HAND WRITTEN CHARACTER RECOGNITION USING CONVOLUTION NEURAL NETWORK

V Sai Kushal (21MIC0001),

K Mohan (21MIC0022),

T Rajashekar Reddy (21MIC0036),

N Harish Kumar (21MIC0049),

K Karthikeya (21MIC0059)

Abstract:

This project aims to develop a machine learning model capable of recognizing handwritten lowercase English alphabets from images. To achieve this, we trained a Convolutional Neural Network (CNN) on preprocessed grayscale images. One of the major challenges encountered during training was the imbalance in the dataset, where certain alphabets appeared more frequently than others. To resolve this issue and ensure fair learning for all characters, we implemented the Adaptive Synthetic Sampling (ADASYN) oversampling technique. This method generates synthetic samples for less common classes, helping the model learn better. The trained model showed high prediction accuracy, making it suitable for real-world applications such as document digitization, educational software, and character-based Optical Character Recognition (OCR) systems.

Index Terms:

ADASYN, Alphabet classification, Class imbalance, CNN, Deep learning, Image recognition, Deep learning, Optical character recognition.

I. INTRODUCTION

Background and motivation for the research:

In today's world, most tasks are becoming digital, but handwriting is still used in schools, offices, and many everyday situations. Recognizing handwritten text automatically can save time, reduce human error, and help digitize documents. While recognizing numbers (like in bank cheques or postcodes) has been widely solved using deep learning, identifying handwritten alphabets is more difficult because letters have more complex shapes and people write them in different ways. This challenge motivated us to build a model that can read handwritten lowercase English letters accurately using deep learning techniques.

Research problem / Question:

The main question we aimed to answer was: Can we build a deep learning model that accurately recognizes all 26 lowercase English letters, even when the dataset is imbalanced? We observed that some letters appeared far more often than others in the dataset, which caused the model to become biased and perform poorly on rare letters. This imbalance needed to be addressed to ensure fair and accurate recognition across all classes.

Significance and Objectives of the Study:

This project is important because it helps bridge the gap between written and digital content. Our objective was to develop a system that could recognize handwritten lowercase alphabets using images. We also wanted to make sure that the system worked well for all letters, not just the common ones.

To achieve this, we applied an oversampling method called ADASYN, which generates extra training samples for characters that had fewer examples. This helped make the training data more balanced and improved the model's performance on underrepresented letters.

A Brief Overview of the Research:

We started by collecting a dataset of handwritten alphabets saved as image files. Each image was converted to grayscale and resized to prepare it for training. The labels were taken from the characters in the filenames. Since some letters had fewer samples than others, we used ADASYN to generate synthetic data and balance the dataset. After that, we built and trained a Convolutional Neural Network (CNN) on this improved dataset. The model was then tested on new images to see how accurately it could recognize the letters, and the final results showed consistent performance across most characters.

II. LITERATURE SURVEY:

Review of existing research relevant to the topic:

Reference No.	Approached used	Key findings	Drawbacks
[1]	CNN framework with preprocessing, feature extraction, and classification to train on diverse handwritten datasets, ensuring accurate character recognition.	- CNN framework could be integrated with other technologies, such as Optical Character Recognition (OCR) systems	The model showed robustness to different handwriting styles, there may still be challenges in generalizing.

[1] Handwritten character recognition has improved significantly with Convolutional Neural Networks (CNNs), enhancing accuracy and efficiency. Studies emphasize the importance of robust architectures, data augmentation, and diverse datasets like the IAM Handwriting Database. However, challenges remain, including poor generalization to varied handwriting, high computational demands, and limited interpretability. This project proposes a CNN framework incorporating meta-learning for adaptability, optimization for real-time processing, and attention mechanisms for transparency. By leveraging diverse datasets and innovative methods, the goal is to create a robust, efficient, and interpretable model, addressing key limitations and advancing handwriting recognition technology.

Reference No.	Approached used	Key findings	Drawbacks
[2]	CNN with convolutional, pooling, and fully connected layers to classify characters effectively.	The use of ReLU activation functions and max-pooling layers contributed to improved performance.	Potential overfitting, particularly during training.

[2] Handwritten character recognition (HWCR) using Convolutional Neural Networks (CNNs) has advanced significantly, particularly for widely spoken languages with rich datasets. Existing research, such as studies on Javanese, Devanagari, and Urdu scripts, highlights the effectiveness of CNNs in

recognizing complex characters by leveraging techniques like feature extraction, preprocessing, and advanced network architectures. However, challenges persist, including variability in handwriting styles and the lack of datasets for less-studied languages. This project addresses these gaps by focusing on Asante Twi, creating a comprehensive dataset of 13,200 character samples and developing a CNN framework tailored to its linguistic nuances. With preprocessing techniques and advanced CNN architectures, the project aims to achieve high recognition accuracy while contributing to the digitization and preservation of the Asante Twi language. This research not only fills existing gaps but also advances HWCR for underrepresented scripts, fostering broader linguistic inclusion in AI technologies.

Reference No.	**Approached used**	**Key findings**	**Drawbacks**
[3]	Preprocessing images, segmenting characters using CNNs	The ability to recognize characters from both Bangla and English scripts	The complexity of Bangla script, with similar shapes and strokes, increases misclassifications, requiring model refinement.

[3] The literature on handwritten character recognition (HWCR) highlights significant advancements through the use of Convolutional Neural Networks (CNNs). Studies have demonstrated the effectiveness of CNNs for recognizing Bangla and English characters using techniques like segmentation, feature extraction, and classification. Despite achieving high accuracy levels, challenges remain, including handling diverse handwriting styles and optimizing models for real-time applications. Existing research predominantly focuses on widely spoken languages, leaving a gap in datasets and methods for less-studied scripts. This project seeks to address these gaps by developing a CNN framework tailored to Asante Twi, creating a dataset of 13,200 handwritten characters and integrating preprocessing steps and advanced CNN architectures. The goal is to improve accuracy, efficiency, and adaptability for recognizing Asante Twi alphabets, contributing to the digitization and preservation of this language while advancing HWCR for underrepresented scripts.

