Cureus Journal of Computer Science
https://doi.org/10.7759/s44389-026-00152-5

Original Article

Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques

Divyanshi Trivedi [1], Rashi Agarwal [2]

1. *Data Science, Banasthali Vidyapith, Rajasthan, IND*
2. *Computer Science and Engineering, Harcourt Butler Technical University, Kanpur, IND*

Received: May 18, 2026 | Review began: June 17, 2026 | Review ended: July 07, 2026 | Published: July 07, 2026

Abstract

Optical character recognition (OCR) is an essential technique for transforming document images into readable text. However, the OCR process faces significant challenges when processing moderately noisy, unstructured documents due to practical degradations such as blurring, stain marks, ink degradation, and background noise. In this research, a comprehensive comparative analysis of three state-of-the-art deep learning models, namely EasyOCR, Donut, and LayoutLMv3, was conducted on the Kaggle Denoising Dirty Documents dataset before implementing a hybrid combiner based on mutual similarity voting and three post-processing algorithms consisting of rule-based correction, Retrieval-Augmented Generation (RAG), and Large Language Model-based correction. The experimental analysis was conducted by evaluating all models on 20 test images using Word Error Rate (WER), Character Error Rate, Accuracy, Precision, Recall, F1-score, and Confusion Matrix as performance metrics. The results show that LayoutLMv3 achieved the best F1-score of 0.9168 by integrating text, image patches, and spatial coordinates within a transformer network. EasyOCR attained the lowest WER of 0.2883 using a two-stage approach consisting of CRAFT detection and Convolutional Recurrent Neural Network-based text recognition. Donut also performed poorly on the dataset, achieving a WER of 2.0997, showing that pretraining end-to-end models on structured documents fails when applied to unstructured, noisy paragraphs - an important negative insight regarding model-task alignment. The hybrid combiner model produced consistently strong performance, achieving a WER of 0.3192 and an F1-score of 0.8995 by effectively eliminating inaccurate Donut predictions through a similarity-based voting scheme. In terms of the post-processing methods, rule-based correction achieved the greatest improvement in WER, reaching 0.3185, while the use of RAG led to degraded performance, with a WER of 0.5630, owing to a small corpus size that resulted in incorrect document retrieval. These findings demonstrate that selecting an architecturally appropriate model - specifically one that jointly encodes text, image, and spatial layout - delivers greater performance gains than applying complex post-processing, providing clear design guidance for deploying OCR systems on moderately noisy, unstructured documents.

Categories: Machine Learning (ML), Deep Learning, Natural Language Processing (NLP)

Keywords: ocr, deep learning, easyocr, donut, layoutlmv3, hybrid model, post-processing, rag, noisy documents, text recognition

Introduction

The computational method of extracting text from document photographs is known as optical character recognition (OCR). In the past, OCR technology has been widely used for a wide range of purposes, such as digitizing documents, managing archives, retrieving information, recognizing forms, and digitally preserving old manuscripts. The need for

reliable OCR techniques that can produce high-precision results has increased dramatically in a number of crucial industries, including banking, education, legal consulting, and healthcare, as businesses move more and more toward fully digitalized workflows [1].

The operational accuracy of these frameworks remains largely dependent on the quality of the input image, despite the fact that OCR techniques have come a long way in the last 20 years. In real-world situations, a variety of geometric and environmental aberrations often deteriorate document photographs. These include low contrast, backdrop artifacts, insufficient or uneven lighting, motion blur, structural stains from deteriorating paper, faulty optical lenses, and aging-related ink degradation [2].

Conventional OCR engines, like Tesseract, mostly depend on traditional computer vision methods like structural pattern matching, connected component analysis, and binarization [3]. When exposed to moderately or substantially contaminated inputs, these traditional approaches fail to provide accurate textual outputs, even though they work satisfactorily on flawless, high-resolution scanned texts. Modern deep learning paradigms have offered cutting-edge architectures intended to reliably interpret text from extremely noisy document images in order to lessen these constraints. Standard sequence-to-sequence networks [4] and basic self-attention processes [5] serve as the technological underpinnings of these sophisticated systems, which in turn spurred the creation of deep bidirectional language representations [6].

The Document Understanding Transformer (Donut) system employs an end-to-end transformer architecture, building upon existing deep learning frameworks. This technique eliminates the requirement for a separate, discrete OCR preprocessing step by processing raw visual pixel values directly to carry out text extraction implicitly [7]. Meanwhile, some multimodal frameworks such as LayoutLMv3 concurrently encode spatial layout coordinates, visual cues from document images, and textual tokens within a single transformer backbone. This multimodal approach divides layout images into distinct visual patches to conceptualize documents and uses shifted window methods to extract hierarchical visual information [8,9]. When processing papers with complicated layouts, the model's reading comprehension is much improved by integrating these unified representations [10]. Additionally, the EasyOCR framework utilizes a decoupled two-stage pipeline incorporating Character Region Awareness For Text (CRAFT) detection [11] and a Convolutional Recurrent Neural Network (CRNN) for text recognition [12]. This architecture draws its structural baseline validation from standard scene text recognition benchmarks [13], rendering it highly adaptive for parsing text afflicted with multi-scale noise.

As non-invasive methods to improve text extraction accuracy without changing the underlying OCR model weights, downstream post-processing approaches have been presented in addition to architectural alterations. In order to detect and correct character-level recognition mistakes in OCR outputs, early statistical paradigms showed the great value of linguistic context [14], often using contextual text alignment algorithms [15]. Retrieval-Augmented Generation (RAG), which queries an external source corpus to return corresponding clean textual contexts, and rule-based correction, which uses deterministic heuristic rules for text normalization, are examples of contemporary variations. Additionally, statistical spell correction makes use of deep denoising sequence-to-sequence pre-training frameworks or language models to maximize grammatical fluency and orthographic accuracy [16]. These methods often use specific Siamese network topologies to construct dense sentence embeddings in order to efficiently map noisy text segments onto canonical, clean representations [17]. However, the particular error typologies found in the document have a significant impact on the effectiveness of these correction measures. Previous research suggests that, when applied to moderately noisy documents with only localized character errors that do not affect the underlying sentence structure, intense linguistic post-processing can cause unintentional semantic degradation [18].

To date, most of the research has focused on either severely degraded documents or structured formats exclusive to a given domain, including business bills, receipts, or forms. As a result, there has not been enough scholarly focus on the problem of processing moderately noisy, unstructured paragraph-style text, where degradations occur only at the

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

character level while the global semantic layout remains unaltered. Inspired by this research gap, this paper rigorously, methodically, and comparatively evaluates three different deep learning OCR topologies acting on this understudied document classification.

This study's primary contributions are as follows:

First, using proven noisy OCR normalization corpora as a baseline, the performance of three distinct deep learning OCR architectures - EasyOCR, Donut, and LayoutLMv3 - on moderately noisy unstructured document images from the Denoising Dirty Documents dataset from Kaggle is evaluated [19].

Second, we present a hybrid method that uses a voting mechanism based on mutual similarity between the predictions of all three models to choose the output produced by each of them.

Third, on the hybrid output, we systematically compare three post-processing mechanisms: rule-based correction, RAG-based correction, and (statistical spell correction).

Finally, we empirically demonstrate that conservative corrections outperform complex post-processing techniques on this kind of document using six different criteria over 20 document pictures. While recent developments such as multi-modal transformer frameworks [20,21] and extensive surveys on multi-oriented text recognition [22] have greatly advanced broader document AI benchmarks [23], the specialized application of deep learning for the restoration of localized noisy documents remains a unique challenge [24]. This comparative analysis addresses this important gap.

Materials And Methods

This section describes the complete experimental methodology, including the dataset, all three deep learning models, the hybrid combiner, post-processing techniques, and evaluation metrics, as shown in Figure *1*.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

