SvedbergOpen
DISSEMINATION OF KNOWLEDGE
International Journal of Artificial Intelligence and Machine Learning
Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Designing Channel Attention Guided Stacking-Based Contextual Graph CNN For Telugu Text Categorization And Recognition With Character Segmentation Through Deformable DETR++

M Kiran Sastry[1] , S. Aparna[2]

[1]Department of Computer Science and Systems Engineering , School of Computer Science and Engineering, GITAM Deemed to be University, Rudraram, Hyderabad 502329, India
kmatti@gitam.edu
[2]Department of Computer Science and Systems Engineering , School of Computer Science and Engineering, GITAM Deemed to be University, Rudraram, Hyderabad 502329, India
ashivamp@gitam.edu

Abstract— In the present modern life, character recognition is significant to make the handwritten text into machine-readable data. Several research works have been examined and developed for the handwritten character recognition module. However, no research has investigated the regional languages, especially for the Telugu language. Hence, the character recognition on Telugu handwritten text plays an important and challenging task because of its orientation angle, varying writing pattern and different sizes of characters. The accurate identification is still become the complex task for the compound character. Recent studies developed a neural network to evaluate the discriminative qualities from the huge volume of data, which tends to be an effective solution for handwriting recognition. However, there are still some limitations in achieving an accurate recognition outcome, especially for the Telugu character. Hence, in this work, a novel deep learning-based handwritten Telugu character classification and recognition model is designed with segmentation procedures. In the beginning, the significant Telugu text images for the validation are sourced from the benchmark resources. Next, the collected Telugu text images are forwarded to the character segmentation phase. Here, Deformable DEtection TRansformer++ (DDERT++) is employed to carry out the segmentation procedures. Once the images are segmented, and then forwarded to the Telugu text categorization and recognition phase. In this phase, an Adaptive and Channel attention guided Stacking based Contextual Graph Convolutional Neural Network (ACS-CGCNN) is used to carry out the Telugu text categorization and recognition procedure. Moreover, the parameters of ACS-CGCNN are optimized using Mutated Hunting-based Groupers and Moray Eels Optimization (MHGMEO), which used to improve the accuracy of Telugu text categorization and recognition. Later, different investigation are conducted in the developed Telugu text categorization and model over the existing schemes under different experimental conditions.

Keywords—Telugu Text Categorization; Character Segmentation; Deformable DEtection TRansformer++; Adaptive and Channel attention guided Stacking based Contextual Graph Convolutional Neural Network; Mutated Hunting-based Groupers and Moray Eels Optimization.

I. INTRODUCTION

In recent times, the development of computing and mobile devices has led to the need for effective character recognition. Character recognition is the process of identifying the printed and handwritten text from diverse documents [9]. However, the recognition becomes a challenging task due to the different writing styles of each person. Due to the differentiation in the orientation, forms and writing technique, it is difficult to develop a better classification model [10]. Different approaches are suggested in existing articles for diverse scripts like English, Chinese and Arabic for character recognition [11]. Telugu is one of the Dravidian languages, which is widely used in the Telangana state in India [12]. There are 16 vowels in the Telugu alphabet that contain both the intricate and straightforward character [13]. The offline recognition of Telugu characters is performed by providing the information to the computer based on an image format [14]. Here, the information retrieval equipment, such as a printer, is needed, and it is complex for accurate recognition due to the limited sequential details [15]. Most of the Telugu characters are constructed by overlapping strokes and intricate connections, and it is more complex to separate them accordingly.

There are several challenges in performing the precise recognition, especially for the Telugu character [16]. This is because of the limited evaluation measures developed for the Telugu character recognition [17]. In addition, a lack of a training dataset specifically for the Telugu handwritten characteristic affects the detection performance. Different styles of handwriting tend to complicate accurate segmentation [18]. For effective recognition, the machine learning models are developed in recent studies, which provide quick recognition even for the larger files [19]. There has been tremendous development attained for the machine learning approaches, which perform the Optical Character Recognition (OCR) [20]. However, these models face issues related to processing large documents and detecting the overlapped character [21]. Moreover, the character variation from each document poses more challenges in performing the recognition.

For the image processing application, the deep learning tools are widely used in recent times, especially the CNN models, which perform global feature extraction through their in-built convolution layers [22]. While utilizing this CNN as the base structure, several advanced models are developed specially for the handwritten character recognition, which are highly adaptable to varying image scalability [23]. This enhances the recognition rate based on its multi-scale sampling. The LSTM, a deep learning network, is also used in some research work for character recognition, which manages the different character shapes in varying font sizes [24]. Additionally, the multilingual text in Telugu is harder to detect because it requires a maximum number of scripts to perform effective recognition [25]. In order to address these issues, a novel classification with integrated segmentation is developed in this work for the Telugu handwritten text recognition.

The major contributions of this article are discussed below:

- To construct an intelligent deep learning framework for the accurate recognition and categorization of Telugu handwritten text. Each character from the handwritten images is precisely extracted using the advanced segmentation model. The effective classification is performed using the attention-aided deep learning technique. The framework ensures robust handling of different styles of Telugu characters and provides reliable classification. Overall, the high-precision solution from the presented model for Telugu character recognition helps in a language processing system and acts as an educational tool.
- To build an advanced DDERT++ approach for the accurate segmentation of handwritten Telugu characters based on the image. The technique is proposed to improve the DERT++ with deformable attention to give more attention to informative spatial regions. The Transformer-based encoder-decoder structure effectively processes the input images, and the key points are selectively evaluated using the attention mechanism to enhance the segmentation. The robust and precise character-level segmentation attained using DDERT++ significantly enhances the overall performance in final classification.
- To develop a robust ACS-CGCNN for the accurate classification of Telugu characters from the segmented images. The architecture is the combination of the CGCNN with the channel attention and an added optimization strategy, which helps in evaluating the local and global dependencies to enhance the classification. Here, the informative patterns are emphasized to differentiate the compound structure of the character. The context-aware and high-precision classification attained using the ACS-CGCNN enhances the recognition accuracy of different handwriting styles.
- To design an MHGMEO algorithm for optimal tuning of the hyperparameter from the ACS-CGCNN in achieving accurate classification of Telugu characters. The ACS-CGCNN is influenced by the standard GMEO algorithm, which is well-known for its grouping mechanism. Here, the position updating phase of the search agent is modified by interchanging the random factor with the corresponding fitness value. This adaptive MHGMEO enhance the convergence stability by optimally adjusting the critical parameter in ACS-CGCNN to attain robust recognition of Telugu handwritten characters.

The remaining section of the architecture is arranged below: Section II reviewed the previous research work on handwritten character classification. Section III detailed the existing challenges, workflow of the presented model, along with the dataset details. Section IV discussed the effective segmentation task. Section V provides a eloborated view of the final classification task. Section VI structures the experimental analysis with standard comparative models. Section VII conclude the work.

II. EXISTING WORKS

A. Related Works

In 2025, Limbachiya and Sharma [1] have designed a hierarchical classification model that included a multi-branch convolution network for handwritten character recognition. The pre-trained mobile network was integrated. The sample characters were pre-processed and segmented using the corresponding neural network. The model could effectively prevent the overfitting issue, attained a high classification outcome and reduced the misclassification.

In 2025, Khayati et al. [2] have developed a transfer learning integrated CNN to improve the character recognition of the Arabic handwriting. The involvement of transfer learning effectively tackles the dataset scarcity and computational requirement issues. Further, the lightweight advanced neural network was further initiated to perform the classification with superior accuracy. The outcome clearly defined the potential of the combinational approach in understanding the handwritten character and generating an robust outcome.

In 2023, Li et al. [3] have recommended a new coupling CNN with the integration of a sequence framework for recognizing the handwritten characteristics. This two-stage framework effectively addresses the high fragility, even for the poorly written character. The contextual information fusion model was further implemented to enhance the robustness of the framework by clearly determining the broken strokes and sloppy writing.

In 2024, Kim et al. [4] have suggested a federated learning-based classification framework that could effectively categorize the handwritten characteristics even with the complex structure. The testing was performed to validate the implemented framework in terms of global and local averages. Some of the standard neural networks were utilized to determine the performance of the approach. The outcome confirmed the enhanced characteristic of the suggested model in character recognition.

In 2022, Gonwirat and Surinta [5] have presented a robust generative framework for handwritten character recognition. Here, the generative module was integrated with the CNN model to develop the hybrid classification network. This model could be capable of improving the recognition performance by solving the noise issues. The architecture was evaluated by the experimental investigation, and the result shows the models efficiency.

In 2025, Sastry, and Aparna [6] have designed the segmentation and classification module for the Telugu handwritten character, which could effectively determine the character with a different length sequence. The handwritten images were collected and initially processed through the Yolov5. Here, the segmentation was carried out and generated the optimal outcome. From the segmented image, the ROI was extracted and processed through the CNN-based classification model. This framework effectively determined the sequential feature and categorized the text.

In 2025, Banavatu, and Parthasarathy [7] have introduced the automatic recognition model to evaluate the South Indian languages. The standard source was used to gather the handwritten images. Here, the active contour was involved to segment the phase, where the textual patterns were generated. The generated textual pattern was finally classified using the cascaded network. The model could effectively identify the curves in the corresponding text and generate an accurate outcome.