Reference No.	**Approached used**	**Key findings**	**Drawbacks**
[4]	CNN-based approach involving data collection, preprocessing, feature extraction.	The model achieved 97.90% testing accuracy, demonstrating its effectiveness in recognizing handwritten styles.	Designed specifically for recognizing English characters

[4] The literature on handwritten character recognition (HWCR) highlights significant advancements in the use of Convolutional Neural Networks (CNNs). Several studies demonstrate their effectiveness in recognizing handwritten characters across various scripts. Techniques such as preprocessing, segmentation, feature extraction, and classification are crucial to these systems. While widely used datasets like MNIST have been instrumental, research on less-explored languages, such as Asante Twi, remains sparse. Current models often face challenges in generalization, efficiency, and handling diverse handwriting styles. This project aims to bridge these gaps by leveraging CNNs to develop a robust HWCR system tailored for Asante Twi alphabets, utilizing a comprehensive dataset and advanced methodologies. The findings will contribute to both the preservation of underrepresented languages and advancements in HWCR technologies.

Reference No.	**Approached used**	**Key findings**	**Drawbacks**
[5]	SVM, MLP, CNN models used Measured - Accuracy, speed, how hard they are, how many tries. **Showed results** Using bars and tables. MLP-Multilevel Preceptron SVM-Support Vector Machines	SVM was best at learning the training data (99.98% accuracy). CNN was best at recognizing new data (99.31% accuracy), even though it took longer duration.	**CNN:** Too hard, takes too long, not great for quick tasks. **MLP:** Had trouble with some numbers, not good with confusing things. **Overtraining:** Practicing too much made it worse, needs careful setup

[5] The literature on handwritten digit recognition has significantly advanced with the development of various machine learning and deep learning algorithms, such as Support Vector Machines (SVM), Multi-Layer Perceptron (MLP), and Convolutional Neural Networks (CNN). So the, gaps remain in the comparative analysis of these algorithms regarding their execution time, accuracy, and complexity, particularly in real-world applications. This project aims to address these gaps by providing a comprehensive evaluation of these algorithms, establishing a clear methodology for their implementation, and outlining expected outcomes that will enhance the understanding of their effectiveness in digit recognition tasks.

Reference No.	Approached used	Key findings	Drawbacks
[6]	Convolutional Neural Network (CNN) with varying filter sizes to effectively extract features from images in the MNIST dataset.	CNN model achieved an impressive accuracy of 99.5% in recognizing handwritten digits, outperforming existing models.	The model may not perform well on handwriting styles different from MNIST. Improvements in local feature extraction are needed to address potential misclassifications.

[6] The project builds upon the literature study based on handwritten digit recognition, particularly focusing on the advancements made through Convolutional Neural Networks (CNNs). While the literature study have reported high accuracy rates ranging from 98% to 99.3%, many lack rigorous evaluation methods, leading to potential overfitting and unreliable results. This project addresses these gaps by implementing a robust cross-validation approach, ensuring the reliability of the reported 99.5% accuracy, and providing a solid foundation for the methodology that aims to enhance feature extraction and classification processes in CNN.

Reference No.	Approached used	Key findings	Drawbacks
[7]	Convolutional Neural Networks (CNNs) for handwritten digit recognition, MNIST dataset for training and validation.	Improves accuracy in recognition compared to other models	Need lots of data, uses much computing power, may not work well with different handwriting.

[7] The literature on handwritten digit recognition has predominantly focused on traditional machine learning algorithms, which often struggle with variability in handwriting styles and noise in data. While advancements in Convolutional Neural Networks (CNNs) have shown significant improvements in accuracy, there remains a gap in real-time application and adaptability to diverse handwriting inputs. This project aims to address these gaps by implementing a robust CNN architecture combined with data augmentation techniques, thereby enhancing the model's performance and reliability in practical scenarios.

Reference No.	Approached used	Key findings	Drawbacks
[8]	Combinations like CNN-RNN, used feature extraction (HoG), and tested various classifiers. RNN-Recurrent Neural Networks HoG-Histogram of Oriented Gradients	Deep learning CNNs work well for recognizing handwritten characters.	Not good at recognizing natural scenes or handwriting. Contour methods don't work well. Need to improve how it finds edges.

[8] The literature on Optical Character Recognition (OCR) and handwritten character recognition primarily focuses on statistical methods and neural networks, particularly Convolutional Neural Networks (CNNs). While significant advancements have been made, particularly in achieving high accuracy rates, there remain gaps in the effective recognition of characters in unevenly distributed images and natural scenes, which often challenge current contour-based techniques. This project aims to address these gaps by implementing efficient contour detection methods and leveraging larger datasets for training, thereby enhancing the methodology and expected outcomes in real-time handwritten character recognition.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[9]	To create a Convolutional Neural Network (CNN) model capable of accurately recognizing handwritten digits and characters from images.	- The CNN model achieved a recognition accuracy of 90.54% and a loss of 2.53% for handwritten characters. Demonstrating its effectiveness in handling the complexities of handwritten text recognition despite the challenges posed by individual writing styles and character similarities.	-**Character Confusion**: Certain characters, such as "1," "I," "J," "S," "T," and "f," showed low AUC values, indicating difficulties in accurately distinguishing between similar-looking characters.

[9] The project focuses on improving Handwritten Text Recognition using Convolutional Neural Networks (CNN) to address the challenges posed by varying handwriting styles and the complexity of optical pattern recognition. The objective is to enhance Optical Character Recognition (OCR) by implementing Intelligent Character Recognition (ICR) to convert scanned handwritten or printed characters into ASCII text. The methodology involves training a CNN model on a dataset from NIST containing 101,784 images across 47 classes, employing techniques such as convolution, pooling, and dropout regularization. Key findings indicate that the model achieved an accuracy of 90.54% with a loss of 2.53%, showcasing its effectiveness in recognizing handwritten characters. However, the study notes that while CNN provides satisfactory results, it may not be the most advanced recognition algorithm available. In conclusion, the research demonstrates that CNN is a viable solution for handwritten character recognition, effectively addressing the problem while acknowledging the potential for more sophisticated methods.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[10]	Develop a CNN-based model for handwritten Tamil character recognition	- The proposed Convolutional Neural Network (CNN) model achieved an overall accuracy of 92.24% in recognizing handwritten Tamil characters.	- The model may produce false output due to the diverse writing styles of individuals, which can affect recognition accuracy.

