Mathematical Modelling of Engineering Problems
Vol. 11, No. 11, November, 2024, pp. 3152-3160
Journal homepage: http://iieta.org/journals/mmep

CNN-DEdge: Multilingual Scene Text Detection and Extraction

Mahesha Mahadevappa[1*], V. N. Manjunath Aradhya[1], H. T. Basavaraju[2], Siddesha Shivarudraswamy[1]

[1] Department of Computer Applications, JSS Science and Technology University, Mysuru 570006, India
[2] Department of Computer Applications, Vidyavardhaka College of Engineering, Mysuru 570002, India

Corresponding Author Email: mahesha_m@sjce.ac.in

https://doi.org/10.18280/mmep.111125

Received: 7 May 2024
Revised: 22 July 2024
Accepted: 30 July 2024
Available online: 29 November 2024

Keywords:
bit plane slicing, CNN, double edge, multilingual scene text detection, multi-orientation text, scene text detection

ABSTRACT

In today's ever-changing environment, text appears in a variety of forms and it creates a significant barrier for text detection process. Text detection is still a visionary process due to scene-specific challenges such as text appearing on curved surfaces, varying lighting conditions, occlusions, and complicated environmental backgrounds. In this work we propose an algorithm customized CNN architecture with bit plane for identifying the textual regions from natural image samples. The customized CNN architecture is developed by defining the kernels to extract the accurate text edges. The bit plane slicing task extracts the contributions of all bits in scene image pixels. By removing least significant bit planes, helps to separate the background from the CNN samples. Finally, the double stroke method is applied to extract fine-grained actual text information. The implemented approach is tested on variety of datasets such as MSRA-TD500, Total Text, MLe2e, ICDAR 2015, and MRRC. The effectiveness of the developed procedure is evaluated using precision, recall, and F-measure parameters. The performance of our CNN - Bitplane - Double edge technique found better than available machine learning methods but not surpassed accuracy of the deep learning models. It has proved distinctively simpler and lightweight model than heavy deep learning models.

1. INTRODUCTION

Identification of scene textual contents in an image is a critical procedure of event understanding and has important consequences for artificial intelligence. Scene text identification is an important process in the area of computer vision, which has realistic applications like automatic annotation, indexing systems and information retrieval systems. Multilingual scene text detection also plays vital role in real time scenarios like vehicle number plate detection in traffic violations, text with sign boards to assist tourists and autonomous or self-driven cars with various native language support, robots managed mass production factories in detecting the text on lanes or products and scene understanding, like wise many applications can be found. Text identification is commonly considered as similar process of object identification, however précised outcomes are rarely produced using standard object detection techniques. Text occurs in a various characteristics like length, dimensions, font and orientation. Early studies concentrated on text-specific properties such as texture and separation of text from the background. Over the previous few decades, sliding window concept and connected component models have evolved into common concept to the textual data identification task. Sliding window concept searches possible text spots in pyramid images using different sliding window ratios and sizes, which incur a large computational cost. Connected component-based algorithms, like maximum stable extremal regions (MSERs) and the stroke width transform (SWT) [1].

Deep learning concepts for scene text identification are typically categorized as regression and segmentation methods. In the early phases of deep learning-based systems, handled the scene textual data detection as a general object identification job, regressing the bounding boxes to locate the text regions. Deep Neural Networks also lately applied to a variety of problems such as textual data detection and script recognition in images. For example, EAST [2] detects text regions by regressing text boxes and representing them with quadrilaterals with rotation angles. CNNs form the foundation of most deep learning algorithms for text detection. Algorithms automatically learn hierarchical feature representations from raw pixel data, making them extremely useful for image-related applications. The Inception Text (IncepText) approach detects orientated text using a Deformable Position Sensitive Region of Interest Pooling concept (PSRoI Pool) [3]. IncepText can accommodate a variety of text sizes. Several methods have been undertaken to address the challenge of multilingual e2e textual data and script recognition. A region-based convolution neural networks (R-CNN) [4] and rapid R-CNN are two methods for region-based approaches.

Since still there is lot of scope to improve scene textual data detection process, here we are proposing our custom CNN architecture followed by postprocessing of thresholding, CCA, bit plane and double stroke method. As preprocessing of input image used gaussian blur, convert the image to grayscale and

normalizes the pixel values and calculation of gradient magnitude and orientation done using Sobel filter.

We have used CNN model for edge detection using four custom kernels. Kernel initializers used to extract the 4 different kinds of edges. Edges detected from 4 different convolutional filters are processed to form combined image for further process. ReLU used as activation function.

Combined edge image applied with binarization using OTSU threshold. Connected component analysis applied to extract closely associated pixels. Smaller components are filtered by setting maximum threshold criteria.

Bit plane slicing applied to filtered components, here contributions of each bit plane is analysed, by removing least significant bit planes enhances the processing of textual regions in images. Bitplane output is pipelined with Canny edge detection and dilation process for further refinement. Finally, by finding counters and filled to close any holes present, then masking with original image extracts prominent text parts of the image. We have tested our models on MSRA TD500, ICDAR 2017, MRRC and MLe2e data sets and results are compared with other deep learning models Test snake, EAST, segment link, pixel link, PAN, baseline CNN and ECN etc. Our proposed custom CNN-DEdge technique is faster, lightweight and effective model than heavy deep learning models which needs higher order processing machines and needed tedious training and testing processes.

