International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 15, No. 1, March 2026, pp. 194~203
ISSN: 2089-4864, DOI: 10.11591/ijres.v15.i1.pp194-203

ELLMW: an enhanced vision–language model for reliable text extraction from manually composed scripts

Dhivya Venkatesh, Brintha Rajakumari Sivaraj
Department of Computer Science, Bharath Institute of Higher Education and Research, Chennai, India

Article Info

Article history:

Received Nov 24, 2025
Revised Jan 12, 2026
Accepted Feb 17, 2026

Keywords:

Extracting text
LLMWhisperer
Long short-term memory
Machine learning
Optical character recognition

ABSTRACT

While conventional optical character recognition (OCR) systems can digitize text, they struggle with diverse handwriting styles, noisy inputs, and unstructured layouts, limiting their effectiveness. This study proposes enhanced large language model whisperer (ELLMW), a vision–language framework for accurate text extraction (TE) from fully handwritten scripts. The methodology integrates advanced preprocessing (noise reduction, binarization, and skew correction), deep learning–based handwriting recognition convolutional neural network–long short-term memory (CNN–LSTM), and LLM-based post-correction to ensure context-aware and structurally coherent outputs. The system converts scanned images, portable document formats (PDFs), and irregularly formatted answer sheets into machine-readable text, while automatically correcting errors in spelling, grammar, and layout. Experimental evaluation on a curated dataset of handwritten examination answer scripts (HEAS) demonstrates that ELLMW achieves 97.8% accuracy, 1.04%-character error rate (CER), and 3.24%-word error rate, outperforming widely used OCR tools including Tesseract, EasyOCR, Google Cloud Vision (GCV), PaddleOCR, ABBYY FineReader, and Transym OCR. The results highlight the model's robustness across varying handwriting styles, noisy backgrounds, and complex document structures.

Corresponding Author:

Dhivya Venkatesh
Department of Computer Science, Bharath Institute of Higher Education and Research
Chennai, India
Email: dhivyamasc@gmail.com

1. INTRODUCTION

Background and problem overview: evaluating handwritten examination answer scripts (HEAS) manually remains the standard approach in educational institutions worldwide. While human grading allows teachers to assess nuanced student understanding, reasoning, and creativity, it is inherently time-consuming, inconsistent, and prone to bias, particularly during large-scale examinations with hundreds or thousands of answer scripts. Research indicates that teachers may spend 20–25 minutes per script, depending on length and complexity, which can lead to significant delays in returning results to students [1]-[4]. Furthermore, manual evaluation introduces variability in scoring due to subjective interpretations, fatigue, and inconsistencies across different evaluators, compromising the fairness and reliability of academic assessments [5]-[8].

The advent of digital assessment tools has partially alleviated this burden; however, most systems are designed for multiple-choice or digitally submitted assignments, leaving handwritten responses largely unautomated. In addition, manual transcription or digitization of these handwritten scripts for archival or analytic purposes is costly and error-prone [9]-[12]. Consequently, there is a pressing need for an automated, accurate, and scalable solution capable of converting handwritten text into machine-readable formats, preserving semantic integrity, and supporting downstream educational applications such as automated grading, plagiarism detection, and feedback generation [13]-[16].

Limitations of existing methods: conventional optical character recognition (OCR) systems, including Tesseract, EasyOCR, PaddleOCR, ABBYY FineReader, and Google Cloud Vision (GCV), have been widely used to digitize printed and some handwritten text. While these systems exhibit robust performance on printed documents with standard fonts and layouts, they struggle to handle the inherent variability of human handwriting [17]-[21]. Challenges arise due to diverse handwriting styles, varying stroke thicknesses, inconsistent spacing, cursive scripts, and the inclusion of symbols, diagrams, tables, and mathematical notations.

Moreover, traditional OCR methods generally follow a two-stage pipeline: text region detection followed by character recognition. Errors in either stage propagate, reducing overall accuracy. These systems also lack contextual and semantic understanding, making them incapable of correcting misrecognized words or interpreting domain-specific terminology. For example, OCR engines may misread “O” as “0” or fail to correctly interpret multi-line equations and annotations common in examination scripts. Such limitations result in high error rates, reduced reliability, and poor performance in downstream tasks, including automated grading, keyword extraction, and semantic analysis, highlighting the inadequacy of existing OCR solutions for complex handwritten academic documents [22]-[25].

Recent advances in deep learning have introduced convolutional neural network (CNN), long short-term memory (LSTM), and transformer-based models for handwriting recognition. While these methods improve character-level recognition, they still fail to integrate linguistic context, post-correction, and structured output generation, which are essential for real-world academic applications.