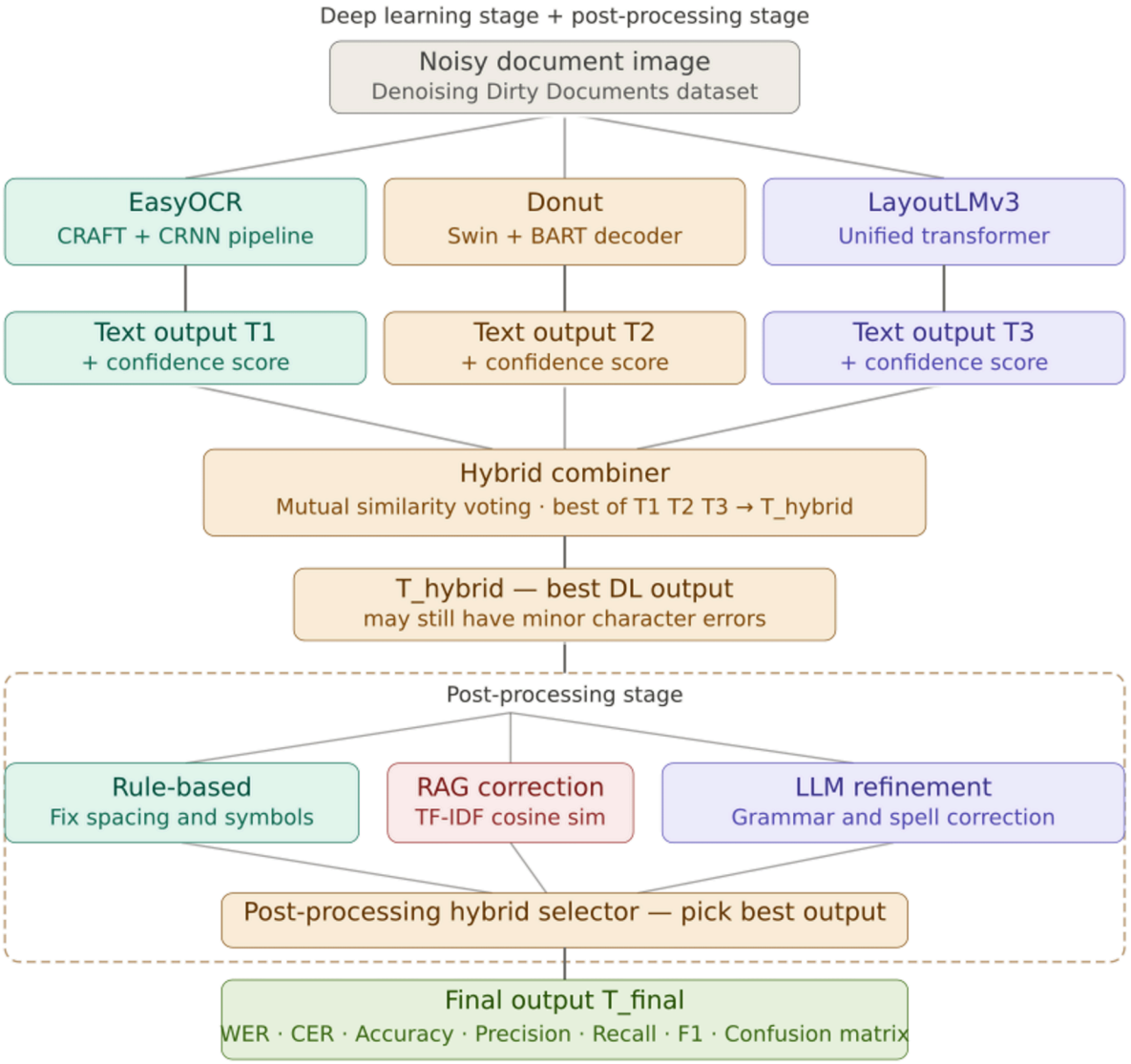

FIGURE 1: Complete system pipeline.

Noisy document images are processed independently by EasyOCR, Donut, and LayoutLMv3. Their outputs are combined by the hybrid combiner using mutual similarity voting. The hybrid output is then refined through three post-processing techniques, and a final selector produces the optimized output.

BART, Bidirectional and Auto-Regressive Transformers; CER, Character Error Rate; CRAFT, Character Region Awareness For Text; CRNN, Convolutional Recurrent Neural Network; DL, Deep Learning; OCR, Optical Character Recognition; RAG, Retrieval-Augmented Generation; TF-IDF, Term Frequency–Inverse Document Frequency; WER, Word Error Rate

Throughout this paper, the following notation is used consistently:

T_{hybrid} denotes the output selected by the hybrid combiner based on mutual similarity voting.

T_{rule} denotes the output produced by applying rule-based post-processing corrections.

T_{rag} denotes the output produced by applying RAG.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

T_{llm} denotes the output produced by applying TextBlob-based statistical spell correction.

T_{final} denotes the optimized final transcription produced by the complete pipeline.

Dataset

The dataset used for this study was obtained from the benchmark Kaggle Denoising Dirty Documents dataset. The dataset consists of high-resolution image pairs containing noisy document scans and their corresponding clean target images of printed textual paragraphs with localized degradation characteristics such as motion blur, dynamic stain marks, fading ink, variable contrast, and background distortion. This dataset is uniquely ideal for the current benchmarking experiment because it explicitly emphasizes unstructured paragraphs of plain text without focusing on structured spatial fields, invoices, or business forms. Moreover, the simulated noise level across these document images is significant enough to trigger semantic character-level substitution errors in standard deep learning architectures, but not so severe that the underlying text becomes completely illegible to a human reader.

There were a total of 30 image pairs extracted for the experimental environment. Specifically, 20 image pairs were utilized for direct cross-model evaluation, while the remaining 10 distinct image pairs were designated exclusively to serve as the external reference corpus for the RAG-based post-processing correction framework. This rigid structural partition is crucial for an objective evaluation setup; if the RAG reference dataset consisted of the identical images used in the pipeline evaluation process, the semantic vector search method would retain direct structural access to the target ground truth, artificially inflating the final performance results.

The baseline ground truth text used to compute all performance metrics was systematically extracted from the clean reference target images using Tesseract OCR configured with Page Segmentation Mode (PSM 6) to treat the layout as a uniform, contiguous text block. To ensure total algorithmic validity, all machine-extracted text blocks were subsequently subjected to manual review and manual character correction, producing flawless, non-empty absolute ground truth strings across all 30 document instances. "To ensure permanent archival stability and total structural reproducibility, the complete experimental code repository and processed dataset splits compiled throughout this study have been deposited in a persistent repository available at: https://github.com/divyatrivedi08/OCR-PostProcessing-Benchmarking and permanently archived with a persistent DOI via Zenodo (https://doi.org/10.5281/zenodo.21157106).

Model 1 - EasyOCR

EasyOCR is a deep learning-based OCR tool that works via a two-stage process where there is a stage for text detection and a stage for text recognition. The model was chosen since it is particularly suitable for processing entire page images of different layouts and noises, fully working offline and showing high efficiency even when dealing with background noises and distortions, as shown in Figure *2*.

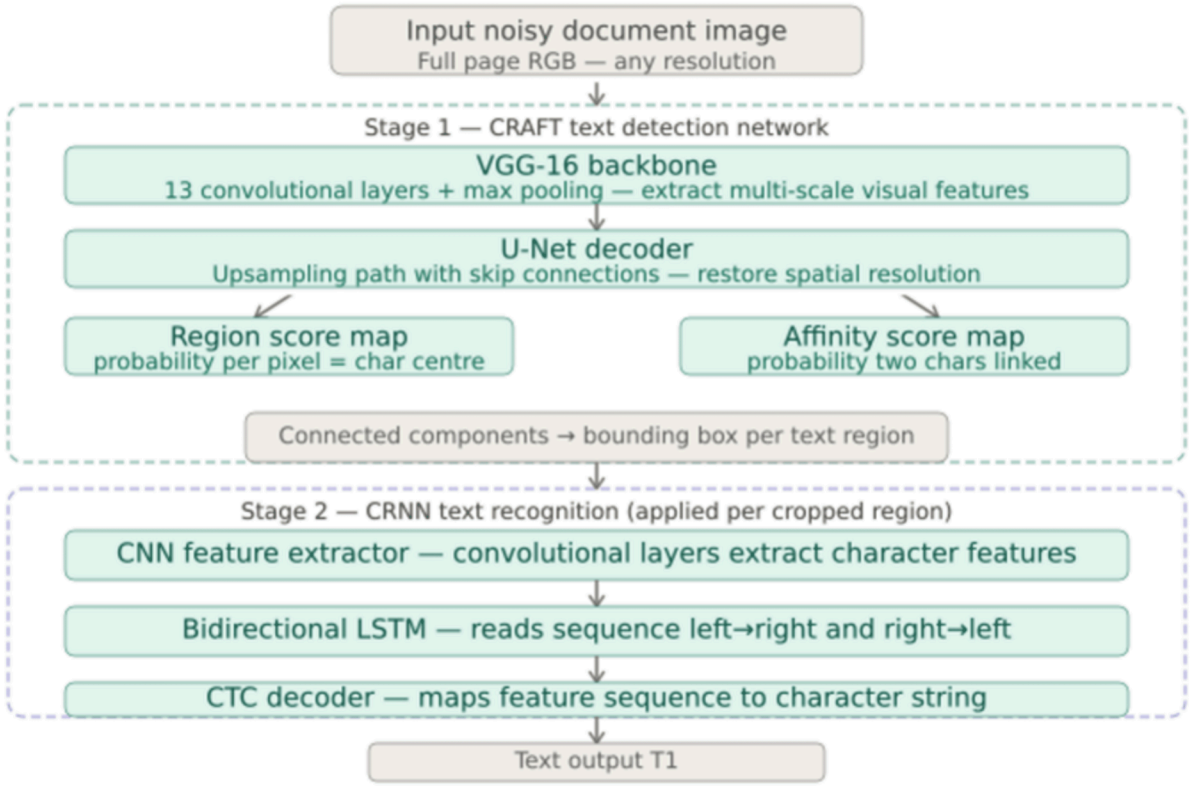

FIGURE 2: EasyOCR architecture.

Stage 1 uses the CRAFT network with VGG-16 backbone and U-Net decoder for text detection. Stage 2 uses the CRNN with bidirectional LSTM and CTC decoder for text recognition.
CNN, Convolutional Neural Network; CRAFT, Character Region Awareness For Text; CRNN, Convolutional Recurrent Neural Network; CTC, Connectionist Temporal Classification; LSTM, Long Short-Term Memory; RGB, Red, Green, Blue

The CRAFT model in Stage 1 operates on the full document input image [5]. The CRAFT network has a VGG-16 architecture comprising 13 convolutional blocks divided into five groups with max pooling in between. As such, the model generates multi-scale features that encode low-level features, such as texture information, and high-level features, such as the structure of the data. The multi-scale features generated by the model are then decoded using a U-Net structure with skip connections between the encoder and decoder. This structure ensures that low-resolution information about character boundaries is preserved during the decoding process. Two maps are then generated at the end of the decoder stage: a region score map, which assigns a probability value to every pixel that can be considered the center of a character, and an affinity score map, which predicts the probability that two adjacent characters form the same word.