In 2025, Livingston et al. [8] have employed the new hybrid recognition framework that integrated the neural network along with the optimization strategy for categorizing the handwritten characteristics. The modified algorithm was utilized here to segment the digitized text after the pre-processing. The varying influential features were extracted using adaptive optimization. The resultant features were further processed through the neural network to perform the handwritten character recognition.

B. Research Gaps and Challenges

Recognizing the Telugu text characters plays a major role in enhancing communication. Conventional character recognition schemes face more complications due to the complex writing structure and similarities, which may affect the accuracy of recognition. Existing text classification and recognition models face more complications due to Gaussian noise and low resolutions. Several limitations take place while designing the Telugu text classification and recognition are given as follows.

- Conventional Telugu text classification and prediction models face more difficulties due to noise penalties such as lower contrast, Gaussian noise, blurriness and lower resolution in the input samples, which affects the overall recognition efficiency.
- Changing the handwritten Telugu text into machine-readable characters is challenging due to intricate characters, complex scripts and poor writing. Moreover, this process consumes more time to observe the handwritten characters in a complex environment.

- Variability issues arise due to varying writing skills of individuals for each letter of the alphabet, which may create confusion. In addition, the overlapping characters generate complications at the training phase, and this makes the execution complex.
- Maintaining contextual understanding while collecting the longer-range formation is difficult, and it also needs higher-quality samples for the experimentation to accomplish more precise outcomes in varying environments.
- Managing the sequential information and managing the variations affects the final outcomes of the system. In addition, the network requires to manage the maximm robustness and generalization for complex classes.

Based on these existing limitations, an advanced Telugu text categorization and recognition framework will be constructed in this article. The merits and demerits associated with the standard Telugu text categorization and recognition models are provided in Table I.

TABLE I. FEATURES AND CHALLENGES OF CLASSICAL TELUGU TEXT CATEGORIZATION AND RECOGNITION MODEL

Author [citation]	Methodology	Features	Challenges
Limbachiya and Sharma [1]	MobileNet-VGG16	• It is efficient in performing the fine-grained classification in the collected samples. • It is good at tackling the misclassifications and minimize the complexities.	• It requires to enhance the accuracy while executing the complicated tasks. • Its processing time is higher due to its complex structure.
Khayati et al. [2]	ShuffleNet	• It is effective in minimizing the variability issues and achieves higher classification accuracy. • It effectively maintains the generalizability and enhances the robustness.	• It faces overfitting issues that arise during the training phase. • It does not have reliability over a wide range of samples.
Li et al. [3]	DSCIFN	• It is capable of maintaining robustness over a diverse samples. • It is effective at understanding the multi-scale features.	• Its validation expense is maximum and also requires deep training. • It faces redundancy issues.
Kim et al. [4]	ResNet50-CNN	• It is efficient in acquiring the required features in higher quality.	• It needs to overcome the vanishing gradient issues. • It requires reducing the training time while processing complex data patterns.
Gonwirat and Surinta [5]	DeblurGAN-CNN	• It is good at eliminating the noisy details presented in the data samples and quickly learning the required feature details.	• Validation expense needs to be reduced while handling the high-quality samples.
Sastry and Aparna [6]	CNN-RA-LSTM	• Computational complexity problem that caused in the network are tackled.	• It required an enormous amount of time to process the required samples.
Banavatu, and Parthasarathy [7]	HSCDNet	• It offers better outcomes by computing the textural patterns.	• It has poor training efficiency in varying classes.
Livingston et al. [8]	FFBNN	• It is capable of classifying the required characters in minimal time and also gains more precise outcomes. • It is good at making better decisions.	• Improving the training speed is essential to eliminate the delay issues. • Understanding the complex data pattern takes more time.

III. DESIGN DETAILS OF THE DEVELOPED TELUGU TEXT CATEGORIZATION AND RECOGNITION FRAMEWORK USING DEEP LEARNING NETWORK

A. Challenges of Traditional Text Categorization Models

Telugu character recognition is the kind of OCR that helps in detecting the handwritten letters of the Telugu language automatically from an image document. Some of the basic elements are used to form the Telugu characters like modifiers, vowels and consonants. This combination of elements is generally formed by a single connected symbol in the handwritten content, which increases the difficulty of the recognition. Stroke variation is one of the other important considerations. Different stroke orders are used to write the single Telugu character, where the continuity and thickness of the strokes are differentiated due to the writing tools and pen pressure. The script contains arcs and loops, where small changes in the corresponding shape change the whole meaning of the character. Most of the Telugu characters are very similar and are differentiated only by minor details, which maximizes the chance of misclassification. Several deep learning models, such as CNN, SqueezeNet and Residual network, are used to perform accurate recognition. These models can determine the complex features from the handwritten images. However, the existing framework still has several limitations. Most of the models require larger labelled training data, which is not sufficient for the Telugu language. There is a chance of misclassification in analysing the similar-looking character, even with the advanced model. The recognition accuracy is highly affected by the average preprocessing in the segmentation phase. Hence, this work designed an deep learning architecture for Telugu handwritten character recognition, which integrated the segmentation and classification, aiming to enhance accuracy in Telugu script.

B. Architectural Illustration of the Presented Telugu Text Categorization Model

This architecture presented an advanced deep learning strategy for the accurate recognition of the Telugu handwritten text. The framework involved the image segmentation and classification task. Initial stage, handwritten Telugu characters are retrieved from available resource, which includes several characters with different orientations and styles. The gatherer images are initially send to segmentation phase, where the process is performed by DDETR++ technique. Each character is effectively localized in this segmentation with the help of the deformable attention. The integration of the attention is effective for the character with the curves, overlapping components and strokes. The effective feature aggregation in the network tends to precise segmentation for both the small and large characters. The character level representation from the accurate segmentation is further provided to the final stage. Here, the classification is performed using the new ACS-CGCNN. The framework takes each segmentation character as a node through its graph structure. From the character, the contextual relationship is determined using the edges. The refined contextual features are extracted progressively using the stacked graph convolution. This helps in aggregating the information from the neighbouring nodes. Moreover, the integration of channel attention helps in suppressing irrelevant information and improving the discriminative features. The incorporation of contextual and attention-guided feature determination enhances the complex character representation. The classification performance is further strengthened by the MHGMEO algorithm which performes the hyperparameter optimization. This optimization improves the overall convergence behaviour and minimizes the false discoveries. Finally, the result is achieved through the final dense classification layer. The results ensure the accurate classification of the Telugu character with varying structure and style. The overall workflow of the implemented deep learning-based Telugu text categorization is difened in Fig. 1 .

C. Description of Telugu Text Image Dataset

It is significant to have a high-quality annotated dataset for the handwritten text recognition to capture the different handwritten styles. The regional Indian Telugu language only has a limited dataset, which has a complex character structure. Hence, this work initiates a comprehensive Telugu dataset to perform the segmentation and recognition task.

The word-level handwritten dataset was taken from the link

"https://cvit.iiit.ac.in/research/projects/cvit-projects/indic-hw-data Access date: 2026-04-01". The dataset includes 120k handwritten Telugu words used for a handwriting recognition task. The word images in the dataset are in JPG format and are stored in the Readme files.

Here, the collected images from the dataset are denoted as Tl_c, where $c = 1,2,3,\ldots C$. The term C represents the total available images from the dataset. The sample Telugu character images from the dataset are depicted in Fig. 2

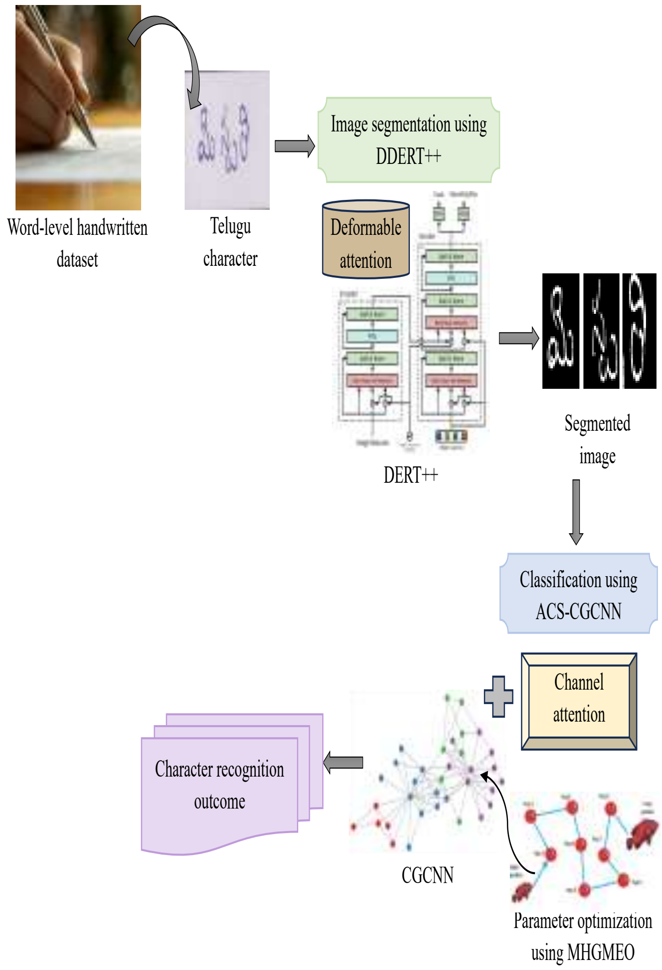


Fig. 1. Overall architectural diagram for the proposed Telugu handwritten character classification model

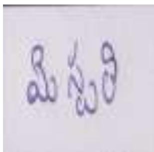
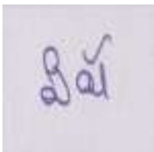
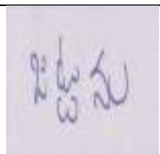
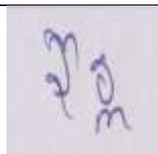
Images	1	2	3	4	5
Telugu Character					

Fig. 2. Example of a different Telugu character from the dataset.

