



Fusion of RNCNN-BRHA for Recognition of Telugu Word from Handwritten Text

Rajasekhhar Boddu^{*}, Edara Sreenivasa Reddy^{*}

Department of Computer Science and Engineering, College of Engineering and Technology, Acharya Nagarjuna University, Guntur 522510, Andhra Pradesh, India

Corresponding Author Email: rajsekhar.cse@gmail.com

<https://doi.org/10.18280/ria.370127>

Received: 7 January 2023

Accepted: 10 February 2023

Keywords:

Word recognition, CNN, HARRIS
Algorithm, BRISK, PSNR

ABSTRACT

With the increasing proliferation of digital pictures, word retrieval (WR) algorithms have been intensively investigated. In general, WR service is relatively costly in terms of computing and storage resources. However, retrieving telugu words from handwritten text is difficult due to the wide range of curves and strokes in handwritten words. Deep learning for handwritten word identification from photos is an ongoing research topic with promising results. In this paper, thinning the words is performed, which helps in extraction of sharp features. After performing thinning operation for given input word, features are extracted using residual network (RN) structure in CNN, BRISK and HARRIS algorithms to identify the word from the handwritten text. These three are combined and termed as RNCNN-BRHA model. The features are evaluated to check the performance of proposed RNCNN-BRHA technique in retrieval of telugu words. The CNN utilized has various innovative features, such as the utilization of several completely linked branches. When applied to all regularly used handwriting recognition criteria, our technique surpasses all current algorithms by a wide margin. The PSNR value obtained using the proposed model is 26.94 which is far better compared to techniques like hilditch algorithm and morphological operation.

1. INTRODUCTION

Handwritten Word Recognition (HWR) is one of the best exciting fields of investigation in computer science, yet it is a tough issue because to the wide variances in the writing styles of people [1]. Though reasonable results for individual word detection and character recognition issues have been obtained, present models are still insufficient for robust handwritten word recognition in an unrestricted context [2]. Furthermore, when the device is required to conduct word recognition in a compressed domain, the difficulty becomes more severe. Preprocessing is a technique used in general methods to handwritten word recognition to lessen the large amount of variance found in numerous options in which the letters are been structured. One of the most typical pre-processing approaches rely on character size normalization and restoration of the slopes of different letters accessible in the provided picture [3].

In the study of networks based on neurons, Convolutional Neural Network (CNN) is the efficient prevalent approach in the present use of deep learning to computer vision [4]. Originally designed to interpret handwritten numbers, this approach is today used for practically every perceptual job involving picture and video input. In many of the challenges CNNs outperform the competition. The techniques with the involvement of CNN which are used for recognition of handwritten words are having a lower level performance. CNNs are taught in a supervised environment. When training it, the first consideration is the supervision style need to be applied. Recent and existing developments of CNN in the area of retrieval of hand written words demonstrated that

word-based encoding outperforms bag-of-n-grams encoding [5]. These binary qualities determine if a given n-gram is present in some region of the word. The techniques like SVM, fishers vectors of SIFTs are been used directly on the raw data of image pixels. Furthermore, many inventions have to be introduced in order to achieve high levels of performance in which six hundred to twelve hundred set of attributes where considered and processed. The network model is rather large, and training it on the comparatively tiny handwriting recognition datasets available seems difficult. To address this, the technique involves progressive training and other principles.

There are two approaches. Canny edge detection [6] and Chu's 3D thinning technique [7] are used to assess overall identification performance and resilience across various typefaces. For many years, handwriting recognition (HWR) has been the focus of study. While many authors have created recognition for single letters or numbers, later recognizers, notably for the English language, concentrated on whole phrases or even sentences. The telugu word have more strokes and curves when compared to English. The research finds telugu hand written words. To make the issue more difficult, we assigned the model a goal of recognizing telugu kinds of words, as illustrated in Figure 1.



