Smart Handwritten Notes Recognition: AI-Powered Solutions for Learning Beyond

Vu Tuyet Anh Pham[1,2][0009−0005−0376−8714], Tan Duy Le[1,2][0000−0001−6597−0209], Nguyen Tan Viet Tuyen[3][0000−0001−8000−6485], and Kha Tu Huynh[1,2*][0000−0001−8262−4703]

[1] International University, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, 700000, Vietnam
[3] University of Southampton, UK
Email: ITDSIU21073@student.hcmiu.edu.vn, ldtan@hcmiu.edu.vn, tuyen.nguyen@soton.ac.uk, hktu@vnuhcm.edu.vn
Corresponding author: hktu@vnuhcm.edu.vn

Abstract. Every day, millions of handwritten notes often record valuable knowledge, critical insights, and critical information. Given the importance of handwritten notes, an effective method to digitize, store, and extract information from these notes would not only greatly enhance the ability to securely preserve and manage information but also unlock a variety of real-world applications. However, handwriting recognition still remains a challenging within Computer Vision (CV) and Artificial Intelligence (AI). Although various studies in Optical Character Recognition (OCR) have achieved remarkable success in recognizing handwritten text at the character level, they still struggle with overlapping characters, inconsistent spacing, and the wide variety of handwriting styles encountered in practice. To address these limitations, this research proposes a Handwritten Word Recognition (HWR) model with a two-stage pipeline, specifically optimized for handwritten English notes. The first stage focuses on accurately detecting the location of each word in an image. The second stage utilizes ResUNet-101 for feature extraction, followed by Bi-LSTM and CTC decoding to recognize entire words. Finally, a post-processing phase leverages Natural Language Processing (NLP) techniques to enhance accuracy further. The proposed method achieves an accuracy of 77 percent on testing data, demonstrating significant improvements over traditional OCR systems in handling handwritten word. This research not only advances handwritten note digitization but also lays the groundwork for automated information extraction systems, benefiting fields such as education, research, and document archiving.

Keywords: Handwritten word recognition · Optical Character Recognition · Image processing · Post - Processing.

1 Introduction

Even though information is increasingly stored and accessed in digital formats, handwritten notes retain significant importance in daily life. Students, researchers, and professionals alike continue to depend on handwritten notes to record ideas, summarize lectures, and document crucial insights. Handwritten notes have been widely recognized to enhance memory retention and comprehension compared to digital note-taking, making it a valuable tool in daily life. Nevertheless, the management's handwritten notes pose significant challenges, including difficulties in retrieval, risk of loss, and lack of organization. Consequently, there is an increasing demand for automated systems that can efficiently convert handwritten notes into structured, searchable text using Optical Character Recognition (OCR). Traditional OCR models, which rely on character segmentation and feature extraction, have achieved high accuracy in recognizing printed text and well-separated handwritten characters [1]. However, real-world handwritten text presents several challenges that hinder the effectiveness of existing OCR solutions. Firstly, handwritten words frequently feature connected or overlapping characters, making character segmentation particularly difficult. Secondly, inconsistent spacing between characters confuses traditional recognition models as they struggle to distinguish between individual letters. Thirdly, variations in stroke width, slant, and writing styles introduce additional complexity, resulting in reduced recognition accuracy [2]. These limitations highlight the necessity for a word-level OCR approach rather than conventional character-level methods, particularly for handwritten notes, where character boundaries are often ambiguous. To address these challenges, this research proposes a two-stage handwritten word recognition framework specifically optimized for digitizing handwritten notes. The principal contributions of this research are articulated as follows:

- **Innovative Two-Stage Recognition Pipeline**: A robust pipeline is proposed, combining YOLOv8 for precise word detection, a ResUNet-101 backbone for feature extraction, Bi-LSTM for sequence modeling, and CTC decoding for word recognition, followed by an NLP-based post-processing phase to enhance accuracy.
- **Optimization for Handwritten Notes**: To be suitable for objective research, the proposed model is trained exclusively on handwritten notes, ensuring superior performance in real-world note-taking applications.
- **Foundation for Digital Transformation**: This research not only addresses the inherent challenges of digitizing handwritten notes, but also paves the way for transformative applications in various domains, including education, research, medical documentation, and archival management.

The remainder of this paper is structured as follows. Section 2 offers a comprehensive review of the existing literature on Optical Character Recognition (OCR), which is the inspiration for the proposed technique. Section 3 provides an in-depth exposition of the proposed model, elucidating its architecture and operational principles. Section 4 presents the experimental results and performance evaluations. Finally, Section 5 reaches the conclusion of the study and discusses future research directions.

2 Literature Review

This section gives an overview of the key developments and identifies existing limitations in recent studies on OCR from 2019 to 2025, including handwritten English text recognition. The research of OCR can be divided into two types: traditional and modern approaches, which are illustrated in *Figure 1*.

