Proceedings of the 2nd International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/109/2024.18132

Research on the Application of Convolutional Neural Networks on MNIST Datasets

Jiazhou Jiang

School of Future Science and Engineering, Suzhou, China

Johnny_jiang_12345@outlook.com

Abstract. This paper explores the application of Convolutional Neural Networks (CNNs) and Fully Connected Neural Networks (FCNNs) on the MNIST dataset for handwritten digit recognition. The study underscores the significance of digit recognition in various sectors, including automation, accessibility, data validation, security, education, and as a foundation for complex AI systems. The research compares the performance of FCNNs and CNNs, highlighting the latter's superiority due to its ability to capture spatial features and resist overfitting. The paper details the structure and evaluation of both network types, with CNNs achieving a higher accuracy of 99.09% compared to FCNNs at 97.8%.

Keywords: Convolutional Neural Networks (CNNs), Fully Connected Neural Networks (FCNNs), MNIST Dataset, Digit Recognition, Computer Vision, Machine Learning.

1. Introduction

1.1. The significance of digit recognition

The advent of the digital age has been marked by the exponential growth of computer technology, with computer vision emerging as a pivotal force in the realm of image processing. At the forefront of this technological revolution are Convolutional Neural Networks (CNNs) and Fully Connected Neural Networks (FCNNs), which have established themselves as cornerstone deep learning models within the field of computer vision.

Digit recognition stands as a fundamental task in the domain of computer vision and pattern recognition, with far-reaching implications across a multitude of sectors. The significance of digit recognition is manifold:

1.1.1. Automation of Routine Tasks

Digit recognition enables the automation of processes that involve reading and interpreting numbers. For example, it is widely used in banking for recognizing handwritten checks, postal services for sorting mail based on handwritten zip codes, and digitizing handwritten forms.

1.1.2. Enhancing Accessibility

For visually impaired individuals, digit recognition technology can help read aloud numbers, making various services and digital platforms more accessible. It can also aid in reading handwritten notes or printed text through assistive technologies.

1.1.3. Data Entry and Validation

In many industries, digit recognition helps in the accurate entry of numerical data. For example, it can be used in inventory management systems, where handwritten numbers on labels need to be converted into digital form for tracking and analysis.

1.1.4. Security and Authentication

Digit recognition is integral to applications like CAPTCHA, where users must identify and input numbers from distorted images. This helps in differentiating between human users and bots, ensuring secure access to online services.

1.1.5. Educational Tools

In education, digit recognition can be used to develop tools for teaching and evaluating students, especially in early learning stages. Applications that automatically recognize and correct handwritten numbers can be used for assignments and tests.

1.1.6. Foundation for Complex Systems

Digit recognition is often a fundamental problem in machine learning and AI research. Techniques developed for digit recognition are applied to more complex problems, such as full-text recognition, object detection, and autonomous driving.

1.1.7. Cost Efficiency and Time Saving

By automating the recognition of digits in various processes, organizations can save significant time and reduce costs related to manual data entry, error correction, and processing.

In summary, digit recognition is vital due to its broad applicability, ability to streamline processes, and its role as a foundational technology in more complex machine learning and AI systems.

1.2. Why digit recognition using Neural Networks (NN) is considered a foundational step in the field of machine learning

1.2.1. Simplified Problem Domain

Digit recognition presents a relatively simple and well-defined problem space, making it an ideal starting point for exploring and developing neural network architectures. Its simplicity allows researchers to focus on learning and improving fundamental concepts in neural networks without the complexity of more challenging tasks.

1.2.2. Benchmark for Model Performance

Digit recognition tasks, like the MNIST dataset, have historically been used as benchmarks to evaluate the performance of various neural network models. Successful digit recognition results provide a baseline for advancing to more complex image recognition tasks.

1.2.3. Proof of Concept

Early successes in digit recognition using neural networks served as proof of concept for the capability of neural networks to learn and generalize from data. These initial successes paved the way for further exploration and adoption of neural networks in other domains.

1.2.4. Scalable Complexity

Although digit recognition is simple, the techniques and models developed can be scaled to more complex tasks. It acts as a stepping stone, where the knowledge gained from solving digit recognition problems is applied to more intricate challenges like full handwriting recognition or even object detection in images.

1.2.5. Educational Value

For students and practitioners entering the field of machine learning, digit recognition is often one of the first tasks tackled using neural networks. It provides a practical, hands-on introduction to key concepts such as data preprocessing, model training, and evaluation, which are fundamental to more advanced applications.

