# Text Detection from Scene Videos having Blurriness and Text of Different Sizes

Mayuri A. Mehta
Department of Computer Engineering
Sarvajanik College of Engineering and Technology
Surat, Gujarat, India.
mayuri.mehta@scet.ac.in

Saloni A. Pote
Department of Computer Engineering
Sarvajanik College of Engineering and Technology
Surat, Gujarat, India.
salonipote10@gmail.com

***Abstract*—Text in videos needs to be detected correctly as it contains important information related to natural scene in videos. Text detection is useful for automatic number plate recognition, street boards reading, Optical Character Recognition (OCR), automatic document scanning and to help blind or visually impaired people. The existing scene text detection techniques are unable to detect text accurately from the videos having low contrast, complex background or excessively small fonts. Hence, we propose a new text detection technique that enhances accuracy of detecting text and reduceses average processing time. Our text detection technique incorporates Edge-enhanced Maximally Stable Extremal Regions (eMSER) method to preserve shape of characters and modified fuzzy C-means clustering to converge faster. In this paper, we provide an experimental analysis of our improved text detection technique. To show the effectiveness of our proposed text detection technique, we have performed experiments considering videos having text of different sizes as well as considering blur videos. The experimental results show that an improved text detection technique outperforms an eMSER based text detection technique.**

***Keywords—Scene text, Text detection, Stroke Width Transform (SWT), Maximally Stable Extremal Regions (MSER)***

## INTRODUCTION

Text detection from scene videos have been receiving increasing attention due to their extensive use in applications such as optical character recognition, wearable camera devices, automatic number plate recognition, automatic document scanning, augmented reality translators and to assist blind or visually impaired people [1-6]. Text embedded in video is classified into two major types [1-2][4][6]: caption text and scene text. Caption text is added manually in video during video processing. It is easily detected due to its good clarity and appearance. Scene text is intuitively available since the time video was captured. Its detection is non-trivial as it may include distortion, complicated background, little contrast, different orientations, color bleeding or blurriness [1-6].

Several scene text detection techniques are available in literature. In [1], Multi-spectral fusion based technique that identifies text of random orientation is proposed. Nevertheless, it fails to detect text correctly from the video containing low contrast, highly composite background, changes in illumination and excessively small font size. Additionally, its text detection time is higher. In [2], Laplacian based technique to detect text of arbitrary orientation is presented. It either fails to detect excessively small or large size text or detects it incorrectly. In [3], authors have proposed an Edge-enhanced MSER based approach that is unable to identify text correctly when video frames are excessively blurred. Multi-strategy tracking based technique to detect text is proposed in [4]. It is inefficient to detect text in case of fast arbitrary movement of camera. In [6], Map-Reduce based text detection technique is proposed that considerably reduces text detection time. However, it represents noisy regions with detected text. Text detection techniques presented in [2][7] either fail to detect text that is extremely small or large in size or partially detect it. Thus, it is needed to design a new technique that to identify different size text correctly even from blur videos.

In this paper, we present a new improved text detection technique to recognize text of different sizes and text from blurred video frames [8]. Furthermore, we present the performance evaluation of an improved text detection technique comparing its results with eMSER based text detection technique [3]. Experimental results show that an improved text detection technique reduces average processing time as it employs distributed canny edge detection that divides video frame into distinct fragments and processes them concurrently. As segmentation restricts noise, it also assists in increasing accuracy. Our improved text detection technique integrates eMSER to deal with blurriness and to preserve the shape of characters. Furthermore, it employs modified-fuzzy C-means clustering that converges faster than the original fuzzy C-means clustering, thereby reduces an average processing time.

The paper is organized as follows: In section 2, we illustrate an improved text detection technique. Experimental results and analysis are discussed in section 3. Finally, the conclusions and future work are presented in section 4.

## AN IMPROVED TEXT DETECTION TECHNIQUE

The distinguished characteristics of our text detection technique are as follows: (1) It detects text of all sizes such as small, medium and large. (2) It detects text from video having poor contrast. Table I describes the notations used through the paper. Our text detection technique consists of four major steps: pre-processing, text localization, text line formation and text identification. As depicted in Fig. 2, input video is first partitioned into various video frames $F_1$, $F_2$, ..., $F_n$. Subsequently, the above mentioned four steps are applied on individual video frame to identify text.

