Int. j. inf. tecnol.
https://doi.org/10.1007/s41870-024-01984-x

ORIGINAL RESEARCH

Text recognition using improved dual attention based on textual double embedding network with aquila optimization algorithm

Harsiddhi Singhdev[1] · Shruti Gupta[1] · Vivek Srivastava[1] · Apoorva Saxena[2]

Received: 27 February 2024 / Accepted: 31 May 2024

Abstract Text recognition is the process that changes an image of text into a system readable text format. Different approaches were suggested related to text detection but the existing methods accuracy is low and error rate is high. Therefore, a text recognition using Improved Dual-attention based on Textual Double Embedding (IDTDE) with aquila optimization algorithm is proposed in this manuscript for effective text recognition. In this method the input image is taken from several datasets. Usually, the images have some kind of noise in it, and to eliminate that, the Structural interval gradient filtering preprocessing technique is used. Then, the ternary pattern and discrete wavelet technique is use to extract best features. Next, residual-based temporal attention convolutional neural network is utilized for the text classification and character identification are effectively attained by and IDTDE network. The Aquila optimization algorithm is used to optimize the network for best extracted text as output. Experimental outcomes demonstrate that the introduced approach accomplishes the high accuracy rates of 98.14%, 98.89%, and 90.47% on the IIIT5K, ICDAR 2013, and ICDAR 2015 datasets, respectively, surpassing the performance of existing frameworks.

Keywords Text recognition · Character recognition · Improved self-attention · Aquila optimizer · Gradient filtering

1 Introduction

Text recognition is essential in computer vision, driven by advancements in Deep Learning (DL) across pattern recognition, computer vision, and machine learning fields. DL drives significant progress in Scene Text Recognition (STR), with segmentation-based and non-segmentation-based techniques as the main branches [1, 2]. Segmentation-based methods identify individual characters, then combine them into a string sequence for recognition [3]. Conversely, non-segmentation-based techniques consider the entire text line as a single entity, mapping input text images directly to target string sequences. This approach avoids single-character segmentation limitations, emerging as the mainstream in natural STR [4].

In recent times, numerous DL schemes have achieved advanced results in STR, particularly for regular text that is typically flat and frontal. Still, these methods often struggle with irregular texts, which may be arbitrarily oriented and curved. Recent research efforts have focused on improving STR performance through various deep network enhancements, such as super-resolution, attention mechanisms, backbone network improvements, and rectification modules [5–7]. While STR methods generally perform well, their accuracy tends to drop significantly with partially occluded and low-resolution images. Current visual models for scene text recognition often prioritize the visual aspects

✉ Harsiddhi Singhdev
harsiddhi.dev@abes.ac.in

Shruti Gupta
shruti.gupta@abes.ac.in

Vivek Srivastava
viveksrivastava@abes.ac.in

Apoorva Saxena
apoorvasaxena@axiscolleges.in

[1] Department of Computer Science and Engineering, ABES Engineering College, Ghaziabad, Uttar Pradesh, India

[2] Department of Computer Science and Engineering, Axis Institute of Technology & Management, Kanpur, Uttar Pradesh, India

Published online: 14 June 2024

of characters, ignoring linguistic context. Integrating linguistic knowledge into STR techniques is crucial for interpreting characters contextually [8, 9].

Various methods explore incorporating linguistic knowledge into STR schemes, inspired by Natural Language Processing (NLP) approaches. Moreover, current techniques have focused on enhancing STR on by incorporating linguistic information [10]. As a result, the prevailing trend in recent technologies is to adopt a two-step framework that combines visual and linguistic modeling. In this technology, the visual strategy primarily considers the visual appearance of text, disregarding linguistic characteristics. On the other hand, language models utilize structures including transformers, Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), to infer associations between characters [11, 12].

Although these approaches have achieved favorable recognition outcomes in recent research, they still face certain challenges. Language models, such as bidirectional inference architectures, enhance linguistic insights but also intensify computational demands. This hinders their real-world effectiveness [13, 14, 34–38]. Ideally, an STR model should balance recognition accuracy, speed, and computational efficiency. The major objectives of this research work are,

- This research work develops a novel IDTDE approach for STR task. Employing an enhanced Self-Attention (SA) mechanism further refines the text feature's representation and increases the efficacy of text feature expression.
- Prior to performing STR, the image must undergo analysis to determine if it contains any relevant information. This task is accomplished using a Residual-based Temporal Attention Convolutional Neural Network (RTACNN) model. The RTACNN model serves as a predecessor to the DL-based character detection process, classifying the image as either non-textual or textual.
- The proposed approach demonstrates best performance by surpassing the existing techniques on both standard and irregular STR benchmarks.

The remaining manuscript is prepared as follows: Sect. 2 reviews the existing STR models, Sect. 3 defines the introduced methodology, Sect. 4 demonstrates the outcomes of the developed STR approach and Sect. 5 indicates the conclusion of research work.

2 Literature survey

Several studies in the literature have explored text recognition. Here, a few recent studies are highlighted as follows, Long S et al. [15] suggested a DL technique to tackle the disadvantages of traditional models for text identification and detection in floor plans. They addressed this by providing extensive testing and training datasets for text identification and creating an end-to-end pipeline for processing text in graphics. However, a notable challenge arose from the poor performance of text detection.

Lee et al. [16] developed an architecture leveraging an enhanced version of the Local Binary Patterns (LBP) shallow deep convolutional neural network for text detection. They incorporated ILBP feature preprocessing into the framework to enhance recognition accuracy. The design included two feature maps aimed at preserving image details while reducing noise. However, a challenge encountered was the decrease in computing and network parameter performance.

Liu C et al. [17] introduced a model using label-to-prototype learning for open-set text detection tasks. This approach aimed to identify and recognize new characters without necessitating retraining. The framework enabled handling of novel characters while maintaining competitive performance on standard benchmarks and respectable speed. However, a challenge encountered was that domain adaptation approaches were ineffective in mitigating domain bias within complex vectors.

Li et al. [31] developed the Unified Text and Table Structure Recognition (UTTSR) model, which comprises four main components. Cascade Faster Region-based CNN (RCNN) with ResNeXt105 were utilized for table detection, employing affine and Thin Plate Spline (TPS) transformations for image correction and accuracy enhancement. Text line detection employed Dual Branch Network (DBNet) for faster training, while text lines were recognized using CRNN, increasing recognition performance.

Yue et al. [12] developed Noise-Robust Scene Text Recognition Network (NRSTRNet) for STR task. NRSTRNet effectively reduced noise in different phases of STR. Firstly, its enhanced text-related features, reducing noise and redundancy. Then, it suggested fine-grained feature coding to minimize the impact of noisy temporal features and partial noise. Finally, a SA module was included to improve connections across temporal features, utilizing global information for noise-resistant features.

Mahadshetti et al. [3] developed the Residual Multi-Feature Pyramid Network (RMFPN), merging ResNet and multi-feature pyramid networks for STR task. RMFPN employed two convolutional pyramids as feature extractors, enhanced feature robustness and semantic information for STR across different scales. However, RMFPN may face challenges in handling highly complex text images, potentially leading to decreased performance in such scenarios

Alshawi et al. [18] suggested a convolutional-based approach incorporating the squeeze and excitation gate to emphasize latent attributes in Persian digit recognition. They introduced a novel convolutional RNN technique with

attention mechanisms to identify digits within the dataset. One limitation was its lack of suitability for noisy text.

