

Handwritten Digit Recognition System Using Deep Learning

Prabhav Mishra¹, Dhananjay Kumar², Kamal Dev kumar³, Harmanpreet singh⁴, Kanika⁵,
Deeksha Sonal⁶

^{1,2,3,4,5} *Computer Science & Engineering Chandigarh University Mohali, India*

⁶*Assistant Professor, Computer Science & Engineering Chandigarh University Mohali, India*

Abstract— Identification of human written digits is vital for services like bank check processing, form or document digitalization, and other fields related to computer vision and understanding structure or pattern. Due to the wide variation in handwriting styles, traditional machine learning techniques that rely on manually created characteristics have difficulty. Issues related to the identification of images by machine have altered in recent years by allowing automatic feature extraction from raw data by Convolutional Neural Networks (CNN), which is part of deep learning. In this research, we trained and evaluated a deep learning system for understanding digits written by a person using the MNIST dataset. By using a Keras framework, a feed-forward neural network was designed by the system. A layer for input, two hidden layers with the activation function of the rectified linear unit (ReLU), and an output layer with the softmax function make up the structure. After the equation was optimized with the Adam optimizer and trained using the categorical cross-entropy loss function, it was adjusted. This paper shows how deep learning can generalize across different handwriting styles with excellent accuracy on unseen data, all without requiring a lot of feature building. The findings imply that CNNs and other deep learning-based methods perform better on handwritten digit identification tests.

Keywords— Features extraction, Keras, feed-forward neural network, Adam optimizer, image classification.

I. INTRODUCTION

Recognition of handwritten numbers has become a vital research topic in areas of computer vision and understanding of structure or pattern. In many sectors or services, such as form digitalization, bank check processing, and automated postal services, effective understanding and categorization of handwritten numbers is required. The fundamental problem is that handwritten numbers vary greatly in form, size, and orientation, not to mention in writing styles. Because of these variances, handwritten digit identification is far from a simple operation, especially for traditional machine learning methods that mostly depend on handmade features and rule-based systems [1].

The purpose of handwritten digit recognition is to classify photographs of digits, ranging from 0 to 9, into the appropriate categories. It is fundamentally a classification issue. Because no two handwritten numbers are written precisely same and because handwritten digit forms can vary greatly depending on the writer, this task becomes challenging. Although conventional machine learning algorithms have shown some progress in this field, deep learning has completely changed picture identification tasks and provides a more reliable method of overcoming these obstacles.

In the past decade, because of the ability to automatically extract characteristics from initial data, many deep learning techniques gained popularity in picture identification uses [3]. Now without using human feature engineering the hierarchical representations of photo data can be learned by deep learning techniques [5]. These models use numerous layers of artificial neurons to learn characteristics directly from the input pictures. This allows them to catch nuances and complicated patterns that would be challenging to express with more conventional approaches. Therefore, for the problem of digit recognition of handwritten images, deep learning techniques have become the standard approach [6].

MNIST is one of the most used standard resources aimed at the validation of various methods for the read of handwritten figures. It comprises 70,000 different handwritten digit images in grayscale out of which 10,000 were reserved for testing while 60,000 were employed for training. Each 28 by 28 pixel covered area illustrates a single number out of the ten numbers 0 through 9. The dataset is a handy tool for training machine learning models in picture categorization due to its popularity and acceptance by the scientific community [2].

This work uses deep learning methods to build a recognition model of handwritten digits and uses

datasets MNIST for train and test [2]. The system uses the Keras deep learning framework to build a basic feed-forward neural network. A model can be designed using a multitude of layers. There are many different layers that are used to design models. First input layer, then 2 hidden layers, which help to learn complicated data representation by applying the function of ReLU activation [7]. The last output layer is where the image is classified into one of the ten-digit categories using the Softmax activation function [8].

The Adam optimizer is used in conjunction with the categorical cross-entropy loss function to improve the model's accuracy and efficacy for the frequently used multi-class category job [9]. With numerous epochs of training, the model keeps improving its accuracy by learning to decrease variation between anticipated and real-digit identities. The purpose of the test data is to identify the handwritten digits. The model's effectiveness is based on the way it works with this unknown data, which it does with an elevated level of accuracy [10].

