Pattern Recognition Letters 129 (2020) 92–100

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Delaunay triangulation based text detection from multi-view images of natural scene

Soumyadip Roy[a], Palaiahnakote Shivakumara[b], Umapada Pal[c], Tong Lu[d,*], Govindaraj Hemantha Kumar[e]

[a] Computer Science And Engineering, Heritage Institute Of Technology, Kolkata, India
[b] Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lampur, Malaysia
[c] Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India
[d] National Key Lab of Novel Software Technology, Nanjing University, Nanjing, China
[e] Department of Studies in Computer Science, University of Mysore, Karnataka, India

ARTICLE INFO

Article history:
Received 29 March 2019
Revised 30 October 2019
Accepted 14 November 2019
Available online 15 November 2019

ABSTRACT

Text detection in the wild is still considered as a challenging issue to the researchers because of its several real time applications like forensic application, where CCTV camera captures images at different angles of the same scene. Unlike the existing methods that consider a single view captured orthogonally for text detection, this paper considers multi-view (view-1 and view-2 of the same spot) of the same scene captured at different angles or different height distances for text detection. For each pair of the same scene, the proposed method extracts features that describe characteristics of text components based on Delaunay Triangulation (DT), namely corner points, area and cavity of the DT. The features of corresponding DT in view-1 and view-2 are compared through cosine distance measure to estimate the similarity between two components of respective view-1 and view-2. If the pair satisfies the similarity condition, the components are considered as Candidate Text Components (CTC). In other words, these are the common components for view-1 and view-2 that satisfy the similarity condition. From each CTC of view-1 and view-2, the proposed method finds nearest neighbor components to restore the components of the same text line based on estimating degree of similarly between CTC and neighbor components using Chi-square and cosine distance measures. Furthermore, the proposed method uses a recognition step to detect correct texts by comparing recognition results of view-1 and view-2. The same recognition step is used for removing false positives to improve the performance of the proposed method. Experimental results on our own dataset, which contains pair of images of different situations, and the standard datasets, namely, ICDAR 2013, MSRATD-500, CTW1500, Total-text, ICDAR 2017 MLT and COCO-text, show that the proposed method outperforms the existing methods.

1. Introduction

The scope of text detection and recognition in video and natural scene images is expanding to new applications and fields. One such new application area is forensic, where text detection and recognition can be used to identify locations of crimes based on text information appearing in images, which includes shop names, addresses of buildings, texts on shirt, etc. Unlike most of the previous applications which consider images captured orthogonally, a forensic application needs to consider images captured at different angles with different height distances by multiple CCTV cameras. As a result, one can expect multi-views of the same spot affected by multiple adverse factors, such as low resolution, perspective distortion, low contrast, arbitrary orientation, font variations, complex background, etc. For example, images captured from two angles with different height distances can be seen in Fig. 1(a), where we can see color, background, font, size variations of texts. For such images, the proposed method gives good results as shown in Fig. 1(b), where all the three text lines are detected. Therefore, text detection and recognition in multi-views captured from different angles is not as easy as text detection in images captured orthogonally. It is evident from the results shown in Fig. 2(a) and (b), where the baseline methods of text detection in scene images are tested. It is noted from Fig. 2(a) and (b) that the methods do not perform well as they miss texts. The baseline methods [1,2] have explored powerful deep learning models for text detection in natural

* Corresponding author.
E-mail addresses: shiva@um.edu.my (P. Shivakumara), umapada@isi.cal.ac.in (U. Pal), lutong@nju.edu.cn (T. Lu).

https://doi.org/10.1016/j.patrec.2019.11.021
0167-8655/

(a) View-1 (left) and View-2 (right) of the same scene as input

(b) The results of the proposed method using view-1 and view-2.

Fig. 1. Example of text detection from multi-view. (a) View-1 (left) and View-2 (right) of the same scene as input (b) The results of the proposed method using view-1 and view-2.

(a) Text detection results of the existing EAST method [1] for view-1 and view-2

(b) Text detection results of existing method (CTD) [2] for view-1and view-2

Fig. 2. Sample text detection results of the baseline methods for multi-views (a) Text detection results of the existing EAST method [1] for view-1 and view-2 (b) Text detection results of existing method (CTD) [2] for view-1and view-2.

scene images. This shows that the existing methods are not effective for images captured by multiple cameras at different angles. This work considers two views, namely, view-1 and view-2 of the same spot/scene (as shown in the sample images in Fig. 1(a)), as the input for text detection.

2. Related work

Many methods are proposed recently in literature for addressing different challenges in text detection, such as complex background, and variations on script, multi-orientation, low resolution, low contrast in natural scene images, etc. Zhou et al. [1] proposed an efficient and accurate scene text detector. The method focuses on arbitrarily oriented text detection based on deep learning models in natural scene images. Shi et al. [3] proposed detecting oriented texts in natural scene images by linking segments. The method focuses on fixing accurate bounding boxes for text detection irrespective of orientations in natural scene images based on deep learning models. Liu et al. [2] proposed detecting curve texts in the wild. The method explores deep learning models for detecting curved text detection in natural scene images. Liao et al. [4] proposed rotation-sensitive regression for oriented scene text detection. The method explores deep learning models for oriented text detection in natural scene images. He et al. [5] proposed multi-oriented and multi-lingual scene text detection with direct regression. This method focuses on both multi-oriented and multi-lingual text detection in natural scene images. Liao et al. [6] discussed a single shot oriented scene text detector. The method focuses on multi-font and arbitrarily oriented text detection in natural scene images. Deng et al. [7] developed a method to detect multi-oriented texts with corner-based-region proposals. The method alleviates the problems of fixed sliding window issues for text detection in natural scene images. NguyenVan et al. [8] applied pooling based scene text proposal technique for scene text reading in the wild. The method focuses on accurate text detection in natural scene images by combining deep learning and histogram oriented gradient features. Gao et al. [9] proposed reading scene texts with fully convolutional sequences modeling. The method explores the bidirectional long short term memory for text detection in natural scene images. Xue et al. [10] used a multiscale shaped regression network, which is capable of detecting arbitrary orientation texts accurately. Bartz et al. [11,12] proposed neural network based method for text detection and recognition. The method aims at proposing single architecture by exploring spatial transformer network for text detection in natural scene images. Shi et al. [13] proposed an attention scene text recognizer with flexible rectification. The method proposes rectification network to transform from input image to new image and then it uses recognition network. Since the methods [11–13] focus on both text detection and recognition, the performance of the method is not consistent for text detection of different datasets.

