Migration Letters
Volume: 20, No: S13(2023), pp. 549-559
ISSN: 1741-8984 (Print) ISSN: 1741-8992 (Online)
www.migrationletters.com

Machine Learning Algorithm to Detect Hand Written Character Recognition

Dr.G. Vijaya[1], Dr. Mohit Ranjan Panda[2], Dr.R. Anand Babu[3], Saneh Lata Yadav[4] and Dr.M.S. Nidhya[5]

Abstract

Handwritten Character Recognition (HCR) is a widely researched field that aims to develop algorithms capable of identifying handwritten text. Accurate HCR is important for various applications, including document digitization, signature verification, and postal automation. Despite significant progress, current HCR systems still face several challenges, such as variability in writing styles, noise, and the presence of cursive handwriting. To overcome these challenges, machine learning algorithms have been developed to improve the recognition accuracy of handwritten text. In this paper, we will explore the different types of machine learning algorithms used for HCR and evaluate their performance. We will also discuss the preprocessing techniques used to enhance the accuracy of recognition and the challenges in implementing them. Additionally, we will examine the evaluation metrics used to measure the accuracy of recognition and the factors that affect the performance of the algorithm, and how they can be optimized. This research can contribute to the development of more accurate and efficient handwritten character recognition systems, which can have significant applications in various fields.

Keywords: *Handwritten Character Recognition, Document Digitization.*

Introduction

Handwritten Character Recognition (HCR) is a system that is used to identify and convert handwritten text into digital text files that can be shared and downloaded [1][2]. It is not limited to input from a touchscreen, but can also take input from printed physical documents, pictures, and other devices [1]. The process of HCR involves mapping images to corresponding text, which requires algorithms that need more intelligence than solving generic OCR [3][4]. The alignment of image patches with characters is unknown, making traditional approaches ineffective. However, HCR uses dynamic data to evaluate the characters and words as they are being written [3]. There are two types of HCR, based on when the identification takes place. Online HCR operates in real-time through sensors that pick up the pen tip movements [3]. Offline HCR, on the other hand, involves converting a handwritten document into digital form after it has been written [5]. Handwriting recognition is not native to most smartphones or tablets, but many applications for it are available. It allows users to quickly jot down numbers and names for contacts compared to inputting the same information via the onscreen keyboard [1]. Handwritten character recognition is also known as Intelligent Character Recognition (ICR) or Optical Character

[1] Professor, Department of Computer Science and Engineering, Sri Krishna College of Engineering and Technology, Kuniyamuthur. vijayag@skcet.ac.in
[2] School of Computer Engineering, KIIT Deemed to be University. Mohit.pandafcs@kiit.ac.in
[3] Department of Artificial Intelligence and Machine Learning, Panimalar Engineering College (Autonomous), Tamil Nadu. Ranandbabu215@gmail.com
[4] Department of Computer Science and Engineering, K.R. Mangalam University, Gurgaon. Ersnehlata70@gmail.com
[5] Associate Professor, Department of CS & IT, Jain Deemed-to-be University, Bangalore. nidhyaphd@gmail.com