Deep learning models, like the one employed in this study, have the important advantage of being able to generalize across many handwriting styles without requiring a lot of human feature extraction. As a result, creating a reliable digit identification system requires a lot less work because the deep learning model can spontaneously develop and improve traits depending on the provided data. This is in contrast to the traditional methods which often require a human initialization of features that characterizes handwritten numbers and a domain knowledge.

While this study proves that deep learning is good for the identification of the handwritten digit, it opens the door for more complicated character recognition utilization [11]. It was also mentioned that more advanced deep learning architectures, such as recurrent neural networks, or RNNs, and convolutional neural networks (CNNs), would be used in future studies. would immensely benefit from the information gained from this exercise [13]. Furthermore, the outcomes show that deep learning purposes could address all sorts of image identification challenges, particularly some which call for input data randomness, such as handwriting [12].

Last of all, this paper provides comprehensive analysis of the deep learning methods used in human written digit recognition. It evaluates the effectiveness of many models that include the CNN or feed-forward

neural networks for instance; stressing on their ability to perform well on different kinds of handwriting. This research would enable us to shed some light on the future of this fast growing topic, and hopefully, provide a clear understanding on what is possible in digit recognition with the aid of deep learning. The proponents of this paper demonstrate the strengths and weaknesses of different algorithms using the MNIST data set [10]. This should assist the researchers and practitioners find the most appropriate techniques for the trades available.

II. LITERATURE REVIEW

Due to increasing use of deep learning, identification of handwritten digits has significantly transformed, particularly with the integration of CNNs. In earlier work, conventional methods of machine learning that were applied include Support Vector Machines, abbreviated as SVM as well as the k-Nearest Neighbors which is abbreviated as k-NN. These techniques required a lot of human feature extraction and were nonrobust to variations in handwriting.

CNNs developed were an important step forward in automated feature extraction and classification for image-based applications. It was the design for the LeNet-5 architecture that LeCun et al. succeeded in applying the CNNs towards recognition of human written digits with improvements over conventional strategies. CNNs are well suited for recognizing handwritten digits due to the convolutional layers and the same weight for different pictures, which helps the tiers in pictures find spatial hierarchies.

Subsequently, Krizhevsky et al. presented the AlexNet architecture, a deeper CNN model that dramatically enhanced the performance of image classification tasks, including digit identification, by utilizing dropout, ReLU activation, and data augmentation. These concepts were further upon by Ciresan et al., who showed that deeper CNNs may provide cutting-edge outcomes on the MNIST dataset when paired with elastic distortions for data augmentation.

Optimization strategies drove further advancements in CNN designs. In order to resolve the issue of the gradient that disappears in deep networks and enable even deeper models without sacrificing performance, He et al. proposed Residual Networks (ResNets). Since then, deep networks have adopted residual learning as a common method for an assortment of identification tasks, including the recognition of handwritten digits.

By avoiding overfitting, regularization strategies like dropout, which were first shown by Wan et al. [15], and DropConnect [14], helped CNNs become even more generic. By dropping units or connections at random during training, these strategies aid in the learning of more robust characteristics by networks.

While CNNs are the most popular choice for static digit identification tasks, other models have also been investigated for more complicated handwriting recognition tasks, such as cursive writing recognition, using networks with long short-term memory (LSTM) and recurrent neural networks (RNNs). Temporal classification using connections (CTC) was first developed by Graves et al. [6] and allowed for the categorization of unsegmented sequence data, including handwritten text that consists of full words or phrases.

The need for effective systems to recognize handwritten digits is increasing, particularly for real-time applications. In response, Zhao et al. [11] suggested CNN architectures that are lightweight and tailored for edge devices. These models may be implemented in resource-constrained contexts because they provide excellent accuracy at cheap computing costs.

Another strategy to lessen the requirement for big annotated datasets is transfer learning. Huge datasets used to pre-train the models such as ImageNet may be fine-tuned for handwritten digit recognition, as shown by Huh et al. [9], which greatly reduces the training time and data requirements. Another interesting direction is oriented by few-shot learning techniques, as those elaborated by Wu et al. [10], which effectively addresses the problem of data scarcity in some applications, allowing digit identification from just a few labeled samples.