Research gap: despite progress in OCR and deep learning-based handwriting recognition, accurately extracting structured and contextually coherent text from fully handwritten examination scripts remains a significant challenge. Existing models are limited in their ability to:

- Handle complex document layouts, including multi-column formats, tables, figures, and margin notes.
- Recognize diverse handwriting styles, including cursive, mixed, and multilingual scripts.
- Correct errors using context-aware or language-aware mechanisms, leaving outputs prone to semantic and syntactic inaccuracies.
- Support downstream educational applications that require structured, machine-readable, and semantically reliable data for automated grading or analytics.

In practice, these gaps mean that current systems either require extensive manual intervention or produce outputs unsuitable for intelligent evaluation, limiting their applicability in academic assessment workflows. This study identifies the need for an integrated framework that combines image preprocessing, handwriting recognition, and large language model-based (LLMs) post-processing to bridge these limitations and provide accurate, structured, and context-aware text extraction (TE).

Objective of the study: the primary objective of this study is to develop a reliable, scalable, and context-aware framework for automatic TE from fully HEAS. Specifically, the system is designed to:

- Preprocess scanned documents to reduce noise, correct skew, and normalize image quality.
- Recognize diverse handwriting styles, including cursive, printed, and mixed forms, while accurately detecting characters, words, and non-text elements.
- Integrate post-processing techniques using LLMs to correct misrecognized words, enhance contextual accuracy, and structure output text in meaningful formats.
- Enable downstream applications such as automated grading, digital archiving, plagiarism detection, and educational analytics, facilitating more efficient and fair evaluation of student performance.

Through these objectives, the study aims to bridge the gap between unstructured handwritten inputs and machine-readable, contextually coherent outputs, providing a practical, AI-driven solution for modern educational assessment systems.

Contribution of the research:

- A novel enhanced LLMWhisperer (ELLMW) framework combining advanced preprocessing, CNN–LSTM–based handwriting recognition, and LLM-based post-correction.
- Demonstrated superior performance over conventional OCR tools, achieving 97.8% accuracy and 1.04% character error rate (CER) on fully handwritten answer scripts.

Enables downstream applications such as automated grading, plagiarism detection, and digital archiving, bridging the gap between handwritten inputs and digital outputs.

2. LITERATURE REVIEW

Using sophisticated preprocessing techniques like binarization as well deskewing, and noise reduction, Patience *et al*. [9] describes to tackles typical problems including low-resolution images illustrations, noisy images, and different TR. Performance evaluations show notable gains in computing speed and accuracy for TR. Furthermore, a comparison with alternative OCR systems is offered to emphasize the benefits of the employed method. Tesseract's abilities can be greatly enhanced via suitable integration and preprocessing, according to the research, making it a potent tool for a variety of TR applications. Crosilla *et al*. [10] describes to deliver an overview of the current capabilities of multimodal large language models (MLLMs) for handwritten TR (HTR), assessing their potential when compared to traditional task-specific, supervised models. The results show that LLMs currently show a strong performance on English texts, yet they demonstrate a weaker performance on languages other than English, and do not possess a significant capability for self-correction. Kampelopoulos *et al*. [11] attempts to fill this vacuum by offering a thorough analysis of the current applications and use examples of LLMs in the architecture, engineering, and construction (AEC) sector that have already been established. In addition, it was feasible to classify them, identify new issues and potential paths for the field, and provide practical suggestions for industry stakeholders by examining the main advantages and disadvantages of these applications as well as by taking into account appropriate studies on the topic.

Li *et al*. [12] provide a simple vision transformer (ViT-based) model solely utilizing the encoder portion of the traditional transducer for HTR. In order to achieve good performance on this task with little changes to the usual ViT architecture, this research suggests a novel ViT like model. According to the initial results, ViT can produce good results, especially on the largest dataset, the LAM dataset [13], which has 19,830 training samples. A one-shot handwritten text synthesis framework called WriteViT is presented by Nam *et al*. [14]. It integrates the ViT family of models, which has demonstrated outstanding results in a variety of computer vision tasks. A lightweight ViT-based recognizer, a multi-scale producer constructed using transformer encoder-decoder modules augmented by conditional positional encoding (CPE), and a ViT-based writer identifier for identifying style incorporation are all included into WriteViT. This paper shows that ViT-based synthesis architectures have great potential to enhance the production of handwritten text, especially in multidisciplinary or low-resource settings. A lightweight architecture called vision and spatially-aware text analysis OCR (VISTA-OCR), which combines text detection and TR into a single dynamic model, is presented by Hamdi *et al*. [15]. In contrast to traditional approaches that need distinct branches with specific TR and detection parameters, this method uses a transformer decoder to produce text transcripts and their spatial coordinates in a single branch with sequential manner. The outcomes demonstrate that the model successfully develops geometric descriptions of spatial tokens, allowing for accurate and fine-grained document OCR. These results include both statistical metrics and qualitative error evaluations. The efficient and accurate scene text detector (EAST) technique is implemented and evaluated by Soni *et al*. [16] for text identification and detection in natural scene photos. Comparing the effectiveness of three well-known OCR methods is the goal of the study. Bounding boxes emphasizing the identified text portions were used to visually display the outcomes of applying the EAST model to a set of test image samples. The algorithm's efficiency was demonstrated by recording the resulting timings for each image, which showed average timing range between 0.439 to 0.446 seconds for the corresponding test images. These findings show that the EAST method works in real-time and is accurate, which makes it appropriate for situations that need instant TR.