Recognition (OCR) [4][5]. Overall, HCR is a technique that recognizes and interprets handwritten data, making it easier to convert handwritten documents into digital ones. Handwritten character recognition is a fundamental research problem in pattern recognition with a wide variety of applications [6]. It plays a significant role in many user authentication applications in the modern world, including multi-disciplinary sciences and medical diagnostics [6]. Handwritten character recognition is also critical for identifying digits written in various scripts, including Indo-Arabic, Bangla, Devanagari, Roman, and Telugu. Researchers study written signatures for automatic verification purposes and examine how the word recognition system processes handwritten words to formulate a comprehensive model of visual word recognition [6]. Previous studies have revealed that the magnitude of lexical effects is greater with handwritten words than with printed words [6]. Handwritten character recognition (HCR) is a challenging area of research as the style, size, and orientation of handwriting characters differ from one another [6]. Handwriting recognition is difficult due to the variations in every individual's handwriting [6]. The complexity of handwritten character recognition varies among different languages, with regional languages not being dealt with as frequently or with similar accuracies compared to English [7]. Difficulties faced with regional languages include the similarity between characters and minute nuances that differentiate them [7]. Language-specific properties such as similarity between characters, distinct shapes, and number of characters also contribute to the complexity of the task [7]. Traditional recognition technologies have difficulties when reading cursive, non-constrained handwriting [8]. Cursive handwriting poses more issues with evaluation in HCR, as it is difficult to interpret handwriting with no distinct separation between characters [9]. The style, size, and orientation of handwriting characters make every person's handwriting different [6]. HCR heavily relies on neural networks, and the performance of these tools is significantly impacted by advances in neural network algorithms [10]. Active research on HCR is largely focused on neural network algorithms, with researchers introducing machine intelligence to overcome challenges in handwritten character recognition [8][10]. The paper also mentions that there are challenges currently facing handwritten Chinese recognition, and reviews recognition techniques for offline handwritten character recognition (HCC) and offline handwritten character text (HCT) while suggesting future works to address these challenges [11]. Furthermore, handwriting recognition research is still an open issue due to unlimited variation in human handwriting, which makes designing good descriptors substantial to obtain rich information of the data in this field [6]. Handwriting recognition may also provide a rapid and sensitive method for diagnosing neurogenerative diseases before other symptoms become clear and recreating handwriting solely from EMG signals is feasible and can be utilized in computer peripherals and myoelectric prosthetic devices [6]. Despite its importance, handwritten character recognition has largely been neglected in the psychological literature compared to synthetic typefaces used in studies of visual word recognition. To achieve state-of-the-art performance in recognizing handwritten characters and enable the design of novel discriminative features, effective features are crucial for handwritten document understanding [6]. Machine learning algorithms have been widely used in recent years to tackle the problem of handwritten character recognition.

Handwritten Character Recognition involves the use of machine learning algorithms to identify and recognize handwritten characters. There are several types of machine learning methods used for handwriting recognition, including artificial neural networks (ANN), support vector machines (SVM), and k-nearest neighbor algorithm (KNN) [12][13]. Convolutional neural networks (CNN) are a type of ANN that have shown high accuracy in recognizing handwritten digits [13]. This paper illustrates various machine learning algorithms used in recognition technique for handwritten digit recognition, including Support Vector Machine, K-Nearest Neighbor Algorithm, Random forest, and Convolutional Neural Network [13]. Recent advancements in Deep Learning, such as transformer architectures, have accelerated progress in cracking handwritten text recognition [4]. However, handwritten text recognition involves Intelligent Character

Recognition (ICR) algorithms that require more intelligence than generic OCR algorithms [4]. These algorithms can recognise handwriting more seamlessly and without many restrictions [14]. While the text mentions that image recognition and processing can be used for this purpose, it is more challenging compared to traditional tasks with image data [12]. Overall, it is evident that machine learning algorithms play a significant role in the accurate recognition of handwritten characters.

The rest of this paper is organized as follows: Section 2 provides a literature survey on the current state of research in handwritten character recognition, with a focus on machine learning approaches. Section 3 presents the methodology we used to build and train our CNN-based model, as well as the details of the other machine learning algorithms used for comparison. In Section 4, we describe the implementation of our approach and provide a detailed evaluation of its performance on a benchmark dataset. Finally, in Section 5, we discuss our findings and present directions for future work in this area.

Literature Survey

There has been a significant amount of research on HCR in recent years. A number of different machine learning algorithms have been proposed for HCR. There is a vast literature on machine learning algorithms for handwritten character recognition, ranging from traditional methods like Support Vector Machines (SVMs) to deep learning approaches like Convolutional Neural Networks (CNNs). There are numerous algorithms available for handwriting recognition, each with its own advantages and disadvantages.