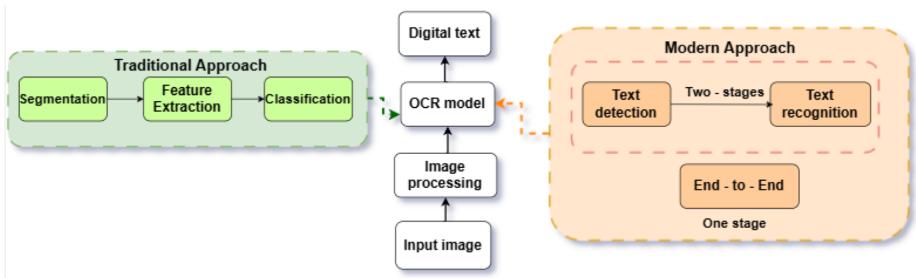

Fig. 1: Comparative Approaches of Traditional and Modern in the OCR.

2.1 Traditional OCR Approach

The traditional approaches simply rely on classical algorithms, generally involving three fundamental steps: segmentation, feature extraction, and classification as *Figure 2*.

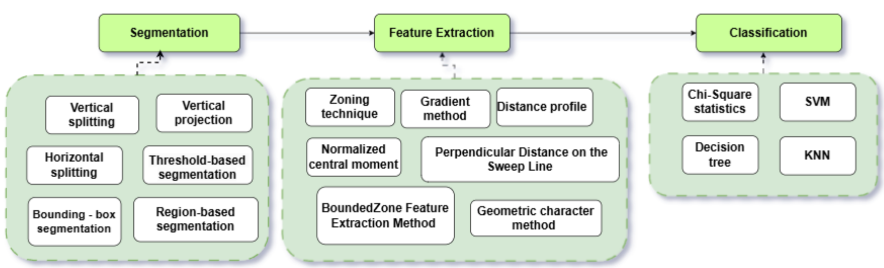

Fig. 2: Detailed Overview of the Traditional Approach.

Segmentation is the process of cutting an image into lines, words, and characters, depending on the specific objective of the research. One common method is vertical projection [3], which separates text by analyzing the width of pixels and determining the best places to cut between sections. Beside that, one approach suggests combining calculations of black pixel distribution with piecewise functions to improve the accuracy and apply piecewise functions. On the other hand, different segmentation techniques may be considered, such as vertical and horizontal splitting, bounding box segmentation, threshold-based segmentation, and region-based segmentation [4]. Additionally, a multi-tiered segmentation approach is proposed as a hierarchical text segmentation method that cuts text images into specific levels. [5].

Feature Extraction provide important and useful details of the image. For instance, distance profile (DP) and normalized central moment (NCM) features are introduced, which capture the shape and structural properties of characters [6]. Additionally, the geometric character method and the gradient method also have been deployed, which focus on shape, structure, and edge information of the text [7]. According to the paper [8] combined statistical and structural features to develop the BoundedZone Feature Extraction Method (BoZFEx). Furthermore, the distribution of black pixels provides valuable information for character recognition, which is applied in Zoning techniques. Another creative method, called Perpendicular Distance on the Sweep Line (PDSL), was proposed by [9]. This method identifies the center point (centroid) of an image, draws perpendicular lines intersecting two straight lines, and records the intersection points. Horizontal lines are then used to locate extreme points, measuring distances from the curve to enhance feature extraction accuracy.

Classification Using the key features identified, the classification algorithms will assign labels for the image. For example, the results of PDSL methods are analyzed and categorized using Chi-Square statistics based on sample proportions [9]. Additionally, some classification algorithms can apply as well as KNN, a decision tree, and SVM. Although the traditional way is simple and fast, it struggles to handle complex backgrounds or unstructured text, which led to the birth of the modern OCR approach.

2.2 Modern OCR Approach

Unlike traditional approaches, modern approaches integrate deep learning, playing a role in solving sequence data better. Researchers choose transformers as an end-to-end approach, which handles all the tasks of OCR at once. In addition, the two-stage method splits the process into detection and recognition steps and selects the most suitable model for each stage, which helps to optimize each stage to make better predictions than one stage. Therefore, this section reviews most of the two-stage methods as the *Figure 3*.

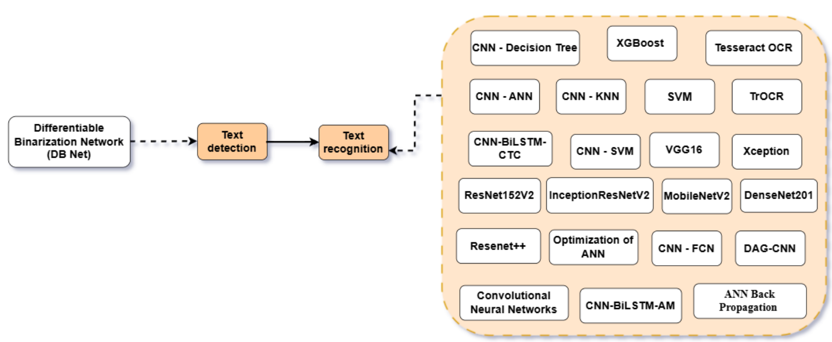

Fig. 3: Detailed Overview of the Modern Approach.

Detection stage: One widespread use of deep learning is approaching a Differentiable Binarization Network (DB Net), which was proposed in [10]. This model network uses pixel segmentation to preserve the integrity of text areas, prevent the text from blending, and reduce text adhesion by scaling the text field.