1.2.6. Low Computational Cost

The relatively small size and simplicity of digit recognition datasets make them accessible for experimentation, even on less powerful hardware. This accessibility encourages innovation and experimentation in the early stages of neural network research and development.

1.2.7. Foundation for Pattern Recognition

The techniques developed for digit recognition, such as feature extraction and convolutional layers, have become fundamental building blocks for more sophisticated pattern recognition tasks, including facial recognition, speech recognition, and natural language processing.

In essence, digit recognition using neural networks is often seen as a gateway to understanding and advancing in the broader field of neural network-based pattern recognition and AI.

The MNIST Dataset includes 70,000 black-white images of hand-written numbers, each image is a 28*28-pixel image. The FCNN/CNN model trained by the dataset can recognize hand-written numbers accurately. As computer vision is becoming more and more popular all over the world, it’s of great importance to learn the basic of computer vision—from the MNIST dataset!

2. Research Methods

2.1. Fully Connected Neural Network (FCNN)

Neural networks are a set of dependent non-linear functions. Each individual function consists of a neuron (or a perceptron). In fully connected layers, the neuron applies a linear transformation to the input vector through a weight’s matrix. A non-linear transformation is then applied to the product through a non-linear activation function f [1].

Structure:

Input Layer

This is the first layer of the network that receives the input data. Each neuron in this layer represents a feature of the input data. Given that the input is a 28x28 image, the input layer consists of 784 neurons.

Hidden Layers

The number of hidden layers and neurons in each layer are crucial hyperparameters. The model features two hidden layers with 256 and 64 neurons, respectively. Activation Function: ReLU.

Output Layer

This is the final layer that produces the network's output. The number of neurons in this layer corresponds to the number of output variables. In this model, the number of neurons is 10, corresponding to the digits 0-9. Activation function: softmax.

2.2. Convolutional Neural Networks (CNN):

Convolutional Neural Networks bear a strong resemblance to ordinary Neural Networks: they are composed of neurons with learnable weights and biases. Each neuron receives some inputs, performs a dot product, and optionally follows it with a non-linearity. The entire network expresses a single differentiable score function: from the raw image pixels on one end to class scores at the other. They still have a loss function (e.g., SVM/SoftMax) on the last (fully-connected) layer, and all the tips/tricks we developed for learning regular Neural Networks still apply.

Input Layer

The input layer in a CNN typically receives image data as a multi-dimensional array.

Convolutional Layers

These layers are the core building blocks of a CNN. They apply convolutional filters (or kernels) to the input data to detect features such as edges, textures, and patterns. Each filter slides over the input, performing a dot product between the filter values and the input, resulting in feature maps. These feature maps highlight different aspects of the input data. The first convolutional layer uses 32 3x3 filters; the second convolutional layer uses 64 3x3 filters. The convolution operation reduces the dimensionality of the data while preserving spatial relationships, making CNNs effective for image processing tasks.

Activation Function is ReLU.

Pooling Layers

Pooling layers reduce the spatial dimensions of the feature maps by summarizing the presence of features in subregions. There are two max pooling layers in this model.

Fully Connected Layers

After several convolutional and pooling layers, the output is typically flattened into a 1D vector and passed through one or more fully connected layers, similar to those in a Fully Connected Neural Network. These layers combine features learned in the convolutional layers to make predictions.

Output Layer

The final fully connected layer outputs the predictions. The output size is 10, corresponding to the number of classes 0-9.

3. Evaluation

3.1. FCNN

Number of hidden layers

The input data isn’t too complex. 3 hidden layers is enough. Apply more hidden layers will cause serious overfit problem, with accuracy descending from 95% to around 60%.

Learning rate

Learning rate controls the size of the steps taken in the direction of the negative gradient during optimization.

Learning Rate (α): The learning rate determines how much the model's weights are updated in response to the error at each iteration of the training process. It essentially dictates the speed at which the model "learns" from the data.

If the learning rate is too high, the model will take large steps toward minimizing the error, potentially overshooting the optimal solution. This can cause the training to diverge and prevent the model from converging to the best solution.

If the learning rate is too low, the model will take very small steps toward minimizing the error. While this can lead to a more precise solution, it might result in a very slow convergence, requiring more iterations and time to reach a satisfactory result.

Applying SGD optimizer, the proper learning rate is approximately 0.1, the accuracy is approximately 97.8%.

3.2. CNN

CNN is highly effective. 3 hidden layers enables the model to have high accuracy, more hidden layers couldn’t contribute to higher accuracy. The proper learning rate (SGD) is 0.001, achieving 99.09% accuracy.