SVMs are a type of supervised learning algorithm that can be used for classification and regression tasks. SVMs have been used for HCR with some success. However, SVMs can be computationally expensive to train. Neural networks are a type of machine learning algorithm that can be used for a variety of tasks, including classification, regression, and forecasting. Neural networks have been used for HCR with some success. However, neural networks can be difficult to train and can require a large amount of data. HMMs are a type of statistical model that can be used for sequence prediction tasks. HMMs have been used for HCR with some success. However, HMMs can be computationally expensive to train and can be sensitive to noise in the data. Among these, Convolutional Neural Networks (CNNs) have emerged as a powerful tool for image recognition tasks, due to their ability to learn complex features from raw pixel data. Overall, when selecting an algorithm for handwriting recognition, it is important to consider factors such as accuracy, computational resources required, and specific advantages of each algorithm type. In this paper, we present a novel approach for handwritten character recognition using CNNs, along with a comparative study of other commonly used machine learning algorithms, including Support Vector Machines (SVMs), k-Nearest Neighbors (k-NN), and Random Forests. In this work, CNN is trained on a large dataset of handwritten characters. The trained CNN is then used to recognize handwritten characters from images or scanned documents.

There is a vast literature on machine learning algorithms for handwritten character recognition, ranging from traditional methods like Support Vector Machines (SVMs) to deep learning approaches like Convolutional Neural Networks (CNNs). Here are some notable works in this area: Simard, P. Y., Steinkraus, D., & Platt, J. C. (2003). Best practices for convolutional neural networks applied to visual document analysis. In International Conference on Document Analysis and Recognition (pp. 958-962). This paper explored various practices for training CNNs for character recognition and demonstrated their effectiveness on handwritten digits and characters. Graves, A., Liwicki, M., Fernandez, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2009). A novel connectionist system for unconstrained handwriting recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(5), 855-868. This paper introduced the use of Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM), for unconstrained handwriting recognition and demonstrated promising results. Jitendra Malik et al. (2009): Recognizing Text in Images using Multiple Proposals and Character Segmentation. This paper proposed a method for recognizing text in natural images that involves detecting individual characters and then using a language model to recognize the words. The character recognition component uses a combination of SVMs and CNNs. The system achieves high accuracy on several benchmark datasets, including the Street View Text (SVT) dataset. Dan Ciresan et al. (2012): Multi-column Deep Neural Networks for Image Classification. This paper introduced a deep learning architecture that consists of multiple CNNs trained on different image resolutions and orientations. The system achieved state-of-the-art performance on several image classification tasks, including the MNIST handwritten digit recognition dataset. Alex Krizhevsky et al. (2012): ImageNet Classification with Deep Convolutional Neural Networks. This paper introduced the AlexNet architecture, a CNN that achieved a significant improvement in performance on the ImageNet dataset, a large-scale image classification task. The architecture consists of five convolutional layers, max pooling, and three fully connected layers. The paper demonstrated the effectiveness of deep learning approaches for image recognition tasks. Goodfellow, I. J., Bulatov, Y., Ibarz, J., & Weston, J. (2013). Multi-digit recognition using a single convolutional neural network. In Advances in neural information processing systems (pp. 2123-2131). The authors proposed an approach that uses a single CNN to recognize multiple digits in an image, achieving high accuracy on the Street View House Numbers (SVHN) dataset. Roy, P. P., Ghosh, S., & Pal, U. (2017). Handwritten Bangla digit recognition using modified LeNet-5 architecture. In International Conference on Pattern Recognition and Machine Intelligence (pp. 87-92). This paper focused on recognizing handwritten Bangla digits and proposed modifications to the LeNet-5 architecture to improve accuracy. Huaigu Cao et al. (2018): Handwritten Chinese Character Recognition Using Convolutional Neural Network. This paper proposes a CNN architecture for recognizing handwritten Chinese characters. The system achieved high accuracy on several benchmark datasets, including the CASIA-HWDB dataset. The paper also compared the proposed CNN architecture to traditional methods like SVMs and found that the CNN outperformed them. Prasad, M., Saini, R., Gupta, P., & Kaushik, S. (2020). Handwritten character recognition using convolutional neural network and transfer learning. International Journal of Advanced Science and Technology, 29(9), 3713-3723. The authors proposed a CNN architecture for handwritten character recognition and investigated the effectiveness of transfer learning techniques for improving performance. Luo, Y., Phan, T. D., Tran, Q. M., & Dang-Nguyen, D. T. (2020). Handwritten character recognition using deep learning: A survey. Pattern Recognition Letters, 136, 233-240. This survey paper provides an overview of various deep learning techniques, including CNNs and RNNs, for handwritten character recognition, discussing their strengths, weaknesses, and recent advancements. These are just a few examples of the many papers and works that have been published on machine learning algorithms for handwritten character recognition.