It is noted from the above existing methods on text detection that none of the methods addresses the challenges of text detection from multi-views of the same scene captured by different CCTV cameras at different angles with different height distances. This is justifiable because most of the methods aim at applications of understanding texts in different scene images (indexing and retrieval) but not forensic applications, where texts can assist forensic investigation. Therefore, text detection form multi-view is still considered as an open issue in this field and this is the first attempt on this issue.

Inspired by the method [14] where Delaunay Triangulation (DT) has been explored for addressing the challenges in determining position and alignment of small text line regions, we explore the same DT for addressing challenges of text detection in multi-view of scenes. This is understandable because DT helps us to extract internal and external features, such as stroke pixels of character components and the spatial relationship between character components. In addition, the similarity between DTs of character components estimated by Chi-square and cosine distance measures in two views is able to extract text components that share common properties. Furthermore, the proposed method uses OCR based on a classifier to eliminate false positives such that text detection performance is enhanced. One of the key contributions of the proposed method to address such complex problem is to explore DT concept for text detection in multi-view images. The way, the proposed work extract the invariant features using DT for detecting candidate text components is new. The way the proposed method uses cosine and chi square distance measures for restoring missing text components and fixing the bounding box for any shaped text line is another novel contribution of the proposed work.

3. The proposed methodology

In this work, we consider view-1 (which is captured at one angle) and view-2 (which is captured at a different angle with a different height variations) of the same scene as the input for the text detection purpose. For each pair of images, the proposed method obtains Canny edge images. The motivation for choosing

Canny edge image is that Canny edge operator is better than other edge operators such as Sobel and Prewitt operators because Canny provides fine details of edges for low contrast and low resolution images, while the other operators are good for only high contrast images. In addition, edge information facilitates subsequent steps with fewer computations compared to pixel level as it gives fine edges for contents in images. To explore Delaunay Triangulation (DT), the proposed method uses Harris corner detection as it is a well-known method for different types of images. We explore DT for extracting corners, triangle areas which are defined in Eq. (1), growing areas which are the regions given by the ring growing step inside triangle areas, and numbers of cavities which are created by growing areas.

It is true that since view-1 and view-2 capture the same scene, both the views must have common text components irrespective of locations. Those text components share the same text properties. Motivated by the method in [15], where text components in different edge images of the same image exhibit common structures and properties, while the components in background do not share the same properties, we explore the same idea for extracting triangles that share the same features, which results in Candidate Text Components (CTC). The features used to obtain CTC are triangle areas, growing areas and the number of cavities.

$$Triangle\ Area = \frac{|a1(b2-c2)+b1(c2-a2)+c1(a2-b2)|}{2} \quad (1)$$

Where (a1,a2),(b1,b2),(c1,c2) are the co-ordinates of the corners of a triangle.

The proposed method chooses CTC as the seed components for respective view-1 and view-2 by comparing the similarity between Histogram of Oriented Gradients (HOG) of the CTCs in view-1and view-2. For estimating the degree of similarity between the two histograms of the CTCs, the proposed method uses Chi-square distance measure. For each seed component of respective view-1 and view-2, the proposed method finds the nearest neighbor CTC, and then checks the similarity between the seed components and the nearest neighbor component as well as the similarity between the nearest neighbor components of view-1 and view-2. Similarly, to strengthen the above similarity estimation, the proposed method estimates the degree of similarity between vectors containing the mean of triangle areas, growing areas and the number of cavities using cosine distance measure. If components give similar degree of similarity using Chi-square and cosine distance measures with a certain threshold, the components are considered as text components.

The reason to use cosine and chi square distance measures is that the cosine distance measure is meant for estimating similarity between two vectors containing numeral values while chi square distance measure is to find similarity between two histograms. In other words, the cosine distance measure helps us to check how two vectors are similar in terms of direction while the chi square distance measure helps us to check how the values are distributed. In our case, since characters are aligned in particular direction according to text line orientation, it is expected feature vectors of CTC should share the same directions. In the same way, the histogram obtained by HOG should exhibits symmetry because of the symmetrical positive and negative gradient angles with respect edges of the CTC. This process results in text detection with some false positives. Therefore, the false positives are further removed based on above features by applying at text line level, using aspect ratio of components and OCR results. The block diagram of the proposed method is shown in Fig. 3.

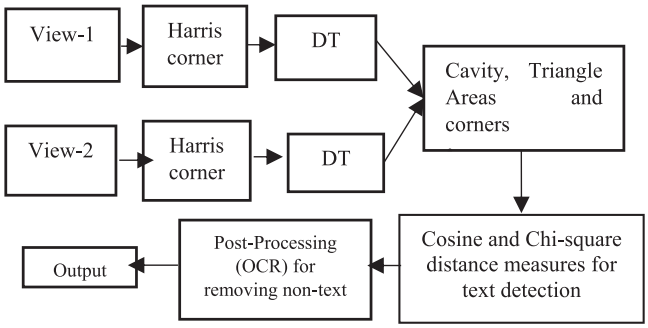


Fig. 3. Block diagram of the proposed method.

3.1. Candidate text component detection

For view-1 and view-2 shown in Fig. 4(a), where it is noted that both the views contain the same text at different locations, the proposed method obtains Canny edge images as shown in Fig. 4(b). For the Canny edge image in Fig. 4(b), the proposed method detects corners using Harris corner method as shown in Fig. 4(c). The proposed method constructs Delaunay Triangulation (DT) for the image using corners as shown in Fig. 4(d), where one can see DT patterns for the texts in view-1 and view-2 appear almost the same, while DT patterns for other background components appear differently. This observation motivated us to extract the feature that represents character components, namely, triangle area, growing area and the number of cavities. For growing area, the proposed method finds the centroid for each triangle, which draws ring around the centroid with one pixel step-size. This ring grows and the ring growing process is continued until it reaches any one of the side of the triangle, which is called growing area as shown in red color in Fig. 4(e). If the triangle is equilateral, one can expect that such growing touches all the three sides of the triangle. This results in thee cavities at corners of a triangle. The number of cavities depends on triangle shape. Therefore, we can conclude that triangle area, growing area and the number of cavities are the features of DT which represent the shape of character components.