During Stage 2, each cropped text area detected by CRAFT undergoes independent recognition using a CRNN recognizer [6]. The CRNN consists of ResNet-inspired convolutional layers, which extract discriminative visual features from each cropped text area. The extracted features are fed into the sequence, which passes through a bidirectional Long Short-Term Memory layer, which reads the input in both forward and backward directions. This allows the network to account for the context in both directions. The result of the bidirectional Long Short-Term Memory layer is decoded by using Connectionist Temporal Classification decoding. Finally, the output text of each recognized text area is concatenated into the final text output T1 for each document image.

Model 2 - Donut

Donut [2] is an end-to-end document understanding model that eliminates the need for a separate OCR preprocessing step. It reads document images directly from pixels and generates text output autoregressively. Donut was selected to evaluate the performance of an end-to-end approach compared to the pipeline-based EasyOCR and the hybrid multimodal LayoutLMv3, as shown in Figure *3*.

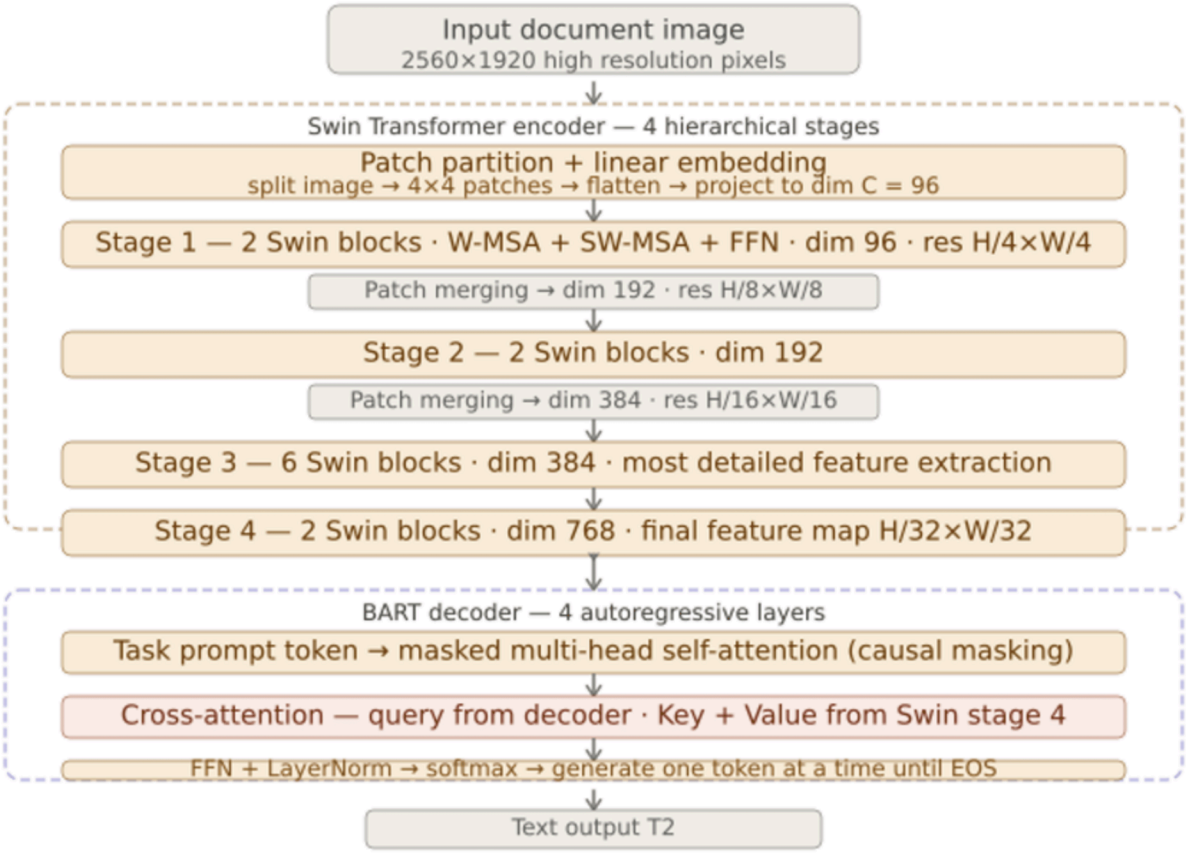

FIGURE 3: Donut architecture.

The Swin Transformer encoder processes the image through four hierarchical stages, with patch merging between stages. The BART decoder generates text autoregressively using cross-attention to the encoder visual features.
BART, Bidirectional and Auto-Regressive Transformers; FFN, Feed-Forward Network; SW-MSA, Shifted-Window Multi-Head Self-Attention; W-MSA, Window-Based Multi-Head Self-Attention

Donut's encoder is based on the Swin Transformer [10], which has four stages of hierarchy. First, the document image of size 2,560 × 1,920 pixels is split into non-overlapping patches of size 4 × 4 pixels. Next, each patch is flattened and linearly projected to an initial channel size of C = 96. In stage 1, two Swin Transformer layers, comprising window-based multi-head self-attention, which runs self-attention inside a local window of 7 × 7, and shifted-window multi-head self-attention, which shifts the window partitioning for exchanging information across windows, are applied before a feed-forward neural network with two layers and a GELU activation function. Patch merging is conducted between consecutive stages to concatenate four neighboring patches, linearly projecting them to double the number of channels while reducing the spatial resolution by half. In stage 2, two Swin Transformer layers of size 192 at a resolution of H/8 × W/8 are employed. In stage 3, six Swin Transformer layers of size 384 at a resolution of H/16 × W/16 are applied. Stage 3 provides the maximum amount of feature extraction by the encoder. Finally, in stage 4, two Swin Transformer layers of size 768 at a resolution of H/32 × W/32 are deployed to extract the final visual feature map.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

The decoder is an autoregressive transformer architecture based on BART (Bidirectional and Auto-Regressive Transformers) and consists of four layers. It receives an input prompt token that is unique to the task. This token provides information about the kind of data that needs to be extracted from the document. At each level of the decoder, there is a masked multi-head self-attention mechanism with causal masking that avoids the model accessing tokens from the future sequence. After that, cross-attention comes into play, where the queries are generated using the decoder, and the keys and values are generated using the Stage 4 encoder feature map. This allows the decoder to pay attention to the relevant portions of the visual data during the generation of the output tokens.

Model 3 - LayoutLMv3

LayoutLMv3 [3] is a multimodal pre-trained transformer model for document AI that simultaneously encodes three types of information: textual content, visual image features, and spatial layout coordinates. This model was selected because it is the only architecture in this study that explicitly models the spatial relationship between text and its visual context, which is expected to provide advantages on noisy document images where understanding layout helps disambiguate corrupted characters, as shown in Figure *4*.

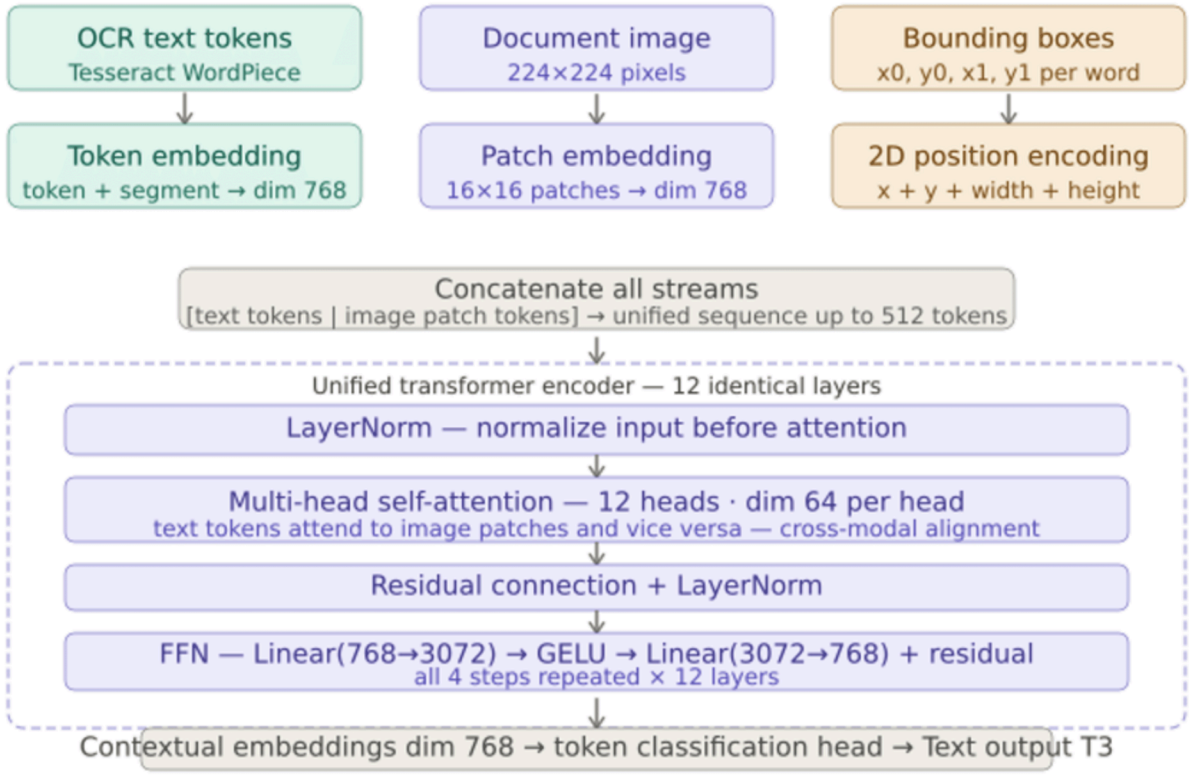

FIGURE 4: LayoutLMv3 architecture.