Recognition stage: includes the integration of feature extraction and classification, therefore, these approaches offer greater robustness and adaptability to diverse text and image types. In Python libraries, Tesseract OCR starts with layout analysis followed by adaptive classifiers [3]. Meanwhile, TrOCR is a pretrained model that excels in context-aware recognition due to its sentence-level depth, though it sacrifices some efficiency. Throughout, Tesseract's superiority over TrOCR due to its faster processing and lower resource demands [11].
Beyond these libraries, XGBoost can also be used for text recognition, as demonstrated in reference [12] This approach involves implementing feature extraction using spatial features and Gabor filters at multiple angles to capture pixel densities and orientation information. The extracted features are then fed into an XGBoost-based classification model.
Starting with neural networks, researchers are advancing OCR through Convolutional Neural Networks (CNNs), which consist of key layers: the Convolutional Layer, a foundational component, is responsible for extracting low-level features such as edges and textures from the input data, as mentioned in reference [13]. This layer is often paired with a ReLU activation function [14] to introduce non-linearity for complex pattern recognition. Batch Normalization is employed to ensure stability during training and mitigate the risk of overfitting, as highlighted in reference [2]. Additionally, Pooling Layers[13, 14] downsample feature maps to reduce computation and focus on prominent features. Dropout layers disabling 0.25 of neurons [2], helping to prevent overfitting. After that, a Flatten operation prepares data for the Fully Connected (Dense) Layer, where ReLU and SoftMax functions refine output classification. To optimize the training process, tools like Adam [14, 15] and RMSprop [2, 16] are utilized to enhance efficiency and performance. Furthermore, robustness is improved through data augmentation techniques, implemented via the 'flow()' method, as described in reference [15].
Researchers have taken the basic Convolutional Neural Network (CNN) architecture and improved it by adding advanced designs and optimizations. As noted in reference [17], several hybrid deep learning models have been proposed to achieve these improvements. These models leverage CNN to extract essential features from data and then train various classification algorithms such as KNN, SVM, and Decision Trees.As cited in reference [18], C. E. Mook and colleagues compared several advanced CNNs, including VGG16, Xception, ResNet152V2, InceptionResNetV2, MobileNetV2, and DenseNet201. The experiment revealed that InceptionResNetV2 has the best accuracy in recognizing both printed and handwritten text. However, these advanced CNNs face challenges in cursive text, which is addressed by Directed Acyclic Graph - Convolutional Neural Network

(DAG – CNN) [19]. The DAG algorithm has a deeper understanding of data patterns when connecting multiple paths, therefore, it can produce overfitting. To address this, a nonlinear flow is introduced, which helps mitigate the risk of vanishing gradients.
For recognizing text sequences, hybrid models like CNN-LSTM-CTC [1] extract features via CNNs, process sequences with LSTMs, and use CTC loss to handle duplicates, though limited to uppercase and numbers. The NWIMR model addresses this problem when adding bidirectional context with stacked BiLSTM layers and a Lambda layer for dimensionality reduction, enhancing temporal dependency capture [20]. In another demonstration, an attention mechanism is applied to the above hybrid model, which helps to focus on relevant parts of the sequence [5]. This approach accurately the output of CTC.
To tackle the problem of text adhesion, the integration of Convolutional Neural Networks (CNNs) and Fully Convolutional Networks (FCNs) [10] is proposed. In this approach, CNNs are used first to identify and extract features and then pass them to FCNs. This method helps to reduce text cohesion through differentiable binarization while remaining consistent and reliable.
Researchers have created some tools to enhance how computers determine and process text through OCR, which can be categorized into both characters and numbers. In one study, [21], a method is proposed that uses CNN to recognize handwriting and SVM to handle digit classification. The final predicted text is displayed above the input text based on the comparison that yields the highest probability. Meanwhile, it uses Support Vector Machines (SVM) for digit classification, which shine when organizing data into distinct categories. Beyond digit classification, SVMs are also applied to font style recognition [6]. This process uses features like Distance Profile Features (DP) and Normalized Central Moments (NCM)—fancy terms for measuring shapes and spacing in characters, extracting and normalizing them before classifying them into font types. As a combination of CNN, another study [8] takes a slightly different tack, pairing CNN for handwriting recognition with an Artificial Neural Network (ANN) to tackle digit recognition.
Referring to ANN optimization, [22] implemented a comprehensive study of three different techniques to optimize ANN with: Generic Algorithms(GA), Particle swarm optimization (PSO), GD, Adam. This research shows the effectiveness of the combination of PSO - ANN. Meanwhile, [23] implements an ANN Back Propagation Algorithm, which includes backpropagation to compute the gradient of the cost function, meaning it helps the network learn and predict more reliably.
A design of single-headed network that introduces residual learning, U-net architecture, and Atrous Spatial Pyramid Pooling (ASPP) [24]. This design enhances feature extraction by emphasizing relevant key details and gathering information across multiple scales. As a result, it is particularly well-suited for tasks that require clear segmentation without the added complexity of breaking down characters into smaller components (graphemes).