2. RELATED WORKS

2.1 Machine learning based methods

In the last two decades, many techniques were introduced for textual data detection process using machine learning and deep learning concepts. Zhong et al. [5] presented a concept for text localization in colour images. Texts were generally localized using horizontal spatial variance, and text components were extracted by colour segmentation inside the localized areas. Wu et al. [6] used Gaussian derivative model for segmentation of textual data. Bounding boxes with the relevant textual data were created using heuristic principles on textual data like similarity height, alignment, and spacing. The outcomes are combined to provide final textual data identification. Li et al. [7] described a model for identifying textual data in videos. Local features are included the mean, as well as central moments of a wavelet decomposition. Zhong et al. [8] recommended that candidate caption text sections be localized in Discrete-Cosine-Transform (DCT) dense domain by utilizing the intensity variation information. Gllavata et al. [9] implemented a concept for statistically separating textual and non-textual areas using distribution of high frequency-based wavelet parameters. Unlike the approaches discussed above, which use filter or transform response coefficients as key properties, Kim et al. [10] developed an algorithm only uses raw pixel intensities. A SVM classifier is trained to create probability maps that search for text places and extents using adaptive mean shift.

2.2 Neural network-based models

Yu et al. [11] suggested a network of semantic reasoning for precise scene textual data identification, which incorporates a module for semantic reasoning to gather the global semantic background. The ResNet50 extracts the foundation of features from inputs, which enhances the results of the model to be deployed on mobile systems. Bagi et al. [12] introduced a concept called crowded TextSpotter, a lightweight scene textual data spotter for crowded environments. This technique is a powerful text spotting tool that may be utilized proficiently on devices with limited resources. It comprises multi-level appropriate information and backbone network. Hu et al. [13] devised concept for scene textual data detection that uses word-based samples. The model is trained with the word samples found in the clipped samples. The two types of convolution layers are employed using convolution layer and arbitrary oriented convolution layer to identify key features of textual data, which recover the accuracy slightly but decreases competence as the quantity of model factor increases.

Baek et al. [14] suggested a four-stage technique for identifying scene textual data in images. The approach evidently demonstrated the tradeoff among the efficiency and accuracy. The method demonstrated that compared to big and complicated neural network architectures such as VGG, ResNet, and others, Recurrent Convolution Neural Networks require far less parameters for feature extraction. Their best model extracts features using ResNet, which improves accuracy but loses efficiency due to computational complexity and memory usage.

Munjal et al. [15] proposed the STRIDE concept, which is a scene textual data identification structure that taken the word samples. The model is implemented with a CRNN approach to identify the textual data in scene samples and significantly decreased the quantity of factors compared to prior scene textual data identification models.

Kang et al. [16] suggested a multi scale residual property pyramid-based network to distinguish between neighboring texts. Ibrahim et al. [17] investigated the residual alteration division to broaden the accessible field and used two stem feature synthesis to improve text compassion. Nevertheless, this method faced the difficulty in a bulky structure, which typically decreases the presumption performance. Wang et al. [18] suggested an inexpensive identification model with post processing technique for efficient textual data detection. Liao et al. [19] developed a distinct binarized component to simplify post processing stage and improve textual data detection performance.

Zhang et al. [20] implemented a FCN to identify textual data regions and identify character candidates using MSER concept. Yao et al. [21] represented single textual data as a number of attributes, including textual data surface and orientation, and then used FCN to forecast the matching heat maps. Lyu et al. [22] used corner identification model to discover appropriate uneven quadrangles for textual data instances. He et al. [23] developed deep convolution networks to detect text in images by integrating multiple cues such as edge features, textural features, and semantic information. The model progressively refines detection results through several layers, enhancing its ability to handle varied text appearances and backgrounds. DeepText's holistic approach to feature integration marks a significant advancement in text detection technology. It’s quite common these days outdoor images appear with text of various sizes, fonts, colour, different orientations, occlusions, blurred, fogged condition and varying lighting conditions. Due to scene text appearance in complex nature detection of text needs to be fast, and robust detection method with less computation and minimum time. Hence, this research work is concentrated on light weighted CNN architecture with customized kernel concept.

3. PROPOSED MODEL

The Bitplane-based CNN architecture is designed with gradient features to detect the area of the textual components in natural scene samples. In the preprocessing step, a Gaussian filter is used to smooth intensity level of the text pixels, and histogram equalization is performed to advance the contrast of textual components in the samples. To determine the actual contour of text regions, a single-layer CNN kernel is developed using gradient magnitude and orientation features of text components. The bit plane slicing process extracts each bit's contribution to disconnect the textual data from the background data. Finally, the double-edge approach is employed to extract fine-grained text from real scene samples. The overall structure of scene textual data detection process is depicted in Figure 1.