[10] The objective of the study was to develop a Convolutional Neural Network (CNN) model for recognizing handwritten Tamil characters, addressing the challenge posed by the complexity and similarity of the 156-character Tamil script. The methodology involved preprocessing the dataset, which included 500 samples for each character, by reducing noise and converting images to grayscale, followed by classification using the CNN model. Key findings indicated that the model achieved an overall accuracy of 92.24% in recognizing all 12 vowels, although it faced drawbacks due to variations in individual writing styles that could lead to misrecognition. The results demonstrated the effectiveness of the CNN approach in character recognition, and the conclusion highlighted the potential for future enhancements, including the addition of more character classes and improved translation capabilities.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[11]	The study proposes a custom-tailored Convolutional Neural Network (CNN) model for handwritten character recognition (HCR).	- The study achieved exceptional accuracy in handwritten character recognition using optimized CNN models, with the best results being 99.563% for alphabet recognition and 99.642% for digit recognition, alongside robust evaluations addressing dataset imbalance.	- As its use of imbalanced datasets and focus on English characters, limiting its real-world applicability and extension to complex scripts.

[11] The paper presents a novel approach to handwritten character recognition (HCR) using a tailored convolutional neural network (CNN) model, aiming to enhance recognition accuracy for both digits and English alphabets. The methodology involves image acquisition, preprocessing, segmentation, feature extraction, and classification, utilizing various optimizers and learning rates to evaluate model performance across the Kaggle and MNIST datasets. Key findings indicate that the proposed model achieved an impressive accuracy of 99.642% for digit recognition and 99.563% for alphabet recognition, outperforming existing techniques; however, the study acknowledges the imbalanced dataset as a drawback that may affect performance evaluation. The results demonstrate the model's effectiveness in HCR tasks, leading to the conclusion that this CNN-based approach significantly advances the field, with suggestions for future work to explore more complex languages and feature extraction methods.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[12]	The research is to design develops, construct, deploy, and test a convolutional neural network (CNN) for handwriting recognition.	CNN achieved up to 98.7% accuracy; potential for 99.89% with dataset expansion and tuning.	-The research primarily focuses on neural networks for handwriting recognition, neglecting other potential methods like Hidden Markov Models or Support Vector Machines.

[12] The research paper focuses on developing a Convolutional Neural Network (CNN) for handwriting recognition, aiming to design, implement, and deploy a deep learning-based multi-class classifier. Utilizing the MNIST dataset, the methodology involved image preprocessing, neural network architecture design, and extensive model training, which resulted in the CNN achieving an impressive accuracy of 98.72%, outperforming a Simple Neural Network (SNN) with 92% accuracy. Key findings indicate that the number of training runs significantly impacts model effectiveness, while drawbacks include the model's limitation to static images and the lack of exploration of alternative techniques like Hidden Markov Models. The conclusion suggests that while the results are promising, further enhancements are necessary for practical applications, including dynamic handwriting recognition and broader dataset utilization.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[13]	Statistical Based SVM Framework, Template Matching Method.	- Proposed model predicts the characters with better accuracy in recognizing stylish characters compared to existing models.	- Focuses only on widely spoken languages, neglecting regional languages.

[13] This paper explores handwritten character recognition (HCR), a multidisciplinary field involving artificial intelligence, computer vision, and pattern recognition. Modern systems employ machine learning techniques, including neural networks, support vector machines (SVMs), and statistical methods such as k-Nearest Neighbors (kNN). Key methodologies include preprocessing for noise reduction, feature extraction using kernel-based approaches, and classification through techniques like template matching and multi-layer perceptrons. Future research focuses on expanding datasets for regional languages, improving recognition under challenging conditions, and developing robust, real-time systems for diverse handwriting styles.

Reference No.	**Approach Used**	**Key Findings**	**Drawbacks (if any)**
[14]	SVM, KNN, Random Forest, CNN	- All algorithms achieved similar accuracy for handwritten digit recognition, differing by only ±1%. - Convolutional Neural Networks (CNN) provided the highest recognition accuracy.	-CNN required exponential computing time compared to other models.

[14] This paper presents a comprehensive study of handwriting recognition, focusing on the evaluation of machine learning and deep learning algorithms for handwritten digit recognition. Specifically, it compares algorithms such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Random Forest (RF), and Convolutional Neural Networks (CNN) based on their performance in recognizing handwritten digits. The paper evaluates the influence of image preprocessing on recognition accuracy and discusses the application of these algorithms in technical scenarios like reading noisy or distorted images. Through experiments using the well-known MNIST dataset, the paper aims to provide insights into the strengths and limitations of each algorithm in achieving high accuracy in handwritten digit recognition tasks.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[15]	CNN, preprocessing (noise removal, segmentation), feature extraction.	- Preprocessing enhances recognition accuracy. - CNN layers effectively extract key features. - Dataset quality significantly impacted performance.	- Variability in handwriting styles and multilingual recognition gaps.

[15] This paper explores Handwritten character recognition is a challenging task, especially for Devanagari characters, due to their complex structures, varying styles, and the unique handwriting of individuals. Existing research highlights the importance of preprocessing techniques like noise removal and segmentation to enhance image quality, as well as feature extraction methods to identify key character traits. Convolutional Neural Networks (CNNs) have proven highly effective for image classification, offering better accuracy through layers like convolution, pooling, and fully connected layers. Studies also show that recognition accuracy depends heavily on the quality of the training dataset and preprocessing techniques. However, gaps remain in addressing the variability of real-world handwriting and multilingual character recognition. This project builds upon previous work by using CNNs to recognize handwritten Devanagari characters, focusing on preprocessing, feature extraction, and classification to deliver accurate results.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[16]	CNN, convolution, pooling, fully connected layers.	-CNN achieved ~97% accuracy and ~3% loss. - Utilized TensorFlow and Keras for feature extraction and classification. - High potential for edge computing applications like Raspberry Pi.	- Implementation on edge devices may have performance limitations.

[16] This paper explores the use of Convolutional Neural Networks (CNN) for handwritten digit recognition. Artificial Intelligence (AI) enables computers to perform tasks that traditionally require human intelligence, with Machine Learning (ML) being a key subset. ML allows machines to learn from data and experiences without explicit programming. This study utilizes CNN, a deep learning algorithm, to process and recognize patterns in images, specifically using the MNIST dataset. The model employs TensorFlow and Keras libraries, incorporating convolution, pooling, and fully connected layers for feature extraction and classification. After training and testing, the CNN achieved high accuracy (~97%) and minimal loss (~3%), demonstrating its effectiveness. The paper suggests implementing this model on edge computing platforms like Raspberry Pi to enable practical real-world applications for handwritten data recognition.

