Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 8s (2025)

Automated LSTM Based Deep Learning Model for Handwritten Telugu Answer Script Analysis

Padmavathi Pragada[1, a], Dhawaleswar Rao CH*[2, b]

[1,2]Centurion University of Technology and Management, R. Sitapur, Odisha

[1]GMR Institute of Technology, Rajam

[a] pragadapadmavathi88@gmail.com

[b] dhawaleswarrao@gmail.com

Article History:

***Received:** 10-11-2024*

***Revised:**15-12-2024*

***Accepted:**03-01-2025*

Abstract: The growing demand for automated evaluation systems in educational environments, especially for languages with complex scripts like Telugu, drives the motivation for this research. Traditional handwriting recognition methods for Telugu have faced challenges with limited accuracy and adaptability, particularly in real-world educational scenarios. These limitations often result in reduced precision in character and sentence recognition, along with increased processing delays. This study proposes a novel system for the automated evaluation of handwritten Telugu answer scripts. The model incorporates advanced preprocessing techniques such as adaptive thresholding for binarization and Gaussian blurring for noise reduction, enhancing the readability of diverse handwriting styles. Robust feature extraction is achieved using Convolutional Neural Networks (CNNs) like ResNet101 and Inception networks. To capture the contextual flow of Telugu scripts, Quad Long Short-Term Memory (LSTM) networks are utilized, with Attention Mechanisms improving focus on intricate character sequences. Additionally, Transformer-based models like BERT, trained on Telugu text, enable the system to better understand the syntax and semantics of the language. For evaluation, Visual BERT embeddings and cosine similarity metrics are employed to ensure precise semantic analysis and answer matching. Testing across multiple datasets demonstrates a significant improvement over existing approaches, with higher precision and accuracy in character recognition and notable enhancements in area under the curve (AUC). Sentence recognition also shows marked improvements in precision, accuracy, and AUC. This work represents a significant advancement in automated evaluation systems for languages with intricate scripts, improving both efficiency and accuracy in educational assessments. Beyond its primary application, the system offers potential for broader uses in document processing and language technologies. This research makes a valuable contribution to the fields of automated handwriting recognition and natural language processing for Indic scripts.

Keywords: Handwriting Recognition, Telugu Language Processing, Convolutional Neural Networks, Sequence-to-Sequence Learning, Semantic Analysis

I. INTRODUCTION

The evolution of automated systems for educational assessment has garnered significant research attention, particularly for languages with complex scripts like Telugu, spoken by millions in India. Telugu poses unique challenges due to its intricate script and expansive character set, which hinder the accuracy, adaptability, and efficiency of traditional handwriting evaluation methods. These limitations underscore the need for advanced approaches to automate and enhance the evaluation of handwritten Telugu documents. Developing an automated system for Telugu handwriting evaluation not only promises efficiency and consistency in educational assessments but also has broader implications for advancing language processing technologies. Such a system can substantially reduce the time and effort required for manual grading while eliminating human bias, thereby ensuring objective and consistent results. Existing approaches to Telugu handwriting recognition and language processing have largely relied on traditional techniques that often struggle with low precision and extended processing times. These methods lack robustness in handling diverse handwriting styles and fail to capture the nuanced semantic content of written responses. Addressing these challenges requires a more advanced and adaptable solution capable of accurately interpreting the variability of Telugu script and its written content. To address these issues, this paper presents a comprehensive system for automating the evaluation of handwritten Telugu answer scripts. The proposed system integrates sophisticated image processing techniques, including adaptive thresholding for binarization and Gaussian blurring for noise reduction, to enhance text clarity and readability. Data augmentation methods, such as elastic distortions and affine transformations, further improve the model's ability to handle diverse handwriting styles. At the core of the system are Convolutional Neural Networks (CNNs) paired with Quad Long Short-Term Memory (LSTM) networks, optimized for feature extraction and contextual understanding of Telugu script. A Bahdanau Attention mechanism further refines the model's accuracy by focusing on nuanced character sequences. This handwriting recognition framework is complemented by a natural language processing module that leverages Transformer-based models, including BERT trained on Telugu text. This module captures the syntactic and semantic intricacies of the language, which are essential for accurate answer evaluation. Extensive testing on varied datasets has demonstrated the system's superior performance over existing methods, with significant improvements in precision, accuracy, and processing speed for both character and sentence recognition in Telugu. This paper discusses the system's design, implementation, and evaluation, showcasing its transformative potential in automated handwriting recognition and language processing, particularly for Indic scripts. This adaptable and efficient system represents a substantial advancement in automated educational assessment, offering a model that could be extended to other complex languages and scripts.

1.1 Motivation

This research is driven by the urgent need to improve the efficiency and accuracy of educational assessments, particularly for languages with intricate scripts like Telugu. In many educational institutions, evaluation processes remain manual, time-consuming, and prone to human error and bias, leading to inconsistencies in grading. These challenges are exacerbated in the case of Telugu due to its complex character set and script structure, making manual evaluation especially difficult and prone to variability. Automating the assessment process for Telugu offers an opportunity to enhance the speed and objectivity of evaluations while ensuring a fairer and more consistent grading system. Additionally, with the growing emphasis on digital solutions in education, an automated Telugu handwriting recognition system could facilitate the digitization of exams, streamlining workflows and simplifying document management. Beyond the immediate benefits in educational assessment, the development of a robust automated system for Telugu handwriting recognition addresses a critical gap in language processing technology for complex scripts. As one of the most widely spoken languages in India, Telugu represents a substantial user base, underscoring the significance of this research not only in education but also in advancing digital document processing and language understanding technologies.

1.2 Novelty of this work

This research introduces a Quad LSTM-based deep learning model tailored for handwritten Telugu answer script analysis, addressing the challenges posed by complex scripts with high context dependencies. It integrates advanced architectures such as Quad LSTMs with Bahdanau Attention Mechanisms and CNNs (ResNet101, Inception) for superior feature extraction and sequence understanding. The novelty lies in leveraging Transformer-based models like BERT, fine-tuned specifically for Telugu, to enhance semantic analysis for subjective assessments. Preprocessing techniques, including adaptive thresholding and data augmentation, ensure robustness across diverse handwriting styles. This approach significantly outperforms traditional systems, setting a new benchmark in automated educational assessments for Indic scripts.