Feature extraction is illustrated in Fig. 5, where for character *A* chosen from view-1 and view-2 shown in Fig. 5(a), the corners are detected as shown in Fig. 5(a) (right side). The results of ring growing using corners are shown in Fig. 5(b), where we can see triangle area, growing area, and number of cavities are almost the same. Therefore, we consider these three features as the feature vector of DT. The proposed method compares the feature vector of triangles in view-1 with that of triangles in view-2 by using cosine distance measure as Eq. (2). The triangles that satisfy the high degree of similarity with a certain threshold are considered as Candidate Text Components as shown in Fig. 5(c).

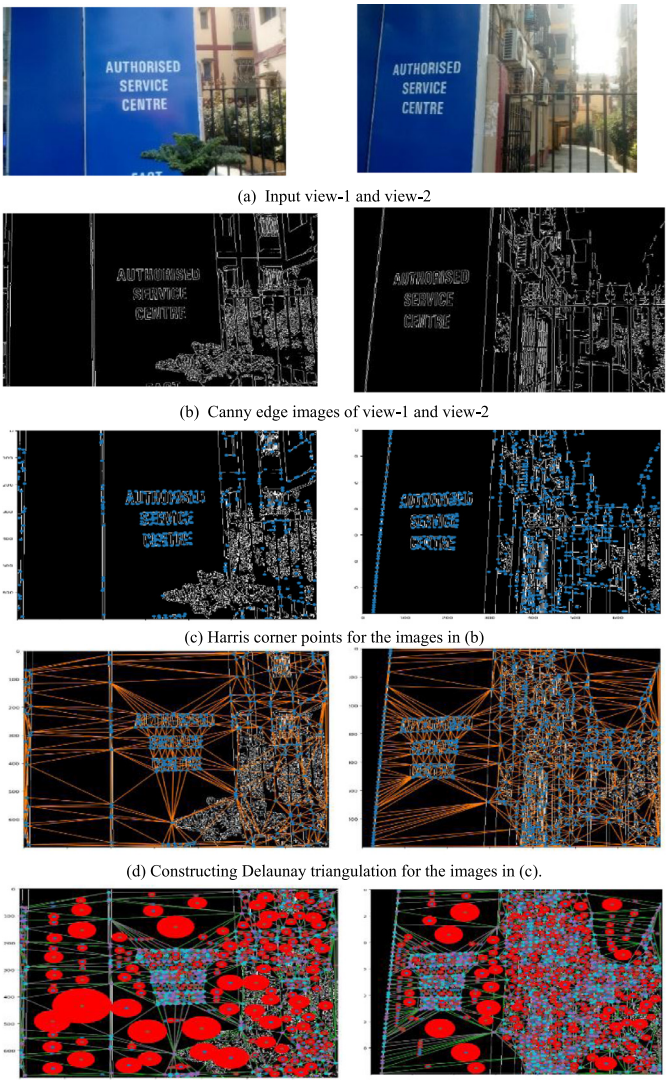
$$Cosine\ Similarity(u,v) = 1 - u.\frac{v}{|u|*|v|} \quad (2)$$

where u and v are two vectors of two triangles.

Since background is often complex, the above feature vector alone is not adequate to remove background information as we can see non-text component in Fig. 5(d). Therefore, we propose a new step called text restoration by removing non-text components, which will be discussed in the subsequent section.

3.2. Text detection from multi-view

The proposed method considers each Candidate Text Components (CTC) as the seed component for respective view-1 and view-2. For each seed component, the method extracts the above-mentioned three features. Next, it calculates the mean of respective features of DT, which results in Mean Feature Vector (MFV).



(a) Input view-1 and view-2

(b) Canny edge images of view-1 and view-2

(c) Harris corner points for the images in (b)

(d) Constructing Delaunay triangulation for the images in (c).

(e) Ring growing for the triangles to extract text properties for the results in (d).

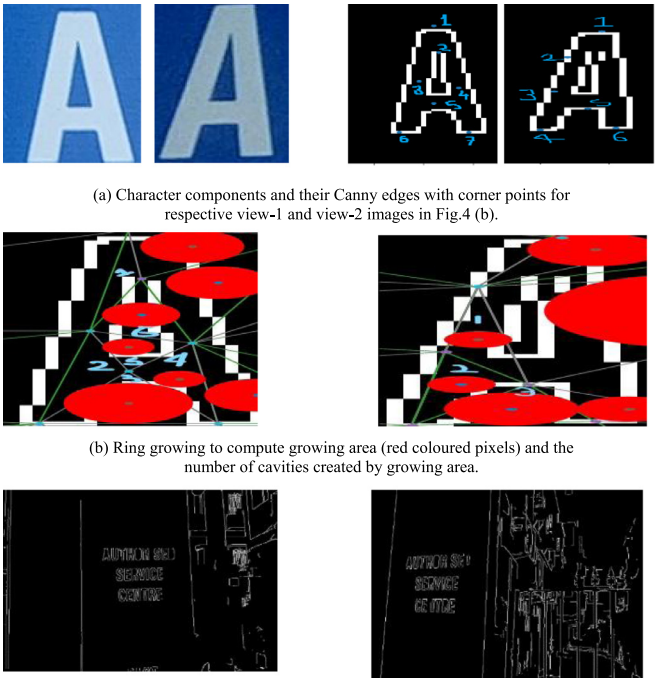
Fig. 4. Extracting properties of Delaunay triangles for differentiating text and non-text in the images. (a) Input view-1 and view-2 (b) Canny edge images of view-1 and view-2 (c) Harris corner points for the images in (b) (d) Constructing Delaunay triangulation for the images in (c). (e) Ring growing for the triangles to extract text properties for the results in (d).