Shylesh *et al* [17] created a model specifically for assessing 40-word responses. By varying the parameters, deep layers, number of neurons, activation function, and bidirectional long short-term memory (BiLSTM) layers, the model is constructed using a range of potential ways. To find the lightest and best model, this paper made numerous adjustments to each parameter and changed the number of layers, LSTMs, or nodes. The model's performance using the test set, achieving an accuracy of about 80%. The accuracy might be raised by improving the training data. This result suggests that a greater degree of analysis and research is needed to create a model for lengthier texts (200–250 words) that include images and equations. In order to increase the model's accuracy and circumvent the time-consuming process of training it on huge datasets, Prerana *et al*. [18] suggest using a Google Cloud Platform (GCP) OCR text extract model that has been trained on enormous volumes of publicly accessible data. In terms of marking accuracy, the suggested language processing method fared better than any of the algorithms. The distinctive feature of this technique is that, before to direct comparison, the keywords are lemmatized, case-folded, stop-words removed, and duplicate verified.

3. METHOD

3.1. LLMWhisperer framework based on handwritten answer script text extraction

The effectiveness of OCR in HEAS TE systems is limited by variations in handwriting and formatting present significant challenges for OCR systems, as they must accommodate diverse styles and inconsistencies across different answer sheets. Additionally, the presence of non-textual artifacts, such as smudges, stains, or stray marks, can further complicate the digitization process, potentially leading to inaccuracies in TR. Recognizing complex characters, especially those found in diagrams, mathematical notations, or specialized symbols, adds another layer of difficulty, requiring advanced feature extraction techniques, and robust algorithms to ensure accurate processing [19]-[21]. To overcome these issues this paper proposes a solution to assist professors by recognizes handwritten image text. This research adopts a multi-phase methodology that combines image preprocessing, segmentation, HR, and LLM based enhancement techniques to accurately extract textual data from scanned document of fully HEAS. Conventional OCR engines, like Tesseract, have trouble with complicated or non-standard layouts due to their rely on static models and preset patterns. The suggested ELLMWhisperer framework employs sophisticated algorithms to adjust dynamically, maintaining document context while guaranteeing text parsing accuracy. The suggested paradigm is a sophisticated text parser that gets ready for LLMs to analyze complicated documents later on, such as scanned portable document formats (PDFs), photos, and tables. Figure 1 displays the overall block diagram for the suggested method.

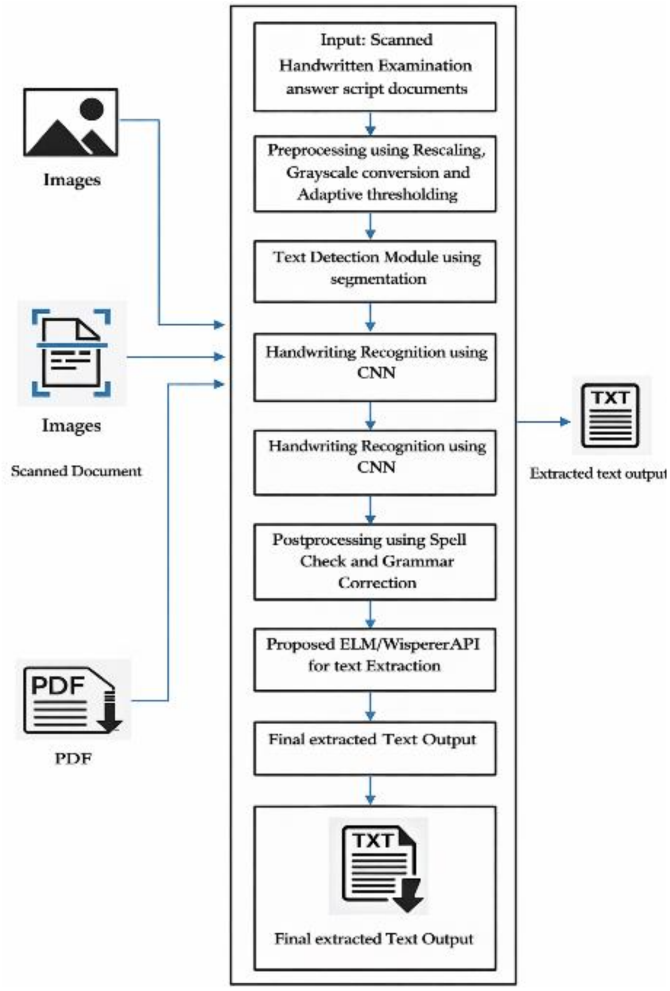
Images
Images
Scanned Document
PDF
Input: Scanned Handwritten Examination answer script documents
Preprocessing using Rescaling, Grayscale conversion and Adaptive thresholding
Text Detection Module using segmentation
Handwriting Recognition using CNN
Handwriting Recognition using CNN
Postprocessing using Spell Check and Grammar Correction
Proposed ELM/WispererAPI for text Extraction
Final extracted Text Output
TXT
Final extracted Text Output
TXT
Extracted text output