Three parallel input streams-text tokens, image patches, and bounding box coordinates-are separately embedded and concatenated into a unified sequence processed through 12 transformer layers where text and image tokens attend to each other. FFN, Feed-Forward Network; GELU, Gaussian Error Linear Unit; OCR, Optical Character Recognition

The LayoutLMv3 model processes three distinct streams of data for each document image input. First, it extracts OCR text tokens from the document image using the Tesseract tool. These tokens are then embedded through the token embedding layer, combined with segment embeddings, and projected to a dimension of 768. Second, it extracts 16×16 pixel patches from a resized version of the document image to a dimension of 224×224 pixels. These patches are linearly

projected to a dimension of 768 to create patch token embeddings. Third, it extracts the spatial bounding box coordinates of each word token in the format (x0, y0, x1, y1). The coordinates are normalized between 0 and 1,000 based on the image dimensions and embedded using four separate learnable layers: one each for the x-position, y-position, width, and height.

The three embedding streams are concatenated together along the sequence dimension to create a single stream of up to 512 tokens consisting of both text word tokens and image patch tokens. This sequence is fed into 12 transformer encoder layers that are identical to one another. Each layer starts off with a LayerNorm applied to its input tensor. Multi-head self-attention with 12 heads of size 64 is applied, where the text tokens and the image patches pay attention to each other at the same time, allowing for actual cross-modal alignment between the two modalities. This is followed by a residual connection and another LayerNorm. Then comes the feed-forward network, where there is first an expansion from dimension 768 to 3072 using a linear transformation, then a GELU activation, and finally a reduction in dimension back to 768 via a linear transformation, with yet another residual connection added afterwards. This process is done for all 12 layers, and the output contextualized embeddings for the text tokens go through token classification to give us the output T3.

Hybrid combiner

Once each model has been applied to all 20 images and generated its own output text T1, T2, and T3, respectively, they are aggregated based on their mutual similarity vote. The similarity between each pair of texts is calculated using Python's difflib SequenceMatcher ratio function, which represents the percentage of common characters between two texts compared to their total lengths. Each output candidate T_i for each image is then evaluated by averaging its similarities with the other two candidates, as expressed in Equation 1:

$$\text{Score}(T_i) = \frac{\text{sim}(T_i, T_j) + \text{sim}(T_i, T_k)}{2} \quad (1)$$

where $i, j, k \in$ {EasyOCR, Donut, LayoutLMv3} and $i \neq j \neq k$. The candidate with the highest score is chosen as the hybrid result T_{hybrid}. This method is based on the premise that if two out of the three models produce similar outputs, then that output is more likely to be accurate rather than an isolated result from a single model. Candidate outputs consisting of empty strings or only blanks are excluded from the voting process.

Post-processing techniques

This hybrid result, T_{hybrid}, is further post-processed using three separate methods, each generating an output candidate for the correct output. The final selection process selects the best output from among these candidates.

Rule-Based Correction

The rule-based correction technique uses three deterministic rules on T_{hybrid} using regular expressions. The first rule removes multiple whitespaces and replaces them with a single whitespace. The second rule reconnects broken words that have a hyphen followed by a whitespace in between them. The third rule eliminates any non-ASCII character introduced due to background noise. This approach does not pose a risk of introducing new errors but focuses on removing systematic errors and generates the output T_{rule}.

RAG-Based Correction

In RAG-based correction, the task of improving OCR is considered a retrieval problem. The hybrid output T_{hybrid} is transformed into a TF-IDF vector with the help of sklearn's TfidfVectorizer where unigrams and bigrams are included along with a maximum size of 5,000 for the vocabulary. The similarity measure between the TF-IDF vectors of T_{hybrid} and the TF-IDF vectors of each of the documents within the 10 documents reference corpus, which was not included within

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

the test set, is then calculated by calculating the cosine similarity measure. If the maximum cosine similarity measure exceeds a certain threshold value of 0.4, then the output is given by the most similar corpus document; otherwise, the output is taken as T_{hybrid}.

Correction Using Statistical Spell Checking (TextBlob)

TextBlob is a traditional NLP library that implements statistical spell correction using a Naive Bayes classifier trained on a general English word frequency corpus. It is not a large language model in the contemporary sense; rather, it performs word-level correction by selecting the highest-probability correction for each input word according to its underlying language model. In this study, this method is referred to as statistical spell correction to accurately reflect its implementation, while the notation T_{llm} is retained for consistency with the pipeline notation introduced in Section "Dataset". This statistical correction involves applying the spell correction capabilities of TextBlob to T_{hybrid} to enhance the accuracy of the spelling and grammatical correctness. TextBlob utilizes its underlying statistical language model to generate the most likely correction for every individual word based on edit distance calculations. Only the first 400 characters of T_{hybrid} are used for corrections, while the rest are appended without modifications, resulting in T_{llm}.

Final Selector

In order to determine the best of T_{rule}, T_{rag}, and T_{llm}, a measurement of how similar each is to the original T_{hybrid} through the sequence matcher is used. The output with the highest similarity to the original T_{hybrid} will be the chosen output T_{final}. It can be said that the selection strategy is very conservative in nature.

Evaluation metrics

The quantitative performance tracking framework across all eight parallel pipeline configurations is systematically established using Word Error Rate (WER), Character Error Rate (CER), Accuracy, Precision, Recall, and F1-score.

$$WER = \frac{S + D + I}{N} \quad (2)$$

where *S* stands for substitutions, *D* stands for deletions, *I* stands for insertions, and *N* stands for total number of words in the ground truth transcription. CER is calculated in the same way but for characters. For the computation of Precision, Recall, and F1-score, word-level confusion matrix components were derived as follows. Each predicted text string and its corresponding ground-truth string were tokenised by whitespace into word sequences. True Positives (TP) were defined as the count of words present in both the predicted set and the ground-truth set (set intersection). False Positives (FP) were defined as words present in the predicted set but absent from the ground-truth set. False Negatives (FN) were defined as words present in the ground-truth set but absent from the predicted set. True Negatives (TN) were not computed, as no fixed bounded vocabulary was defined for this task; accordingly, TN is reported as zero for all configurations, which is consistent with standard practice in word-level OCR evaluation where the vocabulary space is open and unbounded.

In cases where WER exceeds 1.0, the computed accuracy value (1 – WER) yields a negative result. Since accuracy cannot be negative in practical reporting, such values are clipped at zero to reflect complete recognition failure. Accordingly, Donut's Accuracy is reported as 0.000, representing a WER of 2.0997, which indicates the model generated more erroneous word insertions than the total number of ground-truth words.

Results And Discussion

To ensure the thorough validity, transparency, and structural reproducibility of the benchmarking process, all experiments in this study were strictly executed on a unified local CPU-based computing environment. No external cloud infrastructure, high-performance GPU acceleration, or distributed parallel computing networks were employed.

This specific architecture choice reflects a realistic, resource-constrained academic deployment scenario, demonstrating that the proposed three-model OCR and post-processing pipeline can run successfully using standard institutional hardware setups without relying on expensive computing clusters.

The detailed, complete hardware and software tool configurations utilized continuously throughout all image processing sessions are systematically outlined in Table *1* and Table *2*.

Component	Specification	Details
Operating System	Windows 11 Home	64-bit operating system, ARM-based processor
Processor (CPU)	Snapdragon X - X126100 - Qualcomm Oryon CPU	12-Core host processor executing at 2.96 GHz
RAM	16.0 GB	High-speed system memory (15.6 GB usable) at 8448 MT/s
GPU	Not Used (CPU-only)	All models run on CPU. No GPU acceleration used (Qualcomm Adreno X1-45 active for display only)
Storage	477 GB SSD Available	Total capacity 477 GB/512 GB class Solid-State Drive
Execution Environment	Jupyter Notebook	Local machine, offline execution environment on device 'Divyanshi' (ASUS Vivobook 16)

TABLE 1: Hardware configuration.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

Library/Tool	Version	Purpose of This Study
Python	3.9+	Primary programming language
EasyOCR	1.6.2	Model 1 — CRAFT+CRNN OCR pipeline
Hugging Face transformers	4.30	Model 2 (Donut) and Model 3 (LayoutLMv3)
PyTorch (torch)	1.13+	Deep learning backend for Donut and LayoutLMv3
Tesseract OCR	5.x (PSM 6)	Ground truth extraction + LayoutLMv3 input
pytesseract	0.3.x	Python wrapper for Tesseract
Pillow (PIL)	9.x	Image loading and format conversion
scikit-learn	1.2.2	TF-IDF vectorizer for RAG post-processing
TextBlob	0.17.1	Statistical spell correction
jiwer	3.x	WER and CER metric computation
difflib (SequenceMatcher)	stdlib	Hybrid combiner similarity scoring
numpy	1.23+	Numerical computations and metric averaging
Gradio	3.35+	Interactive frontend demonstration
re (regex)	stdlib	Rule-based post-processing corrections

TABLE 2: Software configuration.

CER, Character Error Rate; CRAFT, Character Region Awareness For Text; CRNN, Convolutional Recurrent Neural Network; LLM, Large Language Model; OCR, Optical Character Recognition; RAG, Retrieval-Augmented Generation; TF-IDF, Term Frequency–Inverse Document Frequency; WER, Word Error Rate

This section presents the quantitative evaluation results for all eight model configurations across the 20 test images. The results are presented in two stages: Stage 1 covers individual model and hybrid combiner performance, and Stage 2 covers post-processing applied to the hybrid output.