II. RELATED WORK

Advancements in Optical Character Recognition (OCR) have been transformative across various domains, including handwritten character recognition, scene text detection, and intelligent transportation applications. Yavariabdi et al. presented the CArDIS dataset to enhance OCR for Swedish historical handwritten characters [1]. Rasheed et al. utilized transfer learning and augmentation with AlexNet for Urdu characters and digits, emphasizing the need for tailored OCR systems for underrepresented languages [2]. Buoy et al. developed a low-resource baseline for Khmer OCR, addressing linguistic preservation challenges [3]. Techniques by Li et al. emphasized character-aware sampling for scene text recognition in uncontrolled environments [4]. Zhang et al. and Wang et al. demonstrated the breadth of OCR's utility in industrial applications [5][6]. Coquenet et al. introduced a vertical attention network for recognizing long paragraph handwritten text [7], while Wu et al. and Xue et al. employed novel architectures to enhance scene text recognition [8][11]. Deng et al. focused on end-to-end tag recognition for industrial meters, showcasing OCR's applied potential [9]. R. D. R et al. proposed a hybrid approach for ancient Tamil character recognition, underscoring OCR's cultural importance [10]. Hussain et al. presented the PHTI image base for Pashto, and Ott et al. investigated cross-modal representation learning [12][13]. Selvam et al. and Fan and Zhao introduced transformer-based frameworks and methods for real-world license plate recognition, respectively [14][15]. Tayyab et al. and Malhotra & Addis concentrated on real-time Arabic scripting recognition and historical Ethiopic text digitization [16][17].

Fanjie et al. created key datasets for Uyghur scene text recognition, while Zhu et al. proposed an OCR-RCNN framework for elevator button recognition, highlighting practical OCR applications [18][19]. Anbukkarasi et al. advanced text detection across various media, and Chandio et al. tackled cursive text recognition using deep learning [20][21]. Yang et al. demonstrated OCR's potential in security contexts through optical transmitter fingerprint authentication [22]. Improvements in license plate recognition systems by Shi and Zhao, among others, reinforced OCR's pivotal role in transportation [23][24][25]. Nguyen et al. addressed OCR error correction in Vietnamese texts [26]. Jang et al. focused on material identification in industrial contexts [27]. Fu et al. investigated multiple event recognition in fiber optic systems [28], and Seo & Kang worked on layout-independent detection [29]. Gao et al. developed the Group Plate system for multi-category license plate recognition [30], while Chen et al. prioritized cross-lingual text recognition accuracy [31]. Li et al. introduced a dual relation network to improve recognition [32]. Tan et al. presented an air-writing recognition method using Transformers [33]. Comprehensive surveys by Rahman et al. focused on Bengali handwritten digit recognition [34]. Sharmila et al. developed a convolutional neural network-based method for recognizing handwritten Tamil characters [35], while Karthikeyan et al. highlighted OCR accuracy's importance in medical applications [36]. Kumar et al. proposed an innovative approach for recognizing cursive Hindi scripts [37], and Vinotheni & Pandian contributed a model for Tamil handwritten document recognition [38]. Mrinalini et al. introduced a similarity-based evaluation metric for Indian languages [39]. P.P et al. [40] proposed OCR and text to speech recognition. An overview of character recognition studies is provided in Table-1.

Table 1: Overview of Character Recognition Studies

REFERENCE	FOCUS AREA	METHODOLOGY	LIMITATIONS
Rasheed et al. [2]	Urdu Character Recognition	Transfer learning with Alex Net	Potential overfitting due to small dataset size.
R. D. R et al. [10]	Ancient Tamil Character Recognition	Hybrid approaches	Complexity of ancient scripts can hinder accuracy.
Tayyab et al. [16]	Arabic Script Recognition	Real-time processing	Complexity of Arabic script fonts can affect performance.
Chandio et al. [21]	Cursive Text Recognition	Deep learning	Cursive variations can complicate recognition.
Sharmila et al. [35]	Handwritten Tamil Character Recognition	Convolutional Neural Networks	Requires large datasets for practical applications.
Yavariabdi et al. [1]	Historical Handwritten OCR	CArDIS dataset	Limited to Swedish language; may not generalize well.
Buoy et al. [3]	Khmer OCR	Low-resource baseline	Limited data availability for Khmer language OCR.
Malhotra and Addis [17]	Historical Ethiopic Text Digitization	Deep learning techniques	Difficulties in obtaining comprehensive historical data.
Vinotheni and Pandian [38]	Tamil Document Recognition	Specialized recognition systems	Limited data availability restricts OCR enhancements.
Mrinalini et al. [39]	Indian Language Translation	Similarity-based evaluation metric	May not accommodate all regional dialects.

III. PROPOSED DESIGN

To address the challenges of lower efficiency and higher complexity in current OCR models, this section presents the design of an efficient Automated Quad LSTM-based Deep Learning Language Processing Model for analysing handwritten Telugu answer scripts. As illustrated in Figure 1, the proposed model includes the following phases: Image processing, Data Augmentation, feature extraction process, Quad Long Short-Term Memory (LSTM) networks, Bahdanau Attention mechanism, Visual BERT. In continuation, Figure 2 gives an overview of Internal Architectural Details of the Proposed Model for OCR Analysis. This comprehensive approach not only demonstrates remarkable improvements in key performance metrics such as precision and accuracy but also significantly reduces processing delays. The overall proposed flow is depicted in the following algorithm 1 - Handwriting Recognition and similarity scores using Quad LSTM. As a result, the proposed model stands out as a beacon of innovation in OCR technology for complex scripts.