During preprocessing step, each video frame F is transformed from RGB to gray level. RGB represents each pixel using red, green and blue components. Whereas, gray level represents each pixel using two components: black and white having intensity values ranging between [0, 255]. Thus, transformation from RGB to gray is useful to process and manipulate video frame easily and speedily.

After converting each video frame into gray level, all the boundaries present in video frame are detected using distributed canny edge detection method [9-10]. Considering chosen threshold value, each video frame is partitioned into multiple segments. The segments are

TABLE I. DESCRIPTION OF NOTATIONS

| Notation | Description |
|---|---|
| V | Video |
| F | Set of frames created from video |
| $F^g$ | Set of frames converted to gray scale |
| $Fdcann_y$ | Set of frames having identified edges |
| $F^{MFCM}$ | Set of frames having identified text pixels |
| $F^{eMSER}$ | Set of frames having identified text regions |
| $F^{SWT}$ | Set of frames with stroke width transform |
| $F^{TF}$ | Set of frames with textline formation |
| F' | Set of frames with detected text |

**Input:** Video V

**Output:** Set F' of video frames with detected text

Text Detection (V)

Begin

1. Segment video V into multiple frames $F_1, F_2, \ldots, F_n$
2. for j = 1 to n video frames
3. $F_j^g$ = rgbTogray($F_j$)
4. $F_j^{d_{canny}}$ = Distributed_Canny_Edge ($F_j^g$)
5. $F_j^{MFCM}$ = Modified_Fuzzy C-Means ($F_j^{d_{canny}}$)
6. $F_j^{eMSER}$ = Edge_Preserving_MSER ($F_j^{MFCM}$)
7. $F_j^{SWT}$ = StrokeWidthTransform($F_j^{eMSER}$)
8. $F_j^{TF}$ = Textline_Formation($F_j^{SWT}$)
9. F' = F' ∪ Textline_Verification $F_j^{TF}$

End

Fig. 1. Psuedo code of an improved text detection technique.

processed concurrently using gradient mask of small size to identify the boundaries. Concurrent processing of segments greatly decreases the average processing time compared to other edge detection techniques. Moreover, distributed canny edge detection technique assists in increasing text detection accuracy as it involves segmentation.

The next step in text detection process is text localization which identifies text location in video frame. It comprises of three sub-steps: text pixels detection, text regions detection and stoke width transform. The candidate text pixels are detected using Modified Fuzzy C-Means (MFCM) clustering technique. It converges faster than the other clustering techniques available in literature [1-2][5]. It involves the concept of compression that consists of quantization and aggregation. In quantization process, common intensity values are assigned to several feature vectors. In aggregation process, the feature vectors having common intensity values are grouped and subsequently, mean value is calculated. The rest of the functioning of MFCM is same as that of the conventional Fuzzy C-Means (FCM) clustering. Thus, unlike conventional Fuzzy C-Means clustering technique, in MFCM technique, the rest of the steps are applied on reduced dataset and therefore, it comparatively converges faster.

Pixels identified as candidate text pixels are merged into text regions. MSER is widely used and efficient text region detector available in literature [1-3][11]. However, it is inadequate to preserve the shape of lost characters or the characters that are falsely connected. Additionally, it is unable to deal with blur in video frame. Hence, text regions are detected using edge preserving MSER. In eMSER, noise is removed using guided filter. Subsequently, gradient amplitude map is computed and is normalized to [0, 255]. Finally, the original MSER is applied. eMSER assists in preserving the shape of candidate text pixels, however, it is unable to eliminate non-text pixels. Therefore, we use stroke width transform that assists in removal of non-text candidate pixels [4][6]. Using SWT, each pixel is assigned stroke width value. Subsequently objects possessing high stroke width value are removed to identify character candidates. Finally, text lines are produced by combining characters having same stroke value and height value.

## EXPERIMENTAL RESULTS AND ANALYSIS

We test the effectiveness of an improved text detection technique via performing experiments using MATLAB. Videos from ICDAR 2015 dataset are used to conduct the experiments. Specifically, we consider following two types of videos from this dataset:

- Videos with small, medium and large text size
- Videos with blurriness

The performance is evaluated using two parameters: accuracy and Average Processing Time (APT). The accuracy is calculated using True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) that are defined as follows:

TP: Count of characters identified correctly.

TN: Count of characters that are not identified.

FP: Count of characters identified incorrectly.

FN: Count of characters inappropriately rejected.