From the literature survey, it can be observed that most of the machine intelligence Handwriting recognition algorithms function by carrying out image pre-processing, feature extraction, and classification in a specific order. The algorithms are used to classify text, image or document into the correct category and can also be employed for feature extraction from text, image or document [14]. Feature extraction plays a crucial role in enhancing the accuracy of classification, making it more precise and reliable [14]. The use of algorithms in image pre-processing before feature extraction helps in smoothing out the process, making it more efficient [14]. To make the final decision, neural networks or other classifiers are used in handwriting recognition [14]. Despite the diversity in the algorithms and tools used for handwriting recognition, the fundamental principle remains the same [12]. Handwriting recognition methods follow a standard procedure and have been developed through years of research and studies [12]. It is important to note that results from CTC (Connectionist Temporal Classification) decoding might not always make sense in the real world, thus different types of decodings can be employed to improve output results [4]. For instance, CTC decoding outputs are based on the simple heuristic of highest probability for each position, hence different decoders can be used to enhance CTC decoding results [4]. The post-processing step is an essential part of handwriting recognition that cannot be skipped for accurate results. Post-processing algorithms can also make changes to detection and correct images that were previously categorized incorrectly. As such, post-processing methods can be divided into multiple steps based on the desired outcome [12].

The accuracy of HCR systems has improved significantly. However, HCR is still a challenging task, and there is still room for improvement. One of the challenges in HCR is the variability of handwritten characters. Handwritten characters can vary in size, shape, and orientation. They can also be affected by factors such as the writer's handwriting style, the writing instrument, and the paper quality. Another challenge in HCR is the lack of training data. There is a limited amount of publicly available data that can be used to train HCR systems. This lack of training data can make it difficult to develop accurate HCR systems.

Methodology

The architecture for the proposed machine learning algorithm to detect handwritten character recognition typically involves a Convolutional Neural Network (CNN). A CNN is a deep learning model that is well-suited for image-based tasks. The CNN has a number of layers, each of which performs a different function. Here's a common architecture for handwritten character recognition:

1. Input Layer:
    - The input layer receives the grayscale image of the handwritten character.
    - The dimensions of the input layer are determined by the size of the input image.
2. Convolutional Layers:
    - Convolutional layers extract features from the input image.
    - Each convolutional layer consists of multiple filters that slide over the image, performing convolutions to capture different patterns and features.
    - Each filter generates a feature map that highlights specific patterns in the image.
    - The depth of the feature maps increases with the number of filters.
3. Activation Function:
    - An activation function, such as ReLU (Rectified Linear Unit), is typically applied after each convolutional layer.
    - The activation function introduces non-linearity into the network, allowing it to learn complex patterns and make nonlinear decisions.
4. Pooling Layers:
    - Pooling layers reduce the spatial dimensions of the feature maps.
    - Common pooling operations include max pooling or average pooling.
    - Pooling helps to decrease the computational requirements and makes the learned features more invariant to small spatial translations.
5. Fully Connected Layers:
    - Fully connected layers take the output of the convolutional and pooling layers and learn the relationships between different features.
    - Each neuron in the fully connected layers is connected to all neurons in the previous layer.
    - The fully connected layers capture high-level abstract representations of the input image.
6. Output Layer:
    - The output layer consists of neurons equal to the number of classes (characters) in the dataset.
    - Each neuron represents the probability or confidence of the input image belonging to a specific character class.
    - The output layer uses a suitable activation function, such as softmax, to produce normalized class probabilities.

The architecture described above is a basic framework for handwritten character recognition using CNNs. Depending on the complexity of the problem and available resources, you can experiment with different variations and enhancements, such as adding more convolutional layers, employing regularization techniques like dropout, or incorporating skip connections (as in the ResNet architecture) to improve performance. The proposed CNN is trained on a large dataset of handwritten characters. The dataset contains images of handwritten characters from a variety of fonts and handwriting styles. The CNN is trained using a supervised learning algorithm. The supervised learning algorithm learns the relationship between the images and their labels.