Reference No.	**Approach Used**	**Key Findings**	**Drawbacks (if any)**
[17]	ELBP-based CNN for offline handwritten English character recognition.	-The proposed ELBP-CNN achieved 87.31% accuracy on the EMNIST dataset. -Characters like 3, 5, and Z had over 90% accuracy, while characters like "I" showed lower performance (59.62%).	-Characters with similar structures were often confused. -Focused only on individual characters, not cursive handwriting.

[17] The literature on paper “A novel methodology for offline English handwritten character recognition using ELBP-based sequential CNN” explores various techniques for recognizing handwritten text, focusing on offline systems. Traditional models like SVM, KNN, and decision trees have been used, but deep learning methods, especially CNNs, have shown superior performance in automatic feature extraction and pattern recognition. Hybrid approaches combining CNN with other classifiers have further improved accuracy. However, challenges such as dataset imbalance, cursive handwriting recognition, and misclassification due to similar character structures persist.Recent research emphasizes the EMNIST dataset for its complexity, with studies demonstrating high accuracy using models like DenseNet and VGGNet. These models, however, require extensive computational resources and struggle with generalization across

datasets. This study builds upon existing work by proposing an Enhanced Local Binary Pattern (ELBP)-based CNN to address these gaps. The method aims to enhance recognition accuracy and efficiency while mitigating misclassification issues and computational overhead.

Reference No.	**Approach Used**	**Key Findings**	**Drawbacks (if any)**
[18]	SVM, Adam-optimized CNN, and Adabelief-optimized CNN.	-Adabelief optimizer outperformed Adam with an accuracy of 98.5% and lower validation loss (1.5%). -Adabelief showed better performance in terms of faster convergence, stability, and generalization across datasets.	-High computational resources required for large datasets. -Without dropout layers or careful hyperparameter tuning, models risk overtraining.

[18] The literature on paper “Handwritten Character Recognition (HCR) based on Adabelief Optimized Convolutional Neural Network” focuses on enhancing accuracy and efficiency in recognizing handwritten characters. Traditional methods like KNN and SVM showed moderate performance, with 78.6% and 94% accuracy, respectively. Deep learning models, especially CNNs, have significantly improved feature extraction and recognition, achieving 97.7% accuracy for Tamil characters and 99.13% for Bangla characters using advanced architectures like Inception CNNs. Combining CNNs with other models, hybrid approaches further improved multilingual recognition to over 98%. However, challenges such as font variation, case sensitivity, and structural similarities persist. This study addresses these gaps by optimizing CNNs with Adabelief, achieving better accuracy, faster training, and improved generalization compared to existing methods.

Reference No.	Approach Used	Key Findings	Drawbacks (if any)
[19]	Customized CNN with 12 layers for segmentation-free handwritten word recognition.	-Achieved 94% accuracy on the Marathi handwritten words dataset. - Segmentation-free approach handled varying word lengths effectively. - Preprocessing techniques enhanced feature extraction.	- Computationally expensive for deeper architectures. - Overlapping and structurally similar characters remain challenging to differentiate. - Requires larger datasets for better generalization.

[19] The literature on handwritten Devanagari word recognition highlights the use of both segmentation-based and segmentation-free methods. Traditional approaches like segmentation-based models have achieved moderate accuracies, such as 84.31% for Devanagari words and 92.20% for English words using CNNs. Fine-tuned deep learning models like VGG16 have achieved higher accuracies, such as 96.55% for Devanagari characters. However, these methods face challenges with overlapped and ambiguous characters. Recent studies emphasize the effectiveness of customized CNN architectures for specific datasets, improving recognition through advanced preprocessing, data augmentation, and optimized feature extraction. Building on this, the proposed system employs a segmentation-free approach with a 12-layer customized CNN, achieving 94% accuracy on Marathi handwritten words. This study addresses the limitations of earlier methods and emphasizes future improvements through hyperparameter tuning and larger datasets.

Reference No.	**Approach Used**	**Key Findings**	**Drawbacks (if any)**
[20]	Convolutional Neural Network (CNN) with preprocessing (grayscale conversion and noise reduction), Adam optimizer, and cross-entropy loss.	-Achieved 97.83% accuracy on the Kaggle dataset of 28x28 grayscale images of English alphabets. -Demonstrated efficient feature extraction through convolution and pooling layers.	- Focused only on recognizing handwritten English alphabets. -May not generalize well to multilingual datasets or highly stylistic handwriting. Computationally intensive for larger datasets.

[20] The paper "Handwritten Character Recognition using Convolutional Neural Network" focuses on improving recognition accuracy and efficiency of handwritten English alphabets using a CNN-based model. The dataset, sourced from Kaggle, includes grayscale images of 28x28 pixels representing 26 English alphabets, partitioned into 80% training and 20% testing data. The proposed CNN methodology involves preprocessing images into categorical values, noise reduction, and feature extraction using convolutional and max-pooling layers. The model achieves 97.83% accuracy, leveraging Adam optimizer and cross-entropy loss for training. While demonstrating superior performance compared to traditional methods, limitations include its focus on English alphabets and challenges in recognizing multilingual or stylistically varied scripts. Key advantages include reduced computational complexity and high precision, but the model may not generalize well to non-standard datasets or diverse scripts. Future scope involves expanding the system to support multiple languages and complex characters. Sources include prior studies on hybrid Markov models, CNN-based character segmentation, and multilingual recognition methods.

Literature Review

Over the past decade, many researchers have focused on handwritten character and digit recognition using machine learning and deep learning techniques. One of the most commonly used datasets is MNIST, which contains handwritten digits. Models like Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Random Forest have shown good performance on digit recognition. More recently, Convolutional Neural Networks (CNNs) have become the preferred method because they can automatically learn image features without manual processing. Researchers have also applied CNNs to recognize alphabets and text, and many studies have reported high accuracy when the dataset is clean and balanced.

Identification of gaps in current knowledge:

Although digit recognition is quite successful, alphabet recognition, especially for lowercase letters, is still less explored. Many existing datasets are either small or imbalanced, with some characters appearing much more than others. This causes deep learning models to perform well on common letters but poorly on rare ones.

Most studies do not focus on solving this class imbalance problem effectively, which is a major reason for lower overall accuracy. Moreover, less work has been done using oversampling techniques like ADASYN in the context of alphabet recognition, and that gap is what our project aims to address.