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

Accuracy = (TP + TN) / (TP + TN + FP + FN)

APT represents the total time taken by our text detection technique to detect text correctly. The performance of An Improved Text Detection Technique (AI_TDT) is compared with eMSER based Text Detection Technique (MSER_TDT).

### *A. Performance Evaluation Considering Video Frames with Different Text Size*

In our first experiment, we analyze the performance of AI_TDT considering three types of text: small, medium and large. Fig. 2(a) shows the input video frame of size 1450 KB. It includes small size text. Fig. 2(b) represents the output


(a) (b)

Fig. 2. (a) Input video frame with small size text. (b) Output video frame with identified text.

TABLE II. RESULTS FOR VIDEO FRAME WITH SMALL FONT SIZE

| Parameter | AI_TDT | MSER_TDT |
|---|---|---|
| Precision | 0.913 | 0.82 |
| Recall | 0.954 | 0.9 |
| Accuracy | 0.9 | 0.8 |
| APT (Sec) | 8.59 | 9.74 |

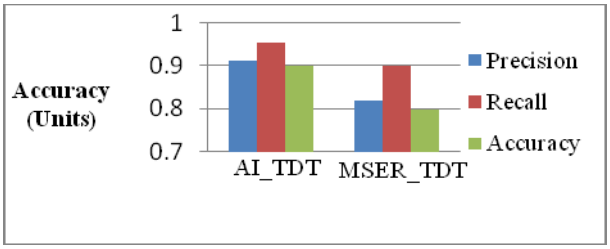



Fig. 3. Performance evaluation of video frame containing small size text.

frame with detected text. Table II and Fig. 3 show the performance of AI_TDT and MSER_TDT. Results show that AI_TDT increases accuracy by 10% and reduces APT by 1.15 seconds compared to MSER_TDT. It has been observed that though AI_TDT performs better than MSER_TDT, it is unable to accurately detect few smaller size fonts as MFCM uses compressed video frames.

Fig. 4(a) and Fig. 4(b) show the input video frames of size 177 KB and 499 KB respectively. They contain text with medium font size. Fig. 4(c) and Fig. 4(d) show the video frames with detected text. Table III, Table IV, Fig. 5(a) and Fig. 5(b) show the performance of AI_TDT. Experimental results show that an improved text detection technique detects medium size characters. For input video frames shown in Fig. 4(a) and Fig. 4(b), it increases accuracy by 1.21% and 2% respectively. Moreover, it reduces APT marginally. Increase in accuracy is marginal because few characters are detected inappropriately due to use of smoothing process in eMSER.


(a) (b)


(c) (d)

Fig. 4. (a) and (b) Input video frames with medium size text. (c) and (d) Output video frames with identified text.

TABLE III. RESULTS FOR VIDEO FRAME SHOWN IN FIG. 4(a)

| Parameter | AI_TDT | MSER_TDT |
|---|---|---|
| Precision | 0.8902 | 0.86 |
| Recall | 0.9865 | 0.9 |
| Accuracy | 0.8795 | 0.8674 |
| APT (Sec) | 11.51 | 11.81 |

TABLE IV. RESULTS FOR VIDEO FRAME SHOWN IN FIG. 4(b)

| Parameter | AI_TDT | MSER_TDT |
|---|---|---|
| Precision | 0.9 | 0.83 |
| Recall | 0.9 | 0.9 |
| Accuracy | 0.9 | 0.88 |
| APT (Sec) | 11.95 | 10.95 |

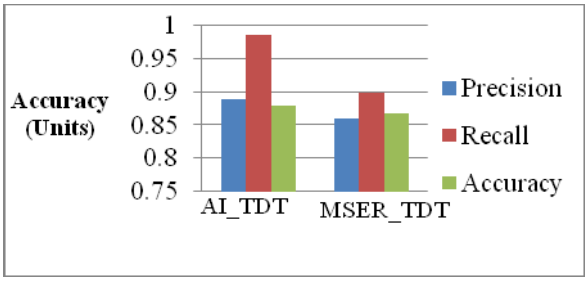



Fig. 5. (a): Performance evaluation of video frame (Fig. 4(a)) containing medium size text.