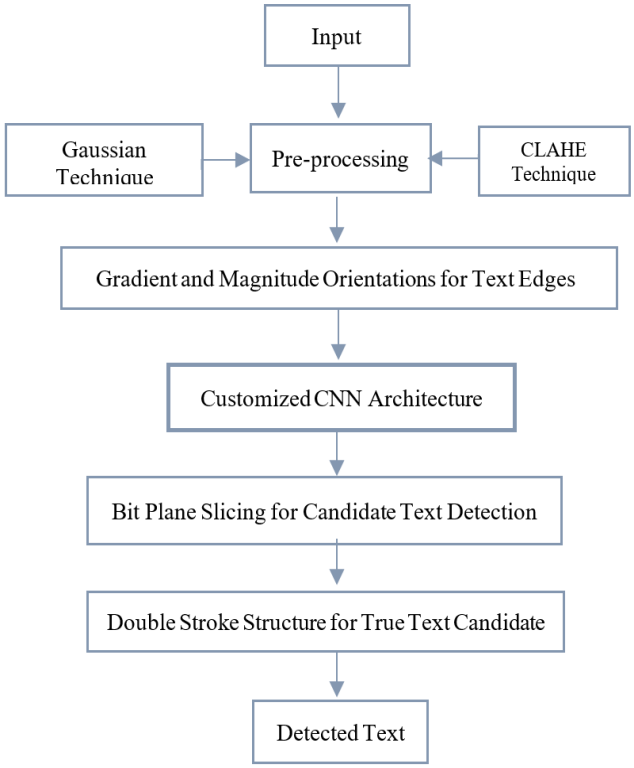

Figure 1. Proposed model for scene text detection process

3.1 Pre-processing

3.1.1 Gaussian filter for text smoothing

The Gaussian filter is a significant step in the preprocessing phase of textual data identification in real scene samples. The Gaussian technique helps to smoothen the pixels by reducing pixel unwanted distributions in the image. The Gaussian combines the pixels together whichever having most similar regions. This clue helps to combine the text pixels together by avoiding noises, and it retains the actual edge information.

$$\mathrm{I}'(\mathrm{x},\mathrm{y}) = (I * G)(x,y) = \sum_{i=-k}^{k}\sum_{j=-k}^{k} I(x-i, y-j).G(i,j) \quad (1)$$

In this process the given input image is considered as $I(x, y)$, and then k is a kernel with size 3×3 to discriminate the textual data, non-textual data and background pixels. The Gaussian G uses the standard deviation to control the width of the Gaussian kernel to group the deviation of text pixels across non-text regions and backgrounds. Finally, the Gaussian filtered image $I'(x, y)$ in Eq. (1) is obtained to process for further text detection task.

3.1.2 CLAHE technique for text components enhancement

Contrast Limited Adaptive Histogram Equalization (CLAHE) process is employed for enhancing the textual data regions in real scene text samples, CLAHE increases the contrast of textual data to distinguish from the complex backgrounds. Hence, CLAHE method focuses on local contrast development of textual data by controlling the noise with limiting adaptive threshold technique.

$$H_{ij}(p) = \min\left(I'(x,y)\, T_{ij}, clip\_limit\right) \quad (2)$$

The filtered Gaussian image $I'(x, y)$ is divided into tiles T_{ij} of size m×n. For each tile T_{ij}, the histogram $H_{ij}(p)$ is calculated using equation to convert into normalized intensity levels (p). The maximum value of histogram H_{ij} is clipped using clip limit value. The remaining excess are redistributed among the histogram. Figure 2(a) represents the initial state before any processing is applied. The outcome of preprocessing step is indicated in Figure 2(b). Figure 2(c) showcases the results after processing the pre-processed image through a customized CNN architecture.

(a) Input (b) Pre-processing (c) Customized CNN

Figure 2. Intermediate results of preprocessing and customized CNN architecture

3.2 Gradient and magnitude orientations for text edges

The gradient and magnitude are calculated and on outcome of preprocessing stage to extract the contour of the text components. Both gradient and magnitude process extracts the directional changes of all intensities in the histogram equalized image. It is computed using Eq. (3) derivative operators like the Sobel operator, which provides an approximation of the text direction in natural scene images.

In order to detect variations in pixel intensity across the image, the gradient magnitude $GM_{gradient_{magnitude}}$ is calculated using Sobel filters:

$$GM_{gradient_{magnitude}}(x,y) = \sqrt{\left(H_{gradientX}(x,y)\right)^2 + \left(H_{gradientY}(x,y)\right)^2} \quad (3)$$

where, $H_{gradientX}(x,y)$ and $H_{gradientY}(x,y)$ are calculated through convolution Sobel filter operations. Gradient and magnitude distinguish the textual data direction among the background data directions. So that textual data edges are identified by suppressing the background edge information. But still some of the non-text edges are also extracted due to sharpening nature as like text components.

3.3 Customized CNN architecture for possible text candidates

CNN architecture for text detection generally consists of several layers designed to progressively extract features from the input image and identify regions containing text. The main layers implicated in the model are convolutional, pooling, and fully connected layers as represented in Figure 3.