Figure 1. Example of different telugu words

The HWR assists in converting the written content in the papers into a text document format, also known as a legible electronic format. Historical information may be easily shared, evaluated, and archived in this fashion. Arabic numerals and English letters are made out of basic forms. The quantity of characters varies by class, with larger words frequently involving more complex forms and a greater number of character levels. The suggested model can be used on variety of shapes with varied complexity levels. The PSNR value obtained using the proposed model is 26.94 which is far better compared to techniques like hilditch algorithm and morphological operation. The approach described in this paper is developed to achieve simpler targets with low cost of computation.

The discussion of paper includes the discussion regarding different type of approaches for HWR in section 2. In section 3 the methodology utilized to give best results in recognition of telugu word. The experimental results which are evaluated using the matlab tool is discussed in section 4. Different parameters have been evaluated in which the proposed method gives good results. Finally, the conclusion of the work is discussed in section 5 and also provides the future work.

2. RELATED WORK

Handwritten character identification from photographs is a well-studied subject with several ways achieved thus far by applying various technical solutions. This problem was initially addressed using a hand-engineered solution. For recognition of hand written text a word gap technique is utilized to separate the word from a line of text picture [8]. The authors [9] created the neural network based model and Hidden Markov Model (HMM) models to improve offline handwritten letter detection in photos. The artificial neural network is combining with HMM. Deep learning techniques for handwritten character and word identification have seen an increase in popularity in recent years.

The utilization of CNN-RNN is discussed by the author [10]. This model depends on previous presented models with small addition in work to improve the retrieval of word in better standard. The model was used to achieve good efficacy by taking several crucial factors into account like assignment of weights, normalization of image etc. The author [11] suggested a new model with test error rate of 0.25% and test is performed on UW3 dataset. The model suggested in obtaining the test rate is CNN-LTSM which is a hybridization of CNN and 1D LTSM model. Some of the models had a significant drawback in that they could not decode arbitrary special characters regardless of vocabulary size.

The author [12] utilized a word beam search-based decoding technique to deal with such a circumstance. This method had a significant impact on the outcomes since the words were limited to those that were already in the collection of known terms. On the other hand some are permitted to recognition of terms that were not in the dictionary. In the scalable online handwritten text recognition system [13], an LSTM-based model is also used. Despite advances in technology solutions for the purpose of identification of handwritten, the focus on the domain compressed has been rather restricted.

The study of deep learning is emerging recently in the field

of optical character recognition (OCR), which helps in solving the problems incurred in OCR. Multilayer perceptron-based CNN is been discussed by the author [14] which provide good rate of accuracy. According to our findings [15], no segmentation is performed for detection of printed urdu word with the use of hybrid CNN-RNN model. The use of synthetic datasets is believed to benefit deep neural network training by decreasing over-fitting through data augmentation.

The studies used different methods for identification of hand written text. The present approaches use different type of feature extraction techniques to evaluate every stroke of word which can effectively recognize the word.

3. METHODOLOGY

The overall work performed in the paper is shown in Figure 2. In this section, the input data is preprocessed and is given for thinning the image and improve the quality. A residual network is utilized for performing the process of CNN for extracting the features. Images are synthesized to obtain more data for CNN. Then feature extraction process is performed. Finally, the techniques like HARRIS and BRISK which are used for feature extraction is described. The parameters are evaluated and compared with other feature extractions algorithms.