IV. Detailed Workflow of An Efficient Transformer-based Character Segmentation Process

A. Detection Transformer++ Model

DETR++ [27] is the advanced version of the detection transformer, which is specially designed for the object detection task. Initially, the feature extraction performer for the given image is termed as $F \in \Gamma^{H \times W \times C}$. The features are flattened by the position encoding into sequences to generate the spatial structure based on Eq. (1).

$$Y_0 = Flatten(F) + P \tag{1}$$

Here, the input of the transformer encoder is indicated as Y_0. The contextual representation generated from the encoder is $Y_e \in \Gamma^{(HW) \times C}$ with the involvement of multi-head self-attention, as defined in Eq. (2).

$$Y_0 = Encoder(Y_0),\ \ Attn(Q,K,V) = soft\max\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

Further, the transformer decoder is introduced with the fixed set of learnable objects Q, which determines the encoder feature to form the object-level embedding given in Eq. (3).

$$Y_d = Decoder(Q, Y_e) \quad (3)$$

Each decoder embedding is related to the potential object instance. In addition, the iterative bounding box employed by the DETR++across the decoder layer to improve the localization stability, which tends to progressive adjustment on detection using Eq. (4).

$$g^l = g^{l-1} + \Delta g^l \quad (4)$$

Here, the refined bounding box is represented as g^l at the l^{th} decoder layer. The term Δg^l is the predicted offset. The set of object instances resulted as the final output, which effectively captures the global contextual relationship to perform accurate segmentation.

B. Character Segmentation using DDETR++

In performing the Telugu handwritten recognition task, it is crucial to have character segmentation because of the complex structure of the Telugu character. Hence, the DDETR++ is developed for the accurate segmentation of the Telugu character, which is essential for subsequent recognition.

The designed DDETR++ is developed by the integration of deformable attention with the DETR++. Here, the original Telugu character images $Tl_c \in I \in \Gamma^{H \times W \times 3}$ are given as input to the DDETR++. The images are processed through the positional encoding to flatten the images, which also preserves the spatial information. Instead of the conventional attention, the deformable attention is included to give more attention towards the sparse set of key sampling. This enhances the sensitivity with less computation, even for the small component. This manages the precision localization with the use of less computation. The deformable attention is in Eq. (5).

$$DeforAtt(z) = \sum_{n=1}^{N} X_n \sum_{k=1}^{K} B_{nk}.F(q + \Delta q_{nk}) \quad (5)$$

Here, the number of attention heads is denoted as N. The attention weight and the sampled key point are specified as B_{nk} and K, where the query vector is $z \in \Gamma^c$. The informative regions are dynamically attended in terms of stroke intersection, small diacritics and fine curves. In addition, the deformable sampling learnable offset is indicated as Δq_{nk}. The informative spatial location is ensured by this query feature to evaluate the irregular pattern in the writing pattern.

The bounding boxes are refined effectively by the decoder after the deformable attention. The precise segmentation Tl_c^{Seg} is generated by the iterative refinement for each character. The system also combines multi-scale feature aggregation, which ensures better detection of the characteristic. The feature map is adaptively weighted through this attention mechanism for recognizing both the larger and smaller characters accurately. The original spatial relationship is retained by this segmented character, which also preserves the contextual arrangement of the text.

The presented DDETR++ can manage different characters with varying sizes, and even for overlapping strokes. Through the deformable attention, the model effectively focuses on the informative spatial region, which also minimizes the redundancy in the feature processing. The high-quality character segmentation attained from the DDETR++ directly improves the recognition. The structural representation of the DDETR++ for the segmentation is given in Fig. 3.

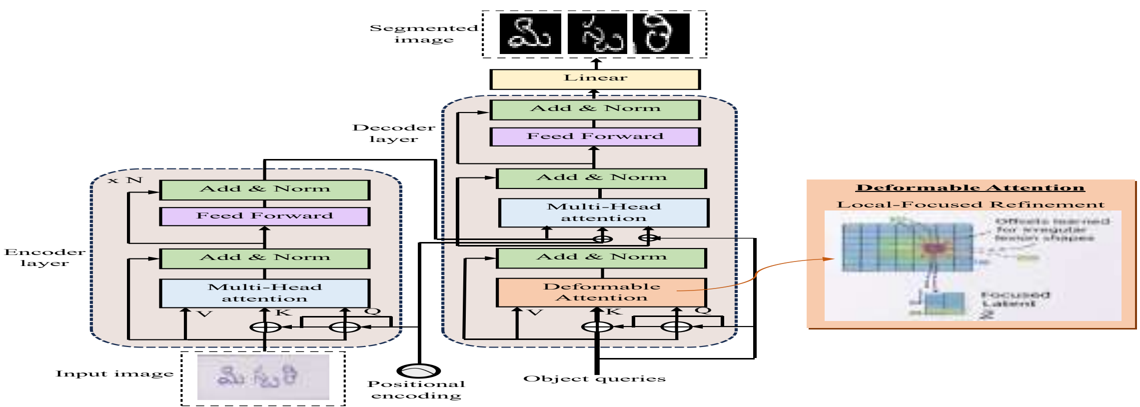


Fig. 3. The schematic representation of DDETR++ for image segmentation

V. Schematic Representation of Telugu Text Categorization and Recognition Process with its Objective Function

A. Contextual Graph CNN

CGCNN [28] is mainly constructed to evaluate the complex dependencies and relationships among the structured data through the nodes in a graph. While performing the detection task, each detected region or object is considered a node. The contextual relationship encoded the edges among the nodes in terms of interaction patterns and semantic similarity. The graph $\zeta = (\nu, \varepsilon)$ is formed with the edges ε and nodes $\nu = \{1,2,\ldots,N\}$, which define the contextual connectivity of the object. The graph convolution operation in CGCNN is performed by information propagation. Here, the nodes are updated through the aggregation of information from their neighbours. The graph convolution is mathematically derived in Eq. (6)

$$G^{l+1} = \sigma\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}G^{l}X^{l}\right) \qquad (6)$$

Here, the adjacency matrix A is added with the self-connection with the degree matrix D. The non-linear activation function and the learnable weight are termed as $\sigma(.)$ and X. Each node is updated by this operation to aggregate the context from the neighbours. The adaptability of the system is enhanced through the attention mechanism that dynamically allows the system to weight contributions. The attention-guided graph convolution is formulated in Eq. (7).

$$g_i^{l+1} = \sigma\left(\sum_{j\in N(i)} \alpha_{ij} X^{l} g_j^{l}\right) \qquad (7)$$

Here, the set of neighbour nodes and the attention coefficient are represented as $N(i)$ and α_{ij}. The global and local contextual dependencies are captured using the CGCNN based on the integration of attention-guided aggregation with the graph convolution. This is a powerful tool for the detection task. Illustration of the CGCNN model is depicted in Fig. 4.

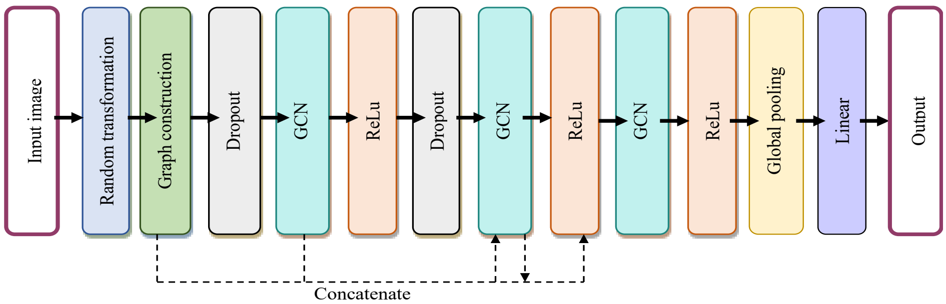


Fig. 4. Diagram structure of CGCNN

B. Developed ACS-CGCNN for Telugu Text Categorization

In a handwritten text recognition task, it is important to have an accurate classification from the segmented image for a robust downstream application. The subtle contextual variation, compound structures and diverse writing style pose more challenges to recognize the Telugu text. Hence, the advanced model called ACS-CGCNN is developed in this work to enhance the convergence stability and classification accuracy.