Zhou et al. [19] developed a lightweight and efficient backbone model for STR. This model achieved robust feature extraction while significantly minimizing the parameter count, facilitating easier implementation of text recognition. However, one of the challenges encountered was that the suggested model exhibited lower recognition accuracy.

Banerjee et al. [20] developed an end-to-end model known as E2EMVSTR (End-to-End Model for Multi-View STR) that recognized text by refining several views of the same scene. The system employed DL techniques for text detection, and refinement. A difficulty raised from various variables, including varying camera angles and capacities, distortion, degradation, poor quality, occlusion, and loss of information. These factors contributed to the complexity and difficulty of recognition model in different scenarios. Table 1 provides the summary of existing frameworks.

2.1 Problem statement

In the context of STR, the challenge lies in developing robust and accurate methods capable of accurately detecting and identifying text from images captured under diverse conditions, such as varying camera angles, lighting conditions, image distortions, occlusions, and varying text qualities. Existing approaches often struggle to handle these complexities effectively, leading to reduced recognition accuracy and robustness. Therefore, there is a need to develop advanced techniques that address these challenges and achieve high performance (accuracy) text recognition in multi-view scenarios.

3 Proposed methodology

The IDTDE method is proposed for an effective text recognition. Fig. 1 shows the procedure of IDTDE method and the explanation of each block is given below.

In the IDTDE method, the input image is taken from IIIT5K, ICDAR 2013(IC13) and ICDAR 2015 (IC15) datasets. For removing unwanted noises from the image structural interval gradient filtering-based preprocessing method is used. After removing the unwanted noises to extract the important features Ternary Pattern and Discrete Wavelet Transform (TP-DWT) based feature extraction is used. Then the data is classified as text and non-text using RTACNN. The character is recognized with the help of IDTDE. Finally, an Aquila Optimization Algorithm (AOA) is used to optimize IDTDE or an effective text recognition.

Table 1 Summary of existing frameworks

References	Methods	Advantages	Drawbacks
[15]	DL	Offered a large number of training and testing datasets for text recognition	The text detection performance is weak
[16]	Deep neural network	Improved character recognition	Both computation and network parameter performance are minimal
[17]	Open-Set text detection	A label-to-prototype learning framework that could handle new characters	Domain adaption techniques were unable to solve complex task
[31]	UTTSR	Minimized time costs and recognition speed	The method still face limitations in detecting complex table structures and addressing the issue of spanning cell recovery
[12]	NRSTRNet	The ability to characterize text images was improved	The training process for NRSTRNet may require a significant amount of time
[3]	RMFPN	Improved the feature extractor's ability	Faced challenges in handling highly complex text images, potentially leading to decreased performance in such scenarios
[18]	Convolutional RNN	A novel model was used to the dataset's digit recognition	One limitation was its lack of suitability for noisy text
[19]	Recursive Residual Transformer Network (RRTrN)	The reduction in parameters made scene text recognition easier to implement	Lower recognition accuracy (94.26%)
[20]	E2EMVSTR	The system improved text recognition and filled in the gaps in text content from a variety of perspectives by combining natural language processing with DL models based on genetic algorithms	The performance of the system was more complex and harder

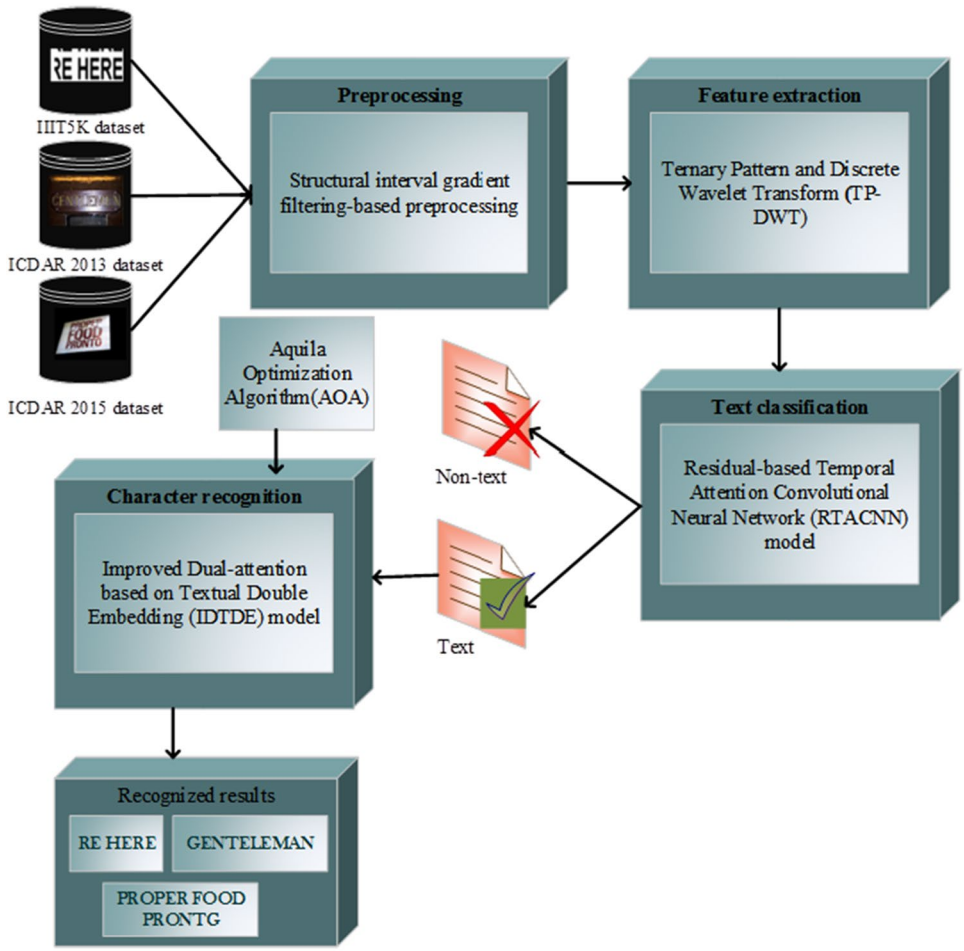


Fig. 1 Block diagram of IDTDE method

3.1 Preprocessing by structural interval gradient filtering

Initially, the input images are collected from the IIIT5K, IC13 and IC15 datasets. These text images contain a lot of noise, and the sizes of the images are dissimilar [21]. Then, the images are filtered to improve the contrast. In this step, the text images are filtered using structural interval gradient filtering to remove the noise and enhance the image quality. Gaussian smoothing filtering is initially utilized to generate the guided image, represented as Ga_{Image} and it is given in the following Eq. (1),

$$Ga_{\mathrm{Image}} = Input_{image} * Gaussian_{filter} \tag{1}$$

After that, the pre-processed image's output is attained by Eqn. (2):

$$Output_{image} = u_1 * Ga_{\mathrm{Image}} + v_1 \tag{2}$$

where linear coefficients are represented by u_1 and v_1, which computed based on Eq. (3):

$$F''(u_1, v_1) = \left[Output_{image} - Ga_{image}\right]^2 + \delta * u_1^2 \tag{3}$$

where δ represents a control parameter. By using Eq. (3), the input noises are removed and the contrast of the images is improved. Then, the pre-processed images are passed to the subsequent process.