Our proposed customized CNN model is designed with four custom convolutional layers defining four edge kernels based on the various structural features image as represented in Figure 4. The convolution layers with single filter convolve on edge information in four different orientations (viz., vertical, horizontal, and two diagonal directions) using these four kernels of size 3×3 to extract the possible text candidates.

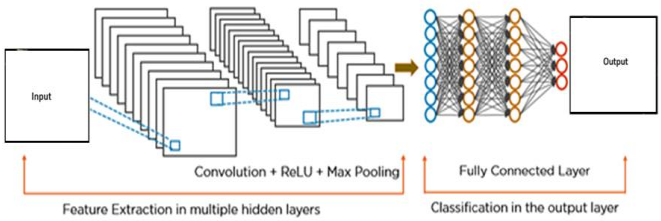

Figure 3. Architecture diagram of the CNN model

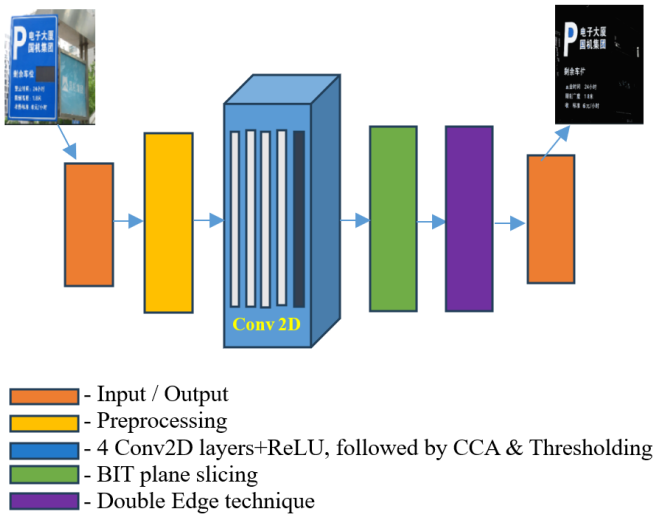

Figure 4. Structure of custom CNN-DEdge

The custom kernels are initialized and given to the convolution layers to study the text contours. The core of this method entails applying convolution operations to the preprocessed gradient magnitude and orientation of image while utilizing these unique kernels created especially for identifying the possible textual candidates. The mathematical illustration of the convolutions is given by:

$$I_{ed}(x,y) = \sum_{i,j} I_{\text{Normalized}}(x-i, y-j) \cdot K_{\text{edge}}(i,j) \quad (4)$$

In addition, these convolution layers' outputs undergo Rectified Linear Units (ReLU) activation function. The gradient and magnitude of image is processed by the custom CNN model in the inference stage to identify the text region. The resulting CNN image is the applied with binary thresholding by OTSU and CCA. Further, maximum pixel threshold criteria is applied, to filter probable text components and contours of filtered components also found. A mask is created to isolate the text regions and components with specified height and width limits are filled, it helps to extract candidate text regions from the image. The intermediate results of preprocessing and customized CNN architecture are as depicted in Figure 2.

3.4 Bit plane slicing for candidate text detection

Bitplane slicing is a technique that divides an image into its constituent bit planes. Every pixel in the sample is encoded by an 8-bit binary format, each bit has its contribution in the representing some properties of a pixel. Specific bit planes are extracted using bitwise AND operation between gray scale image and a binary mask. By manipulating least significant bit planes, visibility of certain features or details of the image can be improved. By retaining only most significant bit planes, helped to separate the fine grain text from background image by considering high frequency textual properties. The output of Bitplane slicing process applied with adaptive thresholding technique to retain the prominent text candidates. Final outcome of Bitplane slicing is shown in Figure 5(a).

(a) Bitplane Slicing

(b) Double Stroke Structure

(c) Final Output

Figure 5. Results after the application of Bitplane slicing and double stroke structure

3.5 Double stroke structure for true text candidates

The important inspection in any circumstance is that textual data in images is composed of double-stroke structures. It is readily apparent to identify the double stroke structures [24] in order to retrieve the real textual data candidates. This proof of textual data property has helped to consider the contour information as a major property. As a result, textual data have double-stroke-shaped components, but the majority of non-textual data does not have such structures. If the linked component's starting and ending points are the identical, it is classified as circular connected information and is considered a true textual candidate. Or else, it is classified as a non-textual information. The double stroke structure and final outcome is represented in Figure 5(b) and 5(c) respectively.