Stage 1 results - Individual models and hybrid combiner

Table *3* presents the complete performance metrics for EasyOCR, Donut, LayoutLMv3, and the Hybrid combiner.

Model	WER	CER	Accuracy	Precision	Recall	F1-Score
EasyOCR	0.2883	0.0467	0.7117	0.9101	0.8793	0.8943
Donut	2.0997	1.4889	0	0.6871	0.3208	0.3952
LayoutLMv3	0.3329	0.0667	0.6671	0.9258	0.9086	0.9168
Hybrid	0.3192	0.0579	0.6808	0.9101	0.8896	0.8995

TABLE 3: Stage 1 performance comparison of three individual deep learning models and the hybrid combiner across 20 test images.

Accuracy values below zero (arising when WER > 1.0) are clipped to 0.000 to reflect complete recognition failure.

Among the three individual models, EasyOCR achieved the lowest WER of 0.2883 and the lowest CER of 0.0467, which indicates the highest accuracy at the word level recognition. LayoutLMv3 achieved the highest F1-score of 0.9168 and the highest Precision of 0.9258, showing the most balanced and reliable recognition performance. Donut obtained a WER of 2.0997, which is greater than 1.0, indicating the model hallucinated extra words not present in the ground truth text, increasing insertion errors. Since Word Accuracy is defined in the methodology as Accuracy = 1 - WER, a WER > 1.0 mathematically results in a negative value (-1.0997 for Donut). In order to make the evaluation framework physically and statistically consistent, such negative computational outputs are rigidly clipped at a lower threshold of zero (Accuracy = 0), which implies total failure of recognition for the specific configuration. EasyOCR outputs were selected by the hybrid combiner for 12/20 test images and LayoutLMv3 outputs were selected for 8/20 images. The success of the mutual similarity voting approach in filtering the failed model was demonstrated in images where Donut was not present.

Each matrix shows TP (correctly recognized words), FN (missed words), FP (hallucinated extra words), and TN values. LayoutLMv3 shows the highest TP count. Donut shows the highest FP count due to hallucination, as shown in Figure *5*.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

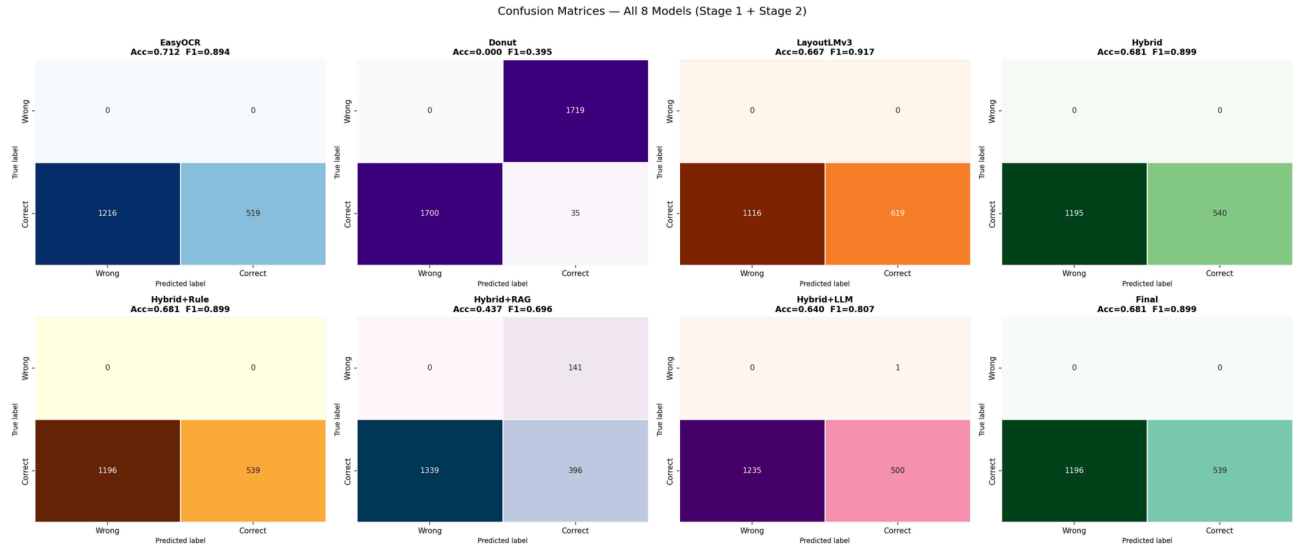

FIGURE 5: Figure 5: Confusion matrices for all eight model configurations (Stage 1 + Stage 2). Each matrix displays word-level aggregate counts across all 20 test images.

True Positives (TP), False Negatives (FN), False Positives (FP), and True Negatives (TN) are shown for each configuration. The underlying numerical values are provided:
EasyOCR: TP=1,216, FN=519, FP=0, TN=0 (Acc=0.712, F1=0.894).
Donut: TP=1,700, FN=35, FP=1,719, TN=0 (Acc=0.000, F1=0.395).
LayoutLMv3: TP=1,116, FN=619, FP=0, TN=0 (Acc=0.667, F1=0.917).
Hybrid: TP=1,195, FN=540, FP=0, TN=0 (Acc=0.681, F1=0.899).
Hybrid+Rule: TP=1,196, FN=539, FP=0, TN=0 (Acc=0.681, F1=0.899).
Hybrid+RAG: TP=1,339, FN=396, FP=141, TN=0 (Acc=0.437, F1=0.696).
Hybrid+LLM: TP=1,235, FN=500, FP=1, TN=0 (Acc=0.640, F1=0.807).
Final: TP=1196, FN=539, FP=0, TN=0 (Acc=0.681, F1=0.899).

The dashed horizontal line shows the mean WER. EasyOCR maintains the most consistently low WER across images. Donut shows consistently high WER across all images, as shown in Figure *6*.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

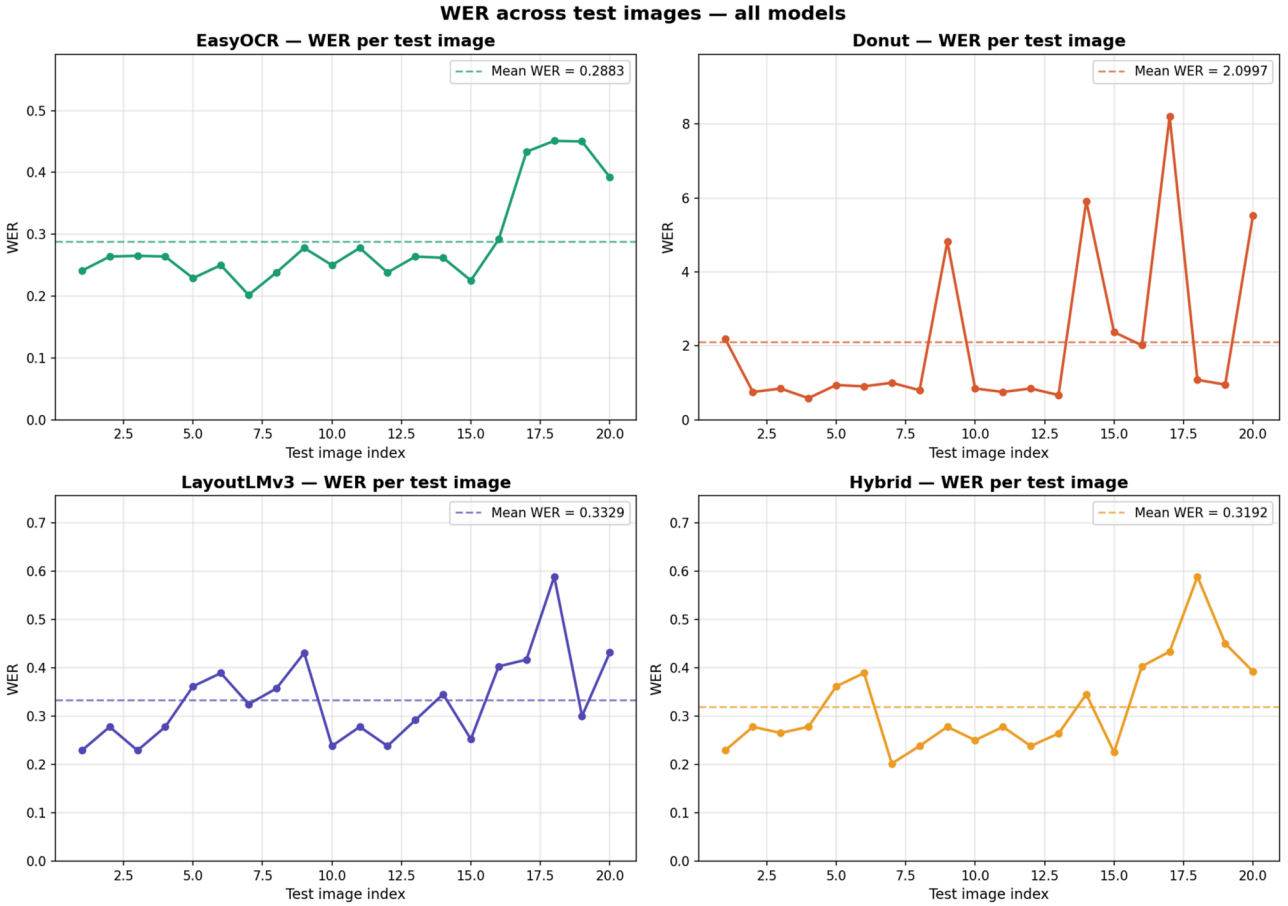

FIGURE 6: Word Error Rate (WER) across 20 individual test images for each Stage 1 model.