To increase the strength of MFV, the proposed method obtains Histogram of Oriented Gradients (HOG) with 8 bins for each seed component in respective view-1 and view-2. The number of bins is determined empirically. Since HOG provides histograms, we propose to use Chi-square distance measures as defined in Eq. (3) for estimating the similarity between histograms. However, for MFV, we use the same cosine distance measure. The proposed method uses these two distance measures for estimating similarity between two components. If components satisfy the similarity with a certain threshold, they are considered as text ones. Threshold is determined through experiments.

$$Chi2 = \frac{\sum (O[i] - E[i]) * (O[i] - E[i])}{E[i]} 1 \leq i \leq 8 \quad (3)$$

where O denotes histogram of orientations, and E is the orientation histogram normalized within 0–1.

Initially, the proposed method finds the common seed component that satisfies the above similarity condition in view-1 and view-2. It then finds the nearest neighbor CTC for the common



(a) Character components and their Canny edges with corner points for respective view-1 and view-2 images in Fig.4 (b).

(b) Ring growing to compute growing area (red coloured pixels) and the number of cavities created by growing area.

(c) Candidate components of the view-1 and view-2 images

Fig. 5. Common components detection for the view −1 and view-2. (a) Character components and their Canny edges with corner points for respective view-1 and view-2 images in Fig.4(b). (b) Ring growing to compute growing area (red coloured pixels) and the number of cavities created by growing area.

seed component in respective view-1 and view-2. Next, the proposed method estimates the similarity distance between the seed component and the nearest neighbor component in respective view-1 and view-2. In addition, it also estimates the similarity between the nearest neighbor component in view-1 and the nearest neighbor component in view-2. If the nearest neighbor of view-1 and view-2 and the seed components of view-1 and view-2 satisfy the similarity criteria, the process merges the seed component and the nearest neighbor component as one single component. This process is continued in both view-1 and view-2 until it fails to satisfy the similarity criteria. If the similarity criterion fails in one view and satisfies in the other view, the process does not stop and continues until the process terminates in both the views. Then the above procedure is repeated for all the seed components in view-1 and view-2. In this way, the proposed method eliminates non-text components, at the same time, it restores missing characters. In other words, if any one of the view contains characters and other view does not, the proposed method also works well. This is the advantage of the proposed method and hence it is different from the exiting methods. The effect of the above step can be seen in Fig. 6(a), where we can see almost all the non-text components are removed without affecting text components.

However, it is noted from Fig. 6(a) that the results still contain non-text components. Furthermore, the method uses OCR presented in [16] for obtaining recognition results. Such et al. [16] explores deep Convolutional Neural Networks (CNN) for recognizing handwriting symbols, thus we explore the same for character recognition. Based on recognition results, we remove non-text components as shown in Fig. 6(b), where view-1 and view-2 contain only text components. The proposed method chooses the text line that gives the minimum edit distance as a correct text among the text lines in view-1 and view-2. The reason to choose CNN based OCR for eliminating false positives is that we do not expect the OCR to recognize texts correctly because false positive elimination requires to identify the presence of text. Therefore, we be-


(a) Effect of similarity criteria using Chi-square and cosine distance.


(b) Text components extraction after removing false positives

(c) Text detection based on recognition results

Fig. 6. Text detection from multi-view. (a) Effect of similarity criteria using Chi-square and cosine distance. (b) Text components extraction after removing false positives (a) Effect of similarity criteria using Chi-square and cosine distance. measures (c) Text detection based on recognition results.

lieve that the OCR is enough to identify the presence of the text in the image irrespective of situations. For fixing bounding boxes for text lines of any orientation, the proposed method uses the above nearest neighbor finding and merging along with directions of text lines. Since the proposed method involves the nearest neighbor and direction of text lines, the method can fix bounding boxes for arbitrarily-oriented text lines in the image. The algorithmic steps are shown in Algorithm.

Algorithm: Text Detection from Multi-Views

1: View-1 (V1) and View-2 (V2) are the input images.
2: Obtain Canny edge images for V1 and V2.
3: Obtain the Harris corner points for the V1 and V2.
4: Get Delaunay Triangulation (DT) for V1 and V2 using step-3.
5. Find the centroid of the triangle.
$centroid = (\frac{x_1+x_2+x_3}{3}, \frac{y_1+y_2+y_3}{3})$ where $(x1, y1)$, $(x2, y2)$ and $(x3, y3)$ are corners of the triangles.
6. From centroid grow a circle where $area\ of\ circle = \prod r^2\ where\ 1 \leq r \leq p$ and $p = \min(d_1, d_2, d_3)$ where d_1, d_2 and d_3 are respectively the perpendicular distance of the 3 sides of the triangle from the centroid.
7: Extract features, namely, Corners (c), Triangle Area (TA) and the Number Cavities (NC) from each DT in both the V1 and V2. This outputs Feature Vectors, FV1, FV2 for respective V1 and V2.
8: For each DT in V1 and V2,
a. Estimate similarity between FV1 and FV2 using cosine distance measure.
b. If the DT satisfies the similarity condition with certain threshold, the DT is considered as Candidate Text Components (CTC)
For end.
9. *For* $\forall\ x \in CTC$from entire Canny Component of x.
10: For each CTC in V1 and V2,
a. Find nearest neighbor for CTC in V1, say N1 and nearest neighbor for the CTC in V2, say N2.
b. Compute mean of the FV for CTC of V1 and CTC of V2, N1 and N2, which outputs Mean Feature Vectors, MFV1 and MFV2 for respective V1 and V2.
c. Estimate the similarity between CTC of V1 & N1, CTC of V2 & N2 and N1 and N2 using Cosine Distance (CD) measure.
d. Perform HOG operation for CTC in V1 and CTC in V2, which gives histograms.

(*continued on next page*)

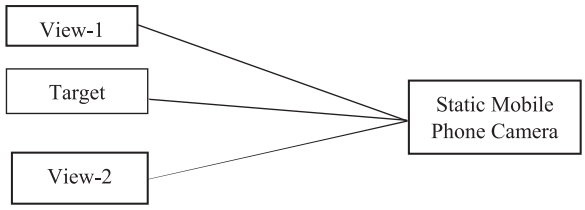


Fig.7. Illustrating the process of capturing view-1 and view-2 images of the same spot.

e. Estimate the similarity between HOG histograms for the same as step c using Chi-square Distance (ChD) measure.
f. If CD and ChD satisfy the similarity condition with certain threshold, CTC is considered as text component and it merges nearest neighbor components with CTC components. If above is satisfied then goto step g else goto step h.
g. The newly found components N1 and N2 are used for finding nearest neighbours and processes a-f are repeated again.
h. The process continues until the similarity condition fails in both V1 and V2 or when all the CTC are explored. This step gives text detection with bounding box.
For end.
Step-11: Apply recognition step using CNN for eliminating false positives and choosing the correct text from V1 and V2.
Algorithm ends.