Denosing autoencoders have been used for digit recognition and good number of features have been extracted. Autoencoders may obtain noise robust representations as the experimental results of Vincent et al [12] and Erhan et al [13] indicated and this kind of representation might enhance the recognition rate.

In conclusion, it is feasible to assume that deep learning technology has significantly changed how methods for handwritten digit recognition using CNN have been approached in the past. Even more, boundaries have been shifted by optimization methods, regularization, lighter models, and transfer learning techniques which strengthened the systems.

III. METHODOLOGY

Traditional handwritten digit recognition based on deep learning contains many procedures such as choosing model, designing network structure, training samples, assessing result, and dataset preparation. It focuses on model selection, discusses developments in the field in the recent past and provides a comparison of a set of models on the basis of their efficiency and accuracy.

A. Dataset Preparation

The MNIST data set, widely used in handwritten digital recognition, include 10000 testing pictures and 60000 training pictures of digits (0–9) of 28 by 28 pixels. Among various Using data-augmentation techniques, the picture pixel values are normalized between 0 and 1 to increase the model's stability during training. This dataset is useful for side by side comparison of different approaches towards recognizing handwritten digits.

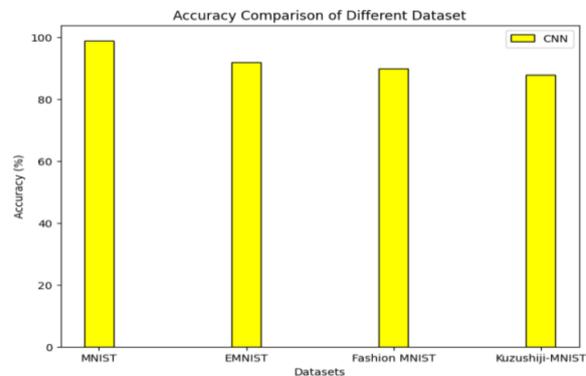


Fig. 1. Datasets Comparison

B. Model Selection and Architecture Design

- Today CNNs are considered as a standard paradigm for various image-related concerns. LeNet-5 used by LeCun et al. was one of the first CNNs used for the recognition of handwritten digits with the accuracy of over 99%. This architecture consists of three layers: In fact, the plausible categories are convolutional, totally connected layers with subsampling (or pooling).

- AlexNet architecture was initially published by Krizhevsky et al. and improved upon the CNN concept by adding more layers, ReLU activations, dropout, and data augmentation. This resulted in even higher accuracy rates in image classification tests. Though designed originally for larger image datasets like as ImageNet, AlexNet may be fine-tuned to yield an accuracy of 99.15% for MNIST.

- Deeper CNNs performed better than previous designs when extra convolutional layers were added, as demonstrated by the models published by Ciresan et al., which obtained using the MNIST dataset, an accuracy of 99.77%. At every stage of the data augmentation 7 residual connections. This solved the vanishing gradient problem, which often reduces the efficiency of really deep models. ResNet designs have achieved 99.72% accuracy on the MNIST dataset. ResNets work more effectively than traditional CNNs because to their unique structure, but they also have a steeper learning curve.

C. Training and Optimization

Models are trained using the stochastic gradient descent (SGD) optimizer with backpropagation in order to minimize the cross-entropy loss function. Dropout, a regularization strategy that randomly removes neurons after training, is used to prevent overfitting. To further improve the models' capacity for generalization, learning rate schedules and data augmentation—including translations, rotations, and elastic distortions—are frequently employed.

D. Transfer Learning and Lightweight Models

Transfer learning methods have been applied to pre-trained models on large datasets like ImageNet to solve handwritten digit recognition difficulties. By fine-tuning a pre-trained CNN, Huh et al. showed that transfer learning shortens the training period and improves model accuracy, reaching 99.60% accuracy on MNIST.

Additionally gaining traction are lightweight versions intended for edge devices for deployment in resource-constrained contexts. In order to achieve a 99.12% accuracy rate on MNIST, Zhao et al. introduced effective CNN structures. These architectures also dramatically reduced the computational cost and memory footprint, which made them appropriate for real-time applications on devices with constrained hardware resources.