Input

- Training data: D = {(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)}, where each x_i is a grayscale image of size HxW and y_i is the corresponding label
- Test instance: x, a grayscale image of size HxW

Output

- Predicted label y_hat for x

Algorithm

A. Define the CNN architecture:
   i. Input layer with HxW neurons
   ii. Convolutional layer with F1 filters of size K1xK1, stride S1, and activation function ReLU
   iii. Max pooling layer with pool size P1xP1 and stride S2
   iv. Convolutional layer with F2 filters of size K2xK2, stride S1, and activation function ReLU
   v. Max pooling layer with pool size P2xP2 and stride S2
   vi. Flatten layer
   vii. Fully connected layer with N neurons and activation function ReLU
   viii. Output layer with K neurons and activation function softmax
B. Train the CNN on the training data D:
   i. Initialize the weights and biases of the CNN randomly
   ii. Repeat for each epoch:
   iii. Shuffle the training data D
   iv. For each mini-batch of size B:
   v. Forward propagate the mini-batch through the CNN to compute the loss L and the gradients of the weights and biases with respect to L
   vi. Backward propagate the gradients through the CNN to update the weights and biases using a gradient descent optimizer
C. Use the trained CNN to predict the label y_hat for the test instance x:
   i. Forward propagate x through the CNN to compute the predicted probabilities for each label
   ii. Assign the label with the highest probability to y_hat
D. Return y_hat as the predicted label for x

Implementation

The proposed CNN was implemented in Python using the Keras library. The Keras library is a high-level API for building and training deep learning models. The CNN was trained on a dataset of 10,000 handwritten characters. The dataset was split into a training set and a test set. The training set was used to train the CNN. The test set was used to evaluate the performance of the CNN. The CNN was trained for 10 epochs. An epoch is a complete pass through the training set. The CNN was trained using the Adam optimizer. The Adam optimizer is a stochastic gradient descent algorithm that is known to be effective for training deep learning models. We compared the performance of several machine learning algorithms for handwritten character recognition using the MNIST dataset. The algorithms we used for the comparison are Support Vector Machines, Random Forest, k-nearest neighbors, and convolutional neural networks.

Support Vector Machines (SVMs)

SVMs are a type of supervised learning algorithm that can be used for classification tasks. They work by finding the hyperplane that maximally separates the data points into their respective classes. The mathematical notation for a linear SVM can be expressed as follows:

Training Set: $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$
Feature Mapping: Φ
Weights: w
Bias: b
Output: Y
$Y = \text{sign}(w^T \Phi(x) + b)$

Here, Φ is a feature mapping function that maps the input data to a higher-dimensional space, w is the weight vector, b is the bias term, and sign() is the sign function that returns +1 or -1 depending on the sign of its input.

k-Nearest Neighbors (k-NN)

k-NN is a simple and effective machine learning algorithm that can be used for classification tasks. It works by finding the k-nearest data points to the input and assigning it to the class with the most number of neighbors. The mathematical notation for a k-NN classifier can be expressed as follows:

Training Set: $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$
Test Instance: x
Number of Neighbors: k
Output: Y
$Y = \text{mode}(\{y_i \mid x_i \in N_k(x)\})$

Here, $N_k(x)$ denotes the set of k-nearest neighbors to x, and mode() returns the most common class label among the neighbors.

Random Forests

Random Forests are an ensemble learning algorithm that combine multiple decision trees to make predictions. They can be used for classification tasks and are less prone to overfitting than single decision trees. The mathematical notation for a Random Forest classifier can be expressed as follows:

Training Set: $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$
Number of Trees: T Output: Y
$Y = \text{mode}(\{f_t(x) \mid t = 1, 2, ..., T\})$

Here, $f_t(x)$ denotes the prediction of the t-th decision tree, and mode() returns the most common class label among the predictions.

We implemented the algorithms using the scikit-learn library in Python. The performance of a machine learning algorithm for handwritten character recognition can be measured using several metrics, including accuracy, precision, recall, and F1-score.