The main aim of this ACS-CGCNN is to evaluate the semantic and spatial dependencies among the characters by the combination of an effective attention mechanism and parameter optimization to enhance the accuracy. The segmented character Tl_c^{Seg} from the DDETR++ is given as input to the ACS-CGCNN. The graph convolution initially processes this input. Here, the segmented image is converted into a rich contextual representation through the neighbouring nodes, which aggregate the information. The feature is progressively refined by the stacking of successive layers. Each layer is combined with the channel attention [29] mechanism. Here, the significance of the different feature channels is recalibrated. The global average pooling is applied to develop the channel attention module using Eq. (8).

$$z_c = \frac{1}{H \times W} \sum_{o=1}^{H} \sum_{j=-1}^{W} F_c(i,j) \tag{8}$$

Here, the channel dimension is termed as c. The two-layer fully connected block further processes the descriptor with a nonlinear activation and reduction ratio to generate the channel attention weight. Here, the less relevant features are suppressed by the informative feature channels. The network ability is enhanced by the adaptive recalibration to differentiate the subtle variation. The CS-CGCNN framework is generally sensitive to hyperparameters and suffers from suboptimal convergence. This led to an increase in the false discoveries. Hence, to solve these complexities, the MHGMEO is initiated to fine-tune the critical hyperparameter and enhance the classification capability. The objective function for the MHGMEO is in Eq. (9).

$$Ob = \underset{\{Hn^{Gc}, Lr^{Gc}, Ne^{Gc}\}}{\arg\max} \left[\Pr e + \frac{1}{Fdr} \right] \tag{9}$$

Here, some of the hyperparameters from the GCNN model, such as hidden neuron Hn^{Gc}, learning rate Lr^{Gc} and number of epochs Ne^{Gc} are optimally adjusted using the MHGMEO algorithm within the range [5-255], [0.01-0.99], and [5-50]. This optimization process lead to improve precision $\Pr e$ with the less False Discovery Rate Fdr.

Precision is the evaluation measure implemented to analyze the accuracy of the classification network in detecting the relevant instance, and it is measured using Eq. (10).

$$\Pr e = \frac{Al^p}{Al^p + Un^p} \tag{10}$$

FDR calculates the false positives proportions among the predicted instances, and it is formulated in Eq. (11).

$$FDR = \frac{Un^p}{Un^p + Al^p} \tag{11}$$

The term Al^p and Al^n represent the true positive and true negative. Un^p and Un^n denotes the false positive and false negative. The optimization strategy tends the system towards a configuration that enhances the classification precision. IN addition, stable and faster convergence is ensured during the training through this adaptive concept. With the attention-guided feature, the tich contextual modelling is combined to perform the accurate character-level segmentation. The final graph convolution layer resulted in the classification outcome, which is followed by a dense layer, which maps the learned node embedding to class probabilities.

The technique better captures the global and local dependencies using the graph convolution, which helps in determining the complex Telugu character. This combinational framework of intelligent parameter optimization, attention-guided recalibrating and context-aware feature learning effectively evaluates the intricate and diverse Telugu handwritten characters. The superior recognition accuracy was attained through this ACS-CGCNN for different writing styles. The overall architecture of the ACS-CGCNN in the Telugu character recognition task is given in Fig. 5

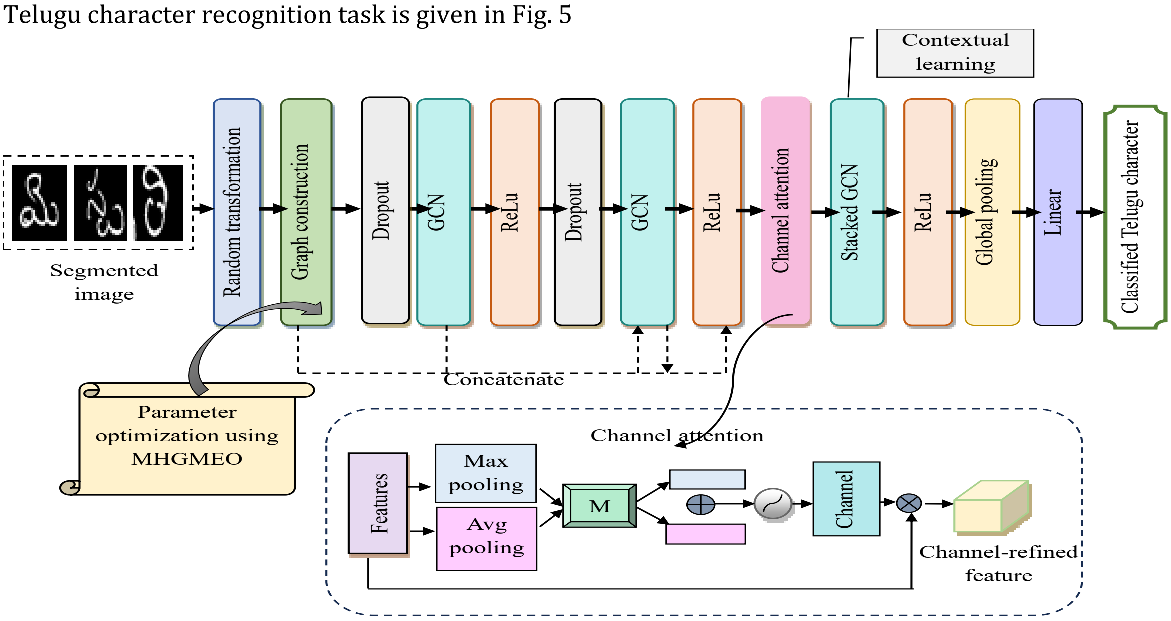


Fig. 5. Overall diagram of ACS-CGCNN for Telugu character recognition.

C. Model Updating using MHGMEO

In general, it is crucial to have the hyperparameter selection in deep learning models to attain a high-performance outcome. Hence, this work adopted the metaheuristic optimization MHGMEO for the efficient parameter tuning to enhance the overall robustness.

Purpose: The introduction of MHGMEO in the ACS-CGCNN-based classification framework is mainly to facilitate and enhance classification performance through effective parameter adaptation. The parameter tuning in the graph-based deep learning models is generally complicated because of the vast search space and interdependencies among the variables. In order to address these issues, the MHGMEO has been developed to perform robust optimization in the corresponding parameter space. During the training, the parameter values are continuously refined to advance the capability of learning. In addition, it ensures the ACS-CGCNN adapts to varying input patterns and enhances the overall consistency in achieving higher accuracy classification in Telugu handwritten character recognition.

Novelty: The developed MHGMEO is build by modifying the foundational GMEO. This GMEO [26] is influenced by the predatory strategy and hunting behaviour of the groupers in the marine ecosystem. This behaviour helps in addressing the complex issues without any gradient information. However, the standard GMEO often faces limitations like inefficient search patterns and premature convergence while dealing with high-dimensional spaces. These cause the unstable convergence due to the uncontrolled updates. To address these issues, the new MHGMEO is constructed by performing the adaptive grouping mechanism. Here, the position updating process is regulated by the modification of the random factor to enhance the convergence stability based on Eq. (12).

$$\kappa = \frac{wgc + byt}{wgc^2 + cdf + mkv} \tag{12}$$

Here, the best and worst fitness functions are denoted as byt and wgc . Similarly, the current and mean fitness function is termed as cdf and mkv . The term κ is the newly generated random value, which is updated in the primary search space of standard GMEO. Here, the eels and groupers are randomly distributed in the search space by generating a standard random value between 0 and 1. In the search space, the agent position is updated based on the particular upper ut and lower lt boundaries using Eq. (13).

$$\begin{aligned} Y_{ij}^{initial} &= lt_j + \kappa.(ut_j - lt_j), \\ i &= 1,2,3,\ldots,N,\ j = 1,2,3,\ldots,D \end{aligned} \tag{13}$$

Here, the initial position of the i^{th} search agent in j^{th} the dimension is specified as $Y_{ij}^{initial}$. The term N and D defines the number of search agents and dimensions. The random value κ in GMEO is updated using Eq. (12). The developed MHGMEO ensures an better balance among the local and global exploitation, to enhance the ability of the algorithm to escape from local optima. It is also suitable for the high-dimensional optimization problem and enhances the solution diversity. As a result, MHGMEO enhance the generalization of the overall network. The pseudocode for the MHGMEO is shown in Algorithm 1.