4. EXPERIMENTATION RESULTS AND DISCUSSION

Experimentation is conducted using proposed method with

datasets such as MSRA -TD500, Total text, ICDAR -2015, MRRC, and MLe2e (Derived dataset) and obtained significantly better results than typical machine learning algorithms. The proposed CNN-Bitplane-DoubleEdge method flourishes in a variety of ways, including a simple and very successful text detection strategy in which our single layer CNN alleviates to be light weight, yet effective approach than the heavy deep learning models such as TextBox++, EAST, RRPN, TextSpotter, CRAFT. The performance metrics such as Precession (P) (Eq. (5)), Recall (R) (Eq. (6)), and the F-measure (F) (Eq. (7)) is used to test the efficacy of the proposed technique. For the calculation of performance metrics, three characteristics are utilized such as TDR (True Detected Text Regions), APTR (Actually Text Regions Present) in the image, and FIR (False Identified Text Regions), which is incorrectly identified non-text regions as text. The following subsections depicts the comparative analysis of proposed method with other existing methods. The performance fact of text detection rate of the proposed method found better than the machine learning approaches but not surpassed performance of the deep learning models in some cases. In terms of advantages, CNN-Bitplane DEdge approach has simplicity, lightweight model, less computation time, faster execution, no training and testing required and it has produced results in few milli seconds. Hence proposed method is better solution than bulky deep learning models and their long iterations of training and testing process.

$$\text{Precision (P)} = \frac{TDR}{TDR+FIR} \tag{5}$$

$$\text{Recall (R)} = \frac{TDR}{APTR} \tag{6}$$

$$\text{F} - \text{Measure (F)} = \frac{2*p*r}{P+R} \tag{7}$$

4.1 Experimental results using MSRA-TD500

MSRA-TD500 dataset is compiled and made as a public benchmark for evaluating text detection algorithms, with the goal of tracking recent advances in the field of text detection in natural images. This dataset has 300 training and 200 testing images. The external outdoor images include visitor information boards and billboards against a complicated background, and the indoor images include signs, doorplates, and caution plates. This dataset presents significant challenges due to its diverse contents and complex backdrops in images. The textual data in images appears with various languages such as Chinese, English, or a combination of both, with varieties of fonts, sizes, colours, and orientations. The outcome of the proposed algorithm on MSRA-TD500 dataset is represented in Figure 6. Table 1 represents the quantitative performance comparison of the developed method verses previous methods on MSRA-TD500 dataset. In our approach we have used edge enhancement techniques like Sobel and Canny and our simple custom CNN built with gradient features and orientation identifies true text edges. Bit plane slicing does the clearly distinguishing text and non-text background. Precision found that its better than symmetric patterns method but its lesser than HOCC, EAST and Pixel Link as shown in table. In CNN DEdge method no need of training and testing like any other deep learning models and its slimmer model and executes faster than DNN models.

Figure 6. Outcome of CNN DEdge on MSRA-TD500 dataset

Table 1. Performance analysis of CNN-Bitplane DEdge on MSRA dataset

Methods	Precision	Recall	F-Measure
Symmetric patterns [25]	70	68	69
HOCC [26]	72	62	66
EAST [27]	87	67	76
Pixel Link [28]	83.0	73.2	77.8
CNN-Bitplane-DEdge	**71.23**	49.87	58.67

4.2 Experimental results on total text

Total text dataset has much images with various multi orientation texts and curved texts apart from horizontal texts. Previous other data sets like ICDAR2013 and MSRA TD500 do not have much curved texts and multi orientation text images. Hence Total text is a fulfilment of the non-existence of required data. Results after experimentation of the proposed algorithm is illustrated in Figure 7. Table 2 depicts the experimental process of the implemented method on Total Text dataset. The results of the implemented method perform efficiently on Total Text dataset as compared to previous approaches. Bit plane slicing does the remarkable performance in our algorithm, it refined each bit contribution of text pixels, results shows that even in complex and various lighting conditions it has produced good results. Our single layer custom CNN Bitplane method found better in precision and average case in F measure value. Our technique comes with slim and light weight approach, will not consume much memory and very much faster compared to other heavy deep network models.

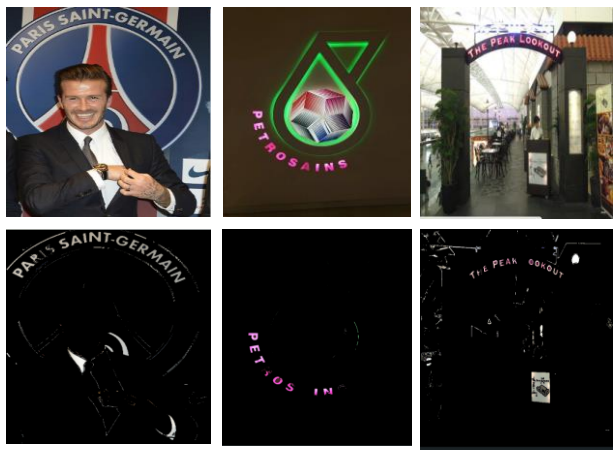

Figure 7. Outcome of CNN DEdge on total text dataset

Table 2. Performance analysis of CNN-Bitplane DEdge on total text dataset

Methods	Precision	Recall	F-Measure
PAN-320 [18]	85.6	75.0	79.9
TextSnake [29]	82.7	74.5	78.4
Baseline CNN [30]	80.06	85.45	82.67
Baseline Resnet-50 [31]	87.44	84.93	86.16
CNN-Bitplane-DEdge	**87.7**	73.06	79.69