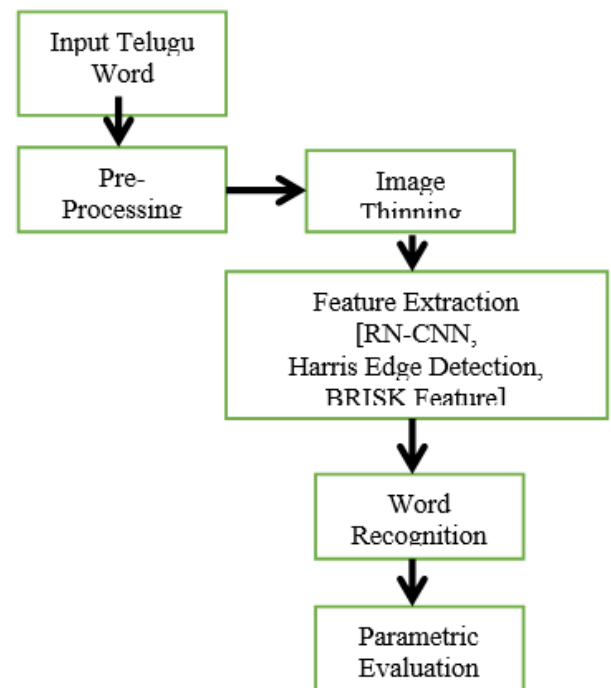


Figure 2. Methodology of word recognition

3.1 Pre-processing

The input picture is retrieved and processed. Preprocessing of image is one of the important stages to improve the visibility of the text in the picture and to guarantee that the input image has the appropriate dimensionality for going to next step of processing. Visual contrast is improved by converting a restricted intensity values of particular range into a larger range with the process of contrast stretching. As a result, it increases the quality level of image in terms of accessibility. In this paper Ostu method is proposed in pre-

processing. In Ostu model of thresholding all the available values of threshold are involved iteratively and evaluate the spread of pixel values each side of threshold i.e., falls of pixels either in background or foreground. The involvement of proposed Ostu model is designed using four steps,

- Processing of the input image
- Retrieving the histogram of image (pixels distribution)
- Computing the threshold value.
- If the value of saturation is higher, replace the pixels of image into white for required regions and black if saturation level is low

3.2 Image thinning

Thinning algorithms are used to create an object's skeleton. The lines in the thinned picture are one pixel wide. Image complexity is reduced via thinning or skeletonization. The thinning technique is commonly utilized in vectorization approaches based on thinning. The object's border is removed from the object during thinning. Thinning aids in readily distinguishing the characteristics. As a result, thinning is done before to extracting the features. It is a morphological process that deletes black foreground pixels layer by layer until a skeleton of one pixel width is achieved. It is the process of shrinking a thing to its smallest possible size. In this process the boundary of the image which is considered as input is subtracted for the given image. The example of thinning can be defined as, $A \oslash B = A \cap (A \otimes B)C$. In this define function 'A' is said to be the image for input and 'B' is said to be a structuring element which is composite and the B is given as $B = (B1, B2)$. In this step the telugu word pattern will be identified. Due to the process of image thinning the next step feature extraction can be enhanced as the thinning process helps by reducing the pattern of digitized word to a skeleton form to evaluate the results with a pixel thickness of value one.

3.3 Feature extraction

3.3.1 ResNet Convolutional Neural Network (RNCNN)

The working of neural network is performed by considering a picture and the weights are been generated depending on the nature of image complexity and dividing the weights from each consecutive values. The CNN network the ease of doing pre-processing for the given input is better when compared to other deep learning algorithms. The training of classifiers in CNN utilizes a simple technique with fundamental capabilities. This will help to identifies the similar features of the object which is been targeted. The structure of CNN is composed of human brain in which the neuron's structure is built mainly the visual cortex. The response of every neuron is important in a particular section of visual area which is called as field of receptive.

One of the models created is the deep residual network, or ResNet. This design was created to overcome difficulties in the convolution network model since the time and the number of layers while performing network training is high. The connection skipping or creating shortcuts is the operation of ResNet and is widely used in such applications. The ResNets model has a benefit over other architectural approaches in that its efficiency does not degrade as the design becomes deeper. Furthermore, the computational level of complexity is low and the training capability of the network is been drastically improved. One of the advantages

of proposed model is its level skipping functionality, where it can skip two to three level that effects the ReLU and batch normalization. This paper uses residual learning to apply to several levels of layers.