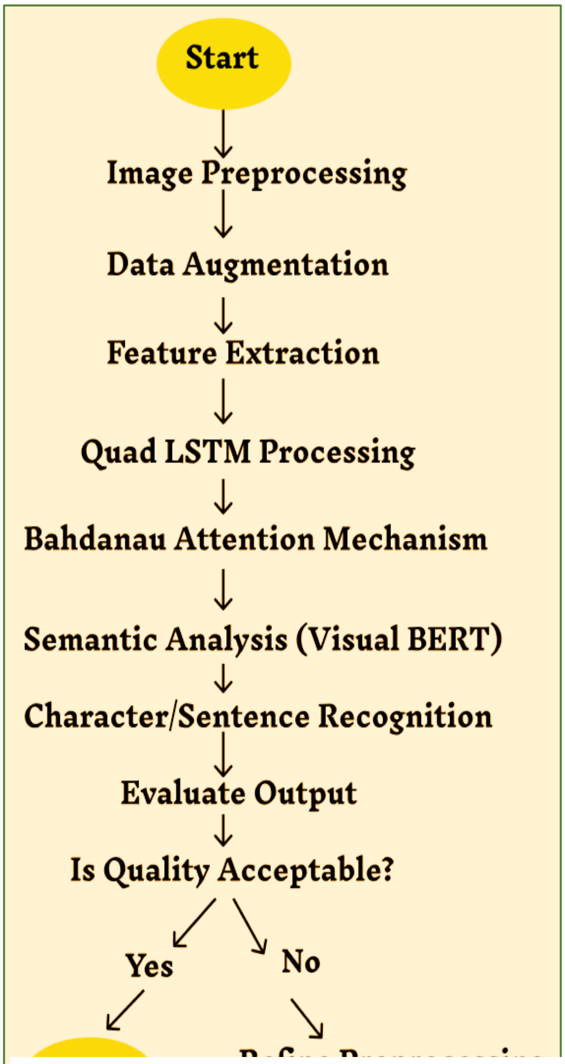

***Figure 1**. Overall flow of the Proposed Model for OCR Analysis*

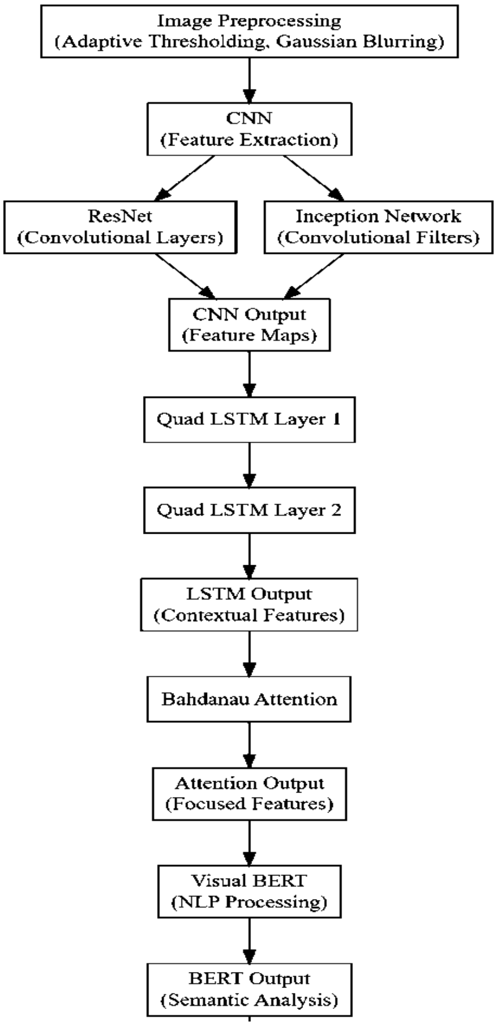

***Figure 2**. Internal Architectural Details of the Proposed Model for OCR Analysis*

Algorithm 1: Handwriting Recognition and similarity scores using Quad LSTM

Input: Images of handwritten text.
Output: Recognized text and similarity scores.
Procedure:

1. Initialize the environment:
   Import libraries: cv2, numpy, keras, transformers, etc.
   Define image paths.
2. Process each image:
   For each image in the paths:
      Load and preprocess:
      - Convert to grayscale.
      - Apply adaptive thresholding and Gaussian blur.

      Detect text regions:
      - Identify bounding boxes

3. For each bounding box:
   Crop the image to isolate the text region.
   Apply data augmentation techniques.
4. Build the recognition model:
   CNN Backbone: Use ResNet101 or InceptionV3 for feature extraction.
   Quad LSTM:
   - Stack four Bidirectional LSTM layers (with attention mechanism).
   - Add output layer with Softmax activation.
5. Recognize text:
   Feed cropped images into the CNN.
   Pass features to the Quad LSTM for text recognition.
6. Post-process results:
   Tokenize recognized text with BERT tokenizer.
   Obtain optional BERT embeddings.
7. Evaluate similarity:
   Load answer embeddings.
   Calculate cosine similarity against recognized text embeddings.

End Procedure.

A. Image Preprocessing

The initial phase of image preprocessing is critical in enhancing the quality of handwritten Telugu text images and comprises adaptive thresholding and Gaussian blurring operations. Adaptive thresholding converts grayscale images to binary format, improving contrast between text and background. The threshold value, $T(x, y)$, is dynamically calculated via equation 1,

$$T(x,y) = \mu(x,y) - C \quad (1)$$

where $\mu(x, y)$ represents the mean intensity in a neighborhood around the pixel at (x, y), and C is a constant. The mean intensity is computed via equation 2,

$$\mu(x,y) = \frac{1}{W^2} \sum_{i=-\frac{W-1}{2}}^{\frac{W-1}{2}} \sum_{j=-\frac{W-1}{2}}^{\frac{W-1}{2}} I(x+i, y+j) \ldots (2)$$

Where, W is the size of the neighborhood (block size), and i(x+i, y+j) is the intensity of the pixel at coordinates (x+i, y+j) for different scans. Following adaptive thresholding, the model applies Gaussian blurring to reduce noise and smoothen the image samples. Gaussian blurring is implemented using a Gaussian filter, via equation 3,

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \ldots (3)$$

Where, $G(x, y)$ represents the Gaussian function, where σ is the standard deviation of the Gaussian distribution, and x, y are the distances from the origin in the horizontal and vertical axes. The blurring effect is achieved by convolving the Gaussian filter with the image, as described via equation 4,

$$B(x,y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} G(i,j) \cdot I(x+i, y+j) \ldots (4)$$

Where, $B(x, y)$ is the blurred image, k is the kernel radius, and i(x+i, y+j) is the intensity of the pixel at coordinates (x+i, y+j) for the image scans. The combination of adaptive thresholding and Gaussian blurring results in an enhanced binary image, which is instrumental for the feature extraction stages. The samples displayed in Figure

3 provide an overview of the image preprocessing techniques, with original image as (a), After adaptive thresholding as (b), After Gaussian Blurring- the preprocessed image (c).

Figure 3. (a) *Original Image* ***(b)*** *After Adaptive Thresholding* ***(c)*** *After Gaussian Blurring*

B. Data Augmentation

Prior to data augmentation, segmentation and recognition are applied to identify text regions and bounding boxes. These bounding boxes are drawn on the preprocessed image, as shown in Figure 4(a), and visualized in Figure 4(b). In the subsequent phase, data augmentation plays a crucial role in text recognition systems by enhancing model robustness through the generation of varied training samples. Techniques such as elastic distortion and affine transformations simulate differences in handwriting styles and text orientations, as depicted after segmentation in Figures 4(c) and 4(d). This strategy improves generalization, reduces overfitting, and allows models to perform more effectively in real-world scenarios. Implementing data augmentation significantly boosts the accuracy and reliability of text recognition applications. The preprocessed image was selected for segmentation using bounding boxes, followed by the application of data augmentation techniques. The results of this process are illustrated in Figure 4.

C. Feature Extraction

The preprocessing phase and augmentation phase ensures robust feature extraction by addressing the variability in handwritten texts. These pre-processed images are further analyzed using an efficient fusion of Convolutional Neural Networks (CNNs) with ResNet101 and Inception Net, which plays a pivotal role in deciphering the complexities of the Telugu scripts. This fusion is designed to capture the script's nuances, distinguishing between the finely varied strokes and shapes of characters and words. ResNet101 uses residual blocks defined via equation 5:

$$F(x,\{Wi\}) + x = y \dots (5)$$

while Inception Net combines filters of varying sizes:

$$I(y) = \sum_{i=1}^{n} Hi(x) \dots (6)$$

Their fusion is expressed as:

$$M(x) = R(x) + I(x) \dots (7)$$

The segmented image, after undergoing data augmentation, has been subjected to feature extraction using ResNet50, as shown in figure-4(e). Top 3 predictions from ResNet50 model:

- menu with probability 0.87
- binder with probability 0.06
- envelope with probability 0.03

Figure 4. (a) Preprocessed Image **(b)** Segmented Image with Bounding Boxes **(c)** Elastic Distortion **(d)** Affine Transformation (Data Augmentation functions) **(e)** Feature Extraction

D. Quad LSTM Processing

From the Internal Architectural Details shown in Figure-2, the proposed model leverages a QuadLSTM architecture to capture contextual information by processing the segmented characters in four orientations: forward, backward, top-down, and bottom-up. The QuadLSTM's forward LSTM is tasked with processing the segmented characters in their natural reading order, akin to how a person reads text. The LSTM layers are applied in multiple directions (forward, backward, top-down, bottom-up), as described below. These layers process the reshaped input and produce sequences as outputs.