Algorithm 1: Designed MHGMEO
Input: Hidden neuron, learning rate and number of epochs in GCNN **Output:** Optimized hidden neuron, learning rate, and number of epochs in GCNN

Initiate the set of N search agents in MHGMEO.
Evaluate the objective function to maximize or minimize
Calculate total number of search iterations
Evaluate the number of movements in the iteration.
  While starting the iteration
    Update the random value κ using Eq. (12)
      Across the search space, distribute the groupers randomly.
      Movement of groupers in zig-zag form.
      Update the new position of each grouper based on movement using Eq. (13)
      Select the best position of the grouper.
      Calculate the objective function in assigning the best position.
      Order the group in descending order.
      Distribute the moray eels randomly.
      Find the best position
  Save the best solution

```
    end while
Return the optimal solution
```

VI. RESULTS AND DISCUSSION

A. Experimental Setup

The presented deep learning architecture for the Telugu handwritten character recognition framework was implemented in Python. The optimization strategy using the MHGMEO attained 10 population number, 3 chromosome length and 50 as the maximum iteration. From the experimental analysis, the standard models were taken to perform the comparison. For the Segmentation, models like Unet [30], Unet++ [31], ResUnet [32], and DenseUnet [33] were employed. The optimization algorithms like Artificial Bee Colony (ABC) [34], Wombat Optimization Algorithm (WOA) [35], Osprey Optimization Algorithm (OOA) [36] and GMEO [26]. Similarly, other classification techniques like MobileNet-VGG16 [1], ShuffleNet [2], ResNet50-CNN [4] and CS-CGCNN [28] [29] were used.

B. Evaluation Metrics

Accuracy, False Negative Rate (FNR), and False Positive Rate (FPR) were evaluated using Eq. (14) to (16).

$$Acy = \frac{Al^p + Al^n}{Al^p + Al^n + Un^p + Un^n} \tag{14}$$

$$FNR = \frac{Un^n}{Un^n + Al^p} \tag{15}$$

$$FPR = \frac{Un^p}{Un^p + Al^n} \tag{16}$$

Matthews Correlation Coefficient (MCC), sensitivity, and dice coefficient were determined using Eq. (17) to (19).

$$MCC = \frac{Al^p \times Al^n - Un^p \times Un^n}{\sqrt{(Al^p + Un^p)(Al^p + Un^n)(Al^n + Un^p)(Al^n + Un^n)}} \tag{17}$$

$$Sty = \frac{Al^p}{Al^p + Un^n} \tag{18}$$

$$Dic = \frac{2Al^p}{2Al^p + Un^p + Un^n} \tag{19}$$

C. Segmented image outcome of DDERT++

In order to determine the robustness of the DDERT++ in performing the Telugu handwritten text segmentation, three types of output are determined, such as original, detected and segmented images. The Fig. 6 illustrates the performance of the DDERT++ in achieving the segmentation task.

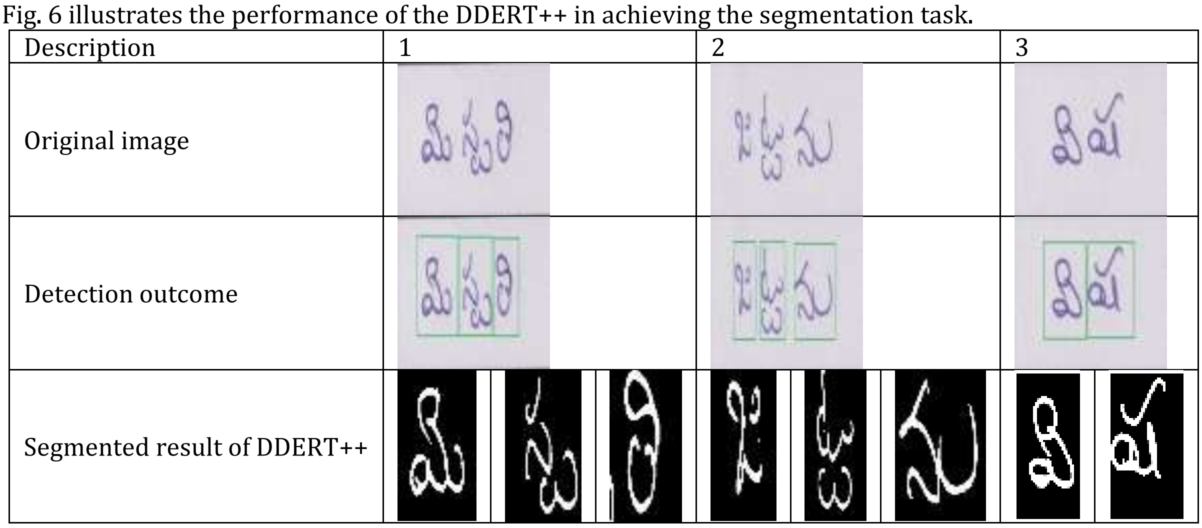
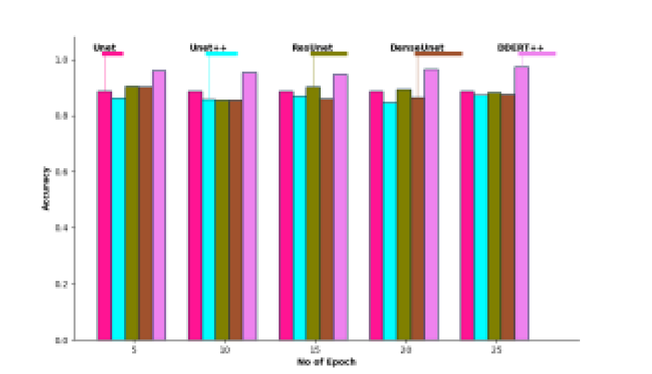


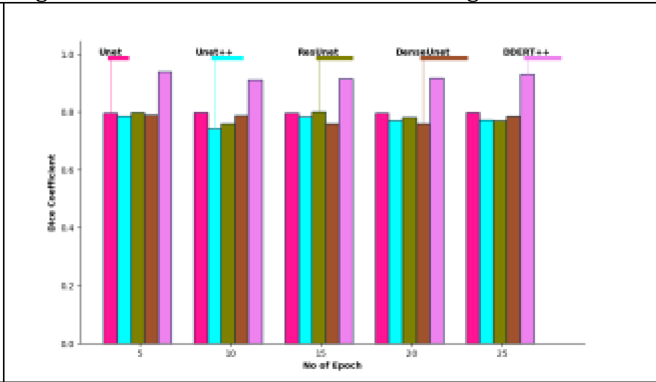
Fig. 6. Segmentation result of DDERT++ for handwritten Telugu text

D. Number of epoch-based estimation of DDERT++ for the segmentation task

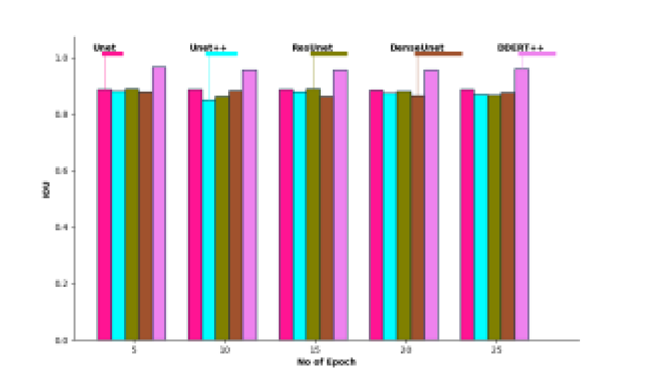
The epoch-based performance assessment of DDERT++ is mainly to evaluate the segmentation capability of the system in performing the handwritten character recognition. The result from the DDERT++ compared with other segmentation models in varying training iterations is defined in Fig. 7. Some of the key evaluation metrics are taken to measure the performance of the DDERT++. The number of epoch-based analyses generally helps in evaluating the learning efficiency of the model. From the investigational observation, the designed DDERT++ achieved higher accuracy than 18% of Unet, 14% of Unet++, 24% of ResUnet, 24% of DenseUnet. From the result, the presented DDERT++ has a faster learning rate, which shows the stability of the model. In addition, the DDERT++ also minimize the segmentation error. This makes the DDERT++ highly suitable for the character segmentation task for Handwritten Telugu words.



(a)



(b)



(c)

Fig. 7. Number of epoch-based comparison of DDERT++ for Telugu handwritten character recognition compared with other segmentation approaches in terms of "(a) Accuracy, (b) Dice coefficient, and (c) IoU"

E. Comparative estimation of DDERT++ for the segmentation task

The statistical measure-based estimation of DDERT++ with the previous segmentation techniques is presented in Table II. The comparison considered the key metrics to calculate the performance of each model. From the differentiation, the DDERT++ outperforms the standard segmentation model. In most places, the DDERT++ achieved superior value, which defines the consistency of the DDERT++ in performing the segmentation. The table result effectively confirms the efficiency of the DDERT++ in the segmentation task. From the result, the DDERT++ performed a better value than 9.1% of Unet, 7.4% of Unet++, 9.1% of ResUnet, 6.1% of DenseUnet, The outcome shows the robustness of the DDERT++ even for the challenging condition. Overall, the comparative outcome showed that the DDERT++ has better capability in performing segmentation.