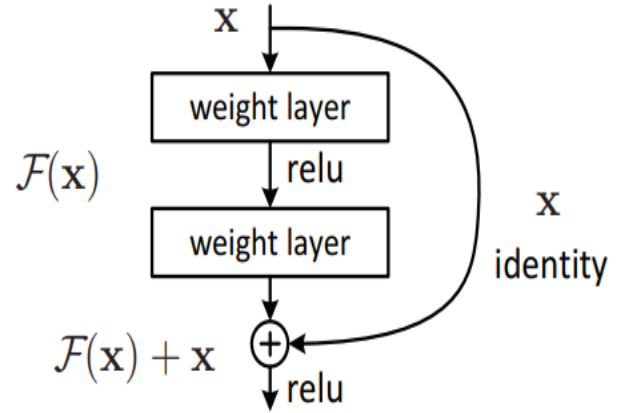


Figure 3. ResNet functioning

As shown in the Figure 3. The approach behind this network is instead of layers learning the underlying mapping, we allow the network to fit the residual mapping. So, instead of say $H(x)$, initial mapping, let the network fit,

$$F(x) := H(x) - x \text{ which gives } H(x) : \\ = F(x) + x \quad (1)$$

In ResNet the function of residual block is given as follows:

$$y = F(x, W + x) \quad (2)$$

where the layer of the input is termed as 'x'; the output layer is given as 'y'; and the function F is related by the residual map.

Assumptions of residual block

1. Addition of more / new layers would not degrade the performance model enables skip of layers if they were not needed.
2. If there is benefit with the additional layer and if regularization is present, the weights or kernels of the layers will be non-zero, and the performance of the model may improve marginally.

As a result of the "Skip connection" / "residual connection," adding more layers ensures that the model's performance does not degrade but may somewhat improve. You may create a very deep network by making arrangements in a way the network layers on top of one another. The identity function can be derived for every block in simple process due to the presence of ResNet blocks. This implies that you may add more ResNet blocks without affecting training set performance. In a ResNet, two types of blocks are employed, depending on the dimension levels of the input and output.

- **Identification block**

This block is termed to be the standard block in residual networks and one of the important considerations in this block is that dimensions of input and output activations are same.

• Block of convolutions

The utilization of this block refers to mismatch of dimensions for input and output activations and the layer i.e., CONV2D have shortcut paths and it is different from identification block.

The input and output dimension need to be same to obtain the residual block using the ResNet. Furthermore, each ResNet block is made up of two or three number of layers in which ResNet-18 is having two layers and ResNet-34 is having two layers and ResNet-50 and 101 are termed to be three layered. The two layers present in the starting of ResNet utilizes 7×7 convolution operations with a size of 3×3 of max-pooling and finally 277 number of strides. In the suggested work, Network-18 and Network-20 is investigated. The input image considered for performing the process is resized into a 224×224 grid. ResNet weights are initialized utilizing Stochastic Gradient Descent (SGD) and typical momentum settings. The proposed network structure is shown in Figure 4.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2.x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3.x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4.x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5.x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$

Figure 4. ResNet description

3.3.2 BRISK features

The BRISK methodology is divided into three parts: Detection of key-point, description of key-point and matching of descriptor. Firstly, a pyramid with a scale space is been built and AGAST extracts the stable extreme points of sub-pixel accuracy in continuous scale space. The gray level relation of the randomly sampled pixel pairings close to the local picture is then used to build the basic features descriptor of the picture is localized. Lastly, characteristics are matched using the Dissimilarity measure.