This is mathematically represented via equation 8, where $h(t,f)$ ais the hidden state at time t for the forward LSTM via equation 8,

$$h(t,f) = LSTM(h(t-1,f), x(t,f)) \ldots (8)$$

In contrast, the backward LSTM processes the characters in reverse, providing aa retrospective context to each character via equation 9,

$$h(t,b) = LSTM(h(t+1,b), x(t,b)) \ldots (9)$$

The top-down LSTM interprets the characters from the top of the page to the bottom, mimicking a vertical reading pattern via equation 10,

$$h(t, td) = LSTM(h(t-1, td), x(t, td)) \ldots (10)$$

Conversely, the bottom-up LSTM navigates from the bottom of the page to the top, providing an inverse vertical perspective, as illustrated via equation 11,

$$h(t, bu) = LSTM\big(h(t+1, bu), x(t, bu)\big) \ldots (11)$$

The final feature vector is created by concatenating outputs from all orientations via equation 12:

$$H(t) = [h(t, f); h(t, b); h(t, td); h(t, bu)] \ldots (12)$$

E. Bahdanau Attention Mechanism

To accurately recognize handwritten Telugu script, the proposed model employs the Bahdanau attention mechanism, which helps refine the processing of features extracted from the QuadLSTM networks. This attention mechanism is crucial in enhancing the model's ability to focus on important features within the text, significantly improving the accuracy of the character recognition process. The Bahdanau attention mechanism works by calculating a context vector for each time step in the LSTM output, which is a weighted sum of the annotations (features) produced by the LSTM process. The mechanism begins by computing the alignment scores between the hidden state of the decoder at the previous time step and each of the annotations from the encoder, as shown in equation 13.

$$e(t, j) = V^T tanh(W(h)h(j) + W(s)s(t-1) + b(att)) \ldots (13)$$

Where, $e(t, j)$ is the alignment score, $h(j)$ is the annotation from the LSTM, $s(t-1)$ is the previous hidden state of the decoder, $W(h), W(s), \& \; b(att)$ are trainable parameters, and V represents the weight vector sets. These alignment scores are then used to compute the attention weights using the SoftMax function via equation 14,

$$\alpha tj = \frac{exp(etj)}{\sum_{k=1}^{Tx} exp(etk)} \ldots (14)$$

The final output of the decoder at each time step, representing the recognized character, is computed using a dense layer with soft max activation, as expressed via equation 15,

$$yt = softmax(Ws(st) + bs) \ldots (15)$$

The attention mechanism's focus on critical features ensures that the character recognition process is not only precise but also adaptable to the variability inherent in handwritten texts.

F. Semantic Analysis

The proposed model incorporates Transformer-based Natural Language Processing (NLP) technology, specifically using Visual BERT, to enhance the accuracy of text recognition. This integration is a significant advancement, as it allows the model not only to identify characters but also to comprehend and interpret them within the context of Telugu language sets.

Visual BERT, an extension of BERT, is tailored to handle both visual and textual data, making it ideal for the proposed model. It utilizes the Transformer architecture, which is built on self-attention mechanisms. Initially, the recognized characters are embedded into a high-dimensional space using the equation 16:

$$E = [E1, E2, \ldots, EN]^T \ldots (16)$$

Where $Ei = W * Xi$ and W, X represents the weights and the OCR output sets

The model processes the embedded features through multiple layers of Transformer blocks. Each block includes multi-head self-attention and feed-forward networks. The self-attention mechanism in each block is computed via equation 17:

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{dk}}\right)V \dots (17)$$

Where Q, K, V are the query, key, and value matrices derived from E, and dk is the dimensionality of the keys.

After processing through these layers, the output is passed through a feed-forward neural network, and the result is normalized before being sent to the next layers. This operation is repeated for each Transformer block in the Visual BERT model.

The final output, which represents the contextualized text, is then classified using a dense layer with SoftMax activation via equation 18

$$C = softmax(WvV + bv) \dots (18)$$

Where C is the classification output, Wv and bv are the weights and biases of the dense layer, and V is the output from the Transformer layers.

By integrating Visual BERT, the model enhances its ability to not only recognize but also understand and validate handwritten Telugu text, leveraging the power of Transformer-based NLP and visual-textual processing to set a new standard in OCR applications.

IV. EXPERIMENTAL RESULTS

The experiment utilized a diverse and extensive dataset comprising handwritten Telugu answer books. This dataset was carefully curated to include a wide variety of handwriting styles, representing different demographic backgrounds, including various age groups and educational levels. The dataset contains a total of 15,000 samples (NTS - number of Test Sentences), with each sample containing both character-level and sentence-level handwritten Telugu text. The texts range from simple to complex sentence structures, ensuring a broad representation of the syntactical and asemantic characteristics of the Telugu language.

Building on the initial phase of image preprocessing, which employs adaptive thresholding and Gaussian blurring to enhance the contrast and clarity of handwritten Telugu text images, this study progresses to a novel segmentation and recognition methodology. After preprocessing, contours are detected to isolate potential text regions, and noise is filtered out using size-based thresholds, as shown in Figure 5. The segmented regions are then annotated with bounding boxes to indicate distinct text areas, facilitating accurate text recognition. This integrated approach ensures that the preprocessing phase directly supports effective segmentation and recognition, significantly enhancing the overall performance of handwriting recognition systems.