3.2 Feature extraction using ternary pattern and discrete wavelet (TP-DWT)

TP-DWT [22] method is employed to extract text features. This technique extract features from non-overlapping nearby blocks. The feature extracted Eqn. is given in Eqn. (4),

$$TP^{Features}(Fst, Secd) = \begin{cases} -1, & Fst - Secd < -thres \\ 0, & -thres \leq Fst - Secd \leq thres \\ 1, & Fst - Secd > thres \end{cases} \tag{4}$$

where, $TP^{Features}$ indicates the ternary rate to extract features, *thres* specifies threshold value,*Fst* and Sec*d* indicates as the initial and next input parameters. TP-DWT uses blocks of size 9 instead of 3×3 matrices.

TP-DWT extracts the arithmetical features including mean, median, standard deviation. After that, the extracted fearures are specified in Eqn. (5),

$$thres^{Features} = SD\left(text_{data}\right) = \frac{text_{data}}{10}, text_{data} = \{1, 2, 3, 4, ...10\} \tag{5}$$

where $SD\left(text_{dats}\right)$ specifies the standard deviation feature of the text data. After that, the extracted mean and median features are specified in Eqns. (6–8),

$$Mean\left(text_{data}\right) = Max(h) - Min(h) \tag{6}$$

$$[U\, S\, V] = SVD(N) \tag{7}$$

$$N = reshape(h,\ 16 \times 16) \tag{8}$$

In Eq. (8), h indicates the lower or upper ternary rates with the size of 256, $SVD(.)$,$Min(.)$,$Max(.)$, and $reshape(.,.)$ represents the singular value decomposition, minimum, maximum, and vector to matrix transformation, U is represented as the unity, S is represented as the vertical, V is represented as the singular matrix. By this procedure, the features are extracted using the images. After extracting the image, the output is given to RTACNN for classification.

3.3 Residual-based temporal attention convolutional neural network (RTACNN) for text classification

The RTACNN [23] classifies the scene images into text and non-text categories. The Residual-based Temporal Attention block (RTA-block) is employed to direct attention to intervals. By utilizing the attention mechanism, additional contextual features may be obtained. The RTACNN's convolutional unit, described in Eqn. (9), focuses on text classification.

$$E(Q) = \rho(Z(v \times Q + c)) \tag{9}$$

In this, Q represents the characteristics of the unit's input, where $Q = \left|q^1 q^2 .....q^v\right| q^i \in R.R$ represents the real-number. ρ denotes the size and distribution of activation channels, while v signifies the convolution's weight. Z denotes the normalization layer, and the rectified linear unit $\rho(Q)$ induces non-linearity. A residual connection incorporates the multiplication into the original feature. The trunk branch initializes features U'' with attention weights A, distributing multiple $A \otimes U''$. Another residual connection incorporates the multiplication to the original feature. Equation (10) yields the RTA-final block output and trunk branch's final output.

$$\begin{aligned} T &= A \otimes U'' + U \\ &= \left[b_1 \times u_1'' + u_1,\ b_2 \times u_2'' + u_2, ..., b_w \times u_w'' + u_w\right], \\ & u_i'', u_i \in \Re^{1 \times C},\ b_i \in [0, 1] \end{aligned} \tag{10}$$

Equation (10) illustrates that the key factor for sentence classification is determined by the users. Ultimately, the RTACNN classifier categorizes the documents into text and non-text.

3.4 Improved Dual-attention based on textual double embedding (IDTDE) network for character recognition

The classified text is given to IDTDE for character recognition. The weight parameter of IDTDE is optimized to improve the efficiency of proposed character recognition strategy. To recognize each character individually, three basic modules are constructed: an improved Textual Self-Attention(TSA) component, a Visual SA(VSA) component, and a Textual Visual Co attention (TVC) component.

- Textual dual Embedding with improved SA component

In character detection for text sentences, RNNs like Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) layers are typically employed for representation learning to construct sentence schemes. Moreover, these schemes often struggle to capture internal word information and dependencies between positions effectively, resulting in longer processing times, especially for lengthy sentences. In contrast, the attention mechanism performs better at capturing internal dependencies between words. By incorporating SA, key text features are highlighted, leading to more accurate representations. Introducing the position weight parameter (*Weight*) enhances the SA model [24] by redistributing the computed weight probability values. This adjustment

optimizes the text feature representations and improves their expression. The improved SA mechanism, outlined in Eq. (11), assigns weight information to further refine text feature representation.

$$Self_{Attention} = Soft\max\left(\frac{QK^T}{\sqrt{D_k}}\right)V.Weight \tag{11}$$

where d_k indicates the key dimension. Q, K and V indicates the query, key and value matrix. Then, sentence information is integrated into word pair modelling to preserve semantic details. Combining embedded words and sentences ensures that TUDE(Transformer with United Positional Encoding) selects relevant keywords for interacting with visual features, enhancing character recognition. Utilizing WordPiece facilitates text processing for improved word and sentence information extraction.

- Image self-attention module

In this research work, the VSA method is employed to leverage internal dependencies within images, enhancing visual representations by weighting and aggregating features across image regions.

For an input image (I), ResNet50 [25] is utilized for extracting regional data $R = \in \mathfrak{R}^{d_r \times k}$. Prior to SA computation, the feature dimension is reduced from the extracted image. Next, the size of feature mapping is scaled to 1/8 of the input image and passed through a convolutional layer with normalization and ReLU layers. This process generated dual feature maps, $I_A = \left\{i_1^a, i_2^a, ...i_k^a \in \mathfrak{R}^{d_r \times k}\right\}$ and $I_B = \left\{i_1^b, i_2^b, ...i_k^b \in \mathfrak{R}^{d_r \times k}\right\}$, representing W_{query} and W_{key}. Both I_A and I_B are reshaped as $\mathfrak{R}^r \times n_k$, where the image region is specified by $n_k = k \times k$. To assess their interdependence, separate matrix multiplications are performed on the transpositions of I_A and I_B, followed by applying a softmax layer to determine the spatial location of the SA map $S \in \mathfrak{R}^{d_r \times d_r}$. Every element in V is computed by Eq. (12):

$$S_{qp} = \frac{\exp\left(r_p^a \times r_q^b\right)}{\sum_{p=1}^{n_k} \exp\left(r_p^a \times r_q^b\right)} \tag{12}$$

where S_{qp} measures the influence of the p^{th} area on the q^{th} area.

- Text visual co attention module

In the text image dual co attention section, dual co attention blocks are combined in parallel to create the co-attention layer. The co-attention layer operates similar to the SA block, with separate computations for query, key, and value within every co attention block. This layer facilitates inter-modality interactions by information exchange. Subsequently, outputs from both co attention blocks are merged and forwarded to a fully connected layer for text image feature recognition, as depicted in Eqn. (13),

$$F = B\left([T, R]; \theta_B\right) \tag{13}$$

In which, B indicates the mapping function, θ_B represents training parameter set for the text detection module and F indicates the representation of the text image feature. T indicates the textual feature, and R indicates the features extracted by ResNet50.