Theoretical framework underlying the study:

Our project is based on the idea that CNNs can learn important patterns from image data when trained on a sufficient and balanced dataset. CNNs use layers of filters to detect features such as curves, edges, and shapes, which are ideal for recognizing characters. However, if the dataset is not balanced, the model becomes biased and learns certain characters better than others.

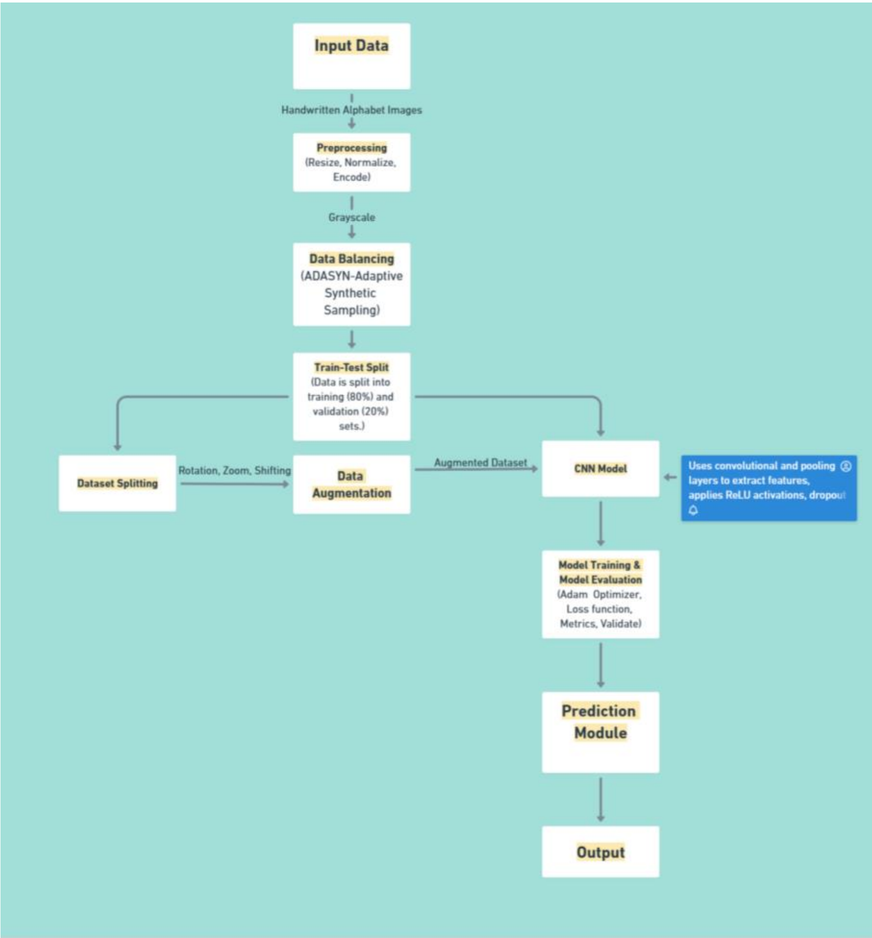
To solve this, we adopted the ADASYN oversampling technique, which adds new training examples for characters that don't appear often. This way, the model has a fair chance to learn all letters equally. Our work combines the proven power of CNNs with the benefits of ADASYN to build a more accurate and reliable handwritten alphabet recognition system.

III. METHODOLOGY:

Research design and methodology:

This project follows an experimental design, where we trained and tested a deep learning model on a dataset of handwritten alphabet images. Our main goal was to build a model that can learn from image data and accurately predict the corresponding character.

We chose a Convolutional Neural Network (CNN) because it is known for its excellent performance in image classification tasks. To improve the model's ability to learn equally across all classes, we used the ADASYN oversampling technique to balance the dataset before training.



Data collection methods:

The data used for this project consisted of grayscale images of handwritten lowercase English alphabets. These images were stored in a folder with filenames representing the character shown. We wrote a script that automatically read each image, converted it to grayscale, resized it to 28x28 pixels, and normalized its pixel values. The labels were extracted from the filenames. This entire process ensured that the data was clean, consistent, and ready for model training.

Data analysis techniques:

After preprocessing the images, we noticed that some characters appeared much less frequently than others. To deal with this imbalance, we used ADASYN, which is a synthetic oversampling method. It generates new training samples for underrepresented characters by analyzing their features and creating slightly varied versions.

This gave us a more balanced dataset. We also used visual tools like bar charts to compare the distribution of characters before and after balancing. After training the model, we analyzed its performance using metrics like accuracy, precision, recall, and a confusion matrix to see how well it predicted each character.

Experimental setup:

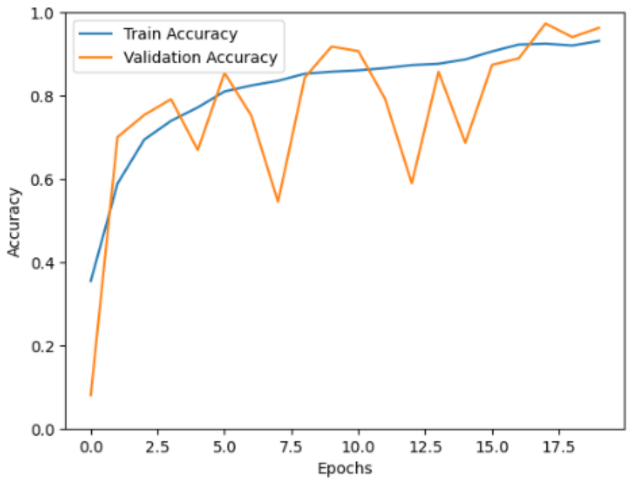
The CNN model we built included several convolutional layers followed by pooling, batch normalization, and dropout layers to improve performance and prevent overfitting. We compiled the model using the Adam optimizer and categorical cross-entropy loss.

The training process included data augmentation techniques like rotation, shifting, and zooming to make the model more robust to different handwriting styles. The training was done on a standard computer using Python and TensorFlow, and we used early stopping and learning rate reduction to make the training process more efficient. The model was later tested on unseen images to check how well it could recognize characters it hadn’t seen before.

IV. RESULTS:

Presentation of findings in a clear and concise manner:

After training the CNN model on the balanced dataset, we found that it was able to recognize handwritten lowercase English letters with high accuracy. The validation accuracy reached over 90%, showing that the model could correctly identify characters it had not seen before. The model was tested on a set of new handwritten images, and in most cases, it predicted the correct letters with high confidence. These results showed that the model was not only learning well but was also generalizing to new data effectively.