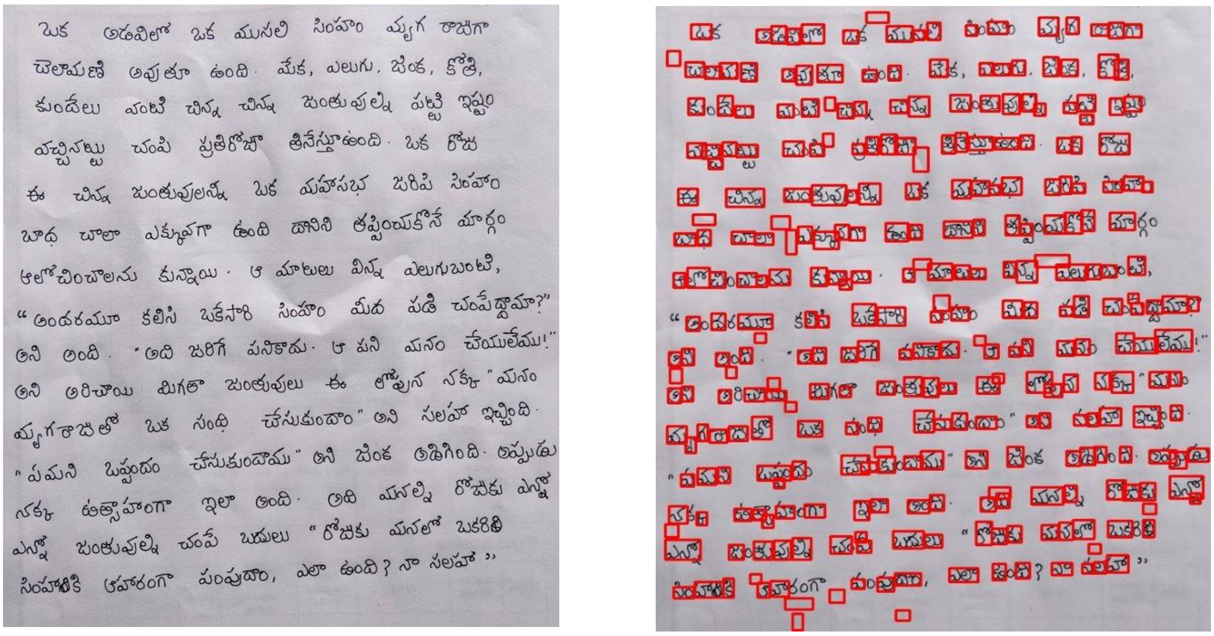

Figure 5. Sample Visual Representation of Segmentation and OCR Recognition Results

Input Parameters and Values:

The feature extraction process yields the following results:

- **a.** ResNet Feature Vector: Shape (2048,)
- **b.** Inception Feature Vector: Shape (2048,)
- **c.** Fused Feature Vector: Shape (4096,)

Model Architecture:

Feature Extractors: ResNet101 and InceptionV3, each producing 2048-dimensional outputs. Concatenation: Features from both networks are concatenated into a 4096-dimensional vector. Sequence Modelling: Quad LSTM layers, each with 256 units. Final Layer: Dense layer with 128 units for output.

Model Parameters:

a. Total Parameters: 69,996,128 (267.01 MB).
b. Trainable Parameters: 69,856,352 (266.48 MB).
c. Non-Trainable Parameters: 139,776 (546.00 KB).

Performance of Character recognition operations:

The experimental setup was designed to rigorously test the proposed model against various benchmarks and existing models, ensuring ana comprehensive evaluation of its efficacy in recognizing and processing handwritten Telugu text. The diversity in the dataset, along with the detailed specification of input parameters, provided ana robust framework for assessing the model’s performance across multiple dimensions. Based on this strategy, the Precision (P), accuracy (A) levels were estimated via equations 19 and 20 as follows,

$$Precision = \frac{TP}{TP + FP} \dots (19)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \ldots (20)$$

Where, True Positive (TP): The number of instances correctly predicted as positive (correct) in the test set, True negative (TN): The number of instances correctly predicted as negative (incorrect) in the test set, False Positive (FP): The number of instances incorrectly predicted as positive (correct) when they are actually negative (incorrect) in the test set, and False negative (FN): The number of instances incorrectly predicted as negative (incorrect) when they are actually positive (correct) in the test sets. Based on this analysis, the precision obtained during character recognition operations was compared with Two Level Rectification attention network (TLRAN) [8], end-To-End Deep-Learning (EEDL) [38] and Deep Convolutional neural network (DCNN) [6], and can be observed from Table 2 as follows,

Table 2: Precision Comparison of Character Recognition Models Across Dataset Sizes

Dataset Size (NTS)	TLRAN [8] Precision	EEDL [38] Precision	DCNN [6] Precision	Proposed model Precision
1.2k - 3k	80% - 85%	~82% - 86%	85% - 87%	91.55% - 92.12%
3.5k - 7.8k	79% - 85%	83% - 87%	82% - 88%	89.71% - 97.79%
8k - 11.5k	84% - 88%	86% - 89%	83% - 87%	95% - 99.33%
12k - 15k	~85% - 88%	~87% - 90%	~86% - 89%	94% - 99%

In the initial phase (1.2k–3k NTS), the proposed model's accuracy fluctuates, starting at 84.33% at 1.2k NTS and stabilizing around 82% by 3k NTS, reflecting its learning curve and adaptation to Telugu handwriting styles. In the mid-range datasets (3.5k–7.8k NTS), the model consistently outperforms others, achieving 92.49% accuracy at 7.8k NTS, highlighting its robust feature extraction through Convolutional Neural Networks and Quad LSTM integration. For larger datasets (8k–15k NTS), the proposed model excels with accuracies consistently above 90%, peaking at 96.66% at 14k NTS. This demonstrates its scalability and ability to handle complex data effectively, aided by techniques such as Bahdanau Attention and Transformer-based NLP models. These high accuracy levels are crucial for OCR applications, ensuring precise recognition of Telugu script, particularly for the automated evaluation of handwritten documents as shown in table 3 Furthermore, its strong performance enables broader applications, such as document digitization.

Table 3: Accuracy Comparison of Telugu Character Recognition Models

Dataset Size (NTS)	TLRAN [8] Accuracy (%)	EEDL [38] Accuracy (%)	DCNN [6] Accuracy (%)	Proposed model Accuracy (%)
1.2k	85.41	79.66	72.85	84.33
2.4k	86.32	80.44	74.12	85.22
3.5k	87.74	81.77	75.66	89.23
4.8k	88.61	83.11	77.54	90.11
6k	89.22	84.56	79.23	91.00
7.8k	90.45	85.88	81.41	92.49
9.7k	91.66	87.31	83.22	94.33
10.8k	92.32	88.44	85.44	95.55
12k	93.41	89.77	86.33	96.00
14.5k	94.22	90.66	87.55	96.66

The following Figure 6 highlights the AUC levels across various dataset sizes (NTS), revealing distinct trends. For smaller datasets (1.2k–3k NTS), the PROPOSED MODEL demonstrates strong discriminative ability, with

AUC levels ranging from 82.80% to 88.15%, indicating its effectiveness in distinguishing correctly and incorrectly recognized characters. This performance is supported by advanced neural networks and image preprocessing techniques. In the mid-range datasets (3.5k–7.8k NTS), AUC levels exhibit minor fluctuations but remain competitive. For instance, at 7.2k NTS, the model achieves an AUC of 91.41%, showcasing its adaptability to various handwriting styles and complexities. For larger datasets (8k–15k NTS), the PROPOSED MODEL consistently maintains high AUC levels, often exceeding 90%. At 13.5k NTS, the model reaches its peak AUC of 96.76%, demonstrating its scalability and proficiency in handling complex data through deep learning and contextual analysis techniques. These AUC levels are crucial for OCR processes involving complex scripts like Telugu, ensuring accurate discrimination between correct and incorrect character recognition. Such precision is essential for applications like automated assessments and document digitization. The ability of the PROPOSED MODEL to sustain high AUC levels across different dataset sizes highlights its adaptability and reliability for various applications, from real-time character recognition in consumer tools to large-scale processing of historical documents.