4. Experimental results

For evaluating the proposed text detection from multi-view, since there are no standard datasets available, we create our own dataset by capturing different scenes at different angles with different height distances. As distance varies, the quality of image changes. This is the main challenge of our dataset for text detection, unlike the existing standard datasets where most of the images are captured at orthogonal angles but not at different height distances. In this work, we consider two views of the same scene, namely, view-1 and view-2 as the input for text detection. We use our own mobile phone camera with the configuration of LENOVO A7000 at a resolution of 1280×720 for capturing view-1 and view-2 as shown in Fig. 7 where we can see view-1 and view-2 captured at different angles using fixed position. In this way, we create our own dataset by varying angles and height distance for different spots. Note that most of the multi-views images are captured in day with sun light. Since images cover natural scenes of environments especially streets, markets, shopping malls and building views, view images have complex background. Therefore, every pair of images poses many challenges. In total, our dataset includes 500 pair of images of English and Bengali scripts.

To test the objectiveness of the proposed method, we also consider the benchmark datasets, namely, ICDAR 2013, MSRATD-500, CTW1500, Total-text, ICDAR 2017 MLT and COCO-text. **ICDAR 2013 [4]:** This dataset provides 229 training images and 233 test images for experimentation. Most texts are in horizontal directions. This is a simple dataset for text detection compared to the other ones. **MSRA-TD500 [5]:** This dataset provides 300 training images and 200 test images for evaluation. This dataset includes multi-oriented texts of English and Chinese languages. **CTW1500 [2]**: This dataset provides 1000 images for training and 500 images for testing. This is created basically for evaluating curved text detection in scene images. Every image in this dataset contains at least one curved text line. **Total-Text [17]**: This dataset provides 1255 images for training and 300 for testing. This is also the same as CTW1500 dataset with more variations, which includes low resolution, low contrast and complex backgrounds. **ICDAR 2017 MLT [5]**: This dataset provides multi-lingual text images, which includes 7200 training images, 1800 validation images and 9000 testing im-

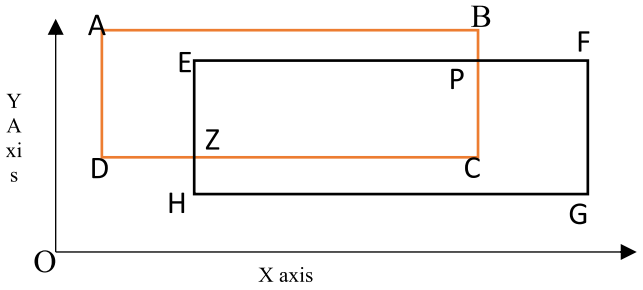


Fig.8. Determining the matching area between ground truth area and text area detected by the method.

ages. The dataset consists of images of 9 languages in arbitrary orientations. This dataset is good for evaluating the ability of the methods on multi-lingual text detection in natural scene images. **COCO-Text [4]**: This dataset is created not with the intention of text detection, hence texts in images are more realistic and there are a large number of variations compared to all the other datasets. As a result, this dataset is much more complex for text detection. It provides 43,686 images for training, 20,000 for validation and 900 for testing.

For evaluating the proposed and existing methods using the standard datasets, we follow the standard instructions and evaluation scheme mentioned in respective datasets. However, for our dataset, since there is no ground truth, we manually count the measures. The measures used for evaluating the proposed and existing methods are Recall (r), Precision (p) and F-measure (f) for all the experiments in this work as defined in Eqs. (4) and (5).According to Eqs. (4) and (5), the evaluation scheme finds the best match between the area of ground truth and the text detected by the methods as illustrated in Fig. 8, where ABCD represent bonding box of the ground truth text and EFGH denote the bounding box of the detected text by the methods. This results in common region and it is considered as the best match, which is denoted by EPCZ in Fig. 8. If the best match is larger than 0.5 (50%) then the detected text is considered as correct count (true positive) else it is considered as false negative.

$$\boldsymbol{p} = \sum_{r_e \in E} m(r_e, T)/|E| \text{ and } r = \sum_{r_t \in T} m(r_t, E)/|T| \quad (4)$$

where $m(r, R)$ is the best match for a block r in a set of block. E and T are our estimated block and the ground truth block, respectively. The f measure is defined using recall, precision as

$$\boldsymbol{f} = \frac{1}{\frac{\boldsymbol{a}}{\boldsymbol{p}} + \frac{\boldsymbol{a}}{\boldsymbol{r}}} \quad (5)$$

where a is 0.5 for counting true positives.

In order to show effectiveness of the proposed method, we run the benchmark methods for comparative study on our dataset. That are Shi et al. [3] proposed detecting oriented texts in natural scene images by linking segments, Zhou et al. [1] proposed an efficient and accurate scene text detector, and Liu et al. [2] proposed detecting curve texts in the wild. Liao et al. [6] proposed single shot oriented scene text detector. Bartz et al. [11,12] proposed a single neural network for text detection and recognition in natural scene images. Shi et al. [13] proposed an attentional scene text recognizer for text detection in natural scene images. The reason to consider the above methods for comparative study is that they explore deep learning models and address almost all the challenges in text detection including orientation or script variations, low contrast, low resolution, complex background, etc. However, the methods are not tested on the dataset created by capturing images from different angles. Note that for running the codes, we use predefined deep learning models for all the experiments.