Figure 1. Overall block diagram for proposed model

The application of ELLMWishperer for the HTR task involves a different workflow than the one based on pre- processing, segmentation, and TR. While preprocessing may be required in some cases where models require image resizing or corrections to orientation, segmentation is no longer necessary, nor are specific ground truth (GT) annotations related to both text and layout. This study primarily focuses on TE from scanned document of HEAS investigates the potential of incorporating ELLMW as a post-correction step in the workflow, to assess whether LLMs could detect and correct errors in their initial predictions and potentially improve accuracy. Finally, models' predictions are evaluated using traditional metrics such as CER, word error rate (WER), and bag of words-WER BoW-WER, guaranteeing a comparable evaluation with traditional OCR models.

3.2. Dataset description

This dataset is curated to sustenance the growth and assessment of the enhanced LLMWhisperer framework for automatic TE from fully HEAS. It aims to address the challenges of unstructured, diverse, and natural handwriting styles across various student populations and academic subjects. The dataset comprises scanned images of real or simulated handwritten examination answer sheets collected from academic institutions or generated using synthetic handwriting tools to protect privacy. Therefore, the images vary in quality and resolution, often as a result of student differences in handwriting style. As such, the image datasets are susceptible to noise, blurry registration, or distortions. The scanned images are pre-processed for quality improvements, such as noise reduction and alignment. It includes samples from students of varying ages, genders, and writing styles (cursive, print, and mixed). The page layouts contain single-column, multi-paragraph responses, figures, tables, and margin notes. The noise factors include single-column, multi-paragraph responses, figures, tables, and margin notes. This dataset is important for training the model for automatic student response recognition and assessment.

3.3. Text detection and segmentation

The initial stage in the text detection procedure is to locate the text area on the page. This involves looking for groups of pixels in a handwritten answer script image comprising elements, characters; each of these elements has a class assigned to it. It is necessary to utilize any thresholding technique to enable further investigation. Adaptive thresholding often works best when the proper settings are used. The following is how the segmentation process is executed in (1).

$$Prb\ den\ (I)\ =\ \frac{Hist\ (I)}{MN}\ +\ threshd\ +\ intense(I) \tag{1}$$

For I= 0, 1….N-1. In (2) and (3) compute a feature for the adaptive pixel set of an immediate layer.

$$O_I = \eta\ \mathrm{P(a)} + \mu\ \mathrm{(a, b)\ for\ a = 1,2}\ldots\ldots\ldots \mathrm{n} \tag{2}$$

$$O_I = \eta\ \mathrm{P(b)} – 2(\mathrm{x}) + \mu\ \mathrm{(a, b)\ for\ a = 3,4,5}\ldots\ldots\ldots \mathrm{m} \tag{3}$$

In order to arrange the layers in a sequential fashion for accurate and unique TR̥ is the layer significance that is taken into consideration. The outcome is estimated by the constructed single layer as the sum of all the pixels that comprise a set, with a fixed total. A new layer is created using (4).

$$O_I=\frac{\sum W\ (a)\ h(I)+\theta}{\sum W\ (a)}+\mu(\mathrm{a,b})F_a \tag{4}$$

3.4. Handwritten text recognition

In this step HTR is the process of converting handwritten answer script in images or scanned documents into machine-readable text. In this case, many hyperparameters are active. These either have default values specified or have been generated using the training data. The handwritten sample have resized all of the word segmented image data to 32×128 pixels in order to feed our dataset into the CNN layer. After that, these words are fed into a CNN layer, which has 64 nodes and a kernel size of (3, 3). A pooling layer with a kernel size of (2,2) received the output from the first CNN layer, reducing its form to 16×64. Two CNN layers, each having 256 nodes, receive the output from the preceding pooling layer. A pooling layer comes after each of these CNN layers. Following these two CNN and pooling layers, the output data's form is shrunk to 4×32. Once more, two CNN layers with 512 nodes get this data, and then two batch normalization layers follow. A CNN layer receives the output of the last batch normalization layer after it has accepted through one pooling layer. In order to identify a word and ultimately convert it into a text file, this research work employed an LSTM layer with 256 layers in the final layer.