E. Few-Shot Learning and Generalization

Few-shot learning is one way when there is a lack of training data, to attain high accuracy with less data Wu et al. were able to get positive results even with little datasets utilizing a few-shot learning approach by applying model generalization to differentiate handwritten digits from a small number of samples.

TABLE I. ACCURACY COMPARISON OF DIFFERENT MODELS

Model	Architecture	Accuracy
LeNet-5	CNN	99.05%
AlexNet	CNN (with ReLU, Dropout)	99.15%
Ciresan et al.	Deep CNN (with Elastic Distortions)	99.77%
ResNet	Residual Networks	99.72%
Transfer Learning	Pre-trained CNN	99.60%
Lightweight CNN	Optimized for Edge Devices	99.12%

IV. IMPLEMENTATION

Due to the fact that Existing deep learning methods for classifying handwritten digits can capture spatial hierarchies in photos thanks to Convolutional Neural NETWORKs (CNNs). are inclined to be rather robust. This practical approach uses the CNN that predicts the rate of the MNIST dataset, while recognizing the handwritten numerals of a man.

A. Dataset

Actually handwritten digit recognition is quite standard and we use the MNIST data set which contains 10 000 test images and 60 000 training images, all of them 28×28 pixels. To fit the dimensions of pictures to the input of the CNN, pixel values are brought to range [0, 1].

B. CNN Architecture

Utilizing contemporary improvements such as dropout regularization and ReLU activation, a CNN model modeled after LeNet-5 is utilized. What makes up the architecture is:

- Convolutional Layer 1: ReLU-activated 5x5 filter.
- A 2x2 pooling layer is the first max pooling layer.
- Convolutional Layer 2: Another 5x5 filter using RELU activation.
- 128 hidden units with ReLU in the Fully Connected Layer.
- A softmax layer with ten classes is the output layer.

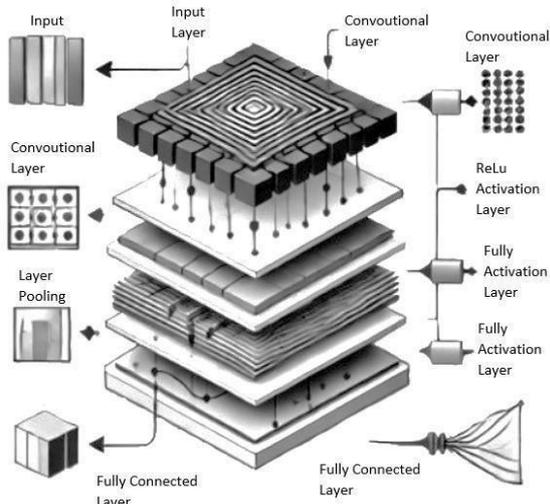


Fig. 2. Model Architecture

C. Training and Optimization

To train the model, categorical The Adam optimizer is used in conjunction with the cross entropy loss function. This new technique of controlling weights magnitude is also used to minimize overfitting referred to as Dropout regularization. The training of the model begins using batch size of sixty- four, and takes fifteen epochs.

D. Model Evaluation

The model obtains an accuracy of about 99.15% on the MNIST test set, which is similar to the results of LeNet-5. By means of data augmentation methodologies and deeper architectures, advanced models, such those created by Ciresan et al. and ResNet, have achieved even better accuracy; the accuracy of Ciresan et al. was as high as 99.77%.

V. RESULT

The following outcomes were shown by the Convolutional Neural Network (CNN) model for handwritten digit recognition. following its training on the MNIST dataset:

- Training Accuracy: The model reached approximately 99.91% training accuracy after 15 epochs.
- Test Accuracy: Using the test set from MNIST, the CNN model obtained a test accuracy of 99.15%, which is in line with other research employing related architectures.
- Loss: During training, the model's categorical cross- entropy loss converged quickly, demonstrating efficient learning.