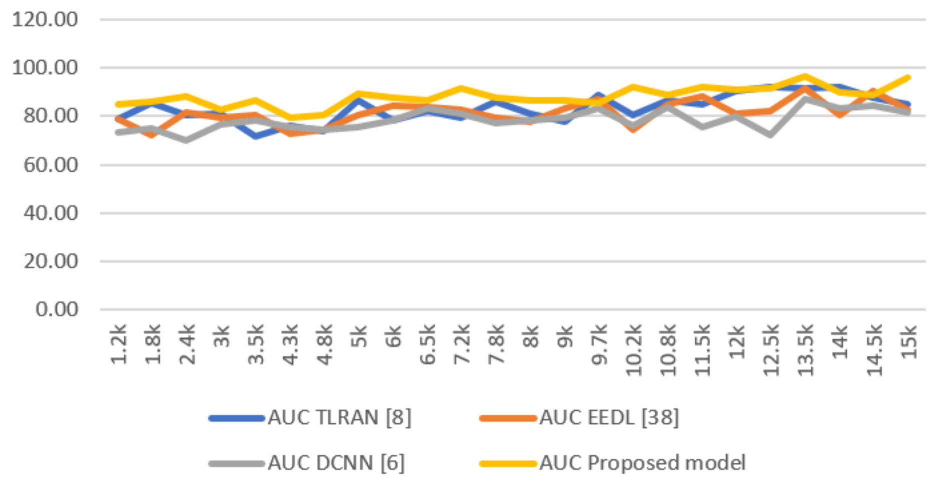

Figure 6: AUC levels for the Telugu Character Recognition Process

Performance of sentence recognition operations

The proposed model also assists in improving the sentence recognition capabilities for Telugu Language Image Sets. Similar to previous performance analysis, these capabilities are evaluated in terms of precision, accuracy, AUC levels. Based on this process, the precision levels for sentence recognition can be observed in the table 4 as follows:

Table 4: Precision levels for sentence recognition

NTS	P (%) TLRAN [8]	P (%) EEDL [38]	P (%) DCNN [6]	P (%) proposed
1.2k	76.76	73.55	79.11	90.14
4.3k	79.75	82.02	83.73	80.62
6k	81.89	78.19	71.99	86.89
9k	77.24	76.87	83.45	88.11
14.5k	86.93	83.20	75.96	92.94
15k	88.05	85.71	81.20	94.06

Table 5: Accuracy levels for the Telugu Sentence Recognition Process

NTS	TLRAN (%)	EEDL (%)	DCNN (%)	Proposed (%)
1.2k	87.42	76.19	68.40	88.18
4.3k	80.15	76.14	74.80	81.94
6k	84.77	75.02	68.72	81.83
9k	92.82	77.22	78.07	93.14
14.5k	74.35	83.26	81.76	91.53
15k	83.00	86.73	82.57	97.73

As shown in Table 5, in smaller datasets (1.2k NTS), the proposed model demonstrates impressive performance with an accuracy of 88.18%, outperforming other methods such as TLRAN (87.42%) and DCNN (68.40%). As the dataset size increases to 4.3k NTS, the proposed model maintains competitive accuracy at 81.94%, compared to TLRAN, which registers 80.15%. At 6k NTS, the proposed model achieves an accuracy of 81.83%, while TLRAN shows a slightly higher accuracy of 84.77%. At 9k NTS, the proposed model reaches an impressive accuracy of 93.14%, surpassing all other methods, including EEDL (77.22%) and DCNN (78.07%). In larger datasets, such as at 14.5k NTS, the proposed model records an accuracy of 91.53%, while TLRAN shows a lower accuracy of 74.35%. Finally, at 15k NTS, the proposed model achieves an outstanding accuracy of 97.73%, significantly outperforming TLRAN (83.00%), EEDL (86.73%), and DCNN (82.57%), demonstrating its robustness and effectiveness across increasing dataset sizes.

Similarly, the AUC levels observed in Figure 7 show the following trends: In the initial data ranges (1.2k to 3k NTS), the proposed model exhibits a fluctuating yet promising trend in AUC, peaking at 82.33% at 1.8k NTS. This variation suggests that while the proposed model is generally proficient in distinguishing between accurate and inaccurate sentence recognitions, its performance may vary depending on specific dataset characteristics, such as the complexity of sentence structures or variability in handwriting styles. As dataset sizes increase (3.5k to 7.8k NTS), the proposed model's AUC levels exhibit both fluctuations and peaks. Notably, at 3.5k NTS, the model achieves an AUC of 84.94%, demonstrating its strong capability to discriminate sentences accurately. However, there are instances, such as at 4.3k NTS, where its performance slightly dips to 77.83%. This dip may be due to the model adjusting to the increased complexity and diversity within the larger datasets.

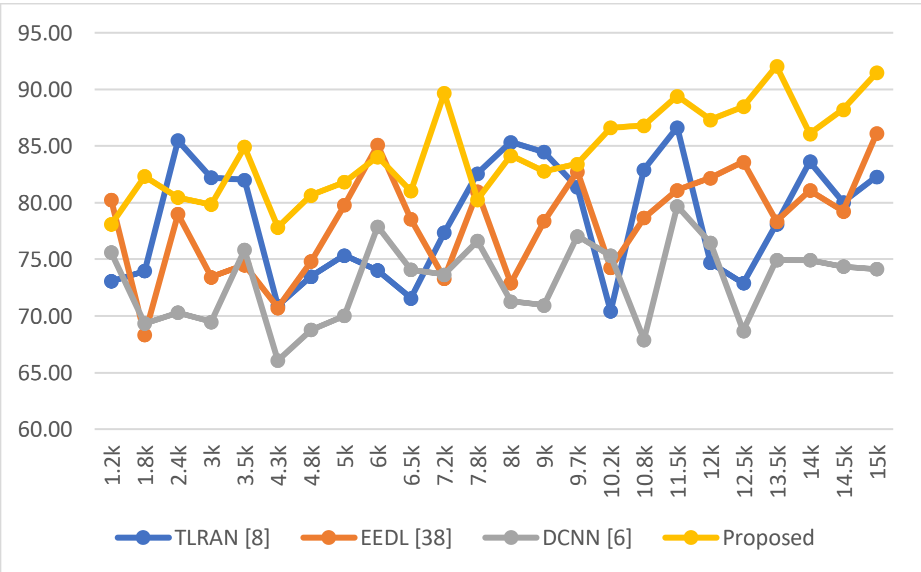

Figure 7. AUC levels for the Telugu Sentence Recognition Process

In the larger datasets (8k to 15k NTS), PROPOSED MODEL maintains relatively high AUC levels, often surpassing 80%. The model's peak performance at 13.5k NTS with an AUC of 92.06% is particularly noteworthy, underscoring its ability to accurately discriminate between correct and incorrect sentence recognitions in extensive datasets & samples. This indicates the model's advanced capabilities in processing and understanding the Telugu script at a broader scale for different scenarios