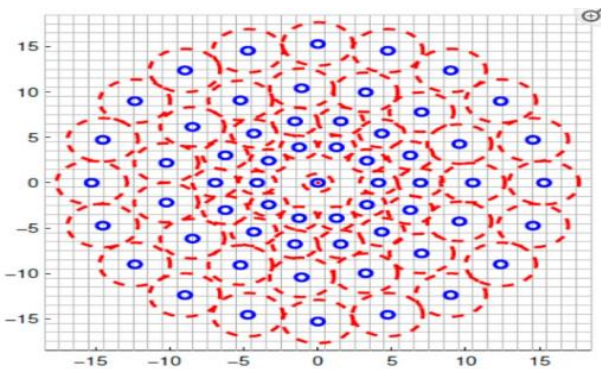


Figure 5. Sampling pattern of scalable key-points

The suggested BRISK algorithm helps in identifying the

feature points by scaling and rotation is shown in Figure 5. The telugu words have different strings and strokes to identify. This algorithm generates binary features by constructing the local feature descriptors. These descriptors are generated based on the grey scale relationship process identifying the random points and pairing with neighborhood points. The BRISK algorithm is an approach for identifying and describing feature points that is scale and rotation invariant. It creates the binary feature descriptor by creating the local picture's feature descriptor utilizing the grey scale relationship of random point pairs in the local image's region.

The detection of interest spots is accomplished by computing the Features from Accelerated Segmentation Test (FAST) score for every value of pixel in the picture. The interested points are easily identified by checking the threshold score which is pre-set for the process. Following some simple illumination comparison experiments, a 512 bits length of binary feature vectors are created. The distance of hamming among the descriptors of features is then used to match interest spots between picture pairs. The curves present in telugu words makes it difficult to extract all features, hence HARRIS algorithm is imposed.

3.3.3 HARRIS features

The HARRIS algorithm (HA) helps in identifying the corner points and extract the corner and infer visual traits of the telugu word. By looking at intensity values inside a limited window, we should be able to quickly identify the location. The appearance changes significantly by shifting the window in any of the direction. The word corner in HA is said to be the location with two dominating and opposing orientation of edges in its local area. The word corner in other terms can be defined as the point where two edges meet, in which the representation of edge involves deep changes in the brightness of the picture. The corners points are one of the most important aspects in the images and are referred as the interest points because these points do not get effected to rotation, translation and illumination. As shown in Figure 6, lets us consider a small window which is surrounded with a pixel 'p' in a picture. So that all such points with unique pixels are initially identified while performing the required action. The change in pixel value is identified by moving the window slightly towards the given direction. The moving direction depends on the change in measuring value.

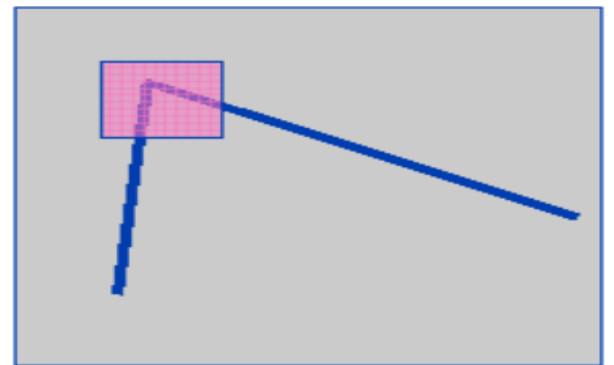


Figure 6. Corner point detection

More number of edges and the corners are present telugu word, this algorithm helps to find the edges and corners more accurately since it adjusts the direction for every 45-degree angle in the input image and immediately takes the difference

in the corner score into account as shown in Figure 7.

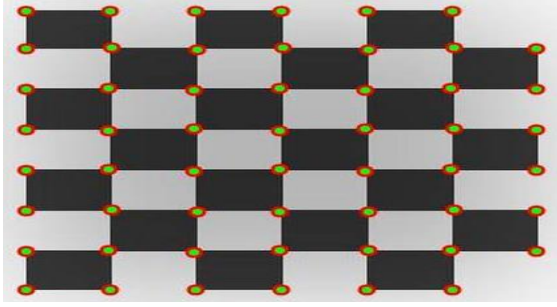


Figure 7. HARRIS corner point detection

For extraction of features a sum squared differences (SSD) is been defined and is given as $E(u, v)$, and in this function the u, v are the said to be the coordinates of the pixels in the image with a window size of 3×3 and the level of intensity of the pixel is termed as I . For detection of corner, the function $E(u, v)$ need to be maximized and is evaluated as shown in Eq (3).

$$E(u, v) = \sum_{x, y} w(x, y) [I(x + u, y + v) - I(x, y)]^2 \quad (3)$$

Finally, the features extracted using RNCNN, BRISK and HARRIS are combined together to identify the telugu word from the hand written text.