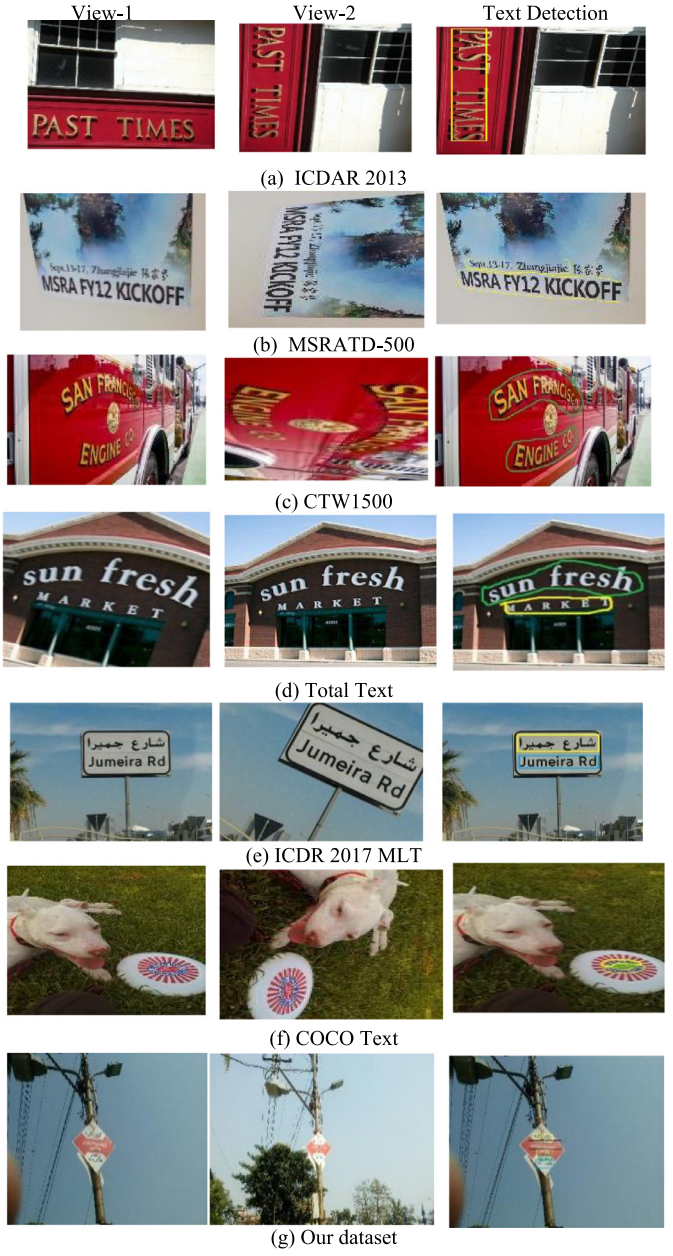


Fig. 9. Qualitative results of proposed method for text detection on our and benchmark datasets. (a) ICDAR 2013 (b) MSRATD-500 (c) CTW1500 (d) Total Text (e) ICDR 2017 MLT (f) COCO Text (g) Our dataset.

4.1. Experiments for text detection

Since the proposed method requires view-1 and view-2 images for experimentation, for conducting experiments on all the standard datasets, we rotate input images randomly to create view-2 for each input image. Sample results of the proposed method for ICDAR 2013, MSRATD-500, CTW1500, Total-Text, ICDAR 2017 MLT, COCO-Text and our dataset are shown in Fig. 9, respectively, where view-1 is an actual input image, while view-2 is created by rotating randomly. When an image is rotated as shown in view-2 column, text location displaced from the actual location in view-1 and also gets affected by distortion compared to the text in view-1. This is the main challenge and makes differences compared to other datasets that are available publicly. Fig. 9 shows that the proposed method detects texts well for all the datasets and hence it is effective and useful.

Table 1
Performance of the proposed and existing methods on our dataset.

Methods	Precision	Recall	F-Measure	APT-Tr (S)	APT-Te (S)
EAST[1]	73	61.5	66.75	3	4
Seglink[3]	65	63.2	64.08	4	6
CTD[2]	70	65.4	67.62	4	5
TextBoxes++[6]	75	70	72.41	8	6
SEE[12]	63	65	64	8	4
STN–OCR[11]	60	62	61	5	5
ASTER[13]	65	60	62.4	4	4
Proposed Method + Sobel edges	78	82	80	–	4
Proposed Method + Canny edges	**86.0**	**83.0**	**84.4**	–	5

Table 2
Performance of the proposed and existing methods on ICDAR 2013 dataset.

Methods	Precision	Recall	F-Measure
CTPN[10]	93	83	88
Zhang et al. [10]	88	78	83
He et al. [10]	92	81	86
SegLink[10]	87.7	83	85.3
SSTD[10]	89	86	88
Lyu et Al[10]	92	84.4	88
RRD [10]	92	86	89
Proposed method	90.4	**88**	**89.1**

The proposed method uses Canny edge image for detecting corners and extracting features from DT. As stated in the proposed methodology, Canny edge is better for the proposed work compared to other edges, we calculate measures using Sobel edge images on our dataset as reported in Table 1. From the results of the proposed method with Canny and with Sobel show that the proposed method with Canny is better than the proposed method with Sobel. Therefore, we prefer to choose the Canny edge in this work. Quantitative results of the proposed and existing methods are reported in Table 1, where it is noted that the proposed method is better than existing methods in terms of Precision, Recall and F-measure. The main reason for the poor results by the existing methods is that they are developed for images captured orthogonally but not at different angles.

Since text detection methods involve lot of computations during Training (Tr) and Testing (Te), analyzing processing time is important. For experimentation, we use HP laptop with configuration of Ram- 8 GB Graphics card – Nvidia Geforce 940 m, Tensorflow 1.3 and ubuntu 18.01. Since our main objective is to detect text in multi-views, our primary focus is solving the problems. Therefore, according to Table 1, the Average Processing Time (APT) in seconds for testing is almost same as the existing methods. This shows that the proposed method does not require high processing time for text detection in multi-view. The advantage of the proposed method is that the method does not involve large number of images for training because the proposed method use features and rules for text detection and predefined CNN for false positive elimination as post processing. However, for determining the parameters and conditions, we use 500 pre-defined sample chosen randomly from across the datasets, which consumes negligible processing time per image. As a result, APT of the proposed method for training is not reported in Table 1.

Quantitative results of the proposed and existing methods for ICDAR 2013, MSRATD-500, CTW1500, Total-Text, ICDAR 2017 MLT and COCO-Text are reported in Table 2, 3, 4, 5, 6 and 7, respectively. It is observed from Tables 2 –7 that the proposed method scores almost consistent results for all the datasets including our datasets, while the existing methods score the lowest for COCO, Total-Text datasets, ICDAR 2017 MLT and our dataset. This shows that though existing methods are capable of handling curved, ori-

Table 3
Performance of the proposed and existing methods on MSRATD-500 dataset.