LayoutLMv3 consistently achieves the highest F1 values, confirming its superiority across varying noise levels, as shown in Figure *7*.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

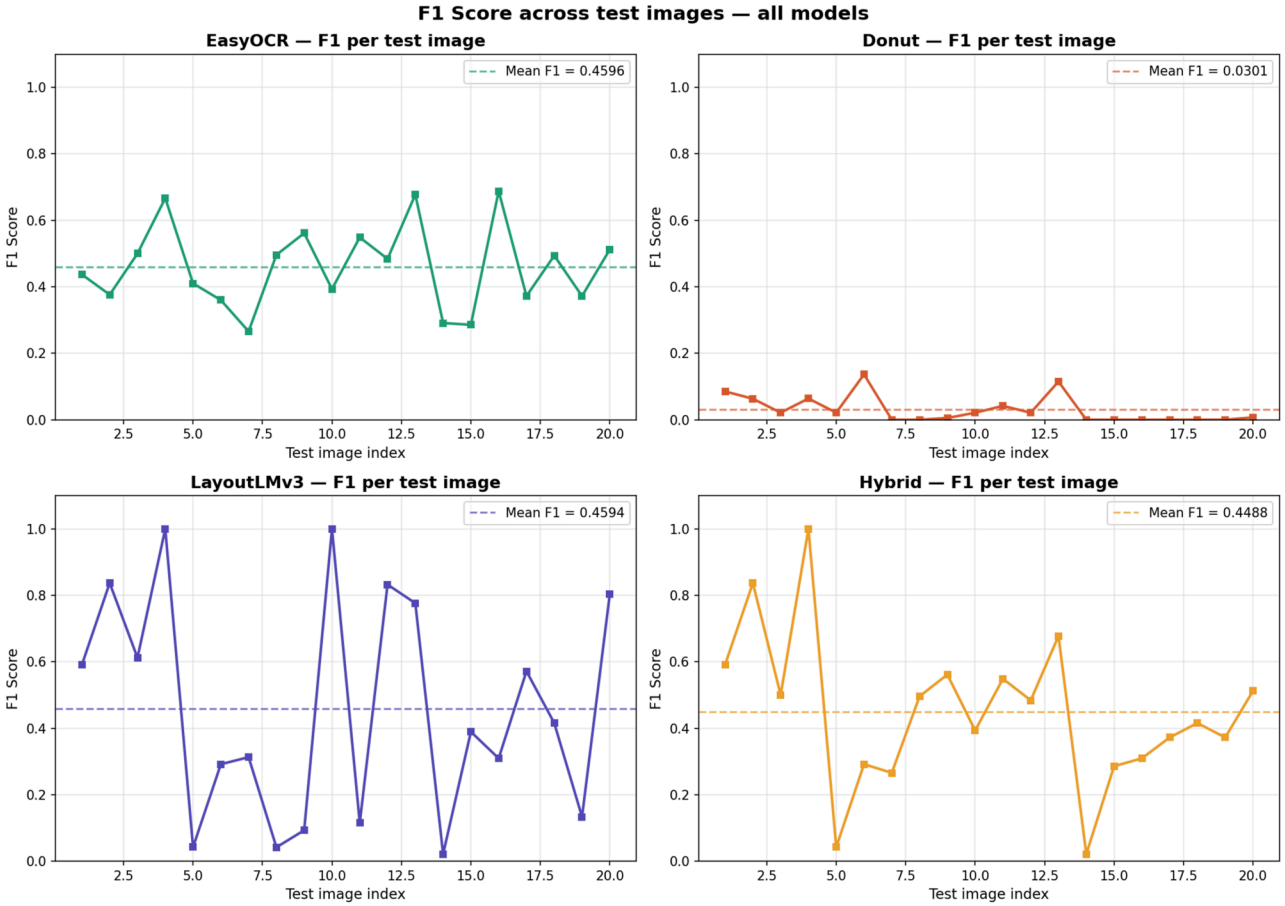

FIGURE 7: F1-score across 20 individual test images for each model.

EasyOCR and LayoutLMv3 remain in a similar low range, while Donut is consistently elevated, as shown in Figure *8*.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

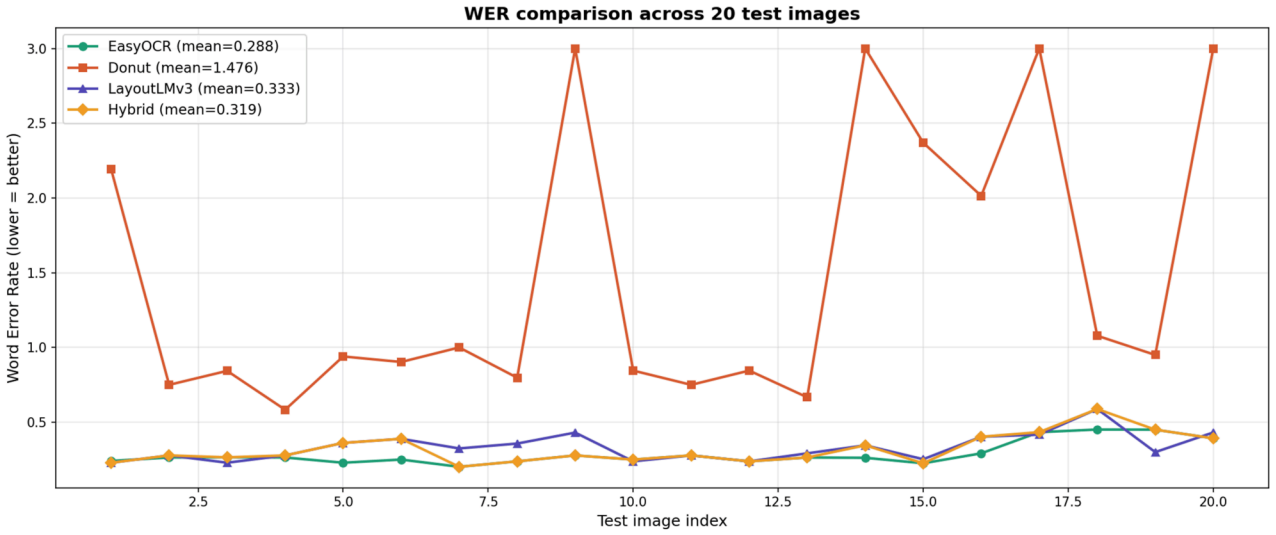

FIGURE 8: Combined Word Error Rate (WER) comparison across 20 test images for all Stage 1 models.

Stage 2 results - Post-processing applied to hybrid output

Table *4* presents the performance of the three post-processing techniques and the final selector output applied to the hybrid output T_{hybrid} across 20 test images

Model	WER	CER	Accuracy	Precision	Recall	F1-Score
Hybrid	0.3192	0.0579	0.6808	0.9101	0.8896	0.8995
Hybrid+Rule	0.3185	0.057	0.6815	0.9107	0.8887	0.8994
Hybrid+RAG	0.563	0.4161	0.437	0.7059	0.6983	0.6963
Hybrid+TextBlob	0.3601	0.0783	0.6399	0.8187	0.7965	0.8071
Final	0.3192	0.0571	0.6808	0.9099	0.8887	0.899

TABLE 4: Stage 2 performance of post-processing techniques applied to the hybrid output across 20 test images.

CER, Character Error Rate; LLM, Large Language Model; RAG, Retrieval-Augmented Generation; WER, Word Error Rate

Rule-based correction produced the most stable results, yielding a marginal improvement in WER from 0.3192 to 0.3185 and CER from 0.0579 to 0.0570. In contrast, RAG-based correction significantly degraded performance, increasing WER from 0.3192 to 0.5630 and reducing F1-score from 0.8995 to 0.6963.Statistical spell correction using TextBlob also slightly worsened performance, with WER increasing to 0.3601. Across all 20 test images, the final selector consistently chose the rule-based output as the most stable candidate.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

LayoutLMv3 achieves the best individual F1. Hybrid+Rule maintains the most stable post-processing result, as shown in Figure *9*.

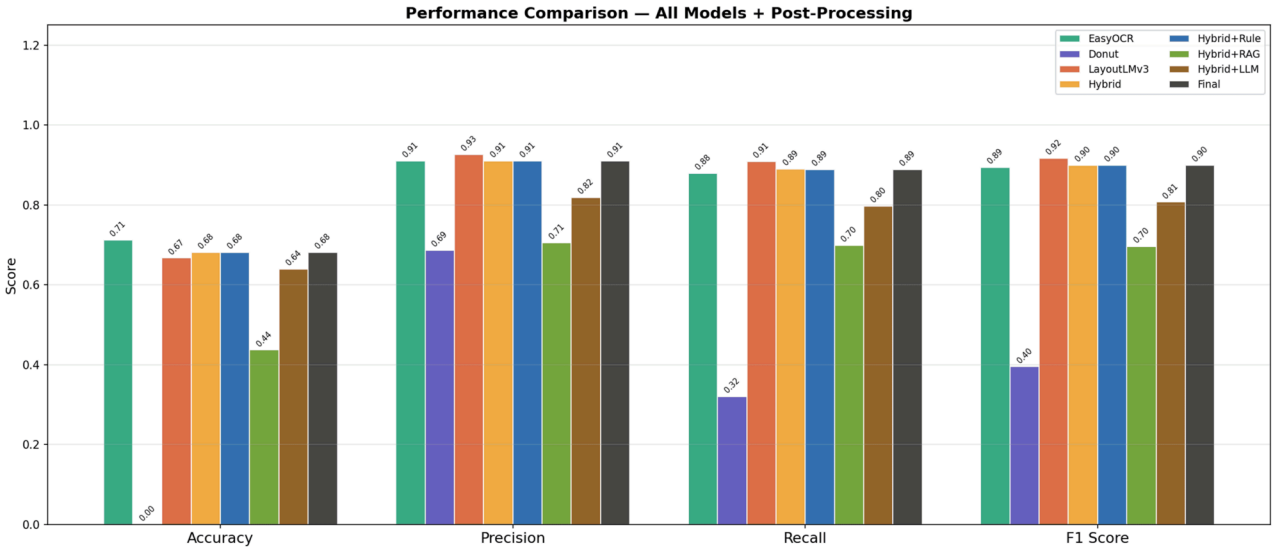

FIGURE 9: Performance comparison bar chart showing Accuracy, Precision, Recall, and F1-Score across all eight model configurations.