4. EXPERIMENTAL EVALUATION

To perform the experimental evaluation telugu words are considered and processed. The results obtained using the proposed RNCNN-BRHA model is compared with other approaches in recognition of telugu images from handwritten text. The performance of proposed model is analyzed by calculating different parameters which are stated below and tabulated. The results are evaluated using Matlab software tool. In this section, two input cases are considered and the outputs are shown in below figures. The telugu word recognition is one of the most prominent languages with many strokes for the purpose of identification.

Input 1.

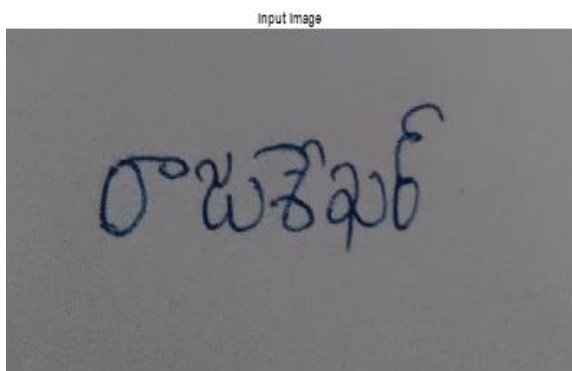


Figure 8. Input image

After preprocessing of input image as shown in Figure 8, the thinning image which is obtained is shown in Figure 9.

The foreground in the image is highlighted to extract the better features.



Figure 9. Thinning image

The features extracted using RNCNN are combined with brisk and harris extracted features. The point and corners detection using BRHA is shown in Figure 10. The combination of all features helps in identification of telugu word and the technique is termed as RNCNN-BRHA. The time taken for retrieval of telugu is lower.

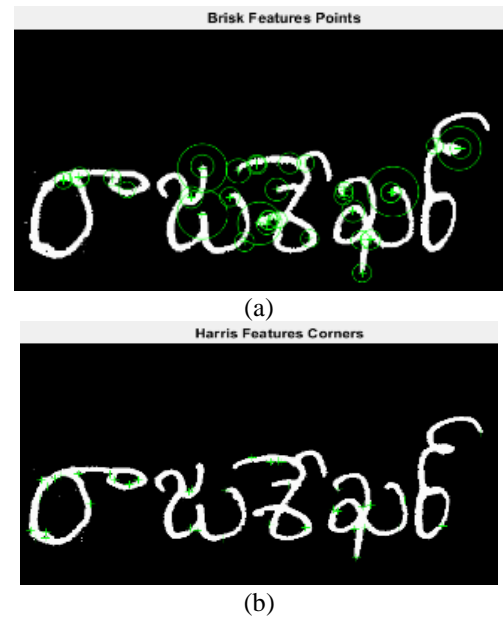


Figure 10. Extraction of features

Input 2.

Now another telugu word is given as input image shown in Figure 11. and proposed methodology is performed in detail. The results obtained like Thinned Image using Hilditch in Figure 12, extracted features of Hilditch thinned image in Figure 13, thinned image using morphological method in Figure 14, extracted features of morphological method in Figure 15, Thinning image using CNN in Figure 16 and extracted features of Harris and Brisk in Figure 17 are shown below.