TABLE II. STATISTICAL COMPARATIVE ANALYSIS OF DDERT++ FOR IMAGE SEGMENTATION

Terms	Unet [30]	Unet++ [31]	ResUnet [32]	DenseUnet [33]	DDERT++
Dice coefficient					
“Best”	8.820677	8.977466	8.814995	9.112395	9.695718
“Worst”	8.297913	8.558449	8.479135	8.549026	9.478475
“Mean”	8.590081	8.721711	8.704814	8.851011	9.586946
“Median”	8.622628	8.69696	8.717024	8.847125	9.608814
“STD”	0.139807	0.130298	0.085607	0.15684	0.077471
Jaccard					
“Best”	7.890171	8.144647	7.881083	8.369512	9.409407
“Worst”	7.090969	7.480148	7.359807	7.465763	9.008651
“Mean”	7.531226	7.73557	7.707664	7.942388	9.207723
“Median”	7.578751	7.694357	7.72584	7.932748	9.247092

"STD"	0.213661	0.206395	0.132673	0.251827	0.142821
Accuracy					
"Best"	8.741913	8.951721	8.845062	9.080811	9.685211
"Worst"	8.372192	8.479614	8.527985	8.653107	9.470215
"Mean"	8.594788	8.733765	8.727768	8.86908	9.589417
"Median"	8.631897	8.75885	8.764572	8.855133	9.61586
"STD"	0.117045	0.137293	0.096634	0.12257	0.073965

F. Convergence estimation of MHGMEO for optimal classification

The convergence behaviour of the MHGMEO-based optimization strategy in performing optimal classification compared with other algorithms is shown in Fig. 8. Here, multiple iterations are taken to evaluate the cost function. Based on the graph result, the MHGMEO exabit less cost function value even for the increasing iteration number, which tends to make the algorithm attain stable and faster convergence. The sharp reduction in cost function attained for the MHGMEO in the initial iteration, which tends to manage the stability without any oscillation. From the graph, the presented MHGMEO attained a better convergence value than 13% of ABC-ACS-CGCNN, 11% of WOA-ACS-CGCNN, 8% of OOA-ACS-CGCNN, and 5% of GMEO-ACS-CGCNN. This prevents the algorithm from getting trapped in the higher cost region. This makes the MHGMEO-ACS-CGCNN suitable for complex classification task.

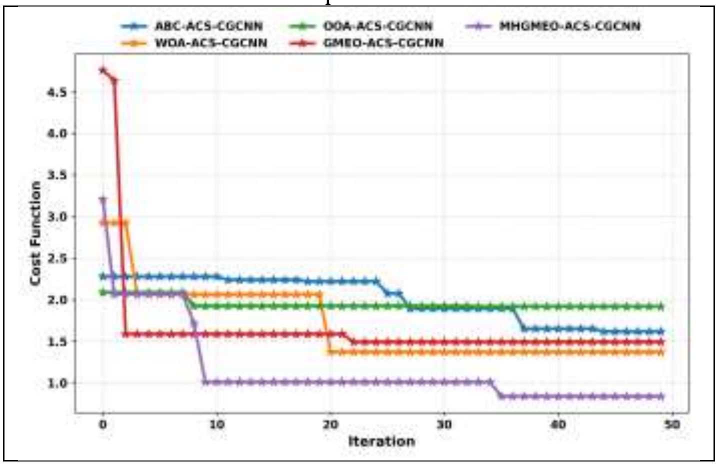


Fig. 8. Convergence analysis of MHGMEO for optimal classification of Telugu handwritten character compared with other algorithms

G. Statistical analysis of MHGMEO for classification

The statistical estimation of MHGMEO-ACS-CGCNN compared with other optimization algorithms for the optimal classification defined in Table III. From the table outcome, the MHGMEO-ACS-CGCNN attained a better value than 92.6% of ABC-ACS-CGCNN, 63.3% of WOA-ACS-CGCNN, 128.5% of OOA-ACS-CGCNN, and 77.7% of GMEO-ACS-CGCNN, which evidences the significance of the proposed model. In order to demonstrate an optimal solution, the superior capability of the MHGMEO-ACS-CGCNN is revealed. In addition, the MHGMEO generate consistent performance, and manages between the exploration and exploitation. Overall, the statistical outcome showed that the MHGMEO provided better classification accuracy.

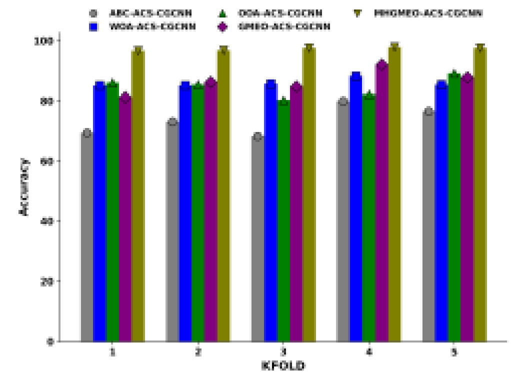
TABLE III. STATISTICAL ANALYSIS OF MHGMEO FOR PERFORMING OPTIMAL CLASSIFICATION DIFFERENTIATED WITH CONVENTIONAL ALGORITHMS

Terms	ABC-ACS-CGCNN [34]	WOA-ACS-CGCNN [35]	OOA-ACS-CGCNN [36]	GMEO-ACS-CGCNN [26]	MHGMEO-ACS-CGCNN
"Best"	1.61781	1.372269	1.919507	1.49298	0.839945
"Worst"	2.277784	2.925123	2.088247	4.759462	3.207754
"Mean"	2.012687	1.70053	1.949431	1.659481	1.165005
"Median"	2.149153	1.372269	1.925359	1.49298	1.010015
"STD"	0.262262	0.446628	0.060641	0.622089	0.497553

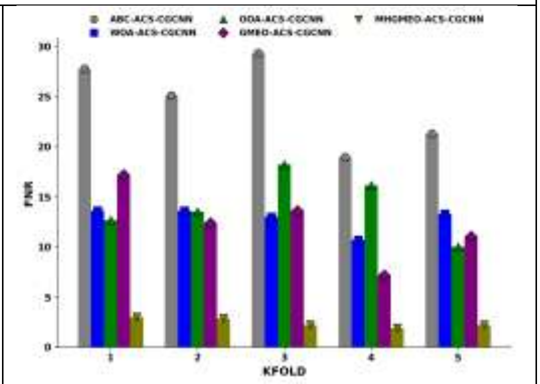
H. K-fold-based performance analysis of MHGMEO-ACS-CGCNN in the classification task.

The K-fold based estimation of MHGMEO-ACS-CGCNN compared with other optimization and classifiers defined in Fig. 9. Here fivefold is evaluated to analyze the classification effectiveness. From the analysis, it is noted that the MHGMEO-ACS-CGCNN provides consistency in the performance across all the folds. From the comparison, the MHGMEO-ACS-CGCNN attained higher accuracy value than 28.5% of ABC-ACS-CGCNN, 13.2% of WOA-ACS-CGCNN, 12.2% of OOA-ACS-CGCNN, and 17.3% of GMEO-ACS-CGCNN. This defines the strong generalization capability of the implemented model. It also reduces the false alarms during the classification phase. The reliable performance attained using the MHGMEO-ACS-CGCNN shows a notable enhancement. The enhanced result shows the efficiency of the MHGMEO-ACS-CGCNN for the Telugu handwritten character classification.

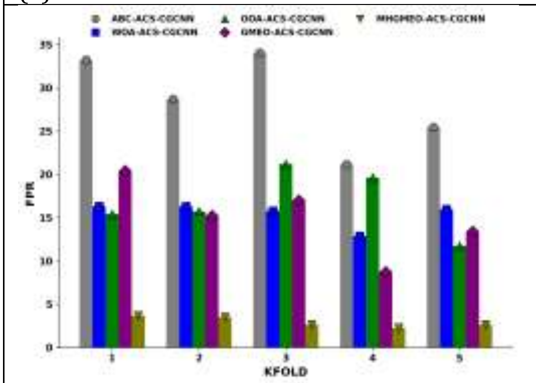
Algorithm comparison.



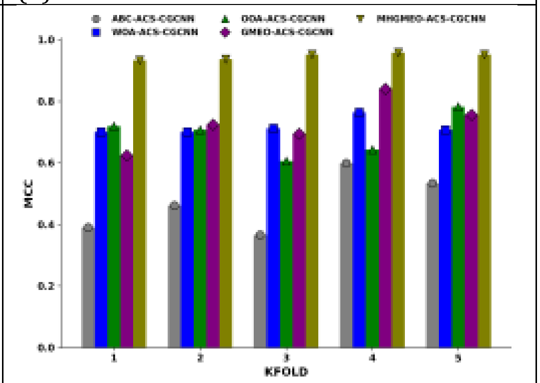
(a)



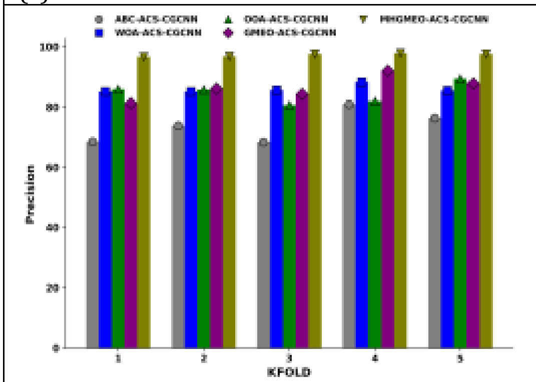
(b)