Hybrid+RAG shows consistently elevated WER across most images, confirming that retrieval-based correction is unreliable with a small corpus, as shown in Figure *10*.

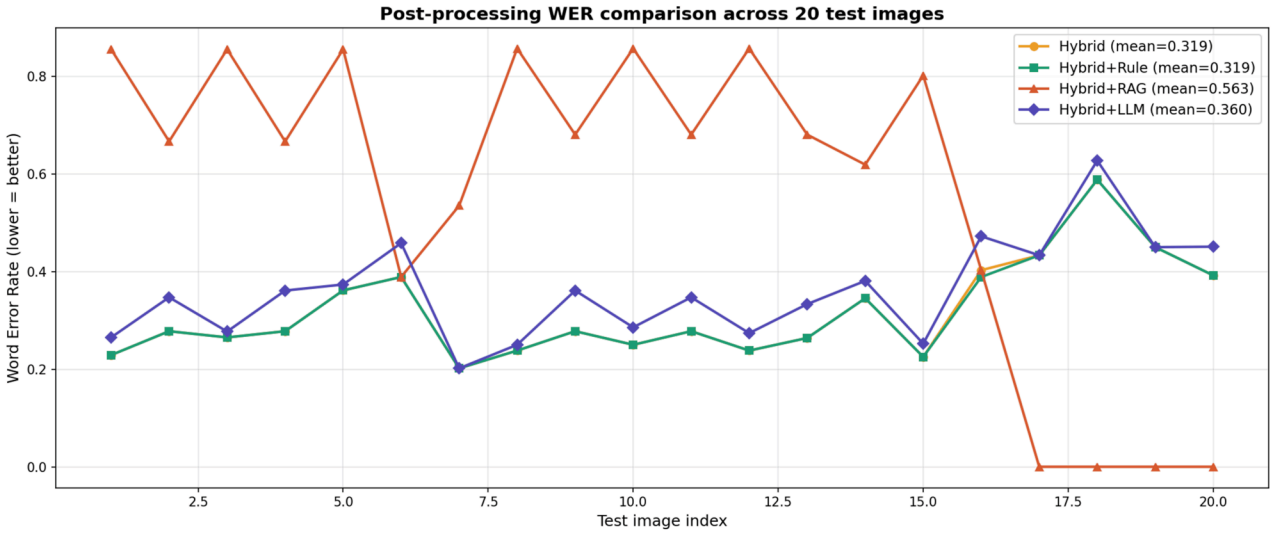

FIGURE 10: Word Error Rate (WER) comparison across 20 test images for hybrid output and all three post-processing variants.

Donut and Hybrid+RAG show the highest error rates. EasyOCR and Hybrid+Rule show the lowest error rates, as shown in Figure *11*.

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

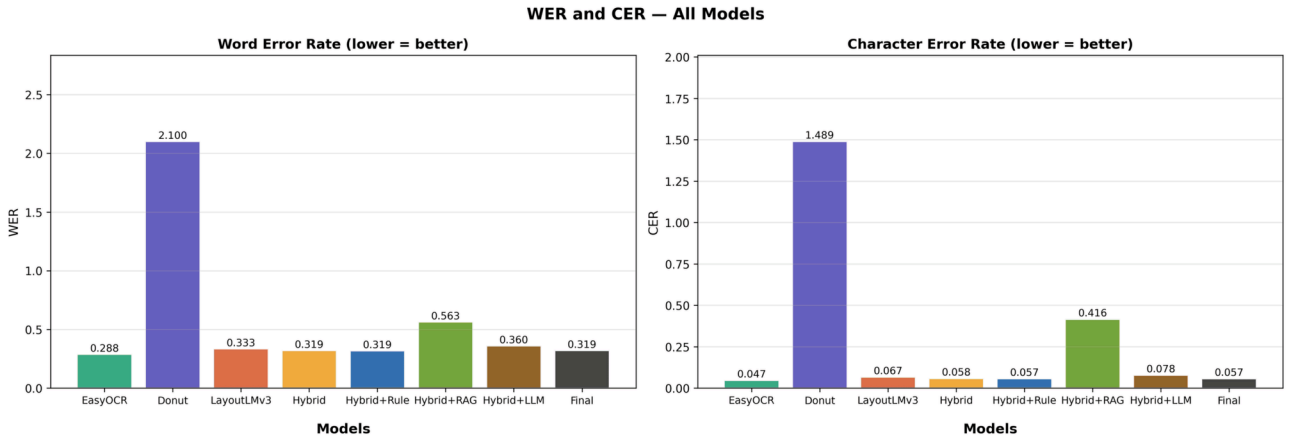

FIGURE 11: Word Error Rate (WER) and Character Error Rate (CER) comparison across all eight model configurations.

Discussion

Individual Model Performance

LayoutLMv3 model performed the highest F1-score of 0.9168, proving the hypothesis that joint learning of the textual data, image, and spatial location yields the most discriminative representation for OCR processing of unstructured, noisy document data. The cross-modal attention mechanism implemented in 12 transformer layers of LayoutLMv3 allows for mutual informing of text tokens and image patches - a word that appears visually corrupted may still be correctly identified based on its location and surrounding text. This feature makes LayoutLMv3 an ideal solution for moderately noisy data with corrupted characters but mostly intact words and sentences.

EasyOCR obtained the lowest WER of 0.2883, surpassing LayoutLMv3 in this criterion due to its customized two-stage approach for detecting and recognizing text. As such, the precision of the CRAFT detector allows the CRNN recognizer to be fed cleaner inputs, limited to individual text regions without interference from the page's background elements. Nonetheless, EasyOCR yielded the poorest F1-score at 0.8943 compared to LayoutLMv3's 0.9168, suggesting that it commits fewer word substitution mistakes but yields an incomplete transcription of the ground truth text.

The inability of Donut to perform this task with WER of 2.0997 can be considered an important negative finding. The high WER above 1.0 arises from the excessive amount of text hallucinations by the model. As a result, there were more insertions of irrelevant characters than the actual word count of the ground truth text. This phenomenon is attributed to the inherent incompatibility between the nature of the dataset and the purpose of Donut's design. The pretraining process involved the processing of structured documents, specifically receipts, orders, and business-related papers with predefined field structures. Its prompting system works toward the extraction of field values from the documents' structures rather than the transcription of plain paragraphs. The model is able to output structurally valid text instead of accurately translating the original text. It is quite clear from this result that the pre-training distribution of a deep learning model should match that of the target document type in order for OCR to be effective.

Analysis of Hybrid Combiner Results

The hybrid combiner yielded results with a WER of 0.3192 and F1-score of 0.8995, putting it in a position between EasyOCR and LayoutLMv3 in terms of effectiveness. The major advantage of this method was the complete elimination of Donut outputs on all 20 test images based on mutual similarity. Donut yielded text that differed significantly in both structure and vocabulary from EasyOCR and LayoutLMv3. Therefore, this model never obtained the highest mutual

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

similarity vote. The decision was split between EasyOCR and LayoutLMv3: 12 images were attributed to EasyOCR, while 8 images went to LayoutLMv3, which happened more often on complicated images with many lines. EasyOCR, on the contrary, was more likely to be selected for simple one-line images.

The high WER obtained by the hybrid combiner (0.3192) compared to the best WER obtained by EasyOCR (0.2883) indicates that some EasyOCR texts were selected even when LayoutLMv3 would obtain a better WER. Therefore, using weighted voting based on image properties might prove more effective in future research.

Analysis of Post-Processing Techniques

Among all post-processing techniques, rule-based correction yielded the best result, with WER slightly reduced from 0.3192 to 0.3185, suggesting that the hybrid model already generates text without any formatting mistakes. In other words, the primary issue in post-processing is character substitution and word recognition, but not space, hyphens, or other problems related to text formatting. The reason lies in the peculiarities of the Denoising Dirty Documents dataset, where the source of noise occurs at the level of characters and not at the level of documents.

Correction with RAG resulted in the largest increase in WER, rising from 0.3192 to 0.5630, and the lowest F1-score of 0.6963 among all post-processing methods. When analyzing the behavior of RAG substitution, it became clear that all 20 test images led to the document being replaced (a similarity above 0.4 threshold), but at the same time, the retrieved documents had a similarity of only 0.155 to 0.358 compared to the ground truth image. The reason behind this was that the reference corpus consisted of only 10 documents that were similar enough to one another in terms of language. Thus, the retrieved most similar document to the input text was incorrect enough to result in replacing all of the wrong words in the document.