Lastly, the character detectors consist of dual fully connected layers along with ReLU and softmax activation functions, respectively. E is described as character detector, which is presented in Eq. (14),

$$E\left(F; \theta_E\right) \tag{14}$$

where θ_E indicates the training parameter. Finally, IDTDE efficiently recognizes the characters of text. After generating the character recognition, the data is given to AOA for optimizing the weight parameters of IDTDE.

3.5 Aquila optimization algorithm for optimizing IDTDE

After character recognition, the data is inputted into the AOA [26] for optimizing IDTDE. This section presents the mathematical model of the Aquila Optimizer (AO). Aquila's algorithm for hunting prey draws inspiration from bird's social behaviour.

- Initialization

Aquila begins with M agents and an initial population of Y, as represented in Eq. (15),

$$Y_{pq} = rand_1 \times \left(U_q - L_q\right) + L_q,\ p = 1, 2, ...M,\ q = 1, 2, ...Dim \tag{15}$$

where the lower and upper bounds in the search boundary are denoted as L_q and U_q, respectively, the population size is represented as Dim, and $rand_1 \in [0, 1]$ is a random

parameter. This algorithm conducts exploration and exploitation actions after initializing the population until the desired outcome is achieved.

- **Random generation**

After initialization, the input parameter is randomly generated using AOA to achieve the best solution.

- **Fitness function**

In this process, the fitness function of the AOA is employed to improve IDTDE. Eqn. (16) is utilized to calculate the fitness function.

$$Fitness = optimize\{\theta_B, \theta_E\} \tag{16}$$

where θ_B and θ_E are the training parameters of IDTDE.

- **Expanded and Narrowed explorations**

In expanded exploration, AO surveys various heights to locate prey in the search area, considering regular agents (Y_N) and the best agents (Y_b). Equations (17–18) illustrate this approach.

$$Y_p(i+1) = Y_b(i) \times \left(\frac{1-i}{I}\right) + \left(Y_M(i) - Y_b(i) \times rand\right) \tag{17}$$

$$Y_M(i) = \frac{1}{M}\sum_{p=1}^{M}(i),\ \forall q = 1, 2, ...Dim \tag{18}$$

where the total iterations are denoted as I, and $\left(\frac{1-i}{I}\right)$ is used to control the search process.

For the attack preparation, AO minimally inspects the target prey's specific location in focused exploration. This approach utilizes the distribution of levy flight $levy(E)$ and Y_b to update agents' exploration. Equations (19–20) depict both narrowed exploration and levy flight strategy.

$$Y_p(i+1) = Y_b(i) \times levy(E) + Y_R(i) + (y - x) \times rand \tag{19}$$

$$levy(E) = s \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}} \tag{20}$$

where v and u are random parameters, β is set to 1.5, and s is set to 0.01. According to Eq. (19), a randomly selected agent is designated as Y_R. Furthermore, y and x are utilized to trace the spiral shape.

- **Expanded and narrowed exploitations**

During the expanded exploitation (Y_p), AO targets a specific area to close in on and attack the prey. Utilizing Y_M and Y_b, the technique updates agents during the exploitation phase, as described in Eq. (21).

$$\begin{aligned} Y_p(i+1) = &\left(Y_b(i) - Y_N(i)\right) \times \beta - rand \\ &+ ((U - L) \times rand + L) \times \delta \end{aligned} \tag{21}$$

The adjustment parameters for the exploitation process are specified by δ and α.

In the final stage of narrowed exploitation, AO attacks the prey. Agents are updated through the quality function QF, Y_b, and $levy$ during this process. Equations (22–23) illustrate this approach.

$$\begin{aligned} Y_p(i+1) = &QF \times Y_b(i) - \left(G_1 \times Y(i) \times rand\right) \\ &- G_2 \times levy(E) + rand \times G_1 \end{aligned} \tag{22}$$

$$QF(i) = i^{\frac{2 \times rand() - 1}{(1-I)^2}} \tag{23}$$

where G_1 describes the movements utilized to track the optimal solution, G_2 indicates decreasing values from 2 to 0 and $rand$ generates random values.

- **Termination**

In this, AOA is employed to improve IDTDE. The algorithm repeats steps 3 to 5 until the termination conditions are met. Algorithm 1 shows the proposed framework that is utilzied for chracter recognition task.

In this research work, the feature extracted image undergoes preliminary analysis to determine the presence of relevant textual information before conducting STR task. This preliminary analysis is carried out using a RTACNN model. The RTACNN model acts as a precursor to the subsequent DL-based character identification process by categorizing the image as either containing non-textual elements or containing text, thereby streamlining and optimizing the overall STR process. Besides, this research introduces a novel approach, called IDTDE, aimed at enhancing the performance of STR tasks. The IDTDE method incorporates an improved SA mechanism to refine the representation of text features, thus improving the expression and effectiveness of these features. Overall, this research contributes to the accuracy of the STR task by efficiently recognizing characters from scene images

Algorithm 1 Proposed methodology for character recognition task

Input: Natural scene image with text Output: Recognized characters
Step 1: Collect input images from IIIT5K, IC13, and IC15 datasets Step 2: Filter images to enhance contrast using structural interval gradient filtering Apply Gaussian smoothing filtering to generate the guided image(Gq_{mage}) Compute linear coefficients based on Eqn. (3) Use Eqn. (2) to obtain the preprocessed image output. Step 3: Extract features using TP-DWT Extract statistical features (mean, median, standard deviation) as per Eqns (5)-(8) Step 4: RTACNN for text classification Based on extracted features, RTACNN calssifies images into text and non-text categories Step 5: IDTDE Network for Character Recognition Take the classified textual part from the previous step. Utilize the IDTDE architecture to analyze and recognize individual characters within the textual content. Employ dual fully connected layers with appropriate activation functions, such as ReLU and softmax, to classify and identify characters accurately. Step 6: AOA for optimizing IDTDE Set population size and search space bounds by Eqn (15) Calculate fitness function using Eqn (16) Explore various heights to locate prey (Eqns (17)-(20)) Target specific areas for attack (Eqns (21)-(23)) Iterate until termination conditions are met.

4 Results and discussions

The research is conducted and evaluated using a personal computer setup featuring an Intel (R) Core (TM) i7-8565U CPU @ 1.80 GHz processor, 8.00 GB of RAM (7.88 GB usable), a 64-bit operating system, and Windows 10 Pro 64-bit. The experiments utilized the open-source TensorFlow library in the python programming language. The efficiency of IDTDE approach is higher than the current techniques.

4.1 Dataset description

IIIT5K dataset [39]*:* The images in this collection differ greatly in terms of their size, change, frequency, layout, blur, colour, noise, typeface, and different lighting. This dataset consists of a total of about 5000 word together images that contain text. 2000 of these 5000 photographs are considered for training, and 3000 images considered for the testing procedure.

ICDAR 2013 dataset (IC13 dataset) [40]*:* Most images in the IC13 dataset (658 samples) were inherited from IC03. Non-alphanumeric words were removed for a fair comparison. This resulted in 1015 cropped word images without lexicons.