3 Methodology

The proposed model aims to identify each word in handwritten English notes. This model is shown in *Figure 4*. The two stages will be trained independently to find the best model at each stage before being merged into the main pipeline (green line). For the goal of word recognition, the sequence model is suitable to develop and exploit based on the underlying structure of CNN - BiLSTM - CTC.

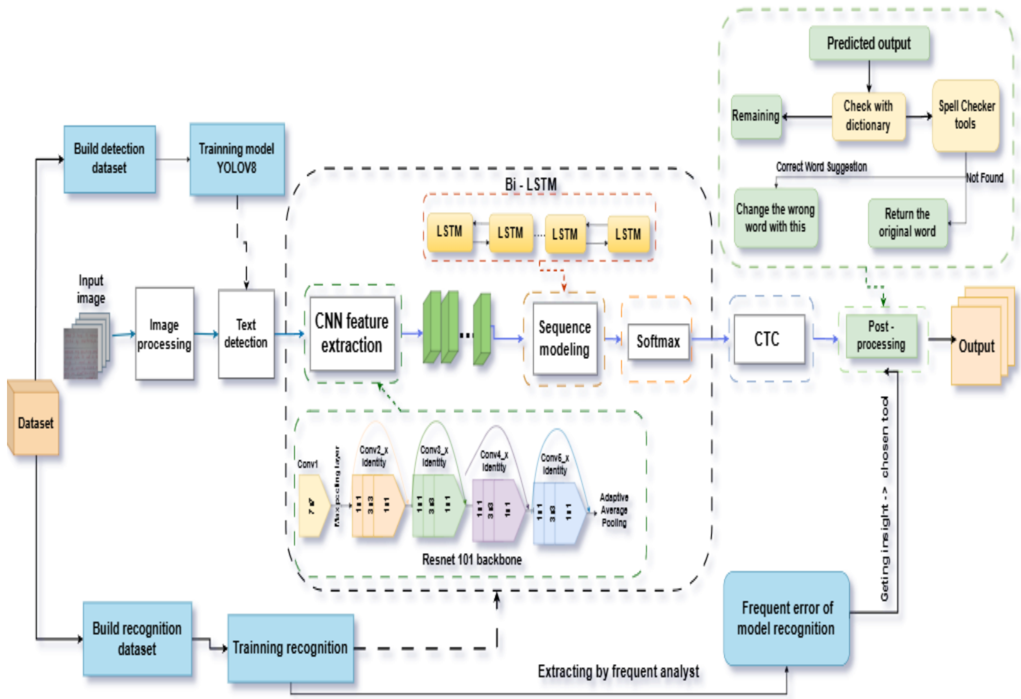

Fig. 4: Overview of proposed technique.

3.1 Data Collection and Organization

The dataset used in this research was precisely collected from the GoodNotes Handwriting Kollection (GNHK), a dataset for English handwriting in the wild [25]. This dataset is specifically designed for text localization and recognition, making it well-suited for our research objectives. It provides a diverse collection of handwritten notes, ensuring practical utility and real-world relevance for developing robust recognition models. To optimize the dataset's relevance and quality, not all images from GNHK were included. Instead, we carefully selected and manually annotated a subset of images that best represent the challenges of handwritten notes, including variations in handwriting styles, spacing inconsistencies, and character overlaps. This manual labeling process establishes a high-quality ground truth, reducing unnecessary noise and refining the dataset to emphasize the most critical elements for accurate recognition. By employing this hands-on approach, we enhance both the quality of the data and the performance of the model, ensuring that the annotations align closely with the specific challenges of handwritten note recognition. The resulting dataset provides a highly reliable benchmark for evaluating word-level recognition models, bridging the gap between theoretical advancements and real-world applications.

3.2 Image processing

After data collection, an essential image processing step is undertaken to ensure consistency, reduce unwanted noise, and optimize model performance. A significant finding derived from the manual curation of the dataset is that handwritten notes typically appear on white paper or lined backgrounds, meaning they are largely unaffected by complex backgrounds. This characteristic simplifies preprocessing since background removal becomes unnecessary. Instead, standard preprocessing steps such as resizing, grayscale conversion, and normalization are applied. These steps enhance model stability, accelerate convergence, and improve computational efficiency. Nevertheless, real-world handwritten notes exhibit natural variations in lighting, alignment, and writing style. To ensure the model generalizes well, additional data augmentation techniques are employed in the training dataset. These include: **Adjustments** to brightness, contrast, and saturation to simulate different lighting conditions. **Affine transformations** (rotation, shear) to account for variations in writing angles. **Random perspective distortion and rotation**, enabling the model to recognize text even when the handwriting deviates from perfect alignment. By incorporating these augmentations, the dataset more accurately reflects real-world scenarios, enhancing the model flexibility to handle the distortions commonly found in handwritten documents. With the data fully prepared, it is poised for use in the next phases of the research. This dataset will be utilized to train two distinct models in two separate stages. Subsequently, the most effective model will be determined, and the final pipeline will be established.