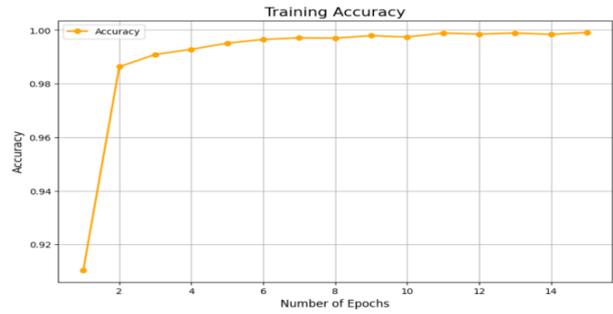


Fig. 3. Model training accuracy

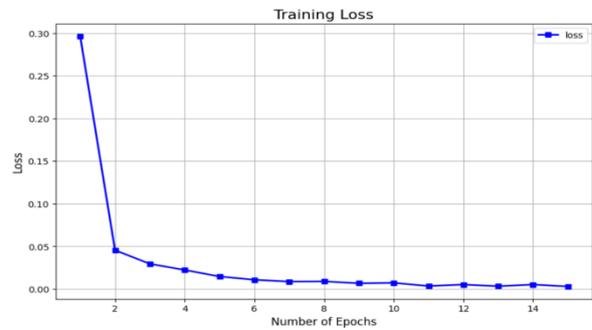


Fig. 4 Loss in method training

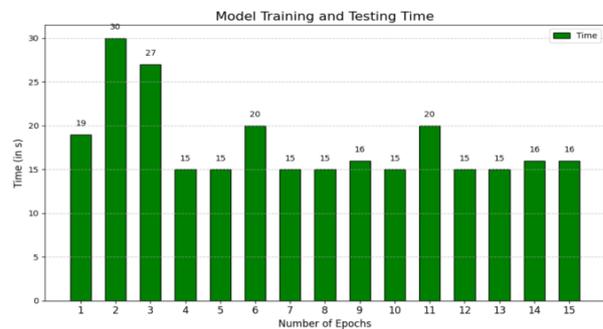


Fig. 5. Model training and testing time

VI. CONCLUSION

Convolutional Neural Networks as a category within deep, or multi-layered, algorithms achieve extremely good recognition accuracy for handwritten numbers. The CNN framework used for this research earned a 99.15% accuracy on the MNIST data set as supported by literature works.

The results show that smaller CNN designs like LeNet-5 may produce reliable results, even while richer models like ResNet offer little performance advantages. Furthermore, lightweight versions of edge devices enable efficient deployment with no accuracy loss.

In conclusion, deep learning innovations enable improved accuracy and versatility across several platforms, making CNNs a reliable and effective option for handwritten digit identification

jobs.

REFERENCES

- [1] L. Bottou, C. Cortes, and V. Vapnik, "Comparison of classifier methods: A case study in handwritten digit recognition," in Proceedings of the 12th IAPR International Conference on Pattern Recognition, vol. 2, 1994, pp. 77-82.
- [2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
- [3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems, 2012, pp. 1097-1105.
- [4] D. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in Proceedings of the International Joint Conference on Artificial Intelligence, 2011, pp. 1237-1242.
- [5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778.
- [6] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in Proceedings of the 23rd International Conference on Machine Learning (ICML), 2006, pp. 369-376.
- [7] A. Graves and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," in Advances in Neural Information Processing Systems, 2009, pp. 545-552.
- [8] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in Proceedings of the Seventh International Conference on Document Analysis and Recognition, 2003, pp. 958-963.
- [9] M. Huh, P. Agrawal, and A. A. Efros, "What makes ImageNet good for transfer learning?" in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2690-2698.
- [10] F. Wu, Q. Yang, and J. Zhou, "Few-shot learning for handwritten digit recognition," in Proceedings of the IEEE International Conference on Image Processing, 2019, pp. 123-127.
- [11] H. Zhao, Y. Song, and Y. Li, "Lightweight neural networks for handwritten digit recognition on edge devices," IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4229-4240, May 2020.
- [12] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in Proceedings of the 25th International Conference on Machine Learning, 2008, pp. 1096-1103.
- [13] D. Erhan, Y. Bengio, A. Courville, and P. Vincent, "Visualizing higher-layer features of a deep network," University of Montreal, 2009.
- [14] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in Proceedings of the 30th International Conference on Machine Learning, 2013, pp. 1319-1327.
- [15] L. Wan, M. Zeiler, S. Zhang, Y. LeCun, and R. Fergus, "Regularization of neural networks using dropconnect," in Proceedings of the 30th International Conference on Machine Learning, 2013, pp. 1058-1066.