ICDAR 2015 dataset (IC15 dataset) [41]*:* This dataset contains 6545 cropped text images, divided into 2077 test images and 4468 training images. It lacks a lexicon. Text in these images often appears in irregular shapes. The dataset comprises images captured by Google Glasses, sometimes lacking proper positioning and focusing.

Table 2. Output images of proposed IDTDE model.

Dataset	Input image	Pre-processed image	Text classified image	Recognized character image
IIIT5K dataset				
IC13 dataset				
IC15 dataset				

Table 3 Accuracy analysis of introduced IDTDE model over existing procedures

Techniques	Accuracy (%)		
	IIIT5K dataset	IC13 dataset	IC15 dataset
OATSA [2]	97.7	98	88.2
RMFPN [3]	96.79	97.38	89.10
CRNN + BCN + LS [27]	91.13	95.15	83.70
HRNet + TransConv2D + SAM [28]	93.7	94.3	82.8
ABINet + + [5]	97.1	97.1	89.2
NRSTRNet [12]	88.5	92.8	73.5
Dual mutual attention transformer [4]	96.6	96.4	81.6
Proposed IDTDE method	98.14	98.89	90.47

4.2 Comparison of output Images of IDTDE method method

This section demonstrates the performance analysis of IDTDE method and the efficiency of IDTDE method is compared with various methods.

Table 2 shows the outcomes of proposed STR model including preprocessing, text classification and character recognition process on three datasets, In Table 2, it is illustrated that the input images encompass various perspectives, including different camera angles, lighting conditions, image distortions, occlusions, and varying text qualities. Furthermore, as depicted in Table 2, the proposed IDTDE model demonstrates accurate character identification across diverse perspectives. This model is structured around three distinct modules. Firstly, the integration of the improved SA module into the TSA module enables the representation of textual features in a richer and more comprehensive manner by embedding the text at multiple levels, enhancing the model's ability to capture intricate textual characteristics. Secondly, the utilization of the VSA module facilitates the capture of internal dependencies between global contextual information and local features. Lastly, the inclusion of the TVC module allows for the capture of visual context and spatial relationships within the image, enhancing the model's ability to accurately recognize and understand scene text. Overall, the combination of these modules facilitates robust and accurate scene text recognition by leveraging both textual and visual cues effectively.

4.3 Performance analysis of IDTDE method with various existing methods

Table 3 shows the classification outcomes of IDTDE method, on three datasets such as IIIT5K dataset, IC13, IC15 dataset.

Table 3 presents the performance of various techniques in the task of STR across three datasets: IIIT5K, IC13, and IC15. Higher accuracy values reflect better performance in accurately recognizing text within scenes. Techniques such as Optimal Adaptive Threshold-based Self-Attention (OATSA), RMFPN, and Autonomous, Bidirectional and

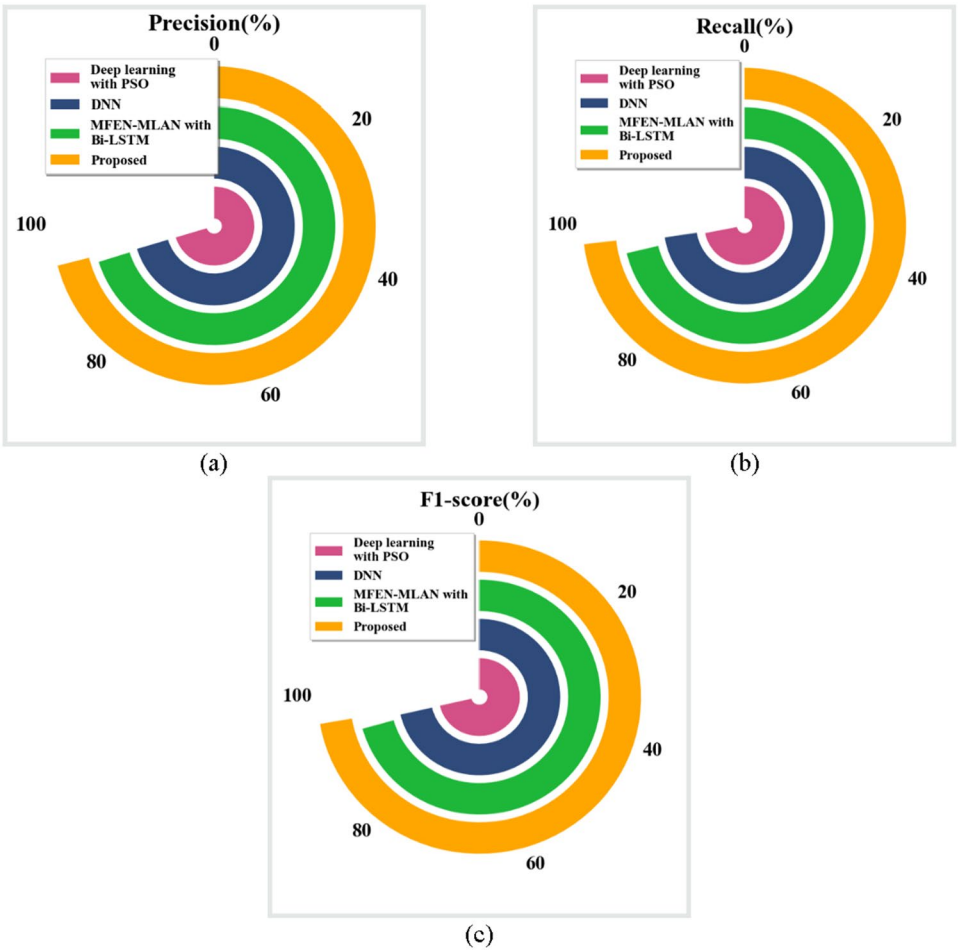


Fig. 2 Performance comparison on IIIT5K dataset **a** Precision **b** Recall **d** F1-score