3.3 Detection stage

The first stage of this research focuses on achieving precise word detection within handwritten notes. This research employs YOLOv8, a state-of-the-art object detection model renowned for its exceptional speed and accuracy. While DBNet has been widely used for text detection due to its ability to segment text with differentiable binarization, it is primarily optimized for printed text and scene text, where character boundaries are well-defined. In contrast, handwritten notes frequently exhibit overlapping characters and indistinct separations, posing challenges to DBNet's ability to accurately localize individual words. In contrast, YOLOv8 offers a more adaptable and resilient approach to text detection. Due to its ability to handle overlapping handwriting, inconsistent spacing, and varying stroke widths. The model is fine-tuned on a custom dataset of handwritten notes, ensuring it learns to detect words in real-world note-taking scenarios. This adaptation enhances its ability to localize handwritten text, even in complex cases involving unconventional writing styles.

3.4 Recognition stage

For building a strong and flexible model for recognizing handwritten words to achieve robust and scalable handwritten word recognition, a proposed advanced deep learning-based OCR architecture integrates feature extraction, sequence modeling, and alignment-free prediction.

– **Feature extraction**: Differing from the studies reviewed, this research utilizes ResNet-101 as the base network for extracting features from input images. ResNet-101 has a deep architecture with 101 layers, shown in the figure below (*Figure 5*). Furthermore, ResNet-101 captures a wide range of features from low-level details, such as edges and textures, to high-level word-specific features, which ensures robustness against handwriting variability, distortions, or noise. Additionally, ResNet-101 incorporates residual connections, which help mitigate the vanishing gradient problem. This architecture is particularly effective in learning complex image patterns, making it well-suited for tasks involving handwritten text. By using the word-level approach, the proposed OCR leverages global word structures rather than segmenting individual characters, making recognition more context-aware and efficient in sequence tasks. This backbone produces a feature map, which is then transformed into a sequence of feature vectors by flattening along the height dimension. This sequence is fed into a Bi-LSTM to capture contextual dependencies, enabling accurate word recognition.

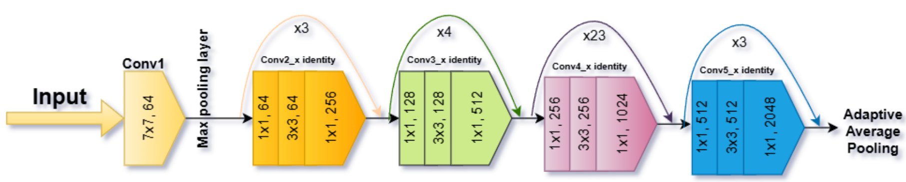

Fig. 5: Resnet 101 architecture.

– **Sequence Modeling**: Handwriting recognition is inherently sequential, requiring an understanding of both past and future character contexts. To address this requirement, the Bidirectional Long Short-Term Memory (Bi-LSTM) network is employed, as it is well-suited for processing sequential data. By analyzing both preceding and succeeding feature dependencies, Bi-LSTM enhances the recognition of ambiguous characters and improves the overall sequence prediction (*Figure 6*). In this research, the sequential feature maps extracted from the ResNet-101 backbone are fed into two Bi-LSTM layers, each comprising 256 hidden units. This configuration enables the model to capture contextual dependencies effectively. This bidirectional processing is particularly advantageous for handwritten notes, where inconsistent spacing, overlapping characters, and diverse writing styles can hinder traditional OCR approaches. The output from the Bi-LSTM layers refines and stabilizes feature representations across the sequence, producing a meaningful feature encoding that is then passed to the next stage for character probability prediction. This structured representation generated serves as the input for Connectionist Temporal Classification (CTC) decoding, facilitating the transformation from feature sequences to readable text.

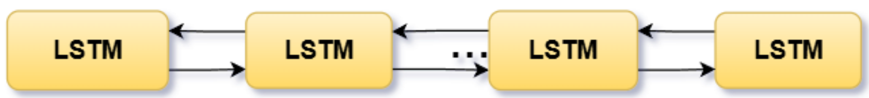

Fig. 6: Bidirectional LSTM Process.

- **Connectionist Temporal Classification (CTC) Loss**: The proposed system employs Connectionist Temporal Classification (CTC) decoding, an alignment - free methodology specifically designed to handle variable-length sequences without requiring explicit character segmentation. This technique allows for variable - length word predictions, making the model robust to different handwriting styles with minimal preprocessing requirements. Furthermore, this approach is particularly well-suited for handwritten word recognition, where inconsistent spacing and overlapping characters make traditional segmentation-based methods ineffective. CTC dynamically aligns the sequence of feature vectors produced by the Bi-LSTM with potential word outputs by learning the best mapping between input frames and character sequences. By leveraging a specialized loss function, CTC enables the system to predict complete words while handling noisy or distorted handwriting effectively.

- **Post-processing**: The output from the recognition stage from both the training and validation stages is analyzed to identify common recognition errors and evaluate the model's weaknesses. A key observation was that most errors involved a single-character misrecognition per character, which significantly impacted overall accuracy. To address this common error, a post-processing step leveraging SpellChecker, an NLP-based tool, was introduced to enhance recognition accuracy. This tool cross-references detected words against a predefined dictionary and suggests corrections based on character similarity. If no suitable match is found, the original word is retained to preserve the writer's intent, particularly in cases of abbreviations or domain-specific terminology.