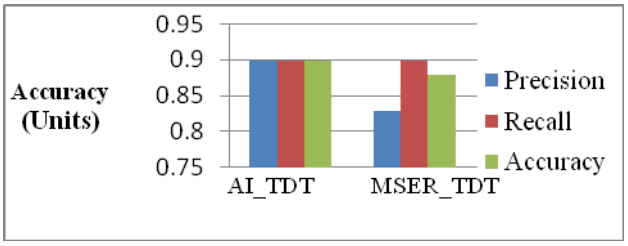
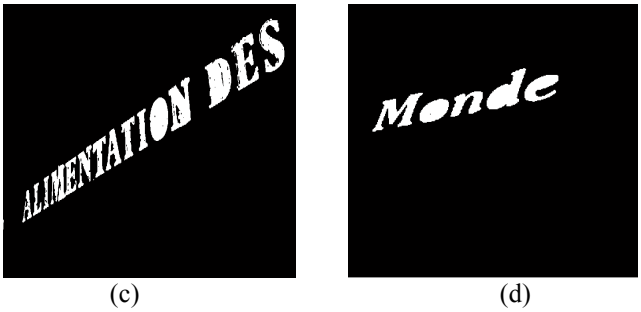



Fig. 5. (b): Performance evaluation of video frame (Fig. 4(b)) containing medium size text.

Fig. 6(a) and Fig. 6(b) show the input video frames of size 925 KB and 769 KB respectively containing large size text. Fig. 6(c) and Fig. 6(d) show the output video frames with detected text. Table V, Table VI, Fig. 7(a) and Fig. 7(b) show the performance of AI_TDT for video frames of Fig. 6(a) and Fig. 6(b) respectively. Results reveal that AI_TDT detects large size text accurately because eMSER preserves the shape of text characters. For input video frame of Fig. 6(a), AI_TDT increases accuracy by 7% and takes 1.99 seconds less processing time compared to MSER_TDT. For input video frame of Fig. 6(b), it increases accuracy by 4% and takes 0.03 seconds less time than MSER_TDT.


(a) (b)




(c) (d)

Fig. 6. (a) and (b) Input video frames with large size text. (c) and (d) Output video frames with identified text.

TABLE V. RESULTS FOR VIDEO FRAME SHOWN IN FIG. 6(a)

| Parameter | AI_TDT | MSER_TDT |
|---|---|---|
| Precision | 0.8 | 0.73 |
| Recall | 0.9 | 0.9 |
| Accuracy | 0.8 | 0.73 |
| APT (Sec) | 10.79 | 12.78 |

TABLE VI. RESULTS FOR VIDEO FRAME SHOWN IN FIG. 6(b)

| Parameter | AI_TDT | MSER_TDT |
|---|---|---|
| Precision | 0.8 | 0.5 |
| Recall | 0.9 | 0.66 |
| Accuracy | 0.8 | 0.4 |
| APT (Sec) | 8.99 | 9.02 |

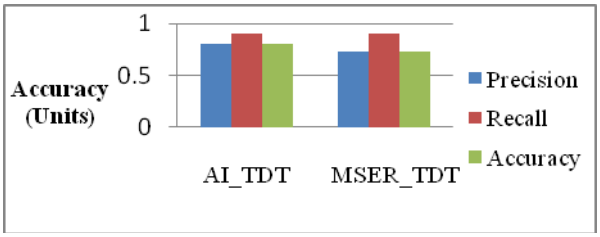



Fig. 7. (a): Performance evaluation for video frame (Fig. 6(a)) containing large size text.

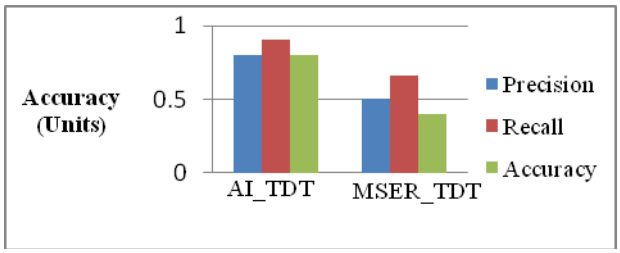



Fig. 7. (b): Performance evaluation for video frame (Fig. 6(b)) containing large size text.

### B. *Performance Evaluation Considering Video Frame with Blurriness*

In this experiment, we evaluate the performance of an improved text detection technique considering video frame with blurriness. Fig. 8(a) shows the input video frame of size 412 KB. Fig. 8(b) represents the output frame with detected text. Fig. 9 shows the performance of AI_TDT and MSER_TDT. Results reveal that AI_TDT increases accuracy by 3.5% and takes less APT compared to MSER_TDT.