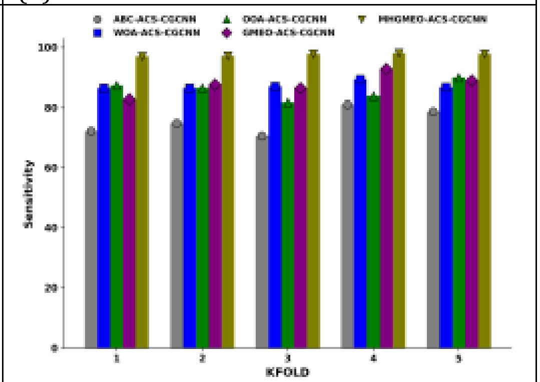
(c)



(d)



(e)



(f)

Classifier comparison

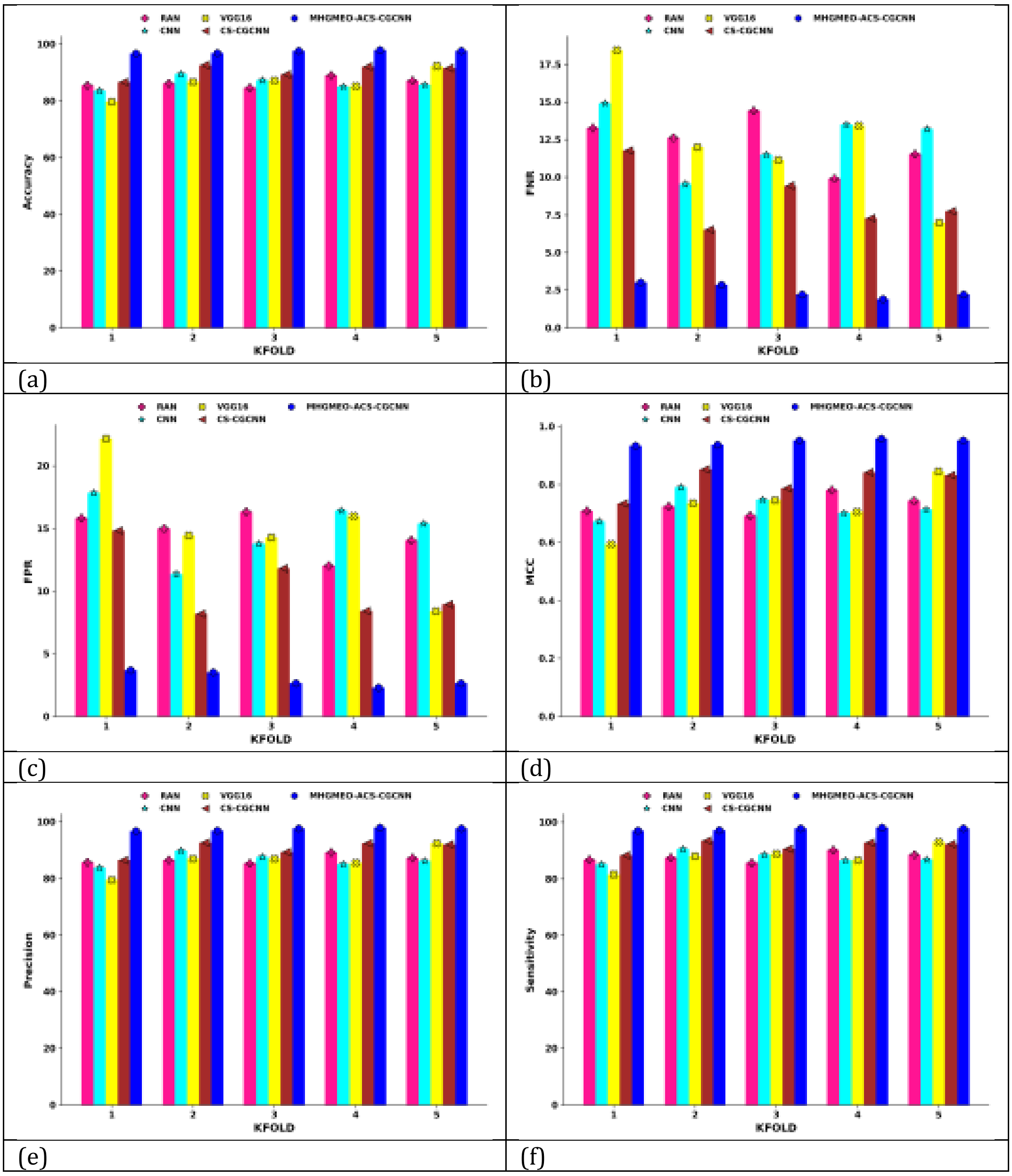


Fig. 9. Kfold-based comparative analysis of MHGMEO-ACS-CGCNN differentiated with conventional algorithms and classifiers for handwritten Telugu text classification in terms of "(a) Accuracy, (b) FNR, (c) FPR, (d) MCC, (e) Precision, and (f) Sensitivity"

I. Activation function-based comparison of MHGMEO-ACS-CGCNN with other models

The activation function-based comparative analysis with other prior algorithms and classifiers is given in Table V and V. Here, the classification functionality is analyzed against different optimization models by varying activation functions. From the table result, it is clearly noted that the designed MHGMEO-ACS-CGCNN significantly outperformed other standard models, even for all the activation functions. From the table, the MHGMEO-ACS-CGCNN achieved a higher accuracy value in terms of linear function than 22.9% of ABC-ACS-CGCNN, 17.2% of WOA-ACS-CGCNN, 13.9% of OOA-ACS-CGCNN, and 11.1% of GMEO-ACS-CGCNN. The outcome defines the stability of the proposed model in maintaining complex variation in the Telugu character. Overall, the validation proved that the MHGMEO-ACS-CGCNN is more efficient for superior classification than other models.

TABLE IV. ACTIVATION FUNCTION-BASED COMPARISON OF MHGMEO-ACS-CGCNN WITH OTHER OPTIMIZATION MODELS FOR TELUGU HANDWRITTEN CHARACTER CLASSIFICATION

Activation function	ABC-ACS-	WOA-ACS-	OOA-ACS-	GMEO-ACS-	MHGMEO-ACS-CGCNN

	CGCNN [34]	CGCNN [35]	CGCNN [36]	CGCNN [26]	
Linear	75.16667	80.75	84	86.75	97.58333
ReLU	80.08333	90.16667	90.33333	89	98.25
TanH	79	83.75	80.5	85.25	97.08333
Sigmoid	79.25	84.75	84.58333	88.08333	97.66667
Softmax	82.08333	86.16667	90.25	90.5	97.5

TABLE V. ACTIVATION FUNCTION-BASED COMPARISON OF MHGMEO-ACS-CGCNN WITH OTHER OPTIMIZATION MODELS FOR TELUGU HANDWRITTEN CHARACTER CLASSIFICATION

Activation function	MobileNet-VGG16 [1]	ShuffleNet [2]	ResNet50-CNN [4]	CS-CGCNN [28] [29]	MHGMEO-ACS-CGCNN
Linear	87.5	88.33333	79.08333	87.83333	97.58333
ReLU	86.5	89.5	88.91667	93.91667	98.25
TanH	78.25	88.25	81.91667	88.41667	97.08333
Sigmoid	78.41667	92.08333	80.08333	88.83333	97.66667
Softmax	76.66667	88.66667	88.91667	94.16667	97.5

VII. CONCLUSION

This framework constructed the advanced deep learning technique for the accurate segmentation and classification of Telugu handwritten text. Initially, the required handwritten images of Telugu characters were collected from available resource. The collected images were initially processed through the DDETR++ to perform the effective character-level segmentation. The high-quality segmentation outcome was attained from the DDETR++ through its integration of the deformable attention mechanism. The segmented characters were further processed through the designed ACS-CGCNN for performing the classification task. Here, the model incorporated channel attention to the contextual graph-based feature learning to enhance the pattern analysis. Moreover, the hyperparameters in the ACS-CGCNN were optimally tuned using the MHGMEO to enhance the overall classification result. The effectiveness of the proposed MHGMEO-ACS-CGCNN was analyzed using extensive experimentation by comparing it with the previous models. From the outcome, the MHGMEO-ACS-CGCNN attained maximum accuracy in classification than 28% of MobileNet-VGG16, 34% of ShuffleNet, 26% of ResNet50-CNN, and 31% of CS-CGCNN. The achieved outcome determines that the implemented framework attained superior performance in Handwritten Telugu character classification.

Limitations and Future Direction: Even though the presented MHGMEO-ACS-CGCNN attained a superior outcome in handwritten Telugu character recognition, the model may exhibit minimal generalization while determining low-quality input and unseen handwriting styles. Hence, in future, it will be significant to focus on a self-supervised learning strategy to enhance the generalizability and adaptability of the system, which makes it efficient for robust real-world applications.

References

[1] K. Limbachiya and A. Sharma, "A Two-Level Multi-Branch Convolutional Neural Network Framework for Handwritten Gujarati Character Recognition," in IEEE Access, vol. 13, pp. 205733-205752, 2025.