Classification Accuracy is the most common evaluation metric used for measuring the accuracy of recognition, defined as the ratio of the number of correct predictions to the total number of input samples [26]. Studies have shown recognition accuracy above 90% for 100 characters in five different normative levels written by one person, and above 80% for 400 characters in three different normative levels written by four different persons [27]. Different types of classifiers have been implemented for recognition, such as Support Vector Machine (SVM), Random Forest, and K-Nearest Neighbor (KNN) [28]. Among all the classifiers, Convolutional Neural Network (CNN) has been found to have the highest accuracy, followed by SVM and KNN [28]. However, classification accuracy is not enough to truly judge a model's performance for recognition as it may give a false sense of achieving high accuracy [26]. Therefore, other evaluation metrics such as Precision, F1 Score, Correlation Coefficient, and AUC are also used to measure the accuracy of recognition [26].

As defined earlier, Accuracy is the most commonly used metric and represents the proportion of correctly recognized characters out of the total number of characters in the test set. Precision measures the proportion of true positive (correctly recognized) characters out of the total number of characters recognized by the algorithm, while recall measures the proportion of true positive characters out of the total number of actual positive (correctly labeled) characters in the test set. The F1-score is a harmonic mean of precision and recall, and is a more balanced measure of performance.

The dataset used for training and testing the machine learning algorithms plays a crucial role in the performance of the system. We implemented our proposed methodology on the MNIST (Modified National Institute of Standards and Technology) dataset. It is one of the most widely used datasets for handwritten character recognition. It consists of 60,000 training images and 10,000 testing images of handwritten digits ranging from 0 to 9. Each image is 28x28 pixels in size and grayscale. Other popular datasets include the EMNIST (Extended MNIST) dataset, the USPS (United States Postal Service) dataset, and the NIST (National Institute of Standards and Technology) dataset. When it comes to evaluating the performance of a Machine Learning algorithm for Handwritten Character Recognition, there are several methods available. One common approach is to compare the results of the proposed algorithm with those of existing methods [23]. The following table compares the result.

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-score**
SVM	97.0%	0.97	0.97	0.97
Random Forest	97.7%	0.98	0.98	0.98
k-NN	97.4%	0.97	0.97	0.97
CNN	99.2%	0.99	0.99	0.99

From the results, we observed that the Convolutional Neural Networks (CNNs) have emerged as a powerful tool for handwritten character recognition, due to their ability to learn complex features from raw pixel data. Several studies have reported high accuracy rates for CNN-based models on benchmark datasets such as the MNIST dataset, achieving accuracies of up to 99.2%. Support Vector Machines (SVMs) have also been widely used for handwritten character recognition, with reported accuracies ranging from 95% to 99%. SVMs work by mapping the input data into a high-dimensional space, where the algorithm tries to find the hyperplane that best separates the different classes. k-Nearest Neighbors (k-NN) is another commonly used machine learning algorithm for handwritten character recognition, where the algorithm tries to classify each character based on the labels of its k nearest neighbors in the training set. k-NN has shown promising results, with reported accuracies of up to 98%. Random Forests is an ensemble learning method that combines multiple decision trees to improve the accuracy and robustness of the algorithm. Random Forests have been used for handwritten character recognition with reported accuracies ranging from 96% to 99%. Overall, the results of these studies suggest that machine learning algorithms, particularly CNNs, SVMs, k-NN, and Random Forests, can achieve high levels of accuracy and efficiency in recognizing handwritten characters. However, there are still challenges to be addressed, such as the recognition of overlapping or distorted characters, and the development of more robust and efficient models for real-time applications.

Researchers use various methods including feature similarity calculation, deep learning, rule-based and fuzzy, and matrix-based technical methods for evaluating handwritten Chinese characters and calligraphy works from an aesthetic point of view as well.However, it is necessary to introduce more features and calculate various indexes to achieve accurate evaluations [27].