Figure 11. Input image

Thinned image using Hilditch

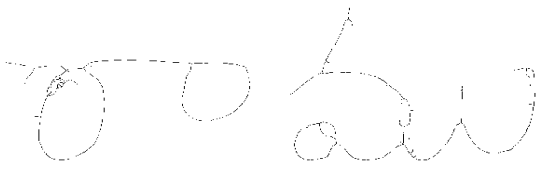


Figure 12. Thinning image using Hilditch

Extracted Features for Hilditch Thinned Image

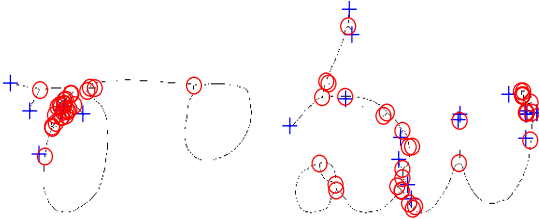


Figure 13. Extracted features of Hilditch thinned image

Thinned Image using Morphology method



Figure 14. Thinned image using morphology method

Extracted Features for Morphology thinned image

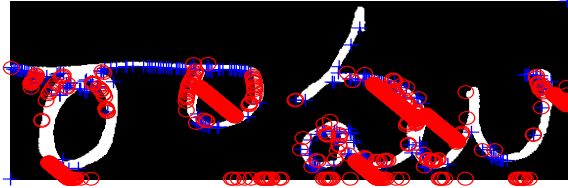


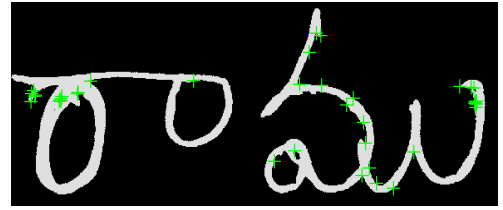
Figure 15. Extracted features for morphology method

Thinning using CNN



Figure 16. Thinning image using CNN

Harris Features Corners



Brisk Features Points

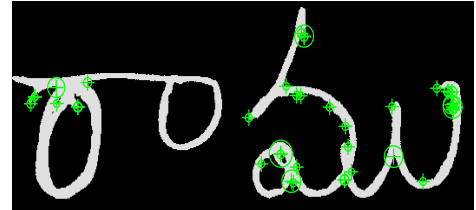


Figure 17. Extraction of features

The parameters which are used to evaluate the process of proposed framework are Connectivity Measurement, Thinning Rate, MSE, PSNR, RMSE, Execution Time, Noise Sensitivity, Hamming Distance. These parameters are compared and shown in Table 1.

Table 1. Comparison of different parameters

Parameter	Hilditch Algorithm	Morphological operations	Proposed RNCNN-BRHA
Connectivity Measurement	4.0	4.0	4.0
Thinning Rate (pixels)	1.0	1.0	1.0
MSE	0.11943	0.10966	0.00202
PSNR	9.22863	9.59945	26.94277
RMSE	0.34559	0.33115	0.04496
Execution Time (Sec)	7.98449	0.06165	0.20872
Noise Sensitivity	0.56679	0.45454	0.59090
Hamming Distance (pixels)	19.86026	16.95749	2.58127

PSNR

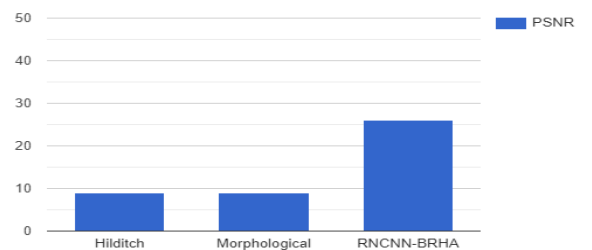


Figure 18. PSNR comparison

PSNR is one of the important parameters which evaluate the performance of the techniques used. PSNR is the ratio of the highest pixel value to the noise (MSE) that influences pixel quality. The greater the PSNR the lower the error and its value which is given on a logarithmic decibel scale. Figure 18 depicts the PSNR contrast.