Methods	Precision	Recall	F-Measure
Kang et al. [10]	71.0	62.0	66.0
Zhang et al. [10]	83	67	74
He et al. [10]	77.0	70.0	74.0
EAST [10]	87.3	67.4	76.0
SegLink[10]	86.0	70.0	77.0
Wu et al. [10]	77.0	78.0	77.0
PixelLink [10]	83.0	73.2	77.8
TextSnake [10]	83.2	73.9	78.3
RRD [10]	87.0	73.0	79.0
Proposed method	**88.0**	**78.0**	**82.6**

Table 4
Performance of the proposed and existing methods on CTW1500 dataset.

Methods	Precision	Recall	F-Measure
EAST [10]	78.7	49.1	60.4
SegLink [10]	42.3	40	40.8
DMPNet[10]	69.9	56	62.2
CTD [10]	74.3	65.2	69.5
TextSnake [10]	67.9	**85.3**	75.6
Proposed method	**85**	82	**83.47**

Table 5
Performance of the proposed and existing methods on Total text dataset.

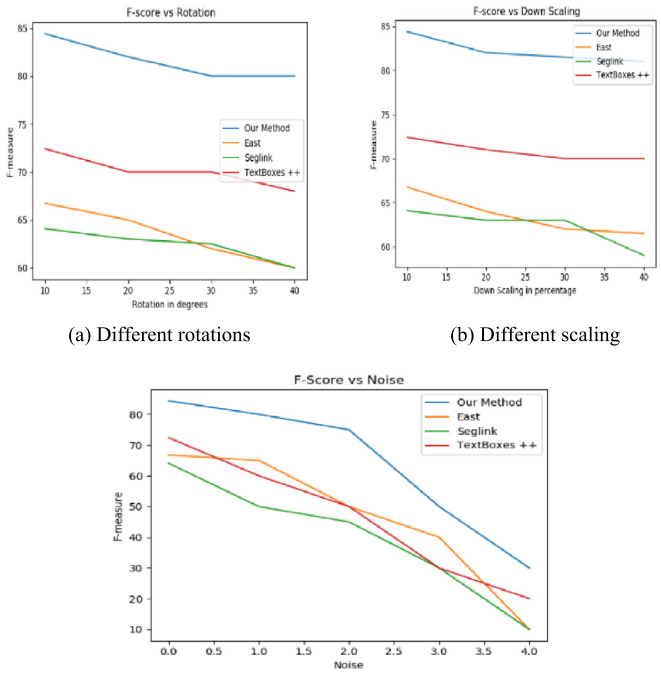
Methods	Precision	Recall	F-Measure
EAST [10]	50.0	36.2	42.0
SegLink[10]	30.3	23.8	26.7
TextSnake [10]	82.7	74.5	78.4
Proposed method	**88.0**	**79.0**	**83.25**

Table 6
Performance of the proposed and existing methods on ICDAR 2017 MLT dataset.

Methods	Precision	Recall	F-Measure
linkage-ER-Flow[5]	44.48	25.59	32.49
TH-DL [5]	67.75	34.78	45.97
SARI FDU RRPN v1 [5]	71.17	55.50	62.37
SCUT_ DLVClab1 [5]	80.28	54.54	64.96
Sensetime_OCR[5]	56.93	69.43	62.56
IDST_CV[5]	31.81	26.02	28.63
Proposed method	**83**	**72**	**77.10**

Table 7
Performance of the proposed and existing methods on Total COCO text dataset.

Methods	Precision	Recall	F-Measure
SCUT_DLVClab[7]	31.6	62.5	42
SARI_FDU_ RRPN[7]	33.3	63.2	43.6
UM[7]	47.5	**65.4**	55.1
Text_Detection_DL[7]	**60.9**	61.8	61.3
Proposed method	60.0	65.0	**62.4**



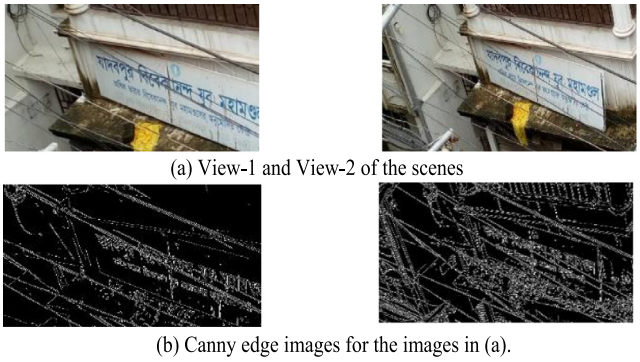
(a) Different rotations (b) Different scaling

(c) Different Gaussian noise

Fig.10. Validating the robustness of the proposed and existing methods for different rotations, scaling and Gaussian noise. (a) F-measure of the proposed and existing methods for different rotations, (b) different scaling and (c) different levels of Gaussian noise.

entations and multi-lingual challenges, still they have inherent limitations of defining the number of labeled samples for tackling large variations of texts. This is justifiable because it is hard to fix the boundary for variations of texts in images as the variations depend on applications and requirements. In other words, as application changes, the requirement changes and hence variations in texts change as the proposed work. However, the proposed method involves the combination of feature extraction, which extracts shapes of text components, thus it works well for all the datasets including our new dataset. One can understand the results of COCO, Total-Text, ICDAR 2017 MLT and our dataset that there are still many challenges and hence there is a scope for improving the results further. For example, in this work, we consider only two views, but one can consider more than two views, such as left, right, front and back views of the same scene. Multi-views can be expected when we capture images from different height distances and different angles. In this situation, each view can pose different challenges. This makes the problem more complex and challenging.

As mentioned earlier the extracted features are invariant to rotation, scaling and noise, we calculate F-measure for text detection in the images of different rotations, scaling and different Gaussian noise levels of our dataset for the proposed and existing methods. The results are illustrated in Fig. 10(a)–(c) for different rotations, scaling and Gaussian noise at different levels, respectively. It is noted that the methods including the proposed method score almost constant F-measure for different rotations, scaling and to some extent to noise levels. Therefore, the methods including proposed method are invariant to rotation, scaling and robust to some extent to noise. However, interestingly, it is also noted from Fig. 10 and the result in Table 1 that the F-measure of the proposed method is almost same while the existing methods do not. Hence, the proposed method is significantly reliable and stable for different situations compared to the existing methods. It is observed from Fig. 10(c) that the F-measure scored by the proposed method is almost constant up to noise level 2.0 (parameter of sigma) while existing methods do not. This indicates that the proposed method can withstand images with Gaussian noise up to 2.0 level.