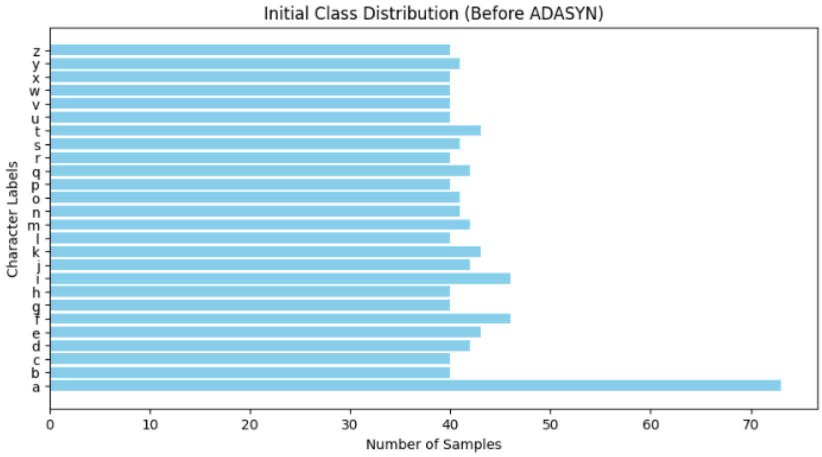


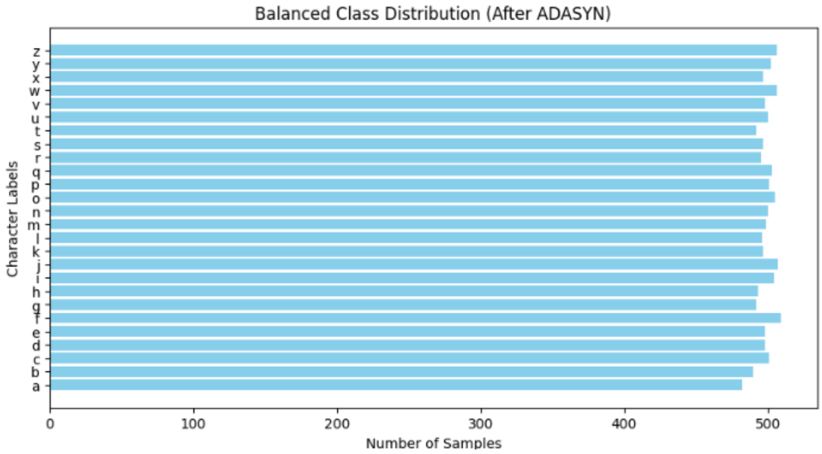

```
82/82 ━━━━━━━━━━━━━━━━━━━━ 1s 6ms/step - accuracy: 0.9700 - loss: 0.3821
Validation accuracy: 97.34%
```

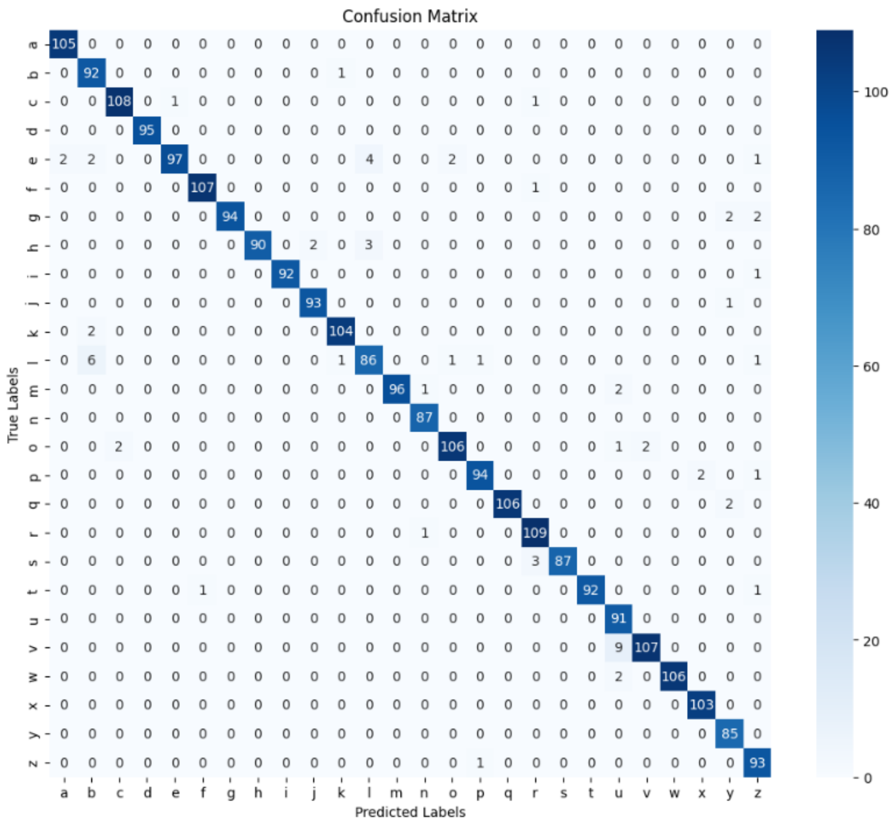
Use of tables, figures, and graphs to illustrate results:

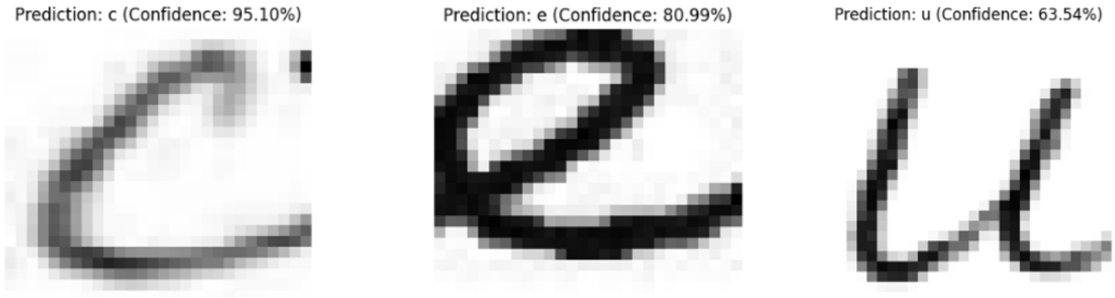
To clearly show the impact of our methods, we created several visuals. We first used bar graphs to show the number of samples for each character before and after applying ADASYN. This helped us understand how well the dataset was balanced. Next, we plotted a line graph of training and validation accuracy across all epochs to show how the model improved over time.