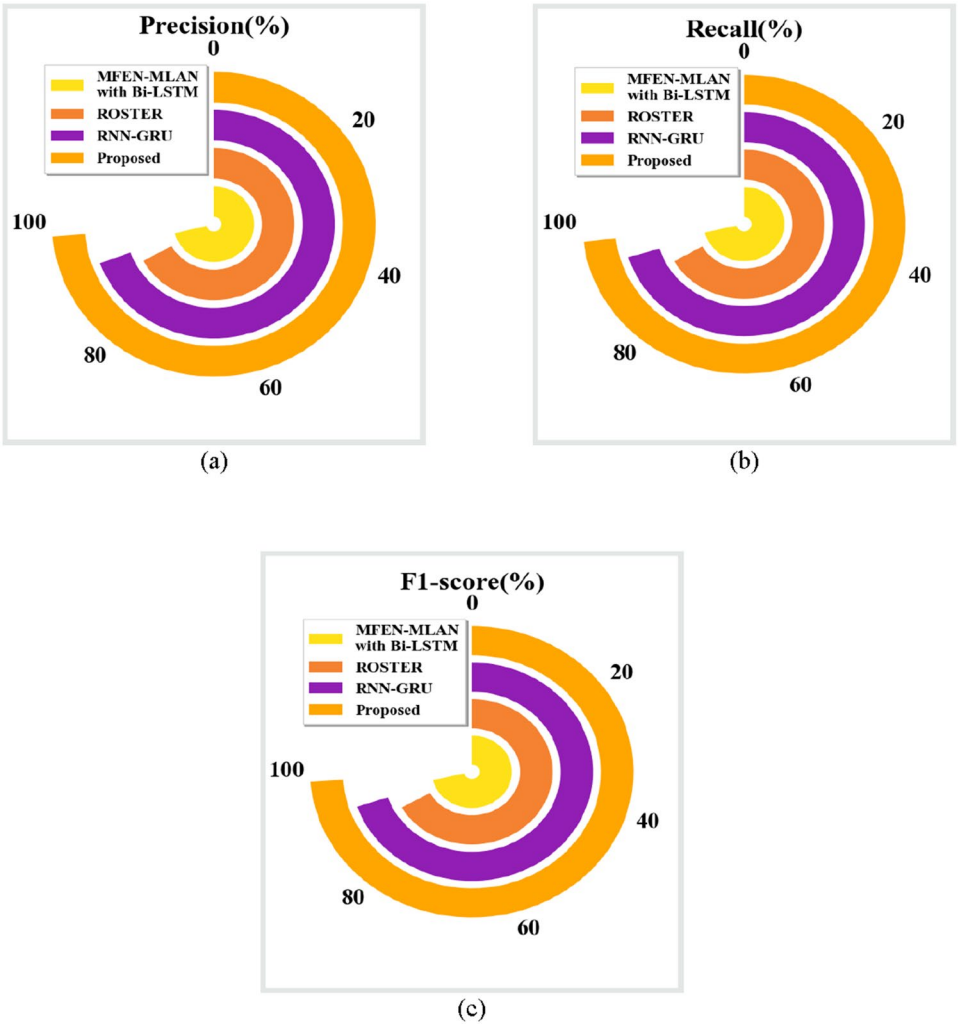


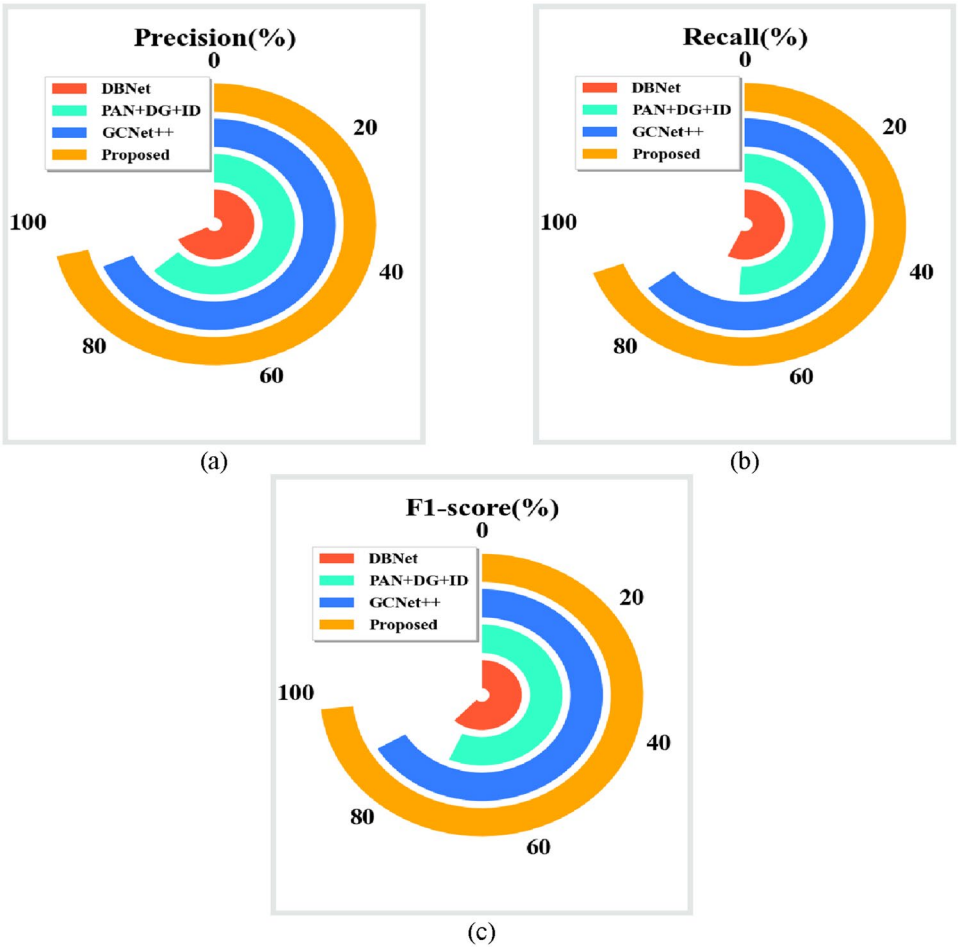
Fig. 3 Performance comparison on IC13 dataset **a** Precision **b** Recall **d** F1-score

Iterative Network (ABINet++) consistently exhibit high accuracy across all datasets, suggesting their effectiveness in STR. Conversely, techniques like NRSTRNet show comparatively lower accuracy, suggesting potential limitations in their ability to recognize text within scenes. Notably, the proposed IDTDE method stands out with exceptionally high accuracy scores across all datasets, underscoring its effectiveness and promise for STR tasks.

Figure 2 compares the performance metrics of various techniques on the IIIT5K dataset for STR. However, the proposed IDTED model outperforms the other techniques such as DL with Particle Swarm Optimization (PSO) [29], Deep Neural Network (DNN) [9] and Multi-Layer Feature Aggregation Neural Network-Manifold Feature Extraction Neural Network (MFEN-MLAN) with Bi-directional Long Short Term Memory (Bi-LSTM) [30], achieving the highest precision of 94.65 %, recall of 97.49 %, and F1-score of 96.34%, showcasing its superior performance in STR tasks on the IIIT5K dataset.

Figure 3 illustrates the performance metrics of different techniques on the IC13 dataset for STR. Compared to existing techniques such as MFEN-MLAN with Bi-LSTM [30], Robust Scene Text Recognition (ROSTER) [10], and RNN-GRU, the proposed IDTDE technique outperforms them all, achieving the highest precision of 95.21%, recall of 96.45%, and F1-score of 95.85%. This demonstrates its superior performance in STR tasks on this dataset.