4 Experimental Result

4.1 Performance of Detector stage

The performance of the YOLOv8 object detection model is evaluated based on key aspects. As the model trains over multiple epochs, its parameters are continuously optimized, and the best-performing version is selected for final evaluation. The performance of chosen YOLOv8 is presented as (*Figure 7*).

```
YOLOv8s summary (fused): 168 layers, 11,125,971 parameters, 0 gradients, 28.4 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95)
                   all          1         14          1      0.996      0.995      0.748
Speed: 3.0ms preprocess, 293.4ms inference, 0.0ms loss, 1.0ms postprocess per image
```

Fig. 7: YOLOv8 Performance Metrics

The YOLOv8 model demonstrates strong detection capabilities with high precision(1) and recall (0.996), maintaining a balance between accuracy and completeness. Next, the primary focus is on mAP@50-95, which achieves 0.748, indicating that the model performs well across different Intersection over Union (IoU) thresholds.

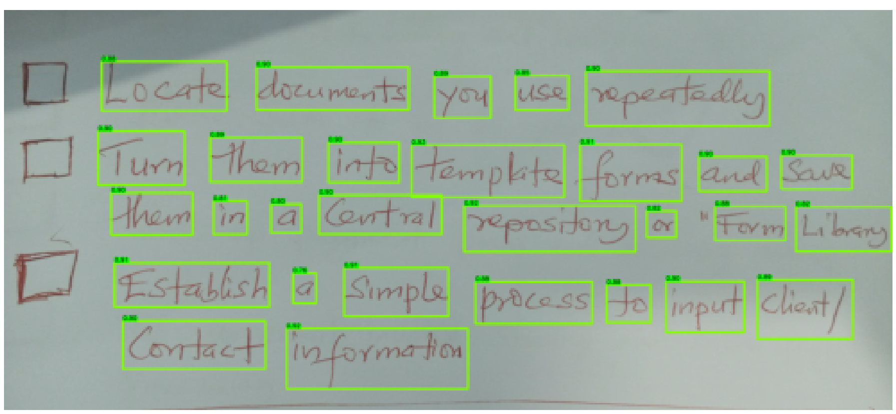

Fig. 8: Output of the Detection Stage

4.2 Performance of Recognition stage and Post-processing

The proposed recognition model is trained by a specified number of epochs. In the end, the chosen model should not overfit the training data.

a) **Impact on post-processing**: The testing data will be used to access the ability of the model in practice and evaluate the effects of post-processing. The three main metrics that typically evaluate the recognition output are Word Error Rate (WER), Word Accuracy Rate (WAR), and Character Accuracy (WA).

Table 1: Evaluating OCR Model Performance: Impact of Post-Processing.

Key metrics	Without Post - processing	With Post - processing
Accuracy	0.65	0.77
WER	0.35	0.23
WAR	0.09	0.08

According to *Table 1*, the performance of the OCR model has significantly improved with the help of post-processing. Although this is a simple post-processing step, the ouput's accuracy on the testing dataset increased from 65 percent to 77 percent. Additionally, the sample output of propose model will be presented at (*Figure 9*).

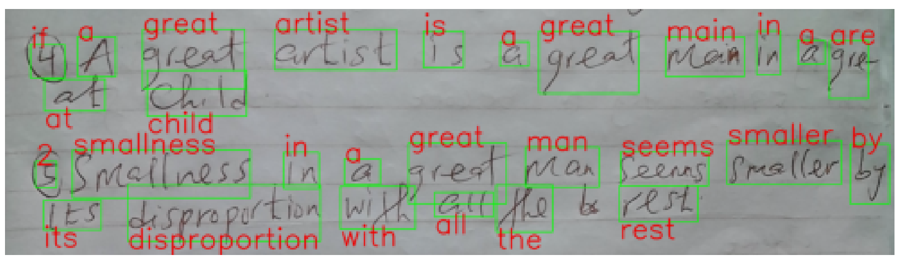

Fig. 9: The output of the Recognition stage.

b) **Evaluating Proposed Model Against Other Approaches**: In this section, three representative images, each exemplifying challenges such as overlapping characters, inconsistent spacing, and diverse writing styles, will be recognized *(Figure 10)*. The proposed model will be employed to recognize the text within these images, and its performance will be compared against three established optical character recognition (OCR) models: EasyOCR, Tesseract, and TrOCR.

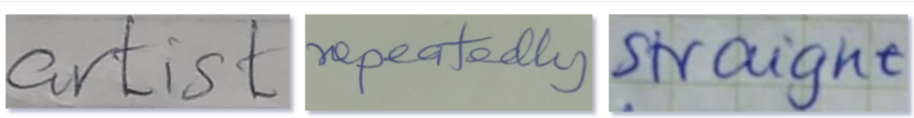

Fig. 10: Three samples of Handwritten Note Challenges

Models such as TrOCR, EasyOCR, and Tesseract are primarily designed for character-level prediction. Therefore, they often identify individual characters and link them into words. However, their performance is may be constrained when dealing with the inherent complexities of handwritten notes, which is present in the *Table 2*. As a result, the proposed system to specifically recognize handwritten notes has been developed.

