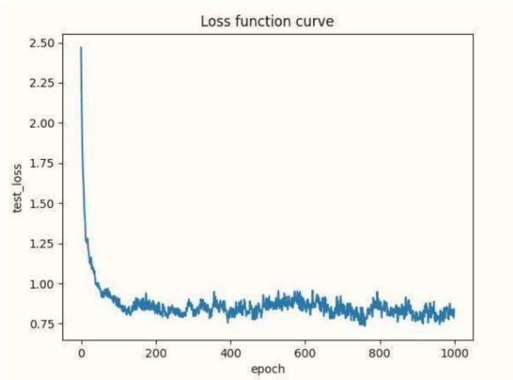

Figure 6: Curve of loss value after transfer learning

The accuracy of the new model after migration learning is shown in Table 2:

Table 2: Experimental accuracy

Epoch	100	200	300	400	500	600	700	800	900	1000
Acc	0.63	0.57	0.70	0.75	0.72	0.72	0.73	0.75	0.74	0.77

3.5 Conclusion of the experiment

A comparison of experimental accuracy tables 1 and 2 shows that the accuracy was higher in the handwritten digital data set of the experimenters after the migration learning was added. Therefore, using transfer learning to optimize handwritten digit recognition, compared with training the model with the MNIST dataset, the effect is significantly improved. The numerical recognition accuracy of handwritten input increased by 7%, and migration learning reduced the time required for the training process, eliminating the need for large data sets while improving the performance of deep models.

4. CONCLUSION

This article briefly introduces the basic concepts of convolutional neural network structures and transfer learning, and describes operations such as convolutional pooling and explains their role. The LeNet-5 network model was cited to implement the application of handwritten digital recognition, and the data on handwritten digital pictures of the experimenters were collected and fed into the trained model to make predictions. The accuracy of the model was not ideal, and based on this, migration learning was used to improve the accuracy. Improvements to experiments can be made by optimizing the network structure, adjusting the learning rate, increasing the size of the data set collected by the experimenter, increasingly the number of trainings, changing the model optimizer, etc. to improve the model accuracy.

REFERENCES

[1] Zheng, Y., Zhou, G., & Lu, B. (2023). Rebar Cross-section Detection Based on Improved YOLOv5s Algorithm. Innovation & Technology Advances, 1(1), 1–6. https://doi.org/10.61187/ita.v1i1.1

[2] Shen, Zepeng, et al. "Research on Application of Whale Optimization Algorithm in Financial Payment Fraud Detection." 2025 4th International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID). IEEE, 2025.

[3] Ge, H., & Wu, Y. (2023). An Empirical Study of Adoption of ChatGPT for Bug Fixing among Professional Developers. Innovation & Technology Advances, 1(1), 21–29. https://doi.org/10.61187/ita.v1i1.19

[4] Hu, H., Zhang, J., & Sun, Y. (2024). The Multiscale Deep Neural Networks: Unveiling New Directions in Text Sentiment Analysis. Innovation & Technology Advances, 2(2), 34–45. https://doi.org/10.61187/ita.v2i2.65

[5] Meng , L. (2023). Research on the Evaluation System of Green Cabling of Cables Based on Neural Network. Innovation & Technology Advances, 1(2), 25–31. https://doi.org/10.61187/ita.v1i2.37

[6] Junxi, Y., Wang, Z., & Chen, C. (2024). GCN-MF: A graph convolutional network based on matrix factorization for recommendation. Innovation & Technology Advances, 2(1), 14–26. https://doi.org/10.61187/ita.v2i1.30

[7] Zhao, X., Zhang, L., & Hu, Z. (2023). Smart warehouse track identification based on Res2Net-YOLACT+HSV. Innovation & Technology Advances, 1(1), 7–11. https://doi.org/10.61187/ita.v1i1.2

[8] Sun, Lingxin. "Designing Inclusive Interfaces: Accessibility Challenges and Solutions in Digital Products." Proceedings of the 2025 International Conference on Artificial Intelligence and Sustainable Development. 2025.

[9] Tang, Yingheng, et al. "Design and Optimization of Shallow-Angle Grating Coupler for Vertical Emission from Indium Phosphide Devices." (2020).

[10] Yang, X., Zheng, X., & Lu, Q. (2025, October). Construction and early warning of multi-dimensional network credit-related transaction risk maps by integrating graph neural network (GNN). In Proceedings of the 2025 2nd International Conference on Digital Economy and Computer Science (pp. 919-923).

[11] Gong, Z., Zhang, H., Yang, H., Liu, F., & Luo, F. (2023). A Review of Neural Network Lightweighting Techniques. Innovation & Technology Advances, 1(2), 1–24. https://doi.org/10.61187/ita.v1i2.36

[12] Zhou, Z. (2025, November). Digital precision distribution strategy for social media content on private domain platforms in the automotive industry: a collaborative filtering model based on user behavior. In Proceedings of the 2025 International Conference on Digital Society and Intelligent Computing (pp. 516-521).

[13] Yang, J., Wu, Y., Liu, J., Liang, P., Yuan, M., Li, X., & Yan, W. (2026). Recursive Multi-Agent Trading System: Iterative Optimized Portfolio Strategy Under Geopolitical Uncertainty. arXiv preprint arXiv:2605.25311.

[14] Li, G., Yuan, H., Chen, S., Hu, Q., Wang, J., & Jiang, K. (2026). MFT: Memory-Aware Fine-Tuning of SAM2 for Efficient Long-Sequence Video Object Segmentation. IEEE Signal Processing Letters.

[15] Wu, Y., Liang, P., Xiang, Y., Yuan, M., Liu, J., Yang, J., ... & Yan, W. (2026, March). Tiny-Critic RAG: Empowering Agentic Fallback with Parameter-Efficient Small Language Models. In 2026 9th International Conference on Advanced Algorithms and Control Engineering (ICAACE) (pp. 2577-2580). IEEE.

[16] Yuan, M., Liu, J., Yang, J., Li, X., Yan, W., Wu, Y., & Liang, P. (2026, March). TA-Mem: Tool-Augmented Autonomous Memory Retrieval for LLM in Long-Term Conversational QA. In 2026 9th International Conference on Advanced Algorithms and Control Engineering (ICAACE) (pp. 2684-2688). IEEE.