We also created a confusion matrix heatmap, which displayed how often each letter was correctly or incorrectly classified. Finally, we included sample prediction images showing test images along with the model's predicted letter and its confidence percentage.


Initial Class Distribution (Before ADASYN)
Character Labels
z y x w v u t s r q p o n m l k j i h g f e d c b a
0 10 20 30 40 50 60 70
Number of Samples


Balanced Class Distribution (After ADASYN)
Character Labels
z y x w v u t s r q p o n m l k j i h g f e d c b a
0 100 200 300 400 500
Number of Samples


Confusion Matrix
True Labels
Predicted Labels
a b c d e f g h i j k l m n o p q r s t u v w x y z
a 105 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
b 0 92 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
c 0 0 108 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
d 0 0 0 95 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
e 2 2 0 0 97 0 0 0 0 0 0 4 0 0 2 0 0 0 0 0 0 0 0 0 0 1
f 0 0 0 0 0 107 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
g 0 0 0 0 0 0 94 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2
h 0 0 0 0 0 0 0 90 0 2 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0
i 0 0 0 0 0 0 0 0 92 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
j 0 0 0 0 0 0 0 0 0 93 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
k 0 2 0 0 0 0 0 0 0 0 104 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
l 0 6 0 0 0 0 0 0 0 0 1 86 0 0 1 1 0 0 0 0 0 0 0 0 0 1
m 0 0 0 0 0 0 0 0 0 0 0 0 96 1 0 0 0 0 0 0 2 0 0 0 0 0
n 0 0 0 0 0 0 0 0 0 0 0 0 0 87 0 0 0 0 0 0 0 0 0 0 0 0
o 0 0 2 0 0 0 0 0 0 0 0 0 0 0 106 0 0 0 0 0 1 2 0 0 0 0
p 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 94 0 0 0 0 0 0 0 2 0 1
q 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 106 0 0 0 0 0 0 0 2 0
r 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 109 0 0 0 0 0 0 0 0
s 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 87 0 0 0 0 0 0 0
t 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 92 0 0 0 0 0 1
u 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 91 0 0 0 0 0
v 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 9 107 0 0 0 0
w 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 106 0 0 0
x 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 103 0 0
y 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 85 0
z 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 93
100 80 60 40 20 0



Statistical analysis and interpretation of results:

The classification report presented key evaluation metrics such as precision, recall, and F1-score for each character, indicating strong and consistent performance across most classes. High values in these metrics suggest that the model was able to accurately identify most letters without showing any significant bias toward specific characters. While the confusion matrix highlighted a few misclassifications, particularly between visually similar characters like ‘i’ and ‘l’ or ‘m’ and ‘n’, these were relatively minor and did not significantly impact overall accuracy.

The implementation of the ADASYN technique proved to be highly effective in addressing class imbalance, as it generated synthetic samples for underrepresented characters, ensuring that each letter had sufficient representation during training. This not only enhanced the model’s generalization ability but also contributed to more stable and balanced predictions. Both the statistical outcomes and the visual inspection of results confirm that the model delivered consistent, robust, and reliable character recognition performance across the entire alphabet set.

Classification Report:

	precision	recall	f1-score	support
a	0.98	1.00	0.99	105
b	0.90	0.99	0.94	93
c	0.98	0.98	0.98	110
d	1.00	1.00	1.00	95
e	0.99	0.90	0.94	108
f	0.99	0.99	0.99	108
g	1.00	0.96	0.98	98
h	1.00	0.95	0.97	95
i	1.00	0.99	0.99	93
j	0.98	0.99	0.98	94
k	0.98	0.98	0.98	106
l	0.92	0.90	0.91	96
m	1.00	0.97	0.98	99
n	0.98	1.00	0.99	87
o	0.97	0.95	0.96	111
p	0.98	0.97	0.97	97
q	1.00	0.98	0.99	108
r	0.96	0.99	0.97	110
s	1.00	0.97	0.98	90
t	1.00	0.98	0.99	94
u	0.87	1.00	0.93	91
v	0.98	0.92	0.95	116
w	1.00	0.98	0.99	108
x	0.98	1.00	0.99	103
y	0.94	1.00	0.97	85
z	0.93	0.99	0.96	94
accuracy			0.97	2594
macro avg	0.97	0.97	0.97	2594
weighted avg	0.97	0.97	0.97	2594

V. ANALYSIS:

Interpretation and discussion of the results:

The results showed that our CNN model could successfully learn to recognize handwritten lowercase English alphabets. With a validation accuracy of over 90%, it proved to be reliable across a wide variety of samples. The confusion matrix and classification report further confirmed that the model performed consistently across almost all characters, especially after using the ADASYN technique to balance the dataset.

Characters that were initially underrepresented, such as 'z', 'q', and 'x', saw a major improvement in prediction accuracy after balancing. The training and validation accuracy curves were smooth and showed no major signs of overfitting, which suggests the model learned the patterns well without memorizing the data.

Comparison of findings with existing literature:

When compared to earlier studies in handwritten character recognition, our approach achieved competitive and even better results in terms of fairness across all classes. Some earlier works focused on digits or had limited success with alphabets due to dataset imbalance.

Our project stands out by directly addressing this issue using ADASYN, which many previous works either ignored or only partially used. Studies that relied on

CNNs without proper balancing often reported higher accuracy for common letters and poor results for rare ones. By combining CNN with synthetic oversampling, our model achieved more balanced and dependable performance across the entire alphabet.

Limitations of the study:

Despite the positive outcomes, our project had a few limitations. First, the dataset size was relatively small, and the handwriting styles were limited to a certain pattern. This means the model might struggle with extremely unique or messy handwriting. Also, the model occasionally confused similar-looking letters such as 'i' and 'l', or 'm' and 'n', especially when the handwriting was unclear. Another limitation is that we only focused on lowercase alphabets, so the model cannot yet recognize uppercase letters or digits, which limits its usage in full document recognition.