(a) View-1 and View-2 of the scenes

(b) Canny edge images for the images in (a).

Fig. 11. Limitation of the proposed method. (a) View-1 and View-2 of the scenes (b) Canny edge images for the images in (a).

In the proposed method, the steps of extracting features like cavity, area of the ring growing and comparing nearest neighbor components to estimate the degree of the similarity are computationally expense compared to other steps. These steps require two loops to process the whole image, therefore, the time complexity of the method is $O(n^2)$ for the worst case. Since the proposed method does not involves a large number of images of training, it does not require noticeable time for training. Sometimes, the proposed method may not work well for the images shown in Fig.11, where it can be seen that texts in Canny edge images are not visible for our eyes due to complex background, height distance variations and limitation of the Canny edge operator. Therefore, there is a scope for the improvement of the proposed method.

5. Conclusion and future work

We have proposed a new method for text detection from multi-view of scenes. Unlike the existing work where the images are captured by orthogonal direction, in this work, the images are captured by different angles and height distances. As a result, the proposed work considers view-1 and view-2 of the same scene as the input for text detection. The proposed method works based on the fact that texts in both views share the same properties. These properties are extracted by exploring Delaunay Triangulation (DT). The feature vectors of view-1 and view-2 are compared using cosine distance measures to detect Candidate Text Components (CTC). For each CTC in view-1 and view-2, the proposed method finds nearest neighbors for respective view-1 and view-2 based on chi square distance measures. This process gives text components with bounding boxes. The recognition step is used to eliminate false positives to improve results. Experimental results on our own dataset and the benchmark datasets, namely, ICDAR 2013, MSRATD-500, CTW1500, Total-Text, ICDAR 2017 MLT and COCO-text datasets, show that the proposed method outperforms the existing method in terms of F-measure.

In this work, the scope is limited to two views of the same spot. For some applications like forensic where multiple CCTV camera captures the same spot from the different height distance and angles. This results in many views for the same spot with more variations in contrast, resolution, blur due to defocus, distortions due to different angle and loss of visibility due to weather and low lights. Text detection in this situation is challenging because this involves many adverse factors. To addresses these challenges, one should investigate enhancement method, which should work for both day and low light images and fusing important information in multiple views to restore the missing information irrespective of above challenges. Further, in order to obtain good recognition results, there is

a need for rectifying the character alignment such that the conventional OCR can give better recognition results. Unlike the methods developed in the past that target traditional retrieval and labeling applications, the present work set for new dimension and direction for future text detection.

Declaration of Competing Interests

We have no conflicts of interest to declare.

Acknowledgments

This work is partially supported by Faculty Grant: GPF014D-2019, University of Malaya, Malaysia and also supported by the Natural Science Foundation of China under Grant 61672273 and Grant 61832008.

References

[1] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, J. Liang, East: an efficient and accurate scene text detector, CVPR (2017) 2642–2651.
[2] Y. Liu, L. Jin, S. Zhang, S. Zhang, "Detecting curve text in the wild: new dataset and new solution", arXiv:1712.02170, 2017.
[3] B. Shi, X. Bai, S. Belongie, Detecting oriented text in natural images by linking segments, in: Proc. CVPR, 2017, pp. 3482–3490.
[4] M. Liao, Z. Zhu, B. Shi, G.-s. Xia, X. Bai, Rotation-sensitive regression for oriented scene text detection, CVPR (2018) 5909–5918.
[5] W. He, X.-Y. Zhang, F. Yin, C.-L. Liu, Multi-Oriented and multi-lingual scene text detection with direct regression, IEEE Trans. IP (2018) 5406–5419.
[6] M. Liao, B. Shi, X. Bai, Textbox++:a single shot oriented scene text detector, IEEE Trans. IM (2018) 3676–3690.
[7] L. Deng, Y. Gong, Y. Lin, J. Shuai, X. Tu, Y. Zhang, Z. Ma, M. Xie, Detecting multi-oriented text with corner-based region proposals, Neurocomputing 01 (2019) 013, doi:10.1016/j.neucom.2019.
[8] D. NguyenVan, S. Lu, S. Tian, N. Ouarti, M. Mokhtari, "A pooling based scene text proposal technique for scene text reading in the wild", arXiv:1811.10003[cs.CV]
[9] Y. Gao, Y. Chen, J. Wang, H. Lu, "Reading scene text with attention convolutional sequence modeling", arXiv:1709.04303[cs.CV]
[10] C. Xue, S. Lu, W. Zhang, "MSR:multi-Scale regression for scene text detection",arXiv1-01.025-6(cs.CV)
[11] C. Bartz, H. Yang and C. Meinel, "STN-OCR: a single neural network for text detection and recognition", arXiv:1707.08831v1 [cs.CV]
[12] C. Bartz, H. Yang, C. Meinel, SEE: towards semi-supervised end-to-end scene text recognition, in: Proc. AAAI, 2018, pp. 6674–6681.
[13] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, X. Bai, ASTER: an attentional scene text recognizer with flexible rectification, IEEE Trans. PAMI (2019).
[14] L. Wu, P. Shivakumara, T. Lu, A new technique for multi-oriented scene text line detection and tracking in video, IEEE Trans. MM (2015) 1137–1152.
[15] A. Risnumawan, P. Shivakumara, C.S. Chan, C.L. Tan, A robust arbitrary text detection system for naturals scene images, Expert Syst. Appl. (2014) 8027–8048.
[16] F.P. Such, S. Pillai, F. Brockler, V. Singh, P. Hutkowski, R. Ptucha, Intelligent character recognition using fully convolutional neural networks, Pattern Recognit. (2019) 604–613.
[17] C.K. Ch'ng, C.S. Chan, Total-text: a comprehensive dataset for scene text detection and recognition, ICDAR (2017) 935–942.