[2] M. El Khayati, A. Maafiri, Y. Himeur, H. Ali Alkhazaleh, S. Atalla and W. Mansoor, "Leveraging Transfer Learning and Mobile-Enabled Convolutional Neural Networks for Improved Arabic Handwritten Character Recognition," in IEEE Access, vol. 13, pp. 166104-166126, 2025.

[3] Yunxin Li; Qian Yang; Qingcai Chen; Baotian Hu; Xiaolong Wang; Yuxin Ding, "Fast and Robust Online Handwritten Chinese Character Recognition With Deep Spatial and Contextual Information Fusion Network," in IEEE Transactions on Multimedia, vol. 25, pp. 2140-2152, 2023.

[4] M. -S. Kim, C. -H. Son and S. -H. Choi, "A Novel Federated Learning-Based Image Classification Model for Improving Chinese Character Recognition Performance," in IEEE Access, vol. 12, pp. 185971-185991, 2024.

[5] S. Gonwirat and O. Surinta, "DeblurGAN-CNN: Effective Image Denoising and Recognition for Noisy Handwritten Characters," in IEEE Access, vol. 10, pp. 90133-90148, 2022.

[6] M. Kiran Sastry, and S. Aparna, "Developing adaptive Yolov5-based Telugu handwritten character segmentation and classification framework using Enhanced Chef-Based Optimization Algorithm and Deep Learning Networks," Computers and Electrical Engineering, Vol. 126, pp. 110533, August 2025.

[7] Triveni Banavatu, and Govindaswamy Parthasarathy, "Heuristic-aided fusion serial cascaded deep network for handwritten character recognition from handwritten images using optimization strategy," Applied Soft Computing, Vol. 174, pp. 112937, April 2025.

[8] L G X Agnel Livingston, D Judson and Prasanth Aruchamy, "A reliable character recognition model based on hybrid feed forward back-propagation neural network and adaptive cuckoo search optimization algorithm," Sadhana, Vol. 50, no. 315, 2025.

[9] H. T. Weldegebriel, H. Liu, A. U. Haq, E. Bugingo and D. Zhang, "A New Hybrid Convolutional Neural Network and eXtreme Gradient Boosting Classifier for Recognizing Handwritten Ethiopian Characters," IEEE Access, vol. 8, pp. 17804-17818, 2020.

[10] M. Shabir, Z. Jan, N. Islam, I. Khan, G. Ali and M. ElAffendi, "TILPDeep: A Lightweight Deep Learning Technique for Handwritten Transformed Invariant Pashto Text Recognition," IEEE Access, vol. 11, pp. 23393-23406, 2023.

[11] H. Muhammad Raza Ur Rehman, S. Arfan Haider, H. Faisal, K. -Y. Yoo, M. Z. Jhandir and G. S. Choi, "A Novel Framework for Saraiki Script Recognition Using Advanced Machine Learning Models (YOLOv8 and CNN)," IEEE Access, vol. 13, pp. 56843-56860, 2025.

[12] R. Sahay and M. Coustaty, "An Enhanced Prototypical Network Architecture for Few-Shot Handwritten Urdu Character Recognition," IEEE Access, vol. 11, pp. 33682-33696, 2023.

[13] Dezhi Peng; Lianwen Jin; Weihong Ma; Canyu Xie; Hesuo Zhang; Shenggao Zhu, "Recognition of Handwritten Chinese Text by Segmentation: A Segment-Annotation-Free Approach," IEEE Transactions on Multimedia, vol. 25, pp. 2368-2381, 2023.

[14] S. Hassan, A. Raza Shahid and M. Asif Naeem, "TDA-ViT: A Transformer-Based Framework for Unified Urdu Text Recognition via Topological and Visual Feature Fusion," IEEE Access, vol. 13, pp. 182940-182959, 2025.

[15] C. Vinotheni and S. L. Pandian, "End-To-End Deep-Learning-Based Tamil Handwritten Document Recognition and Classification Model," IEEE Access, vol. 11, pp. 43195-43204, 2023.

[16] D. Fang and C. Zhang, "Multi-Feature Learning by Joint Training for Handwritten Formula Symbol Recognition," IEEE Access, vol. 8, pp. 48101-48109, 2020.

[17] Y. Zhang, C. Wang, F. Li, W. Yu, Y. Wang and J. Chen, "RF-AIRCGR: Lightweight Convolutional Neural Network-Based RFID Chinese Character Gesture Recognition Research," IEEE Access, vol. 12, pp. 195760-195777, 2024.

[18] N. I. Kajla, M. M. S. Missen, M. M. Luqman and M. Coustaty, "Graph Neural Networks Using Local Descriptions in Attributed Graphs: An Application to Symbol Recognition and Hand Written Character Recognition," IEEE Access, vol. 9, pp. 99103-99111, 2021.

[19] H. Zhao, H. Chu, Y. Zhang and Y. Jia, "Improvement of Ancient Shui Character Recognition Model Based on Convolutional Neural Network," IEEE Access, vol. 8, pp. 33080-33087, 2020.

[20] Z. Li, X. Wang, Y. Liu, L. Jin, Y. Huang and K. Ding, "Improving Handwritten Mathematical Expression Recognition via Similar Symbol Distinguishing," IEEE Transactions on Multimedia, vol. 26, pp. 90-102, 2024.

[21] J. C. Aradillas, J. J. Murillo-Fuentes and P. M. Olmos, "Boosting Offline Handwritten Text Recognition in Historical Documents With Few Labeled Lines," IEEE Access, vol. 9, pp. 76674-76688, 2021.

[22] Y. Chen, H. Zheng, Y. Li, W. Ouyang and J. Zhu, "Online Handwritten Chinese Character Recognition Based on 1-D Convolution and Two-Streams Transformers," IEEE Transactions on Multimedia, vol. 26, pp. 5769-5781, 2024.

[23] M. M. Fuad, A. Faiyaz, N. M. K. Arnob, M. F. Mridha, A. K. Saha and Z. Aung, "Okkhor-Diffusion: Class Guided Generation of Bangla Isolated Handwritten Characters Using Denoising Diffusion Probabilistic Model (DDPM)," IEEE Access, vol. 12, pp. 37521-37539, 2024.

[24] Y. Wang, Y. Yang, W. Ding and S. Li, "A Residual-Attention Offline Handwritten Chinese Text Recognition Based on Fully Convolutional Neural Networks,"IEEE Access, vol. 9, pp. 132301-132310, 2021.

[25] I. Lamaakal, I. Ouahbi, K. El Makkaoui, Y. Maleh, P. Plawiak and F. Alblehai, "A TinyDL Model for Gesture-Based Air Handwriting Arabic Numbers and Simple Arabic Letters Recognition," IEEE Access, vol. 12, pp. 76589-76605, 2024.

[26] Nehal A. Mansour, M. Sabry Saraya and Ahmed I. Saleh, "Groupers and moray eels (GME) optimization: a nature-inspired metaheuristic algorithm for solving complex engineering problems," Neural Computing and Applications, Vol. 37, pp. 63–90, 2025.

[27] Zhang, C.B., Zhong, Y. and Han, K., "DETR++: Instructive Multi-Route Training for Detection Transformers with Mixture-of-Experts", Computer Vision and Pattern Recognition, vol.1, 2024.

[28] Zhang, X., Xu, C. and Tao, D., "Context aware graph convolution for skeleton-based action recognition", Computer vision and pattern recognition, pp. 14333-14342, 2020.

[29] Zhu, H., Xie, C., Fei, Y. and Tao, H., "Attention mechanisms in CNN-based single image super-resolution: A brief review and a new perspective", Electronics, vol.10(10), pp.1187, 2021.

[30] Liu, X., "Enhanced handwriting recognition through hybrid UNet-based architecture with global classical features", Journal of Ambient Intelligence and Humanized Computing, vol.16(1), pp.315-327, 2025.

[31] Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N. and Liang, J., "Unet++: Redesigning skip connections to exploit multiscale features in image segmentation", IEEE transactions on medical imaging, vol.39(6), pp.1856-1867, 2019.

[32] Damayanti, F., Yuniarno, E.M. and Suprapto, Y.K., "Modified ResUNet Architecture for Binarization in Degraded Javanese Ancient Manuscript", Mathematical Modelling of Engineering Problems, vol.11(7), 2024.

[33] Cao, Y., Liu, S., Peng, Y. and Li, J., "DenseUNet: densely connected UNet for electron microscopy image segmentation", IET Image Processing, vol.14(12), pp.2682-2689, 2020.

[34] Wang, H., Wang, W., Xiao, S., Cui, Z., Xu, M. and Zhou, X., "Improving artificial bee colony algorithm using a new neighborhood selection mechanism", Information Sciences, vol.527, pp.227-240, 2020.

[35] Benmamoun, Z., Khlie, K., Dehghani, M. and Gherabi, Y., "WOA: Wombat Optimization Algorithm for Solving Supply Chain Optimization Problems", Mathematics, vol.12(7), pp.1059, 2024.

[36] Mohammad Dehghani and Pavel Trojovský, "Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems", Frontiers in Mechanical Engineering, Vol. 8, 2022.