For accurate TE, the hidden layers are taken into consideration since NN is utilized. In (5) describes the input of each hidden layer.

$$H(I\ (a,b)) = I_{a,b} + \sum F_a * W \tag{5}$$

where W and F stand for the input and hidden layer weights and the bias value of the hidden layer, respectively. Therefore, (6) is used to compute the output of each hidden layer.

$$\text{O (I (a,b))} = \frac{\sum_{b=1}^{n}(X_{a,b}^{M}+Y_{a,b}^{N})x_b}{\sum_{a=1}^{m}\left(X_{a,b}^{M}+Y_{a,b}^{N}\right)} \tag{6}$$

3.5. Post-processing

Then post-processing enhances and corrects the output generated by the HTR system from OCR. It involves several steps: error correction: use context-based algorithms to correct common HTR mistakes (e.g., confusing 'O' with '0' or 'I' with '1').

3.6. Text extraction using proposed ELLMWishperer

In this step, this study explores to increase the model's ability to extract raw text from scanned handwritten answer script accurately, a proposed ELLMW analysis step is taken after processing through first four steps described above. A comprehensive text parser that uses the suggested ELLMWhisperer structure creates complex documents, such as scanned PDFs, pictures, and tables, for LLMs to process later. Its main goal is to transform semi-structured or unstructured data into structured, useful formats like JavaScript object notation (JSON). ELLMW excels at managing handwritten text, noisy scans, and complex layouts, allowing businesses to optimize document parsing processes. Rather, it applies clever parsing techniques to improve OCR results, making the data easier for LLMs and other systems to understand.

The refinement task is carried out in three steps where the model is asked again to analyze both the original image and the output produced at time t–1, which means that in the first iteration the model examines the output of the zero-shot task. In each subsequent step a different user prompt is provided, specifying where the focus of the model is needed, whether in the orthography or spelling refinement or in the layout formatting or in both as the last prompt. Here the extracted raw text T_{raw}=T(X,Y) =($W_1, W_2, \dots, W_n$) is passed to language model of proposed ELLMW for correction, context aware restricting or denoising is mentioned in (7)-(10).

$$T_{final} = f_{ELLMW}\ (T_{raw}) \tag{7}$$

$$T_{final} = f_{ELLMW}\ (\text{T (X, Y)}) \tag{8}$$

$$T_{final} = f_{ELLMW}\ ((W_1, W_2, \dots, W_n\ )\ ) \tag{9}$$

$$\text{Output} = T_{final} - (w_1^*\ , w_2^*\ , \dots \dots , w_m^*) \tag{10}$$

One of the key advantages of ELLMW is their ability to handle both structured and unstructured evaluation tasks. By incorporating well-designed prompts that include questions, rubrics, and example answers, ELLMW can perform nuanced evaluations and provide justifications for their decisions. This transparency enhances trust in the automated grading process. This model also versatile, capable of managing open ended questions and subjective responses effectively. Their scalability and adaptability make them suitable for a wide range of educational contexts.

4. EXPERIMENTAL SETUP

Using a dataset of full HEAS samples, this study evaluates the proposed ELLMW model on standard OCR tasks, specifically text recognition (TR). The objective is to demonstrate that the proposed method outperforms state-of-the-art OCR models in terms of performance while also offering greater flexibility and generalization capability. All experiments were conducted on a desktop PC equipped with an Intel i3 processor, 8 GB of RAM, 512 GB of storage, and a hard disk drive (HDD). To evaluate the accuracy and quality of the text recognized by the proposed ELLMW model for HEAS, several key evaluation metrics and criteria were employed. These metrics are used to assess the efficacy of suggested solutions by determining how well they can convert handwritten or scanned exam answer script text into a machine-readable arrangement. Some common estimation criteria used in the context of HEAS are listed below.

a. CER*:* it is the “inverted accuracy”, meaning that it represents the error rate at character level based on the Levenshtein distance. This suggests that all the metrics that incorporate this aspect depend on the reading order. To calculate the CER, the following formula has been used:

$$CER = \frac{I+S+D}{N} = \frac{I+S+D}{C+S+D}$$

The reference text must be transformed into the given GT by making the following changes: I represent the number of insertions, S represents the number of substitutions, D represents the number of deletions, and N is the total amount of characters in the GT. To give an idea of what these percentages actually represent, a CER below 5% is considered very good, if it falls in a range between 5 to 10% is good, excellence is achieved with a CER below 2.5%.

b. WER: it is the word-level equivalent of CER, calculating the number of additions, deletions, and substitutions in the recognised words as a percentage of all the words in GT.

$$WER = \frac{S_w + I_w + D_w}{N_w}$$

Similarly, the number of term substitutions (S_w), insertions (I_w), and deletions (D_w) required to change one string into another is added up, and the total number of ground-truth terms (N_w) is used to determine the WER. BoW-WER it is based on the reading order, meaning that, if a word is detected but not in the correct positioning, it will not be considered. To address these issues and consider the correctly detected words regardless of their order, which is particularly useful for information retrieval tasks, the WER-BoW metric has been introduced.

$$WER - BoW = \frac{N-(P \cap N)}{N}$$

Where, N is the total number of words in the GT and P the number of words in the prediction. Therefore, this value can be much lower than WER when the reading order between GT and prediction is different. The TE results from each scanned HEAS sample demonstrate that the proposed ELLMW model significantly outperforms conventional OCR APIs, achieving an average accuracy of 97.8%.