4.3 Experimental results on MRRC dataset

This dataset created by IISC professors for their event Multi Script Robust Reading Competition in ICDAR 2013. This competition organised to find the scene text detection algorithm which is independent of scripts, able to extract the texts from given scenic images. This data set contains 4000 images and pictures taken in and around Bangalore, contains texts with many orientations, multilingual, many found with complex environment. Output of the experiment conducted on MRRC dataset is showed in Figure 8. Table 3 shows that quantitative experimental process of proposed approach on MRRC dataset. The outcome of the developed method performs successfully on MRRC dataset as compared to existing models. In our proposed method, post processing of our CNN, the output is applied with OTSU binarization, CCA to fine tune connected regions and threshold method used to filter out unwanted small components. This helps out retain the text candidate regions from non-text part. Various segmentation strategies OTCYMIST and Seglink methods used by Gomez et al. [32] but not yielded any good results. Proposed CNN Bit plane Double edge is significantly yielded better results than other deep models compared, also thrived in terms of simplicity, faster execution time and less usage of memory.

Table 3. Performance analysis of CNN-Bitplane-DEdge on MRRC dataset

Methods	Precision	Recall	F-Measure
Kumar et al. [33]	64	58	61
Kumar et al. [33]	71	67	69
Basavaraju et al. [24]	81.81	69.52	75.16
CNN-Bitplane-DEdge	**87.27**	**73.62**	**79.87**

Figure 8. Output of CNN DEdge on MRRC dataset

4.4 Experimental results on ICDAR 2017 dataset

ICDAR-2017 dataset contains 500 testing and 1000 training images. Here four quadrilateral vertices are used to annotate the texts created for scene text identification and recognition tasks. Basically, these images focused with incidental scene text found with challenges like illumination, complex background, depth, and occlusion. The output of the proposed approach on ICDAR-2017 is represented in Figure 9. In our approach we have applied contour-based filter technique by creating mask of input image, to helps out to identify the protentional text properties with appropriate height and width of characters. Further output will be pipelined to bit plane slicing and double edge technique, to separate the candidate text regions and non-text part of given image. Results of proposed method and other network models tested on ICDAR-2017 listed in Table 4. The proposed model gave better precision than GCN method and less than other models. Other models need many iterations of training and testing, huge memory and higher processing machines, but CNN DEdge does not need any training and testing and works faster due its simpler approach.

Table 4. Performance analysis of CNN-Bitplane DEdge on ICDAR-2017 dataset

Methods	Precision	Recall	F-Measure
GCN [30]	75.0	61.0	67.3
Sequential Deformation Resnet-18 [31]	82.14	70.72	76
TFAM [34]	85.7	70.6	77.4
Transformer based [35]	84.75	63.23	72.4
Swim transformer [36]	86.6	75.3	80.5
CNN-Bitplane-DEdge	**78.32**	57.97	66.62

Figure 9. Output of CNN DEdge on ICDAR-2017 dataset

4.5 Experimental results on MLe2e dataset

Multilingual end-to-end (MLe2e) scene text dataset used as another benchmark dataset to evaluate the scene text detection and recognition algorithms. It is used to find the efficacy and accuracy of text detection, script identification, and text recognition. The MLe2e dataset was compiled from variety of existing scene text datasets, with the images and ground truth updated to ensure homogeneity. This dataset includes 711 scene photos in four distinct scripts like Latin, Chinese, Kannada, and Hangul. The output of the proposed approach on MLe2e dataset is shown in Figure 10. The experimental results conducted on MLe2e dataset is compared with other approaches shown in Table 5, proposed CNN DEdge yielded very good results than other methods in terms of precision and f-measure. In our proposed method we apply double edge method after bit plane slicing. This process generates the fine-grained true text pixels with greater accuracy. Since its

slimmer approach, fastness and no overhead of training and testing, proposed method can used as successful scene text detection technique.

Figure 10. Output of CNN DEdge on MLe2e dataset

Table 5. Performance analysis of CNN-Bitplane DEdge on MLe2e dataset

Methods	Precision	Recall	F-Measure
ECN [32]	51	62	56
Enhanced Receptive Field [37]	80.0	86.0	83
Gradient Morphology [38]	88	90	89
CNN-Bitplane-DEdge	**94.57**	85.33	**89.71**

5. CONCLUSIONS

As appearance of text in the scene images is quite common, many novelties of extraction and recognition of scene text are increasing day by day due to its abundant applications. But still the detection method become critical task due to uneven lighting, background clutter, occlusions, perspective distortions and it appears in multilingual and multi orientation nature. In order to overcome, the proposed approach developed a customized CNN architecture with bit plane slicing and double stroke method. The pre-processing effectively highlights the text regions by employing Gaussian technique with CLAHE technique. The gradient and magnitude orientation groups the unique text pixels directional features and customized CNN architecture is implemented by using kernels, which are defined to identify the contour of the text region irrespective of the language and orientation. Bit slicing helps to retain actual text candidates from the CNN image by isolating and analysing specific bit planes. Finally, the CNN-bit plane associated with double line structure method employed to extract the true text candidates from natural scene images. Proposed CNN DEdge is slimmer model due to simplicity and faster execution time, does not require large memory and high-powered machines like deep learning models and no training and testing required. Hence CNN DEdge stands out better than deep CNN models. Since the current work focused on text detection, future work can be focused on complete text recognition in scene images. Further, development of e2e scene text application for mobiles would serve the public need to the better extent.