4. Application

In the field of medical image processing, Convolutional Neural Networks (CNNs) have a wide range of applications, here are some specific scenarios and related literature:

4.1. Medical Image Segmentation

CNNs are used in medical image segmentation to identify and distinguish different tissues and structures, such as tumors and organs. For instance, 3D CNNs have been utilized to enhance the segmentation accuracy of organs or lesions in MRI and CT images [3].

4.2. Disease Diagnosis

CNNs aid in analyzing medical images for the diagnosis of various diseases, including cancer, brain diseases, etc. Studies have used CNNs for computer-aided diagnosis of breast cancer in ultrasound images [4].

4.3. Pathology Image Analysis

In pathology, CNNs are used for the automatic detection and classification of abnormalities in cellular or tissue samples, such as cancer cells. Research has employed CNNs for the classification of the malignancy of lung nodules [5].

4.4. Water Quality Monitoring

CNNs can be used to analyze water samples images, rapidly detecting and identifying dominant microalgal communities, which is crucial for monitoring water quality and preventing the formation of harmful algal blooms. For instance, a project at the Jean Golding Institute utilized CNNs to identify and count microalgae cells for monitoring algal populations in freshwater bodies [6].

4.5. Unmanned Aerial Vehicle (UAV) Environmental Monitoring

Cameras on UAVs can collect image data, and CNNs are used for image recognition and localization to monitor environmental anomalies. Researchers have developed CNN-based image recognition technology for identifying images collected by UAVs during environmental monitoring processes [7].

4.6. Weather Forecasting

CNNs combined with Long Short-Term Memory networks (LSTM) are used to address weather forecasting challenges. This hybrid model can process and predict weather patterns, providing more accurate forecasts [8].

5. The Future of CNN

The future research directions for Convolutional Neural Networks (CNN) in the field of environmental monitoring include:

5.1. Advanced Image Recognition

Continued development and refinement of CNN models for more accurate image recognition in environmental monitoring, including the differentiation of subtle patterns and changes in environmental conditions .

5.2. Integration with UAVs

Further research into the application of CNN for image recognition and localization in UAV-based environmental monitoring systems, which can be used for emergency rescue, disaster relief, and urban planning .

5.3. Spatio-Temporal Climate Data Analysis

Exploring the use of CNN for classifying and identifying patterns in spatio-temporal climate data, which can aid in weather forecasting, understanding climate change effects, and investigating air pollution transport .

5.4. Ocean Remote Sensing

Applying CNN to ocean remote sensing for the analysis of satellite imagery, which can help in monitoring ocean health, detecting marine life, and studying climate patterns .

5.5. Improving Multi-GCM Ensemble Predictions

Utilizing CNN frameworks to enhance the accuracy of multi-model ensemble predictions of monthly precipitation at local areas, which is crucial for climate adaptation and water resource management .

5.6. Extreme Weather Event Prediction

Developing CNN models to predict extreme weather events, which are becoming more frequent due to climate change, to aid in disaster preparedness and response.

5.7. Real-time Environmental Monitoring

Enhancing CNN algorithms for real-time analysis and prediction in environmental monitoring, which can provide immediate feedback for decision-making processes.

5.8. Data Fusion and Multi-sensor Integration

Research into combining data from various sensors and platforms using CNN to create a comprehensive view of environmental conditions.

5.9. Automated Feature Extraction

Advancing CNN capabilities to automatically extract and learn complex features from environmental data without manual intervention.

5.10. Sustainability and Resource Efficiency

Focusing on the development of more efficient CNN models that require less computational power, making them more sustainable and accessible for a wider range of environmental monitoring applications.

These directions indicate a trend towards more sophisticated applications of CNN in environmental monitoring, aiming to improve predictive capabilities, enhance data analysis, and support decision-making processes in the face of environmental challenges.

6. Conclusion

FCNNs are versatile and can be used for a wide range of tasks, but they are typically computationally expensive and may not be the best choice for problems involving spatial or sequential data, where other architectures like Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs) might be more suitable.

6.1. Connectivity

FCNN

Fully Connected: Each neuron is connected to every other neuron in adjacent layers, making the network fully connected.

Parameter Count: A large number of parameters (weights and biases), especially when dealing with high-dimensional input, leading to increased computational costs and a higher risk of overfitting.

CNN

Locally Connected: Neurons in convolutional layers are only connected to a local region of the input, which reduces the number of parameters and captures spatial dependencies.