Figure 2 illustrates the CER, demonstrating that the proposed ELLMW model consistently achieves the lowest character-level errors among all evaluated OCR frameworks, indicating its superior capability in accurately identifying individual characters in handwritten scripts. Figure 3 presents the WER comparison, confirming that ELLMW effectively recognizes complete words, with minimal substitutions, insertions, or deletions, thereby enhancing overall word-level accuracy. Figure 4 shows the WER-BOW comparison, highlighting the model's strength in preserving contextual integrity and capturing word-level semantics more precisely than traditional OCR methods.

Figure 5 provides a comprehensive comparison of key metrics—precision, recall, F1-score, and accuracy—across all methods. The proposed ELLMW model demonstrates the highest performance across all metrics, with precision of 97.2%, recall of 94.7%, F1-score of 97.4%, and overall accuracy of 97.8%. By contrast, traditional OCR APIs such as EasyOCR, Tesseract, GCV, PaddleOCR, ABBYY FineReader, and Transym OCR achieved lower performances ranging between 92–96% across these metrics.

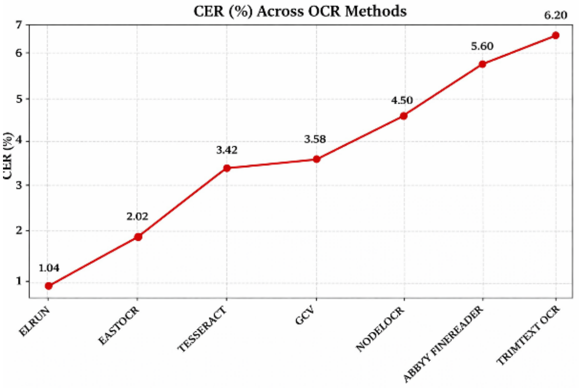

Figure 2. CER across various methods

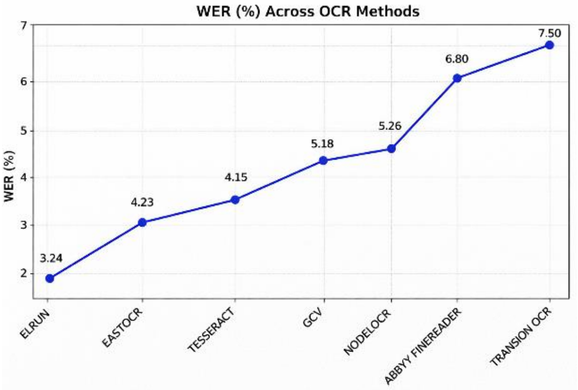

Figure 3. WER across various methods

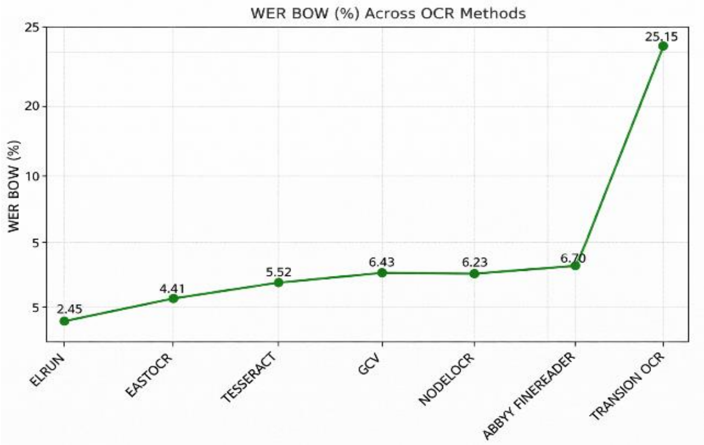

Figure 4. WER BOW across various methods

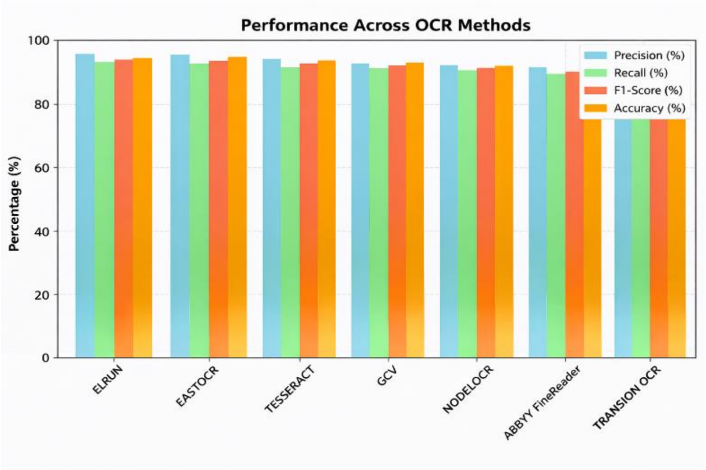

Figure 5. Comparison of metrics across various methods

5. CONCLUSION

This study presented a comprehensive evaluation of the proposed ELLMW framework for automated TE from HEAS. The experiments demonstrated that ELLMW consistently outperforms traditional OCR methods, including Transym OCR, GCV, PaddleOCR, EasyOCR, Tesseract, and ABBYY FineReader, across multiple evaluation metrics. Specifically, the model achieved a CER of 1.04%, WER of 3.24%, WER-BOW of 2.45%, and an overall accuracy of 97.8%, highlighting its superior capability in accurately recognizing both characters and words. The proposed framework also showed high precision, recall, and F1-score, reflecting its robustness in preserving text integrity and contextual information. The results indicate that ELLMW is highly effective in handling diverse handwriting styles, text orientations, and document layouts, making it a reliable solution for large-scale automated assessment and grading systems. By significantly reducing error rates and improving extraction accuracy, the framework can streamline evaluation workflows, provide faster feedback to students, and reduce faculty workload. The limitation of the study is that the model may exhibit reduced performance on extremely noisy, heavily smudged, or multilingual handwritten scripts. Future work could focus on further enhancing the framework's adaptability to multilingual scripts, complex mathematical notations, and heavily degraded documents, as well as integrating the model with real-time educational assessment platforms. Overall, the ELLMW framework represents a significant step toward accurate, efficient, and scalable TE from handwritten examination scripts.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Dhivya Venkatesh	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Brintha Rajakumari Sivaraj		✓				✓		✓	✓	✓	✓	✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis

I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting

Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] Md. A. Rahaman and H. Mahmud, "Automated evaluation of handwritten answer script using deep learning approach," *Transactions on Machine Learning and Artificial Intelligence*, vol. 10, no. 4, pp. 1-16, Aug. 2022, doi: 10.14738/tmlai.104.12831.

[2] M. Ravikumar, S. S. Sampath Kumar, and G. Shivakumar, "Evaluation of handwritten answer scripts using machine learning approaches," *Turkish Journal of Computer and Mathematics Education*, vol. 9, no. 1, pp. 620–633, 2018, doi: 10.17762/turcomat.v9i1.14041.

[3] R. Smith, "An Overview of the Tesseract OCR Engine," in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Curitiba, Brazil, 2007, pp. 629-633, doi: 10.1109/ICDAR.2007.4376991.

[4] Jaided AI, "EasyOCR: Ready-to-use OCR with deep learning," *GitHub repository*, 2020. [Online]. Available: https://github.com/JaidedAI/EasyOCR. (Accessed: Nov. 21, 2025).

[5] J. Puigcerver and C. Mocholí, "PyLaia 2018," GitHub repository. [Online]. Available: https://github.com/jpuigcerver/PyLaia. (Accessed: Nov. 21, 2025).

[6] M. Kišš, K. Beneš, and M. Hradiš, "AT-ST: Self-training adaptation strategy for OCR in domains with limited transcriptions," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2021, pp. 463-477, doi: 10.1007/978-3-030-86337-1_31.

[7] J. Kohút and M. Hradiš, "TS-Net: OCR trained to switch between text transcription styles," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2021, pp. 478-493, doi: 10.1007/978-3-030-86337-1_32.

[8] PaddlePaddle Community, "PaddleOCR: An open-source OCR tool based on PaddlePaddle," GitHub repository, 2021. [Online]. Available: https://github.com/PaddlePaddle/PaddleOCR. (Accessed: Nov. 21, 2025).

[9] O. O. Patience, E. M. Amaechi, O. George, and O. N. Isaac, "Enhanced text recognition in images using Tesseract OCR within the Laravel framework," *Asian Journal of Research in Computer Science*, vol. 17, no. 9, pp. 58–69, 2024, doi: 10.9734/ajrcos/2024/v17i9499.

[10] G. Crosilla, L. Klic, and G. Colavizza, "Benchmarking large language models for handwritten text recognition," *Journal of Documentation*, vol. 81, no. 7, pp. 334-354, 2025, doi: 10.1108/JD-03-2025-0082.

[11] D. Kampelopoulos, A. Tsanousa, S. Vrochidis, and I. Kompatsiaris, "A review of LLMs and their applications in the architecture, engineering and construction industry," *Artificial Intelligence Review*, vol. 58, p. 250, 2025, doi: 10.1007/s10462-025-11241-7.

[12] Y. Li, D. Chen, T. Tang, and X. Shen, "HTR-VT: Handwritten text recognition with vision transformer," *Pattern Recognition*, vol. 158, p. 110967, Feb. 2025, doi: 10.1016/j.patcog.2024.110967.