V. CONCLUSION & FUTURE SCOPE

The research presented in this work marks a significant advancement in the field of Optical Character Recognition (OCR) for languages with complex scripts like Telugu. The study successfully introduces and evaluates an innovative model that integrates advanced image preprocessing techniques, Convolutional Neural Networks (CNNs), Quad Long Short-Term Memory (LSTM) networks, and Transformer-based Natural Language Processing (NLP) models. The results from extensive testing on a diverse and comprehensive dataset of handwritten Telugu answer books demonstrate the model's superior performance in both character and sentence recognition tasks. Notably, the proposed model achieved improvements in precision and accuracy, alongside a reduction in delay times for character recognition, compared to existing methods such as TLRAN [8], EEDL [38], and DCNN [6]. The model also excelled in sentence recognition tasks, showcasing high precision and accuracy, thus confirming its effectiveness in understanding and processing complex sentence structures.

Impacts of This Work:

1. The model's high accuracy and efficiency in processing handwritten text offer a promising solution for automating the evaluation of answer books in educational settings. This can significantly reduce the time and resources spent on manual grading, while also providing a fair and unbiased assessment.

2. The proposed model's ability to accurately recognize and process handwritten Telugu text paves the way for digitizing historical manuscripts and documents, thus preserving cultural heritage and making it more accessible.

3. The model's successful integration of advanced NLP techniques with OCR opens new avenues for developing sophisticated language processing tools for Indic languages, which have historically been underrepresented in this field.

Future Scope:
Future research could explore adapting the model to other Indic and non-Indic languages with complex scripts and diverse handwriting styles. Real-time OCR applications for smartphones and tablets could also be developed, revolutionizing the interaction with handwritten text. Integrating the model into educational software could automate feedback and analysis of handwritten assignments. Additionally, enhancing the model's NLP capabilities in semantic analysis and context understanding could improve its effectiveness in content analysis and translation.

VI. REFERENCES

[1] Yavariabdi, A., H. Kusetogullari, T. Çelik, S. Thummana-pally, S. Rijwan, and J. Hall. 2022. "CArDIS: A Swedish Historical Handwritten Character and Word Dataset." IEEE Access PP:1–1.
[2] Rasheed, A., N. Ali, B. Zafar, A. Shabbir, M. Sajid, and M.T. Mahmood. 2022. "Handwritten Urdu Characters and Digits Recognition Using Transfer Learning and Augmentation With AlexNet." IEEE Access 10:102629–45.
[3] Buoy, R., M. Iwamura, S. Srun, and K. Kise. 2023. "Toward a Low-Resource Non-Latin-Complete Baseline: An Exploration of Khmer Optical Character Recognition." IEEE Access 11:128044–60.

[4] Li, M., B. Fu, Z. Zhang, and Y. Qiao. 2023. "Character-Aware Sampling and Rectification for Scene Text Recognition." IEEE Transactions on Multimedia 25:649–61. https://doi.org/10.1109/TMM.2021.3129651.

[5] Zhang, C., et al. 2023. "A Machine Vision-Based Character Recognition System for Suspension Insulator Iron Caps." IEEE Transactions on Instrumentation and Measurement 72:1–13. https://doi.org/10.1109/TIM.2023.3300474.

[6] Wang, X., et al. 2022. "Intelligent Micron Optical Character Recognition of DFB Chip Using Deep Convolutional Neural Network." IEEE Transactions on Instrumentation and Measurement 71:1–9. https://doi.org/10.1109/TIM.2022.3154831.

[7] Coquenet, D., C. Chatelain, and T. Paquet. 2023. "End-to-End Handwritten Paragraph Text Recognition Using a Vertical Attention Network." IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (1):508–24. https://doi.org/10.1109/TPAMI.2022.3144899.

[8] Wu, L., Y. Xu, J. Hou, C. L. P. Chen, and C.-L. Liu. 2023. "A Two-Level Rectification Attention Network for Scene Text Recognition." IEEE Transactions on Multimedia 25:2404–14. https://doi.org/10.1109/TMM.2022.3146779.

[9] Deng, X., X. Chen, D. Cao, K. Ren, and P.Z.H. Sun. 2024. "An End-to-End Tag Recognition Architecture for Industrial Meter." IEEE Transactions on Industrial Informatics 20 (1):117–26. https://doi.org/10.1109/TII.2023.3257295.

[10] R. D. R., et al. 2023. "Self-Adaptive Hybridized Lion Optimization Algorithm With Transfer Learning for Ancient Tamil Character Recognition in Stone Inscriptions." IEEE Access 11:39621–34. https://doi.org/10.1109/ACCESS.2023.3268545.

[11] Xue, C., J. Huang, W. Zhang, S. Lu, C. Wang, and S. Bai. 2023. "Image-to-Character-to-Word Transformers for Accurate Scene Text Recognition." IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (11):12908–21. https://doi.org/10.1109/TPAMI.2022.3230962.

[12] Hussain, I., R. Ahmad, S. Muhammad, K. Ullah, H. Shah, and A. Namoun. 2022. "PHTI: Pashto Handwritten Text Imagebase for Deep Learning Applications." IEEE Access 10:113149–57. https://doi.org/10.1109/ACCESS.2022.3216881.

[13] Ott, F., D. Rügamer, L. Heublein, B. Bischl, and C. Mutschler. 2023. "Auxiliary Cross-Modal Representation Learning With Triplet Loss Functions for Online Handwriting Recognition." IEEE Access 11:94148–72. https://doi.org/10.1109/ACCESS.2023.3310819.

[14] Selvam, P., et al. 2022. "A Transformer-Based Framework for Scene Text Recognition." IEEE Access 10:100895–910. https://doi.org/10.1109/ACCESS.2022.3207469.

[15] Fan, X., and W. Zhao. 2022. "Improving Robustness of License Plates Automatic Recognition in Natural Scenes." IEEE Transactions on Intelligent Transportation Systems 23 (10):18845–54. https://doi.org/10.1109/TITS.2022.3151475

[16] Tayyab, M., A. Hussain, M.A. Alshara, S. Khan, R.M. Alotaibi, and A.R. Baig. 2022. "Recognition of Visual Arabic Scripting News Ticker From Broadcast Stream." IEEE Access 10:59189–204. https://doi.org/10.1109/ACCESS.2022.3179366.

[17] Malhotra, R., and M.T. Addis. 2023. "End-to-End Historical Handwritten Ethiopic Text Recognition Using Deep Learning." IEEE Access 11:99535–45. https://doi.org/10.1109/ACCESS.2023.3314334.