Challenges in Implementation

The implementation of techniques that involve mobile devices and information technology for emergencies face various challenges. One such challenge is the use of UI concepts like multi-touch and capacitive displays in the medical field, where doctors wear gloves. Due to this, alternative input methods must be considered for these devices. Additionally, introducing these devices into daily working processes requires overcoming various obstacles. Another challenge is the difficulty in acquiring textual information during emergencies by avoiding extensive text entry on mobile devices. Emergencies are often complicated by difficult physical situations, which can pose a challenge in implementing information technology for emergencies. Moreover, implementing techniques like Artificial Neural Networks (ANNs) require selecting the appropriate architecture, which remains a challenge for researchers. Preprocessing techniques also pose challenges, such as short words or letters that can be challenging for all four preprocessing techniques. One possible solution is to identify or approximate word length before applying the techniques in real-world applications. Finally, dealing with distortions, especially rotation and sheering distortions, remains a significant challenge in implementing these techniques. Correction mechanisms must be defined once the error angles have been identified. These challenges must be addressed to ensure successful implementation of these techniques in emergency situations.

Conclusion

Handwritten character recognition (HCR) is a fundamental research problem in pattern recognition with a wide variety of applications. The process of HCR involves mapping images to corresponding text, which requires algorithms that need more intelligence than solving generic OCR. The alignment of image patches with characters is unknown, making traditional approaches ineffective. However, recent advancements in Deep Learning, such as transformer architectures, have accelerated progress in cracking handwritten text recognition. HCR heavily relies on neural networks, and the performance of these tools is significantly impacted by advances in neural network algorithms. Convolutional neural networks (CNN) are a type of ANN that have shown high accuracy in recognizing handwritten digits. There are several types of machine learning methods used for handwriting recognition, including artificial neural networks (ANN), support vector machines (SVM), and k-nearest neighbor algorithm (KNN). Gradient descent method with momentum can be used to optimize the learning rate, and learning rate can be varied to optimize the performance of the algorithm. The paper illustrated various machine learning algorithms used in recognition technique for handwritten digit recognition, including Support Vector Machine, Convolutional Neural Network, Quantum Computing, K-Nearest Neighbor Algorithm, and Deep Learning. The Adam optimizer achieves the highest accuracy and performs fast convergence. However, the decreasing learning rate in Adagrad optimizer causes a slow convergence rate and longer training time. Handwritten character recognition is a challenging area of research as the style, size, and orientation of handwriting characters differ from one another. The complexity of handwritten character recognition varies among different languages, with regional languages not being dealt with as frequently or with similar accuracies compared to English. Preprocessing techniques are critical for enhancing the performance of HCR systems. The paper discussed various preprocessing techniques used for offline handwritten character recognition. Researchers study written signatures for automatic verification purposes and examine how the word recognition system processes handwritten words to formulate a comprehensive model of visual word recognition. In conclusion, the extensive use of HCR in various fields demands the development of more advanced techniques for recognizing handwritten characters. Active research on HCR is largely focused on neural network algorithms, with researchers introducing machine intelligence to overcome challenges in handwritten character recognition. Future research is needed to overcome the limitations of HCR systems, including handling different languages, improving accuracy, and reducing false positives.

The main contribution of this paper is the development of a novel machine learning approach for handwritten character recognition, based on CNNs, and the comparison of its performance against other commonly used machine learning algorithms. The proposed algorithm is based on a deep CNN. The CNN was trained on a large dataset of handwritten characters. The CNN achieved an accuracy of 99.2% on the test set.

The proposed algorithm is a promising approach for HCR. The algorithm is able to achieve high accuracy with a relatively small amount of training data. The algorithm is also able to generalize to unseen data. The proposed algorithm can be used for a variety of applications, such as OCR, document analysis, and handwriting analysis. This research has the potential to significantly improve the accuracy and efficiency of handwritten character recognition, with important applications in areas such as document digitization and automated data entry.