Table 2: Comparison Results of Evaluated Models

Order	Proposed model	EasyOCR	Tesseract OCR	TrOCR
The 1st image	artist	Gr List	None	curtist
The 2nd image	repeatedly	peafadly	None	repeatably
The 3rd image	straight	Sf Qugnt	StY Qughe	surculight

5 Conclusion

The proposed handwritten word recognition system, integrating YOLOv8 for text detection, a deep learning pipeline with ResNet-101, Bi-LSTM, and CTC decoding, and a simple streamlined NLP-based post-processing step, demonstrates robust performance in recognizing handwritten English notes. This system offers substantial potential across multiple applications, such as automating exam grading by identifying handwritten responses, detecting plagiarism, and aiding in scoring to reduce teachers' administrative burdens. Additionally, it lays the groundwork for intelligent study tools capable of summarizing notes, highlighting key topics, and enhancing student comprehension, thereby enhance the learning performance.

Despite its strengths, the system encounters limitations in recognizing mathematical expressions and special symbols. Moreover, the reliance on a dictionary-based post-processing approach poses challenges in handling abbreviations effectively. Future research will aim to address these shortcomings by enhancing the model's ability to interpret complex handwritten elements, including mathematical notation and non-standard characters. Research in post processing is also needed. Expanding these capabilities will unlock a broader range of real-world applications, further improving the system's practical utility and adaptability.

References

1. M. Geetha, R. C. Suganthe, S. K. Nivetha, S. Hariprasath, S. Gowtham, and C. S. Deepak, "A Hybrid Deep Learning Based Character Identification Model Using CNN, LSTM, And CTC To Recognize Handwritten English Characters And Numerals," in *Proc. Int. Conf. Comput. Commun. Informatics (ICCCI)*, Coimbatore, India, 2022, pp. 1–6. https://doi.org/10.1109/ICCCI54379.2022.9740746
2. A. A. Mokhtar *et al.*, "An Intelligent Handwritten Digits and Characters Recognition System," in *Proc. Int. Conf. Innov. Intell. Informatics, Comput., Technol. (3ICT)*, Sakheer, Bahrain, 2022, pp. 670–673. https://doi.org/10.1109/3ICT56508.2022.9990669
3. X. Li, Z. Zhan, C. Li, C. Guo, Z. Li, and J. Li, "Research on English Character Recognition Technology Based on OCR," in *Proc. IEEE 6th Adv. Inf. Manag., Commun., Electron., Autom. Control Conf. (IMCEC)*, Chongqing, China, 2024, pp. 1210–1213. https://doi.org/10.1109/IMCEC59810.2024.10575683
4. C. Varshini, S. Yogeshwaran, and V. Mekala, "Tamil and English Handwritten Character Segmentation and Recognition Using Deep Learning," in *Proc. Int. Conf. Commun., Comput. Internet Things (IC3IoT)*, Chennai, India, 2024, pp. 1–5. https://doi.org/10.1109/IC3IoT60841.2024.10550221.

5. P. K. M, R. Logesh Babu, G. Sunil, J. Sampathkumar, and S. Chakraborty: Attention-Based Deep Learning Algorithm in Natural Language Processing for Optical Character Recognition. In: Proceedings of the 2023 International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT), Bengaluru, India, 2023, pp. 1–5. IEEE (2023). https://doi.org/10.1109/EASCT59475.2023.10393635
6. A. J. Mohammed, H. S. M. Al-Khaffaf: Font Recognition of English Letters Using Normalized Central Moments Features. In: Proceedings of the 2020 International Conference on Computer Science and Software Engineering (CSASE), Duhok, Iraq, 2020, pp. 249–254. IEEE (2020). https://doi.org/10.1109/CSASE48920.2020.9142078
7. N. Murugan, R. Sivakumar, G. Yukesh, J. Vishnupriyan: Recognition of Character from Handwritten. In: Proceedings of the 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2020, pp. 1417–1419. IEEE (2020). https://doi.org/10.1109/ICACCS48705.2020.9074424
8. N. N. Adlin Nik Qausbee, N. I. Raihana Ruhaiyem: BoZFEx – A New Feature Extraction Based on Bounded-Zone Method for Handwriting Character Recognition. In: Proceedings of the 2023 IEEE International Conference on Computing (ICOCO), Langkawi, Malaysia, 2023, pp. 503–507. IEEE (2023). https://doi.org/10.1109/ICOCO59262.2023.10397617
9. N. R. Soora, K. Dorthi, S. V. Mankala: Character Recognition using Perpendicular Distance on Sweep Line and Chi-Square Statistic as Classifier. In: Proceedings of the 2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, 2021, pp. 1–4. IEEE (2021). https://doi.org/10.1109/ICSES52305.2021.9633816
10. W. Jing: Automatic English Character Entry System based on Novel Optical Character Recognition Algorithm. In: Proceedings of the 2024 3rd International Conference on Sentiment Analysis and Deep Learning (ICSADL), Bhimdatta, Nepal, 2024, pp. 393–398. IEEE (2024). https://doi.org/10.1109/ICSADL61749.2024.00070
11. S. S. Mukherjee, A. Chandra, K. Chemburkar, V. Kulkarni: Project Lingua Franca: Democratizing Information through Unified Optical Character Recognition and Neural Machine Translation. In: Proceedings of the 2023 International Conference on Modeling, Simulation Intelligent Computing (MoSICom), Dubai, United Arab Emirates, 2023, pp. 486–491. IEEE (2023). https://doi.org/10.1109/MoSICom59118.2023.10458775
12. J. J. B. Jayachandran, K. Kirubasankar, K. Varsini: Enhancing Handwritten Character Recognition with XGBoost: A Machine Learning Approach. In: Proceedings of the 2023 International Conference on Data Science, Agents Artificial Intelligence (ICDSAAI), Chennai, India, 2023, pp. 1–5. IEEE (2023). https://doi.org/10.1109/ICDSAAI59313.2023.10452531
13. S. Katoch, M. Rakhra, D. Singh: Recognition of Handwritten English Character Using Convolutional Neural Network. In: Proceedings of the 2022 4th International Conference on Artificial Intelligence and Speech Technology (AIST), Delhi, India, 2022, pp. 1–6. IEEE (2022). https://doi.org/10.1109/AIST55798.2022.10064860
14. B. V. V, I. V, V. R. M, N. N, N. V. S. Reddy: Advancing Optical Character Recognition for Handwritten Text: Enhancing Efficiency and Streamlining Document Management. In: Proceedings of the 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1–8. IEEE (2023). https://doi.org/10.1109/ICCCNT56998.2023.10307143