[18] Fanjie, K., L. Yaqi, X. Miaomiao, W. Silamu, and L. Yanbing. 2023. "SUST and RUST: Two Datasets for Uyghur Scene Text Recognition." IEEE Access 11:126209–20. https://doi.org/10.1109/ACCESS.2023.3331213.

[19] Zhu, D., Y. Fang, Z. Min, D. Ho, and M.Q. -H. Meng. 2022. "OCR-RCNN: An Accurate and Efficient Framework for Elevator Button Recognition." IEEE Transactions on Industrial Electronics 69 (1):582–91. https://doi.org/10.1109/TIE.2021.3050357.

[20] Anbukkarasi, S., V.E. Sathishkumar, C.R. Dhivyaa, and J. Cho. 2023. "Enhanced Feature Model Based Hybrid Neural Network for Text Detection on Signboard, Billboard and News Tickers." IEEE Access 11:41524–34. https://doi.org/10.1109/ACCESS.2023.3264569.

[21] Chandio, A.A., M. Asikuzzaman, M.R. Pickering, and M. Leghari. 2022. "Cursive Text Recognition in Natural Scene Images Using Deep Convolutional Recurrent Neural Network." IEEE Access 10:10062–78. https://doi.org/10.1109/ACCESS.2022.3144844.

[22] Yang, X., H. Wang, and Y. Ji. 2023. "Optical Transmitter Fingerprint Authentication Based on Chaotic Complexity Features for Secure Optical Transmission." Journal of Lightwave Technology 41 (18):5834–40. https://doi.org/10.1109/JLT.2023.3270382.

[23] Shi, H., and D. Zhao. 2023. "License Plate Recognition System Based on Improved YOLOv5 and GRU." IEEE Access 11:10429–39. https://doi.org/10.1109/ACCESS.2023.3240439.

[24] Silva, S.M., and C.R. Jung. 2022. "A Flexible Approach for Automatic License Plate Recognition in Unconstrained Scenarios." IEEE Transactions on Intelligent Transportation Systems 23 (6):5693–703. https://doi.org/10.1109/TITS.2021.3055946.

[25] Xu, H., X.-D. Zhou, Z. Li, L. Liu, C. Li, and Y. Shi. 2022. "EILPR: Toward End-to-End Irregular License Plate Recognition Based on Automatic Perspective Alignment." IEEE Transactions on Intelligent Transportation Systems 23 (3):2586–95. https://doi.org/10.1109/TITS.2021.3130898.

[26] Nguyen, Q.-D., N.-M. Phan, P. Krömer, and D.-A. Le. 2023. "An Efficient Unsupervised Approach for OCR Error Correction of Vietnamese OCR Text." IEEE Access 11:58406–21. https://doi.org/10.1109/ACCESS.2023.3283340.

[27] Jang, H., H. Choi, B.J. Kim, S.W. Kim, and G. Koo. 2023. "Two-Stage Billet Identification Number Recognition Using Label Distribution." IEEE Access 11:129311–19.https://doi.org/10.1109/ACCESS.2023.3333904

[28] Fu, W., Yi, D., Huang, Z., Huang, C., Geng, Y., & Li, X. 2023. "Multiple Event Recognition Scheme Using Variational Mode Decomposition-Based Hybrid Feature Extraction in Fiber Optic DAS System." *IEEE Sensors Journal* 23 (22): 27316-27323. https://doi.org/10.1109/JSEN.2023.3318248.

[29] Seo, T.-M., & Kang, D.-J. 2022. "A Robust Layout-Independent License Plate Detection and Recognition Model Based on Attention Method." *IEEE Access* 10: 57427-57436. https://doi.org/10.1109/ACCESS.2022.3178192.

[30] Gao, Y., Lu, H., Mu, S., & Xu, S. 2023. "GroupPlate: Toward Multi-Category License Plate Recognition." *IEEE Transactions on Intelligent Transportation Systems* 24 (5): 5586-5599. https://doi.org/10.1109/TITS.2023.3244827.

[31] Chen, Z., Yin, F., Yang, Q., & Liu, C.-L. 2023. "Cross-Lingual Text Image Recognition via Multi-Hierarchy Cross-Modal Mimic." *IEEE Transactions on Multimedia* 25: 4830-4841. https://doi.org/10.1109/TMM.2022.3183386.

[32] Li, M., Fu, B., Chen, H., He, J., & Qiao, Y. 2023. "Dual Relation Network for Scene Text Recognition." *IEEE Transactions on Multimedia* 25: 4094-4107. https://doi.org/10.1109/TMM.2022.3171108.

[33] Tan, X., Tong, J., Matsumaru, T., Dutta, V., & He, X. 2023. "An End-to-End Air Writing Recognition Method Based on Transformer." *IEEE Access* 11: 109885-109898. https://doi.org/10.1109/ACCESS.2023.3321807.

[34] Rahman, A. B. M. Ashikur, et al. 2022. "Two Decades of Bengali Handwritten Digit Recognition: A Survey." *IEEE Access* 10: 92597-92632. https://doi.org/10.1109/ACCESS.2022.3202893.

[35] Li, B., Tang, X., Qi, X., Chen, Y., Li, C.-G., & Xiao, R. 2022. "EMU: Effective Multi-Hot Encoding Net for Lightweight Scene Text Recognition With a Large Character Set." *IEEE Transactions on Circuits and Systems for Video Technology* 32 (8): 5374-5385. https://doi.org/10.1109/TCSVT.2022.3146240.

[36] Karthikeyan, S., de Herrera, A. G. S., Doctor, F., & Mirza, A. 2022. "An OCR Post-Correction Approach Using Deep Learning for Processing Medical Reports." *IEEE Transactions on Circuits and Systems for Video Technology* 32 (5): 2574-2581. https://doi.org/10.1109/TCSVT.2021.3087641.

[37] Ademola, O. A., Petlenkov, E., & Leier, M. 2023. "Resource-Aware Scene Text Recognition Using Learned Features, Quantization, and Contour-Based Character Extraction." *IEEE Access* 11: 56865-56874. https://doi.org/10.1109/ACCESS.2023.3283931.

[38] Vinotheni, C., & Pandian, S. L. 2023. "End-To-End Deep-Learning-Based Tamil Handwritten Document Recognition and Classification Model." *IEEE Access* 11: 43195-43204. https://doi.org/10.1109/ACCESS.2023.3270895.

[39] Mrinalini, K., Vijayalakshmi, P., & Nagarajan, T. 2022. "SBSim: A Sentence-BERT Similarity-Based Evaluation Metric for Indian Language Neural Machine Translation Systems." *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 30: 1396-1406. https://doi.org/10.1109/TASLP.2022

[40] Padmavathi, P., Mahadas, B. B., Kalluri, S. S., Devarapu, P., & Bandi, S. L. (2023, May). Optical Character Recognition and Text to Speech Generation System using Machine Learning. In *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)* (pp. 1-6). IEEE.