## CONCLUSION

Text needs to be detected correctly for several applications because it is crucial and explicit source of important information. In this article, we have presented an improved scene text detection technique that detects text from videos


(a) (b)

Fig. 8. (a) Input video frame with blurriness (b) Output video frame with detected text.

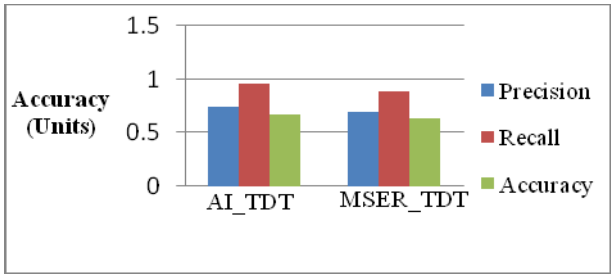



Fig. 9. Performance evaluation for video frame (Fig 8(a)) containing bluriness.

having blurriness and having text of different sizes. It increases text detection accuracy using distributed canny edge detector. Additionally, it incorporates edge preserving MSER that preserves the shape of characters and handles blurriness. Furthermore, it decreases average processing time required to detect text due to following two reasons: First, it partitions video frame into various fragments and process them concurrently. Second, MFCM clustering technique converges faster than the conventional FCM clustering method. Experimental results show that an improved text detection technique marginally increases accuracy and reduces average processing time compared to eMSER text detection technique. In future, our work can be extended considering videos containing text of different orientations and languages.

## REFERENCES


[1] G. Liang, P. Shivakumara, T. Lu and C. Tan, "Multi-Spectral Fusion Based Approach for Arbitrarily Oriented Scene Text Detection in Video Images," IEEE Trans. Image Processing, vol. 24, no. 11, pp. 4488-4501, Nov. 2015.

[2] P. Shivakumara, T.Q. Phan and C.L. Tan, "A Laplacian Approach to Multi-Oriented Text Detection in Video," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 33, no. 2, pp. 412-419, Feb, 2011.

[3] Chen, Huizhong, Sam S. Tsai, G. Schroth, David M. Chen, Radek Grzeszczuk and Bernd Girod, "Robust Text Detection in Natural Images with Edge-enhanced Maximally Stable Extremal Regions," Proc. IEEE, 18th International Conference on Image Processing (ICIP), pp. 2609-2612, 11-12 Sep. 2011.

[4] Z. Zuo, S. Tian, W. Pei and X. Yin, "Multi-strategy Tracking Based Text Detection in Scene Videos," Proc. IEEE, 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 66-70, 23-26 Aug. 2015.

[5] R. Minetto, N. Thome, M. Cord, N. Leite and J. Stolfi, "Snoopertrack: Text Detection and Tracking for Outdoor Videos," Proc. IEEE, 18th International Conference on Image Processing (ICIP), pp. 505-508, 11-14 Sep. 2011.

[6] A. Ayed, M. Halima and A. Alimi, "MapReduce Based Text Detection in Big Data Natural Scene Videos," Procedia Computer Science, vol. 53, pp. 216-223, Elsevier, 10 Aug. 2015.

[7] X. Huang, Q. Wang, L. Zhu and K. Liu, "Video Text Detection Based on Text Edge Map," Proc. IEEE, 3rd International Conference on Computer Science and Network Technology (ICCSNT), pp. 1003-1007, 12-13 Oct. 2013.

[8] S. A. Pote and M. A. Mehta, "An Imporved Technique to Detect Text from Scene Videos," Proc. IEEE, 2017 International Conference on Communication and Signal Processing (ICCSP), pp. , 6-8 Apr. 2017.

[9] R. C. Gonzalez, R. E. Woods, "Digital Image Processing," Pearson,1987.

[10] A. Thombare and S. Bagal, "A Distributed Canny Edge Detector: Comparative Approach," Proc. IEEE, International Conference on Information Processing (ICIP), pp. 312-316, 16-19 Dec. 2015.

[11] M. Shasidhar, V. Raja and B. Kumar, "MRI Brain Image Segmentation Using Modified Fuzzy C-Means Clustering Algorithm," Proc. IEEE, International Conference on Communication Systems and Network Technologies (CSNT), pp. 473-478, 3-5 Jun. 2011.