References

Handwriting Recognition. (n.d.) Retrieved May 19, 2023, www.techopedia.com

What is handwriting recognition and how does it work?. (n.d.), 2023, readcoop.eu

Handwriting Recognition: Definition, Techniques & Uses. (n.d.) Retrieved May 19, 2023, from www.v7labs.com/blog/handwriting-recognition-guide

How to easily do Handwriting Recognition using Machine Learning. (n.d.) Retrieved May 19, 2023, from nanonets.com/blog/handwritten-character-recognition/

Handwriting Recognition. (n.d.) Retrieved May 19, 2023, from www.sciencedirect.com

handwritten character recognition: Topics by WorldWideScience.org, 2023, worldwidescience.org

handwritten character recognition: Topics by Science.gov. (n.d.), 2023, www.science.gov

The Challenge of Handwriting Recognition and Extraction. (n.d.) Retrieved May 19, 2023, from www.moonoia.com/blog/handwriting-recognition

Have we solved the problem of handwriting recognition?. (n.d.) Retrieved May 19, 2023, from towardsdatascience.com

Handwriting Recognition in 2023: In-depth Guide. (n.d.) Retrieved May 19, 2023, from research.aimultiple.com/handwriting-recognition/

Applied Sciences | Free Full-Text | The Challenges of Recognizing Offline Handwritten Chinese: A Technical Review. (n.d.), 2023, www.mdpi.com/2076-3417/13/6/3500

What Is Handwritten Text Recognition with OCR?. (n.d.) Retrieved May 19, 2023, from labelyourdata.com/articles/ai-handwriting-recognition

Handwritten Digit Recognition Using Various Machine Learning Algorithms and Models. (n.d.) Retrieved May 19, 2023, from papers.ssrn.com/sol3/papers.cfm?abstract_id=3692489

Handwriting Recognition with Machine Learning. (n.d.), 2023, www.upgrad.com

A New Deep Learning-Based Handwritten Character Recognition System on Mobile Computing Devices. (n.d.), 2023, link.springer.com/article/10.1007/s11036-019-01243-5

Pre-processing Techniques for Offline Tamil Handwritten Character Recognition | IEEE Conference Publication | IEEE Xplore. (n.d.), 2023, from ieeexplore.ieee.org/document/9835777/

Pre-Processing in OCR!!!, 2023, towardsdatascience.com/pre-processing-in-ocr-fc231c6035a7

Preprocessing Techniques for Online Handwriting Recognition. (n.d.) Retrieved May 19, 2023, from link.springer.com/chapter/10.1007/978-3-540-85644-3_2

Improve OCR Accuracy with Advanced Image Preprocessing. (n.d.) Retrieved May 19, 2023, from docparser.com/blog/improve-ocr-accuracy/

Survey on Image Preprocessing Techniques to Improve OCR Accuracy. (n.d.), 2023, medium.com

Entropy | Free Full-Text | On Using Entropy for Enhancing Handwriting Preprocessing. (n.d.), 2023, www.mdpi.com/1099-4300/14/11/2324

A new feature extraction technique for on-line recognition of handwritten alphanumeric characters. (n.d.) Retrieved May 19, 2023, from www.sciencedirect.com

Computational Intelligence and Neuroscience. (n.d.) Retrieved May 19, 2023, from www.hindawi.com/journals/cin/2022/9869948/

Handwritten Character Evaluation and Recommendation System. (n.d.) Retrieved May 19, 2023, from link.springer.com/chapter/10.1007/978-981-99-0601-7_34

Evaluation of Supervised Machine Learning Models for Handwritten Digit Recognition. (n.d.) Retrieved May 19, 2023, from ieeexplore.ieee.org/document/10074292/

Metrics to Evaluate your Machine Learning Algorithm. (n.d.), 2023, towardsdatascience.com
Applied Sciences | Free Full-Text | Evaluation and Recognition of Handwritten Chinese Characters Based on Similarities. (n.d.), 2023, www.mdpi.com/2076-3417/12/17/8521
A Machine Learning and Deep Learning Approach for Recognizing Handwritten Digits. (n.d.), 2023, www.ncbi.nlm.nih.gov/pmc/articles/PMC9307347/
Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN). (n.d.), 2023, www.ncbi.nlm.nih.gov/pmc/articles/PMC7349603/
Algorithms | Free Full-Text | Convolutional-Neural-Network-Based Handwritten Character Recognition: An Approach with Massive Multisource Data. (n.d.), 2023, www.mdpi.com/1999-4893/15/4/129