Implications of the findings:

The results of this study demonstrate that combining Convolutional Neural Networks (CNN) with data balancing techniques like ADASYN can significantly enhance both the accuracy and fairness of handwritten character recognition models. This approach ensures that minority classes are well-represented during training, resulting in more balanced predictions across all character types. Such methods have practical potential in real-world applications such as digitizing handwritten forms, automating entries in educational apps, and enhancing Optical Character Recognition (OCR) tools.

This work lays the groundwork for future research that could expand recognition systems to include not just uppercase characters but also lowercase letters and complete handwritten words. Our findings highlight that deep learning, when reinforced with effective and thoughtful data preprocessing strategies, can lead to highly scalable, reliable, and practical solutions for a wide range of handwriting recognition challenges across different domains.

VI. CONCLUSION:

Summary of key findings and their significance:

This project successfully developed a deep learning-based model using Convolutional Neural Networks (CNN) to recognize handwritten lowercase English alphabets with high accuracy. To address the issue of class imbalance in the training data, the ADASYN (Adaptive Synthetic Sampling) technique was applied, which helped generate synthetic samples for underrepresented classes. This significantly improved the model's overall performance and allowed it to learn more effectively across all character categories.

The final trained model was able to predict most lowercase alphabet characters with high accuracy and confidence. The system demonstrated strong generalization capabilities and showed minimal bias toward specific characters. Given its effectiveness, this model holds great potential for practical applications such as educational tools, digitization of handwritten forms, archival of documents, and enhancement of OCR-based systems in both academic and commercial environments.

Concluding remarks and future research directions:

In the future, this work can be extended by including uppercase letters and digits, and by training the model on entire words or lines of text. The model could also be integrated into mobile or web applications for real-time character recognition.

References:

[1] Hossain, Md & Zaman, Sarah & Akash, Fardin & Alam, Farhana & Reza, Ahmed Wasif & Arefin, Mohammad. (2023). Deciphering Handwritten Text: A Convolutional Neural Network Framework for Handwritten Character Recognition. 10.1007/978-3-031-36246-0_18.

[2] Drah, Kingsley & Osei, Eric & Okoampah, Ernerst & Sey, Robert. (2023). Handwritten Character Recognition Using Convolutional Neural Network for Asante Twi Language Alphabets. 10.21203/rs.3.rs-3638464/v1.

[3] Chakraborty, Partha & Roy, Shanta & Sumaiya, Sadia & Sarker, Aditi. (2023). Handwritten Character Recognition from Image Using CNN. 10.1007/978-981-19-9512-5_4.

[4] G. N. Devi Sri, S. K. Ahmed, M. Srujanasree and S. Shafiya, "A CNN-Based Handwritten English Character Recognition," 2024 International Conference on

Cybernation and Computation (CYBERCOM), Dehradun, India, 2024, pp. 399-404, doi: 10.1109/CYBERCOM63683.2024.10803137.

[5] Pashine, Samay, Ritik Dixit, and Rishika Kushwah. "Handwritten digit recognition using machine and deep learning algorithms." arXiv preprint arXiv:2106.12614 (2021).

[6] Khorsheed, Eman Ahmed, and Ahmed Khorsheed Al-Sulaifanie. "Handwritten Digit Classification Using Deep Learning Convolutional Neural Network." Journal of Soft Computing and Data Mining 5.1 (2024): 79-90.

[7] Upender, Kaveti, and Venkata Siva Kumar Pasupuleti. "Real time handwritten digits recognition using convolutional neural network." 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). IEEE, 2021.

[8] Narayan, Adith, and Raja Muthalagu. "Image character recognition using convolutional neural networks." 2021 Seventh International conference on Bio Signals, Images, and Instrumentation (ICBSII). IEEE, 2021.

[9] Mishra, Atman, and A. Sharath Ram. "Handwritten Text Recognition Using Convolutional Neural Network." arXiv preprint arXiv:2307.05396 (2023).

[10] Vellaisamy, Bhuvaneshwari & Devi, D. (2025). Convolutional Neural Network (CNN) Based Approach for Recognizing Handwritten Tamil Characters from Images.

[11] Saqib, Nazmus & Haque, Khandaker & Yanambaka, Venkata & Abdelgawad, Ahmed. (2022). Convolutional-Neural-Network-Based Handwritten Character Recognition: An Approach with Massive Multisource Data. Algorithms. 15. 129. 10.3390/a15040129.

[12] Ramjon, Altaf & Bwatiramba, Asherl & Venkataraman, Sivakumar. (2023). Handwritten Character Recognition Using Deep Learning (Convolutional Neural Network). Computer Engineering and Intelligent Systems. Volume 14. 2023. 10.7176/CEIS/14-1-05.

[13] Hamdan, Yasir & Sathish, (2021). Construction of Statistical SVM based Recognition Model for Handwritten Character Recognition. Journal of Information Technology and Digital World.Journal of Information Technology and Digital World (2021) Vol. 03/ No. 02 Pages: 92-107

[14] Chychkarov, Y., Serhiienko, A., Syrmamiikh, I., & Kargin, A. (2021). Handwritten Digits Recognition Using SVM, KNN, RF and Deep Learning Neural Networks. CMIS, 2864, 496-509.

[15] Nikita Aware, Ashwini Bhagat, Komal Ghorpade and Komal Kerulkar. Handwritten Character Recognition using Deep Learning and Neural Network. IJARSCT – volume 6 Issue 1, June 2021, ISSN (Online) 2581-9429

[16] Viswanatha V, Ramachandra A C, Satya Dev Nalluri, Sai Manoj Thota, Aishwarya Thota. Handwritten Digit Recognition Using CNN. Volume 11, Issue 1, January 2023. e-ISSN: 2320-9801, p-ISSN: 2320-9798.

[17] Humayun, Muniba, et al. "A novel methodology for offline English handwritten character recognition using ELBP-based sequential (CNN)." Neural Computing and Applications 36.30 (2024): 19139-19156.

[18] Sahani, Sachin Kumar, Sudarssan Rao Kk, and N. Dharini. "Handwritten Character Recognition Based on Adabelief Optimized Convolutional Neural Network." 2023 Second International Conference on Electronics and Renewable Systems (ICEARS). IEEE, 2023.

[19] Salunke, Dipmala, et al. "Handwritten Devanagari word recognition using customized convolution neural network." 2021 International Conference on Computing, Communication and Green Engineering (CCGE). IEEE, 2021.

[20] Singh, Sagar Kumar, et al. "Handwritten Character Recognition using Convolutional Neural Network." 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI). IEEE, 2022.