[13] S. Cascianelli *et al.*, "The LAM Dataset: A Novel Benchmark for Line-Level Handwritten Text Recognition," in *2022 26th International Conference on Pattern Recognition (ICPR)*, Montreal, QC, Canada, 2022, pp. 1506-1513, doi: 10.1109/ICPR56361.2022.9956189.

[14] D. H. Nam, H. T. D. Khoa, and V. N. L. Duy, "WriteViT: Handwritten text generation with vision transformer," *arXiv preprint*, May 2025, doi: 10.48550/arXiv.2505.13235.

[15] L. Hamdi, A. Tamasna, P. Boisson, and T. Paquet, "VISTA-OCR: Towards generative and interactive end-to-end OCR models," *arXiv preprint*, 2025, doi: 10.48550/arXiv.2504.03621.

[16] V. K. Soni, V. Shukla, S. R. Tandan, A. Pimpalkar, N. K. Nema, and M. Naik, "Performance evaluation of efficient and accurate text detection and recognition in natural scene images using EAST and OCR fusion," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 16, no. 1, 2025, doi: 10.14569/IJACSA.2025.0160144.

[17] A. Shylesh, A. Raafeh, S. Mathin, V. B. Prakash and H. Shanmugasundaram, "Automated Answer Script Evaluation Using Deep Learning," in *2023 International Conference on Computer Communication and Informatics (ICCCI)*, Coimbatore, India, 2023, pp. 1-5, doi: 10.1109/ICCCI56745.2023.10128311.

[18] M. S. Prerana, S. M. Chavan, R. Bathula, S. Saikumar, and G. Dayalan, “Eval: Automatic evaluation of answer scripts using deep learning and natural language processing,” *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, no. 1, pp. 316–323, 2023.

[19] V. Agrawal, J. Jagtap, and M. P. Kantipudi, “An Overview of Hand-Drawn Diagram Recognition Methods and Applications,” *IEEE Access*, vol. 12, pp. 19739-19751, 2024, doi: 10.1109/ACCESS.2024.3357398.

[20] Y. S. Chernyshova, A. V. Sheshkus, and V. V. Arlazarov, “Two-Step CNN Framework for Text Line Recognition in Camera-Captured Images,” *IEEE Access*, vol. 8, pp. 32587-32600, 2020, doi: 10.1109/ACCESS.2020.2974051.

[21] A. Nikitha, J. Geetha and D. S. JayaLakshmi, “Handwritten Text Recognition using Deep Learning,” *2020 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, Bangalore, India, 2020, pp. 388-392, doi: 10.1109/RTEICT49044.2020.9315679.

[22] F. J. Ayrin *et al*., “Enhancing OCR post-processing through vision-language model,” in *International Conference on Emerging Technologies and Computing Innovations*, 2025, pp. 247-253, doi: 10.1007/978-3-031-92854-3_29.

[23] K. Feldhoff, H. Wiemer, P. Träger, R. Kühne, M. Zimmermann, and S. Ihlenfeldt, “Automatic information extraction from scientific publications based on the use case of additive manufacturing,” *Applied Sciences*, vol. 15, no. 17, p. 9331, 2025, doi: 10.3390/app15179331.

[24] X. Gao *et al.*, “Selecting post-processing schemes for accurate detection of small objects in low-resolution wide-area aerial imagery,” *Remote Sensing*, vol. 14, no. 2, p. 255, 2022, doi: 10.3390/rs14020255.

[25] Y. Zhang, G. Ding, D. Ding, Z. Ma, and Z. Li, “On content-aware post-processing: Adapting statistically learned models to dynamic content,” *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 20, pp. 1–23, 2023, doi: 10.1145/3612925.

BIOGRAPHIES OF AUTHORS

Mrs. Dhivya Venkatesh is pursuing Ph.D. in the Department of Computer Science, Bharath Institute of Higher Education and Research, Selaiyur, Chennai. She has got 7 years of experience in teaching. She is an Editor at International Conference for the ISBN Proceedings. She has been awarded the Best Faculty award, Young Researcher award, Nallasiriyar award, Kalvisemmal award and Samuga Arpanipalar award. She has published and presented papers at international and national conferences. She completed NPTEL/SWAYAM course in “Digital Image Processing”. She is an active participant in various FDP, webinar, and workshops. She served as a resource Person for various Southern Universities. She has organized international conference, workshops, and seminars in college. She can be contacted at email: dhivyamasc@gmail.com.

Dr. Brintha Rajakumari Sivaraj is an Associate Professor at Bharath Institute of Higher Education and Research, Chennai, in the Department of Computer Science. She has over 24 years of teaching experience and a robust academic background in computer science, with a specialization in applying machine learning techniques to healthcare and data quality research. She has authored key publications addressing Alzheimer’s diagnosis, ocular disease detection, data mining, and cloud-based computing security. She is an active member of professional bodies such as ISTE and IAENG, and has published more than 45 research papers in Scopus indexed, Web of Science, and UGC CARE listed journals. She can be contacted at email: brintha.ramesh@gmail.com.