REFERENCES

[1] Epshtein, B., Ofek, E., Wexler, Y. (2010). Detecting text in natural scenes with stroke width transform. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, pp. 2963-2970. https://doi.org/10.1109/CVPR.2010.5540041

[2] Zhou, X.Y., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J. (2017). East: An efficient and accurate scene text detector. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, pp. 5551-5560. https://doi.org/10.1109/CVPR.2017.283

[3] Yang, Q.P., Cheng. M.L., Zhou, W.M., Chen, Y., Qiu, M.H., Lin, W. (2018). Inceptext: A new inception-text module with deformable PSROI pooling for multi-oriented scene text detection. In International Joint Conference on Artificial Intelligence, pp. 1071-1077. https://doi.org/10.24963/ijcai.2018/149

[4] Girshick, R. (2015). Fast R-CNN. In IEEE International Conference on Computer Vision, Santiago, Chile, pp. 1440-1448. https://doi.org/10.1109/ICCV.2015.169

[5] Zhong, Y., Karu, K., Jain, A.K. (1995) Locating text in complex color images. Pattern Recognition, 28(10): 1523-1535. https://doi.org/10.1016/0031-3203(95)00030-4

[6] Wu, V., Manmatha, R., Riseman, E., (1997). Finding text in images. In DL97: 2nd ACM International Conference on Digital Libraries, Philadelphia, Pennsylvania, USA, pp. 3-12. https://doi.org/10.1145/263690.263766

[7] Li, H.P., Doermann, D., Kia, O. (2000). Automatic text detection and tracking in digital video. IEEE Transactions on Image Processing, 9(1): 147-156. https://doi.org/10.1109/83.817607

[8] Zhong, Y., Zhang, H., Jain, A.K. (2000). Automatic caption localization in compressed video. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(4): 385-392. https://doi.org/10.1109/34.845381

[9] Gllavata, J., Ewerth, R., Freisleben, B. (2004) Text detection in images based on unsupervised classification of high-frequency wavelet coefficients. In Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, pp. 425-428. https://doi.org/10.1109/ICPR.2004.1334146

[10] Kim, K.I., Jung, K., Kim, J.H. (2003). Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(12): 1631-1639. https://doi.org/10.1109/TPAMI.2003.1251157

[11] Yu, D., Li, X., Zhang, C., Liu, T., Han, J., Liu, J., Ding, E. (2020). Towards accurate scene text recognition with semantic reasoning networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, pp. 12113-12122. https://doi.org/10.1109/CVPR42600.2020.01213

[12] Bagi, R., Dutta, T., Gupta, H.P. (2020). Cluttered Textspotter: An end-to-end trainable light-weight scene text spotter for cluttered environment. IEEE Access, 8: 111433-111447. https://doi.org/10.1109/ACCESS.2020.3002808

[13] Hu, J., Liao, X., Wang, W., Qin, Z. (2021) Detecting

compressed deepfake videos in social networks using frame-temporality two-stream convolutional network. IEEE Transactions on Circuits and Systems for Video Technology, 32(3): 1089-1102. https://doi.org/10.1109/TCSVT.2021.3074259

[14] Baek, J., Kim, G., Lee, J., Park, S., Han, D., Yun, S., Oh, S.J., Lee, H. (2019). What is wrong with scene text recognition model comparisons? Dataset and model analysis. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea (South), pp. 4715-4723. https://doi.org/10.1109/ICCV.2019.00481

[15] Munjal, R.S., Prabhu, A.D., Arora, N., Moharana, S., Ramena, G. (2021). Stride: Scene text recognition in-device. In 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, pp. 1-8. https://doi.org/10.1109/IJCNN52387.2021.9534319

[16] Kang, J.J., Ibrahim, M, Hamdulla, A. (2022). MR-FPN: Multi-level residual feature pyramid text detection network based on self-attention environment. Sensors, 22(9): 3337. https://doi.org/10.3390/s22093337

[17] Ibrahim, M., Li, Y., Hamdulla, A. (2022). Scene text detection based on two-branch feature extraction. Sensors, 22(16): 6262. https://doi.org/10.3390/s22166262

[18] Wang, W., Xie, E., Song, X., Zang, Y., Wang, W., Lu, T., Yu, G., Shen, C. (2019). Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea (South), pp. 8440-8449. https://doi.org/10.1109/ICCV.2019.00853

[19] Liao, M., Wan, Z., Yao, C., Chen, K., Bai, X. (2020). Real-time scene text detection with differentiable binarization. In Proceedings of the AAAI Conference on Artificial Intelligence, 34: 11474-11481. https://doi.org/10.1609/aaai.v34i07.6812