Parameter Efficiency: Fewer parameters compared to FCNNs due to weight sharing (using the same filter across the entire input) and local connectivity.

6.2. Handling Data

FCNN

Data Representation: Treats all input features equally without considering the spatial or sequential relationships between them.

Use Cases: Suitable for structured data (like tabular data) where the relationships between features are not spatial or sequential.

CNN

Data Representation: Specifically designed to handle spatially correlated data, such as images, by capturing local features and building complex hierarchies of features.

Use Cases: Ideal for image processing, computer vision tasks, and any data where local patterns are important, such as time-series analysis and even some natural language processing tasks.

6.3. Feature Extraction

FCNN

Manual Feature Engineering: Often requires manual feature extraction and engineering before feeding the data into the network.

Learning: Learns from raw data but does not inherently capture spatial or hierarchical relationships unless manually encoded.

CNN

Automatic Feature Extraction: Convolutional layers automatically learn hierarchical features from the input data, from simple edges in the initial layers to complex structures in deeper layers.

Learning: Excels at learning spatial hierarchies and extracting features directly from raw data, such as pixels in images.

6.4. Scalability and Performance

FCNN

Scalability: Can become computationally expensive and inefficient with high-dimensional inputs due to the fully connected nature, leading to a large number of parameters.

Performance: Often less efficient and less accurate on tasks that involve spatially correlated data like images.

CNN

Scalability: More scalable and efficient with high-dimensional inputs like images, as it reduces the number of parameters and focuses on local patterns.

Performance: Typically outperforms FCNNs on tasks involving spatial data due to its ability to capture complex patterns and hierarchical features.

FCNNs are versatile and can be used for a variety of tasks, but they are less efficient when dealing with structured grid-like data (e.g., images) because they do not exploit spatial relationships between features.

CNNs are specifically designed to handle tasks involving spatial or hierarchical patterns, making them ideal for image processing and similar tasks. They are more efficient in terms of parameter usage and are generally more accurate for these types of tasks.

The choice between FCNN and CNN depends largely on the nature of the data and the task at hand. Both CNN and FCNN can be used for handwritten digit recognition but with the ability to learn spatial features and avoid overfitting makes CNN the more effective model for this task. This finding shows the importance of model selection based on the nature of the data.

It can be predicted that CNN will perform better on more complex datasets such as Fashion MNIST and CMNIST, as these datasets require more complicated spatial feature extraction, which is the strength of CNNs.

References

[1] Diego Unzueta. Convolutional Layers vs Fully Connected Layers What is really going on when you use a convolutional layer vs a fully connected layer? Convolutional Layers vs Fully Connected Layers (readmedium.com)

[2] CS231n Convolutional Neural Networks for Visual Recognition

[3] S. Niyas, S.J. Pawan, M. An and Kumar, Jeny Rajan. a medical image segmentation with 3D convolutional neural networks: A survey Medical image segmentation with 3D convolutional neural networks: A survey - ScienceDirect

[4] Wenjian Yao, Jiajun Bai, Wei Liao, Yuheng Chen, Mengjuan Liu, Yao Xie. From CNN to Transformer: A Review of Medical Image Segmentation Models [2308.05305] From CNN to Transformer: A Review of Medical Image Segmentation Models (arxiv.org)

[5] Huixin Jia , Jiali Zhang , Kejun Ma , Xiaoyan Qiao , Lijie Ren , Xin Shi .Application of convolutional neural networks in medical images: a bibliometric analysis – PubMed Application of convolutional neural networks in medical images: a bibliometric analysis - PubMed (nih.gov)

[6] 20 September 2021 by Jean Golding Institute. Convolutional neural networks for environmental monitoring – Jean Golding Institute News Convolutional neural networks for environmental monitoring – Jean Golding Institute News (bristol.ac.uk)

[7] Kunrong Zhao, Tingting He, Shuang Wu, Songling Wang, Bilan Dai, Qifan Yang & Yutao Lei . EURASIP Journal on Image and Video Processing volume 2018, Article number: 150 (2018) Application research of image recognition technology based on CNN in image location of environmental monitoring UAV Application research of image recognition technology based on CNN in image location of environmental monitoring UAV | EURASIP Journal on Image and Video Processing | Full Text (springeropen.com)

[8] Michael Fan, Omar Imran, Arka Singh, Samuel A. Using CNN-LSTM Model for Weather Forecasting Using CNN-LSTM Model for Weather Forecasting | IEEE Conference Publication | IEEE Xplore