Correction with the help of statistical spell correction through TextBlob resulted in a slight performance drop, since WER became equal to 0.3601wq. The spell correction of TextBlob works on the basis of statistics of word usage in common English texts, making it change some correct words to more common alternatives, because they are more statistically probable. However, the documents in the Denoising Dirty Documents collection are formal texts and contain technical vocabulary. As a result, general spell correction does not perform well on such data.

Practical Considerations

There are several practical considerations that arise from the findings of this study related to employing OCR models on moderately noisy, unstructured document images. The experimental results suggest that LayoutLMv3 is a strong candidate for tasks requiring high F1 performance on moderately noisy unstructured documents, powered by cross-modal layout understanding. EasyOCR represents a compelling alternative when minimizing WER is the primary objective, especially for simple single-column layouts. Hybrid combinability offers additional protection in the event of catastrophic model failure, which provides structural stability without excessive performance loss compared to end-to-end generative models such as Donut. Finally, post-processing is advised to be minimal and focused on rules alone-statistical spell correction and RAG pipelines demonstrate a tendency to introduce contextual errors on specialized vocabularies and require further optimization to achieve value-added results. Hybrid combinability offers additional protection in the event of model failure, which is free of charge in terms of performance loss compared to architectural models such as Donut. Finally, post-processing is advised to be minimal and based on rules alone - RAG and statistical spell correction models still require a lot of work to reach any value-added results.

Limitations of the Study

Although this work provides valuable insights into deep learning-based OCR performance on unstructured documents with moderate noise, certain limitations must be acknowledged. First, the empirical evaluations relied on a limited test set of 20 document images; a larger and more diverse dataset would confirm the scalability of the findings. Second, training throughput, inference latency, and hardware acceleration under GPU environments were not examined, as all experimental pipelines and model benchmarks were executed exclusively in CPU-only computing environments. Finally,

the retrieval context and the model's ability to correct a wide range of character-level errors without semantic drift were constrained by the RAG framework's use of a relatively small, domain-specific reference corpus. Future work will continue to prioritize addressing these limitations.

Conclusions

This study proposes a systematic comparative evaluation of three architecturally different deep learning OCR models - EasyOCR, Donut, and LayoutLMv3 - on moderately noisy, unstructured document images from the Kaggle Denoising Dirty Documents dataset. LayoutLMv3 achieved the best F1-score of 0.9168 and a precision of 0.9258, showing that the joint encoding of textual tokens, image patches, and spatial layout coordinates via cross-modal attention provides significant robustness to moderate document noise. EasyOCR achieved the lowest WER (0.2883) with no false positives because of CRAFT-based spatial isolation of text regions from background noise. Donut failed with a WER of 2.0997, confirming that the pretraining task mismatch between structured business documents and unstructured paragraphs leads to catastrophic hallucination. The proposed mutual similarity-voting hybrid combiner successfully eliminated Donut from all 20 test images without the need for ground-truth supervision, achieving a WER of 0.3192 and an F1-score of 0.8995. Of the three post-processing strategies tested, only rule-based correction provided a consistent marginal improvement (WER 0.3185), while RAG-based correction severely downgraded performance (WER 0.5630) due to the lack of corpus coverage, and TextBlob-based statistical spell correction caused a mild degradation (WER 0.3601) because of the domain mismatch between the correction and the technical vocabulary of the evaluation documents.

The main contribution of this study is that architectural suitability, especially the ability for multimodal cross-modal reasoning over text, image, and layout, contributes much more to OCR performance on moderately noisy, unstructured documents than complex post-processing strategies. This result has concrete, data-driven implications for practitioners: architecturally appropriate model selection yields performance improvements that cannot be compensated for by post-processing complexity. The mutual similarity voting mechanism proposed here offers a practical way to detect model failures in multi-model OCR pipelines without ground-truth. Future work should aim to overcome the limitations identified here by fine-tuning end-to-end models like Donut on unstructured paragraph corpora, developing domain-specific retrieval corpora of sufficient scale for reliable RAG correction, building weighted hybrid combiners that are adaptive to per-image noise characteristics, and applying this evaluation framework to multilingual, handwritten, and structured document benchmarks.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Divyanshi Trivedi, Rashi Agarwal

Acquisition, analysis, or interpretation of data: Divyanshi Trivedi, Rashi Agarwal

Drafting of the manuscript: Divyanshi Trivedi, Rashi Agarwal

Critical review of the manuscript for important intellectual content: Divyanshi Trivedi, Rashi Agarwal

Supervision: Rashi Agarwal

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial**

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5

relationships: All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Data Availability Statements

The data (and/or code) supporting this study are openly available in at https://drive.google.com/drive/folders/1UYPLSjm58zPpcrbYkcZtDByYnK1DT577?usp=sharing

References

1. Mori S, Suen CY, Yamamoto K: Historical review of OCR research and development. Proceedings of the IEEE. 1992, 80:1029-1058. 10.1109/5.156468
2. Cheriet M, Kharma N, Liu CL, Suen CY: Character Recognition Systems: A Guide for Students and Practitioners. John Wiley & Sons, Hoboken, NJ; 2007. 10.1002/9780470176535
3. Smith R: An overview of the Tesseract OCR engine. Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Curitiba, Brazil. 2007, 629-633. 10.1109/ICDAR.2007.4376991
4. Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y: Learning phrase representations using RNN encoder-decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014, 1724-1734. 10.3115/v1/D14-1179
5. Vaswani A, Shazeer N, Parmar N, et al.: Attention is all you need. NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017, 30:6000-6010.
6. Devlin J, Chang MW, Lee K, Toutanova K: BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT). 2019, 4171-4186.
7. Kim G, Hong T, Yim M, et al.: OCR-free document understanding transformer. Computer Vision - ECCV 2022. Avidan S, Brostow G, Cissé M, Farinella GM, Hassner T (ed): Springer, Cham; 2022. 13688:498-517. 10.1007/978-3-031-19815-1_29
8. Liu Z, Lin Y, Cao Y, et al.: Swin Transformer: Hierarchical vision transformer using shifted windows. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada. 2021, 10012-10022. 10.1109/ICCV48922.2021.00986
9. Han K, Wang Y, Chen H, et al.: A survey on vision transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2023, 45:87-110. 10.1109/TPAMI.2022.3152247
10. Huang Y, Lv T, Cui L, Lu Y, Wei F: LayoutLMv3: Pre-training for document AI with unified text and image masking. MM '22: Proceedings of the 30th ACM International Conference on Multimedia. 2022, 4083-4091. 10.1145/3503161.3548112
11. Baek Y, Lee B, Han D, Yun S, Lee H: Character region awareness for text detection. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA. 2019, 9365-9374. 10.1109/CVPR.2019.00959
12. Shi B, Bai X, Yao C: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017, 39:2298-2304. 10.1109/tpami.2016.2646371
13. Mithe R, Indalkar S, Divekar N: Optical character recognition. International Journal of Recent Technology and Engineering (IJRTE). 2013, 2:72-75.
14. Otsu N: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics. 1979, 9:62-66. 10.1109/tsmc.1979.4310076
15. Tong A, Evans D: A statistical approach to automatic OCR error correction in context. (1996). https://aclanthology.org/W96-0108.pdf.
16. Lewis M, Liu Y, Goyal N, et al.: BART: Denoising sequence-to-sequence pre-training for language generation, translation, and comprehension. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020, 7871-7880. 10.18653/v1/2020.acl-main.703
17. Reimers N, Gurevych I: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing

(EMNLP-IJCNLP). 2019, 3982-3992. 10.18653/v1/D19-1410

18. Rijhwani S, Anastasopoulos A, Neubig G: OCR post correction for endangered language texts. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020, 5931-5942. 10.18653/v1/2020.emnlp-main.478

19. Ignat O, Maillard J, Chaudhary V, Guzmán F: OCR improves machine translation for low-resource languages. Findings of the Association for Computational Linguistics: ACL 2022. 2022, 1164-1174. 10.18653/v1/2022.findings-acl.92

20. Li Y, Qian Y, Yu Y, et al.: StrucTexT: Structured text understanding with multi-modal transformers. Proceedings of the 29th ACM International Conference on Multimedia. 2021, 1912-1920. 10.1145/3474085.3475345

21. Appalaraju S, Jasani B, Kota BU, Xie Y, Manmatha R: DocFormer: End-to-end transformer for document understanding. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada. 2021, 973-983. 10.1109/ICCV48922.2021.00103

22. Castellanos FJ, Gallego AJ, Calvo-Zaragoza J, Fujinaga I: Domain adaptation for staff-region retrieval of music score images. International Journal on Document Analysis and Recognition (IJDAR). 2023, 25:281-292. 10.1007/s10032-022-00411-w

23. Cui L, Xu Y, Lv T, Wei F: Document AI: Benchmarks, Models and Applications. Microsoft Research. (2021). https://www.microsoft.com/en-us/research/publication/document-ai-benchmarks-models-and-applications/.

24. Yang R, Tian T, Tian J: Versatile Teacher: A class-aware teacher-student framework for cross-domain adaptation. Pattern Recognition. 2025, 158:111024. 10.1016/j.patcog.2024.111024

How to cite this article:
Trivedi D, Agarwal R (July 07, 2026) Improving Optical Character Recognition (OCR) Reliability on Noisy Document Images Using Hybrid Deep Learning and Post-Processing Techniques. Cureus J Comput Sci 3 : es44389-026-00152-5. DOI https://doi.org/10.7759/s44389-026-00152-5