Figure 4 presents a comparison of performance metrics for various techniques on IC15 dataset in a STR task. Techniques such as DBNet [31], Pixel Aggregation Network+ Domain Generalization + Inverted Dice loss (PAN+DG+ID) [32], Grouped Channel Composition Network (GCNet++) [33], and the proposed method achieved precision scores of 90.60%, 84.68%, 91.28%, and 92.78%, recall scores of 76.00%, 68.13%, 86.27%, and 88.50%, and F1-scores of 82.70%, 75.51%, 88.71%, and 89.53%, respectively. The proposed IDTDE method stands out with the highest performance metrics, indicating its efficiency



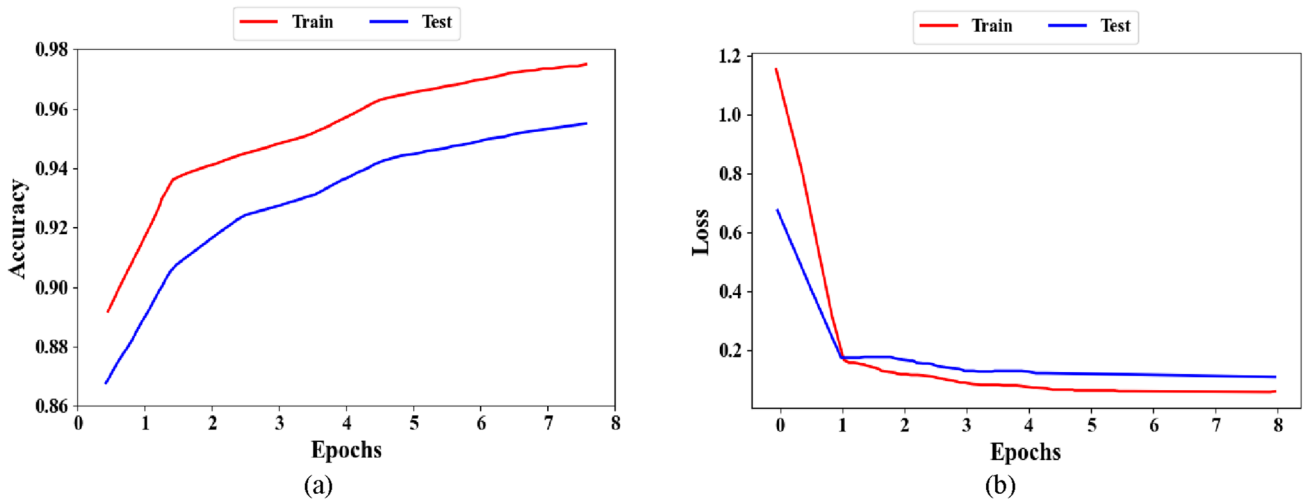


Fig. 4 Performance comparison on IC15 dataset **a** Precision **b** Recall **d** F1-score

Fig. 5 Training and testing graph **a** Accuracy **b** loss

in accurately recognizing scene text compared to the other techniques.

Figure 5 visualizes the accuracy and loss curves to assess the IDTDE model's accuracy during training and testing processes in the context of STR tasks. The accuracy curve represents the IDTDE's ability to accurately identify text in scene images over the course of training epochs or iterations. As the accuracy curve rises, it specifies that the IDTDE

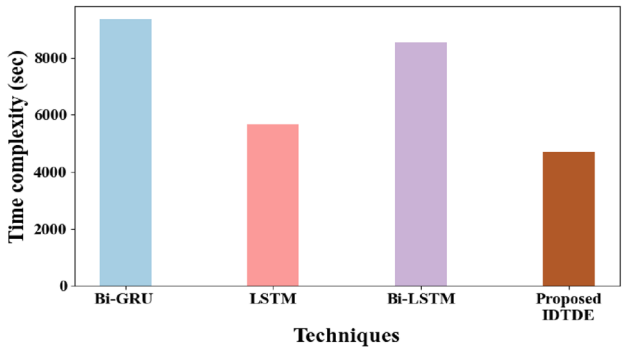


Fig. 6 Time complexity analysis

Table 4 Ablation study on all datasets

Model	Accuracy (%)		
	IIIT5K	IC13	IC15
Improved TSA	90.11	92.25	85.93
VSA	89.45	90.13	83.46
DCA	96.24	95.25	88.61
IDTDE (Proposed model)	98.14	98.89	90.47

technique is improving its ability to accurately recognize text in scenes. Also, the loss curve measures the discrepancy between the model's predicted text and the actual text in scene images. It represents how well the model is minimizing errors during training. A decreasing loss curve indicates that the model is progressively reducing its prediction errors, which is desirable for effective STR.

In the comparison of different techniques, Fig. 6 explores into the time complexities associated with various architectural configurations. It is notable that simpler variants, including models utilizing only Bi-LSTM or LSTM [18], tend to exhibit higher time complexities in comparison to the IDTDE model. This suggests that the IDTDE model offer computational efficiency advantages over alternative architectures, making them more suitable for STR task.

4.4 Ablation study

In the IDTDE model, ablation experiment is conducted on the IIIT5K, IC13, and IC15 datasets to investigate the effectiveness of individual components. Each component is isolated by fixing one part of the IDTDE model while altering the others. The experimental results revealed that every element of the IDTDE technique plays a crucial role in enhancing STR performance.

- The Improved TSA utilized only textual double embedding and enhanced SA without employing text visual co attention.
- VSA only relied on visual SA without text visual co attention.
- DCA indicated the exclusion of SA mechanisms for text images, utilizing only dual text visual co attention for prediction. Table 4 shows the ablation study on three datasets.

Across the IIIT5K, IC13, and IC15 datasets, the experiments highlighted the significance of improved SA networks in learning essential features, with textual SA yielding greater accuracy gains in scene text detection compared to VSA. Additionally, the dual SA mechanism from text images showed superior performance over TSA and VSA. These findings underscore the importance of SA mechanisms for capturing internal and global features of images, thereby aiding in scene text detection.

5 Conclusion

In this research, the IDTDE with AOA method is proposed for effective text recognition. The input image is sourced from various datasets, typically undergoing preprocessing via the structural interval gradient filtering technique. Optimal features are then extracted using the TP-DWT technique. These extracted features facilitate effective text classification and character identification through the RTACNN and IDTDE network. The AOA is employed to optimize the network for optimal text extraction. Experimental outcomes showcase that the developed approach accomplishes remarkable accuracy rates of 98.14%, 98.89%, and 90.47% on the IIIT5K, IC13, and IC15 datasets, respectively. The effectiveness of the IDTDE method is expected to be enhanced by incorporating new optimizations in the future research. In future endeavors, there will be a focus on enhancing the scalability of the algorithm, further optimizing its efficiency, expanding its application to diverse interdisciplinary fields, and integrating features of varying scales to facilitate large-scale data validation.

Author contributions All the authors have contributed equally to the work.

Funding This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability Data will be made available on reasonable request.

Declarations

Conflict of interest The authors declare that they have no potential conflict of interest.

Statement of animal and human rights All applicable institutional and/or national guidelines for the care and use of animals were followed.

Informed consent For this type of analysis formal consent is not needed.