15. P. R. Dash, R. C. Balabantaray, R. Dey: An Approach for Handwritten Alphanumeric Character Recognition: Leveraging CNN for Accurate Recognition. In: Proceedings of the 2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU), Bhubaneswar, India, 2024, pp. 1–6. IEEE (2024). https://doi.org/10.1109/IC-CGU58078.2024.10530699
16. B. K. Pattanayak, A. K. Biswal, S. R. Laha, S. Pattnaik, B. B. Dash, S. S. Patra: A Novel Technique for Handwritten Text Recognition Using Easy OCR. In: Proceedings of the 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), Erode, India, 2023, pp. 1115–1119. IEEE (2023). https://doi.org/10.1109/ICSSAS57918.2023.10331704
17. A. Akbar, M. S. Islam: Hybrid Deep Learning Model for Handwritten Bangla and English Digit Recognition. In: Proceedings of the 2023 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), Dhaka, Bangladesh, 2023, pp. 129–133. IEEE (2023). https://doi.org/10.1109/ICICT4SD59951.2023.10303498
18. C. E. Mook, C. P. Lee, K. M. Lim, J. Y. Lim: Handwritten Character and Digit Recognition with Deep Convolutional Neural Networks: A Comparative Study. In: Proceedings of the 2023 11th International Conference on Information and Communication Technology (ICoICT), Melaka, Malaysia, 2023, pp. 137–141. IEEE (2023). https://doi.org/10.1109/ICoICT58202.2023.10262721
19. P. V. Bhagyasree, A. James, C. Saravanan: A Proposed Framework for Recognition of Handwritten Cursive English Characters using DAG-CNN. In: Proceedings of the 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT), Chennai, India, 2019, pp. 1–4. IEEE (2019). https://doi.org/10.1109/ICIICT1.2019.8741412
20. S. Saikia, R. C. Saritha, J. CH: Deep Neural Networks for Intelligent Word Recognition. In: Proceedings of the 2023 IEEE 5th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA), Hamburg, Germany, 2023, pp. 557–561. IEEE (2023). https://doi.org/10.1109/ICCCMLA58983.2023.10346924
21. P. Latchoumy, G. Kavitha, S. Anupriya, H. S. Banu: Handwriting Recognition using Convolutional Neural Network and Support Vector Machine Algorithms. In: Proceedings of the 2022 6th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2022, pp. 1266–1271. IEEE (2022). https://doi.org/10.1109/ICECA55336.2022.10009150
22. A. Rath, P. Sarangi, A. Rath, and G. Panda, "Development and Performance Assessment of Bio-inspired based ANN Model for Handwritten English Numeral Recognition," in *Proc. IEEE 2nd Int. Conf. Appl. Electromagn., Signal Process., Commun. (AESPC)*, Bhubaneswar, India, 2021, pp. 1–5. https://doi.org/: 10.1109/AESPC52704.2021.9708530.
23. S. Waziry, A. B. Wardak, and J. Rasheed, "Character and Digit Recognition Using ANN Back Propagation Algorithm and Image Segmentation," in *Proc. Int. Conf. Smart Inf. Syst. Technol. (SIST)*, Nur-Sultan, Kazakhstan, 2022, pp. 1–5. https://doi.org/: 10.1109/SIST54437.2022.9945756.
24. A. Das, A. S. Azad Rabby, I. Kowsar, and F. Rahman, "A Deep Learning-based Unified Solution for Character Recognition," in *Proc. 26th Int. Conf. Pattern Recognit. (ICPR)*, Montreal, QC, Canada, 2022, pp. 1671–1677. https://doi.org/:10.1109/ICPR56361.2022.9956348.
25. A. W. C. Lee, J. Chung, and M. Lee, "GNHK: A Dataset for English Handwriting in the Wild," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, 2021.