[20] Zhang, Z., Zhang, C., Shen, W., Yao, C., Liu, W., Bai, X. (2016). Multi-oriented text detection with fully convolutional networks. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 4159-4167. https://doi.org/10.1109/CVPR.2016.451

[21] Yao, C., Bai, X., Sang, N., Zhou, X., Zhou, S., Cao, Z. (2016). Scene text detection via holistic, multi-channel prediction. arXiv Preprint, arXiv:1606.09002. https://doi.org/10.48550/arXiv.1606.09002

[22] Lyu, P., Yao, C., Wu, W., Yan, S., Bai, X. (2018). Multi-oriented scene text detection via corner localization and region segmentation. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, pp. 7553-7563. https://doi.org/10.1109/CVPR.2018.00788

[23] He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J. (2015). Deep residual learning for image recognition. arXiv:1512.03385. https://doi.org/10.48550/arXiv.1512.03385

[24] Basavaraju, H.T., Aradhya, V.M., Guru, D.S., Harish, B.S. (2018). LoG and structural based arbitrary oriented multi lingual text detection in images/video. International Journal of Natural Computing Research, 7(3): 1-16. https://doi.org/10.4018/IJNCR.2018070101

[25] Risnumawan, A., Shivakumara, P., Chan, C.S., Tan, C.L. (2014). A robust arbitrary text detection system for natural scene images. Expert Systems with Applications, 41: 8027-8048. https://doi.org/10.1016/j.eswa.2014.07.008

[26] Kang, L., Li, Y., Doermann, D. (2014). Orientation robust text line detection in natural images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, pp. 4034-4041. https://doi.org/10.1109/CVPR.2014.514

[27] Basavaraju, H.T., Aradhya, V.N.M., Guru, D.S. (2019). Text detection through hidden Markov random field and EM-algorithm. In Information Systems Design and Intelligent Applications. Advances in Intelligent Systems and Computing, vol. 862, pp. 19-29. Springer, Singapore. https://doi.org/10.1007/978-981-13-3329-3_3

[28] Deng, D., Liu, H., Li, X., Cai, D. (2018). Pixellink: Detecting scene text via instance segmentation. In Proceedings of the AAAI Conference on Artificial Intelligence. https://doi.org/10.1609/aaai.v32i1.12269

[29] Long, S., Ruan, J., Zhang, W., He, X., Wu, W., Yao, C. (2018). Textsnake: A flexible representation for detecting text of arbitrary shapes. In Proceedings of the European Conference on Computer Vision (ECCV), pp. 19-35. https://doi.org/10.1007/978-3-030-01216-8_2

[30] Zhang, S.X., Zhu, X., Hou, J.B., Liu, C., Yang, C., Wang, H., Yin, X.C, (2020). Deep relational reasoning graph network for arbitrary shape text detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, pp. 9699-9708. https://doi.org/10.1109/CVPR42600.2020.00972

[31] Xiao, S., Peng, L., Yan, R., An, K., Yao, G., Min, J. (2020). Sequential deformation for accurate scene text detection. In European Conference on Computer Vision, pp. 108-124. https://doi.org/10.1007/978-3-030-58526-6_7

[32] Gomez, L., Nicolaou, A., Karatza, D. (2017). Improving patch-based scene text script identification with ensembles of conjoined networks. Pattern Recognition, 67: 85-96. https://doi.org/10.1016/j.patcog.2017.01.032

[33] Kumar, D., Prasad, M.A., Ramakrishnan, A.G. (2013). Multi-script robust reading competition in ICDAR 2013. In Proceedings of the 4th International Workshop on Multilingual OCR, Washington D.C. USA, pp. 1-5. https://doi.org/10.1145/2505377.2505390

[34] He, M., Liao, M., Yang, Z., Zhong, H., Tang, J., Cheng, W., Yao, C., Wang, Y., Bai, X. (2021). Most: A multi-oriented scene text detector with localization refinement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA. pp. 8813-8822. https://doi.org/10.1109/CVPR46437.2021.00870

[35] Raisi, Z., Naiel, M.A., Younes, G., Wardell, S., Zelek, J.S. (2021). Transformer-based text detection in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, pp. 3162-3171. https://doi.org/10.1109/CVPRW53098.2021.00353

[36] Wang, T. (2023). Research on multilingual natural scene text detection algorithm. arXiv Preprint arXiv:2312.11153. https://doi.org/10.48550/arXiv.2312.11153

[37] Yang, Y.W., Ibrahim, G., Zhu, Y.L., Ubul, K., Mamt, H. (2022). A method of text detection and script identification in natural scene. In International Conference on Virtual Reality, Human-Computer

Interaction and Artificial Intelligence, Changsha, China, pp. 43-48. https://doi.org/10.1109/VRHCIAI57205.2022.00014
[38] Basu, S., Dhar, D., Chakraborty, N., Choudhury, S., Paul, A., Mollah, A.F., Sarkar, R. (2020). Multilingual scene text detection using gradient morphology. International Journal of Computer Vision and Image Processing, 10(3): 31-43. https://doi.org/10.4018/IJCVIP.2020070103