References

1. Lu N, Yu W, Qi X, Chen Y, Gong P, Xiao R, Bai X (2021) Master: multi-aspect non-local network for scene text recognition. Pattern Recogn 117:107980
2. Selvam P, Koilraj JA, Romero CA, Alharbi M, Mehbodniya A, Webber JL, Sengan S (2022) A transformer-based framework for scene text recognition. IEEE Access 10:100895–100910
3. Mahadshetti R, Lee G-S, Choi D-J (2023) RMFPN: end-to-end scene text recognition using multi-feature Pyramid Network. IEEE Access 11:61892–61900
4. Liu Z, Wang L, Qiao J (2022) Visual and semantic ensemble for scene text recognition with gated dual mutual attention. Int J Multimed Inf Retrieval 11:669–680
5. Fang S, Mao Z, Xie H, Wang Y, Yan C, Zhang Y (2023) Abinet++: Autonomous, bidirectional and iterative language modeling for scene text spotting. IEEE Trans Pattern Anal Mach Intell 45:7123–7141
6. Du Y, Chen Z, Jia C, Yin X, Li C, Du Y, Jiang Y (2023) Context Perception Parallel Decoder for Scene Text Recognition. arXiv preprint arXiv:2307.12270
7. Ma J, Guo S, Zhang L (2023) Text prior guided scene text image super-resolution. IEEE Trans Image Process 32:1341–1353
8. Yang X, Silamu W, Xu M, Li Y (2023) Display-semantic transformer for scene text recognition. Sens 23:8159
9. Pandey B, Pandey D, Wariya S, Agarwal G (2021) A deep neural network-based approach for extracting textual images from deteriorate images. EAI Endorsed Trans Ind Netw Intell Syst 8:170961
10. Francis LM, Sreenath N (2022) Robust scene text recognition: using manifold regularized twin-support vector machine. J King Saud Univ Comput Inf Sci 34:589–604
11. Heng H, Li P, Guan T, Yang T (2022) Scene text recognition via context modeling for low-quality image in logistics industry. Complex Intell Syst 9:3229–3248
12. Yue H, Huang Y, Vong C-M, Jin Y, Zeng Z, Yu M, Chen C (2023) NRSTRNet: a novel network for noise-robust scene text recognition. Int J Comput Intell Syst. https://doi.org/10.1007/s44196-023-00181-1
13. Fanjie K, Yaqi L, Miaomiao X, Silamu W, Yanbing L (2023) Sust and rust: two datasets for uyghur scene text recognition. IEEE Access 11:126209–126220
14. Patel G, Kim T, Lin Q, Allebach JP, Qiu Q (2024) Self-attention enhanced recognition: a unified model for handwriting and scene-text recognition with improved inference. Electron Imaging. https://doi.org/10.2352/ei.2024.36.8.image-241
15. Long S, He X, Yao C (2021) Scene text detection and recognition: the Deep Learning Era. Int J Comput Vision 129:161–184
16. Lee S, Yu W, Yang C (2022) ILBPSDNet: based on improved local binary pattern shallow deep convolutional neural network for character recognition. IET Image Process 16:669–680
17. Liu C, Yang C, Qin H-B, Zhu X, Liu C-L, Yin X-C (2023) Towards open-set text recognition via label-to-prototype learning. Pattern Recognit 134:109109
18. Alshawi AA, Tanha J, Balafar MA (2024) An attention-based convolutional recurrent neural networks for scene text recognition. IEEE Access 12:8123–8134
19. Zhou Q, Gao J, Yuan Y, Wang Q (2024) RRTrN: a Lightweight and effective backbone for scene text recognition. Exp Syst Appl 243:122769
20. Banerjee A, Shivakumara P, Bhattacharya S, Pal U, Liu C-L (2024) An end-to-end model for multi-view scene text recognition. Pattern Recognit 149:110206
21. Kumar MP, Poornima B, Nagendraswamy HS, Manjunath C (2021) Structure-preserving NPR framework for Image abstraction and stylization. J Supercomput 77:8445–8513
22. Tuncer T, Dogan S, Subasi A (2020) Surface EMG signal classification using ternary pattern and discrete wavelet transform based feature extraction for hand movement recognition. Biomed Signal Process Control 58:101872
23. Karthik V, Lakshmi R, Abraham S, Ramkumar M (2023) Residual based temporal attention convolutional neural network for detection of distributed denial of service attacks in software defined network integrated vehicular adhoc network. Int J Netw Manag. https://doi.org/10.1002/nem.2256
24. Huang Y, Dai X, Yu J, Huang Z (2023) Sa-SGRU: combining improved self-attention and skip-GRU for text classification. Appl Sci 13:1296
25. Han H, Ke Z, Nie X, Dai L, Slamu W (2023) Multimodal fusion with dual-attention based on textual double-embedding networks for rumor detection. Appl Sci 13:4886
26. Abualigah L, Yousri D, AbdElaziz M, Ewees AA, Al-qaness MAA, Gandomi AH (2021) Aquila optimizer: a novel meta-heuristic optimization algorithm. Comput Ind Eng 157:107250
27. Yu W, Ibrayim M, Hamdulla A (2023) Scene text recognition based on improved CRNN. Inf 14:369
28. Li M, Li X, Sun J, Dong Y (2022) HRNet encoder and dual-branch decoder framework-based scene text recognition model. Int J Antennas Propag 2022:1–10
29. Pandey BK, Pandey D, Wariya S, Aggarwal G, Rastogi R (2021) Deep learning and particle swarm optimisation-based techniques for visually impaired humans' text recognition and identification. Augmented Human Res. https://doi.org/10.1007/s41133-021-00051-5
30. Anbukkarasi S, Sathishkumar VE, Dhivyaa CR, Cho J (2023) Enhanced feature model based hybrid neural network for text detection on signboard, Billboard and news tickers. IEEE Access 11:41524–41534
31. Li M, Zhang L, Zhou M, Han D (2023) UTTSR: a novel non-structured text table recognition model powered by deep learning technology. Appl Sci 13:7556
32. Kim T, Patel G, Lin Q, Allebach JP, Qiu Q (2024) Generalizing handwriting and scene-text detection in images. Electron Imaging. https://doi.org/10.2352/ei.2024.36.8.image-242
33. Liu C, Yang C, Hou J-B, Wu L-H, Zhu X-B, Xiao L, Yin X-C (2021) GCCNet: grouped channel composition network for scene text detection. Neurocomput 454:135–151
34. Sobhanam H, Prakash J (2023) Analysis of fine tuning the hyper parameters in RoBERTa model using genetic algorithm for text classification. Int J Inf Technol 15(7):3669–3677
35. Chaudhary M, Pruthi J, Jain VK, Suryakant (2022) A novel squirrel search clustering algorithm for text document clustering. Int J Inf Technol 14(6):3277–3286

36. Yadav AK, Singh A, Dhiman M, Vineet KR, Verma A, Yadav D (2022) Extractive text summarization using deep learning approach. Int J Inf Technol 14(5):2407–2415
37. Shekar BH, Raveeshwara S (2022) Contour feature learning for locating text in natural scene images. Int J Inf Technol 14(4):1719–1724
38. Mandal S, Singh GK, Pal A (2021) Single document text summarization technique using optimal combination of cuckoo search algorithm, sentence scoring and sentiment score. Int J Inf Technol 13(5):1805–1813
39. Mishra A, Alahari K, Jawahar CV (2012) Scene text recognition using higher order language priors. In: BMVC-British machine vision conference
40. Karatzas D, Shafait F, Uchida S, Iwamura M, i Bigorda LG, Mestre SR, Mas J, Mota DF, Almazan JA, De Las Heras LP (2013) ICDAR 2013 robust reading competition. In: 2013 12th international conference on document analysis and recognition, pp. 1484–1493
41. Karatzas D, Gomez-Bigorda L, Nicolaou A, Ghosh S, Bagdanov A, Iwamura M, Matas J, Neumann L, Chandrasekhar VR, Lu S, Shafait F (2015) ICDAR 2015 competition on robust reading. In: 2015 13th international conference on document analysis and recognition (ICDAR), pp. 1156–1160