5. CONCLUSION

By using the RNCNN-BRHA technique for telugu words, this effort created a baseline for word recognition. To our best knowledge, all earlier efforts concentrated on the recognition of the characters, whereas word recognition technique is relatively little investigated. In the suggested methodology, we created a novel framework using Telugu word picture retrieval and identification system by merging three distinct types of feature extraction processes. When compared to other approaches, the PSNR value obtained by RNCNN-BRHA is good. The value of PSNR obtained using proposed model is 26.94 while other approach having a PSNR of 9.22 and 9.59. The execution time in recognition of word using proposed model is 0.20 seconds which is very effective for fast processing. Furthermore, a telugu dataset may be constructed for word identification by performing classification algorithms. Every letter in the word can be decoded to improve the recognition rate. Parameters like as accuracy and sensitivity can be evaluated.

REFERENCES

- [1] Plamondon, R., Srihari, S.N. (2000). Online and off-line handwriting recognition: A comprehensive survey. *International Journal of Heat and Technology*, 22(1): 63-84. <https://doi.org/10.1109/34.824821>
- [2] Sharma, A., Jayagopi, D.B. (2021). Towards efficient unconstrained handwriting recognition using dilated temporal convolution network. *Expert Systems with Applications*, 164: 114004. <https://doi.org/10.1016/j.eswa.2020.114004>
- [3] Dutta, K., Krishnan, P., Mathew, M., Jawahar, C. (2018). Improving cnn-rnn hybrid networks for handwriting recognition. In 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR). IEEE, NY, USA, pp. 80-85.
- [4] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. (1998). Gradientbased learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278-2324. <https://doi.org/10.1109/5.726791>
- [5] Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A. (2014). Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint*

arXiv:1406.2227.
<https://doi.org/10.48550/arXiv.1406.2227>

- [6] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6: 679-698. <https://doi.org/10.1109/TPAMI.1986.4767851>
- [7] Lee, T.C., Kashyap, R.L., Chu, C.N. (1994). Building skeleton models via 3-d medial surface axis thinning algorithms. *CVGIP: Graphical Models and Image Processing*, 56(6): 462-478. <https://doi.org/10.1006/cgip.1994.1042>
- [8] Kim, G., Govindaraju, V., Srihari, S.N. (1999). An architecture for handwritten text recognition systems. *International Journal on Document Analysis and Recognition*, 2(1): 37-44. <https://doi.org/10.1007/s100320050035>
- [9] Espana-Boquera, S., Castro-Bleda, M.J., Gorbe-Moya, J., Zamora-Martinez, F. (2010). Improving offline handwritten text recognition with hybrid hmm/ann models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4): 767-779. <https://doi.org/10.1109/TPAMI.2010.141>
- [10] Breuel, T.M. (2017). High performance text recognition using a hybrid convolutional-lstm implementation. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), 1: 11-16. <https://doi.org/10.1109/ICDAR.2017.12>
- [11] Scheidl, H., Fiel, S., Sablatnig, R. (2018). Word beam search: A connectionist temporal classification decoding algorithm. In 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR). IEEE, 253-258. <https://doi.org/10.1109/ICFHR-2018.2018.00052>
- [12] Ingle, R.R., Fujii, Y., Deselaers, T., Baccash, J., Popat, A.C.(2019). A scalable handwritten text recognition system. In 2019 International Conference on Document Analysis and Recognition (ICDAR). IEEE, pp. 17-24.
- [13] Edmundson, D., Schaefer, G. (2012). An overview and evaluation of jpeg compressed domain retrieval techniques. In *Proceedings ELMAR-2012*. IEEE, 2012, pp. 75-78.
- [14] Jain, M., Mathew, M., Jawahar, C. (2017). Unconstrained ocr for urdu using deep cnn-rnn hybrid networks. In 2017 4th IAPR Asian Conference on Pattern Recognition (ACPR). Nanjing, China. <https://doi.org/10.1109/ACPR.2017.5>
- [15] Varga, T., Bunke, H. (2003). Generation of synthetic training data for an hmm-based handwriting recognition system. In *Seventh International Conference on Document Analysis and Recognition, proceedings*, Edinburgh, UK, pp. 618-622. <https://doi.org/10.1109/ICDAR.2003.1227736>