A New Common Points Detection Method for Classification of 2D and 3D Texts in Video/Scene Images

Lokesh Nandanwar[1], Palaiahnakote Shivakumara[1](✉), Ahlad Kumar[2], Tong Lu[3], Umapada Pal[4], and Daniel Lopresti[5]

[1] Faculty of Computer Science and Information Technology,
University of Malaya, Kuala Lumpur, Malaysia
lokeshnandanwar150@gmail.com, shiva@um.edu.my

[2] Dhirubhai Ambani Institute of Information and Communication Technology,
Gandhinagar, Gujarat, India
ahlad_kumar@daiict.ac.in

[3] National Key Lab for Novel Software Technology, Nanjing University,
Nanjing, China
lutong@nju.edu.cn

[4] Computer Vision and Pattern Recognition Unit, Indian Statistical Institute,
Kolkata, India
umapada@isical.ac.in

[5] Computer Science and Engineering, Lehigh University, Bethlehem, PA, USA
lopresti@cse.lehigh.edu

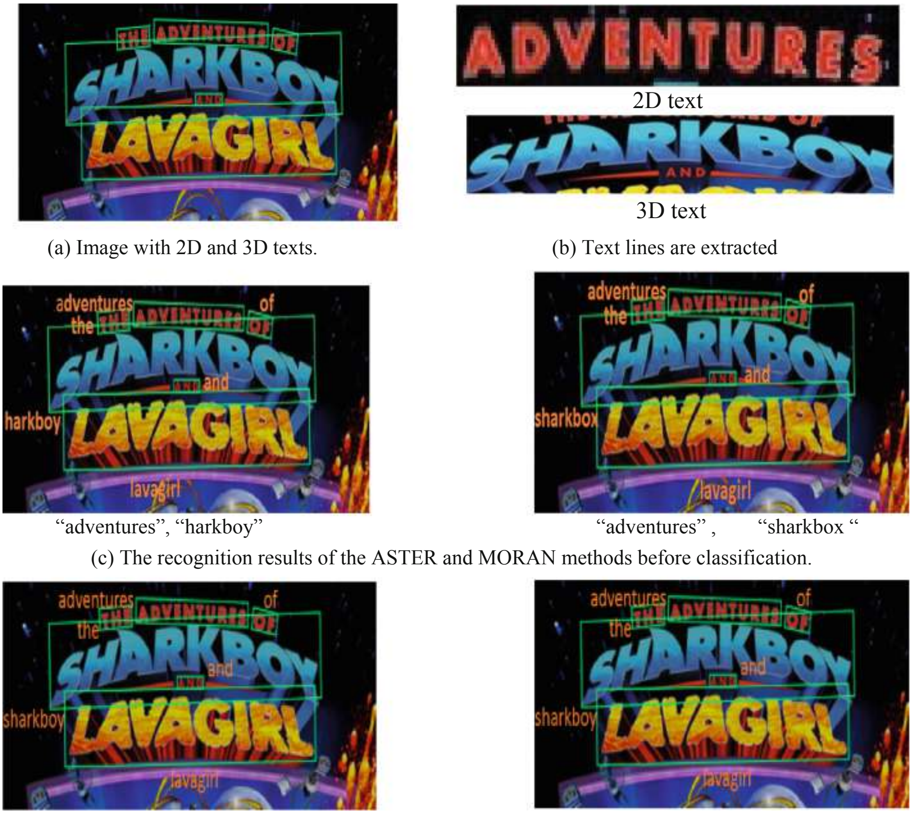
Abstract. Achieving high quality recognition result for video and natural scene images that contain both standard 2D text as well as decorative 3D text is challenging. Methods developed for 2D text may fail for 3D text due to the presence of pixels representing shadow and depth in the 3D text. This work aims at classification of 2D and 3D texts in video or scene images such that one can choose an appropriate method in the classified text for achieving better results. The proposed method explores Generalized Gradient Vector Flow (GGVF) for finding dominant points for input 2D and 3D text line images based on opposite direction symmetry. For each dominant point, our approach finds distance between neighbor points and plots a histogram to choose points which contribute to the highest peak as candidates. Distance symmetry between a candidate point and its neighbor points is checked and if a candidate point is visited twice, a common point is created. Statistical features such as the mean and standard deviation of the common points and candidate points are extracted to feed to Neural Network (NN) for classification. Experimental results on dataset of 2D-3D text line images and the dataset collected from standard natural scene images show that the proposed method outperforms exiting methods. Furthermore, recognition experiments before and after classification show recognition performance improves significantly as a result of applying our method.

Keywords: Gradient Vector Flow · Edge points · Candidate points · 2D text · 3D text · Text recognition · Video/scene images

X. Bai et al. (Eds.): DAS 2020, LNCS 12116, pp. 512–528, 2020.
https://doi.org/10.1007/978-3-030-57058-3_36

1 Introduction

When we look at the literature on text detection and recognition in video and natural scene images, new applications are emerging, such as extracting exciting events from sports videos, finding semantic labels for natural scene images [1] etc. At the same time, the challenges of text detection and recognition are also rising. There are methods for addressing some of these in the literature. For instance, Roy et al. [2] proposed a method for text detection from multiple views of natural scenes by targeting forensic application, where it is expected different views captured by CCTV cameras for the same location. Shivakumara et al. [3] proposed a method for Bib number detection and recognition in marathon images to trace the runner. Xue et al. [4] proposed a method for addressing challenges posed by blur and arbitrary orientation of text in images. Tian et al. [5] proposed a method for tracking the text in the web videos. Shi et al.'s [6] method is developed for solving the issues such as rectification caused by perspective



2D text

3D text

(a) Image with 2D and 3D texts.

(b) Text lines are extracted

"adventures", "harkboy"

"adventures" , "sharkbox "

(c) The recognition results of the ASTER and MORAN methods before classification.

"adventures", "sharkboy"

"adventures", "sharkboy"

(d) The recognition results of the ASTER and MORAN methods after classification

Fig. 1. Example of recognition results for 2D and 3D text before and after classification. The recognition results are displayed over the images for respective methods.

distortion and different camera angles to improve recognition rate. Luo et al. [7] proposed a method for text of arbitrary shaped characters in natural scene images.

It is noted from the above discussion that the main focus of the methods is to find a solution to several new challenges of 2D text recognition but not images that contain both 2D and 3D texts. The 3D text usually provides depth information representing 3D plane instead of standard 2D plane. This makes difference between 2D and 3D text in the images. However, the presence of 3D text in the images does not affect much for text detection performance in contrast to recognition performance. It is evident from the results shown in Fig. 1(a), where the method called CRAFT (Character region awareness for text detection) [8] that employs deep learning for arbitrary oriented text detection in natural scene images detects both 2D and 3D texts well as shown in Fig. 1 (b). At the same time, it is observed from the recognition results shown in Fig. 1(b) that the methods called ASTER [6] and MORAN [7], which uses deep learning models for achieving better recognition results, detect 2D text correctly but not 3D text. This is understandable because the methods are developed for 2D text recognition but not for 3D, where one can expect the following challenge. In case of natural images, 3D effect can be due to real depth of letters, or can be rendered by the artist to draw attention of the viewer in the case of synthetic images. Furthermore, the 3D text can contain shadow information because of capturing images at different angles. The effect can arise due to carving in stone or wood, or through embossing on paper. As a result, the extracted features may not be effective for differentiating text and non-text pixels in the images. One such example can be seen in Fig. 1(a) and (b).

Due to this effect, the above methods are not adequate for recognizing 3D text. This limitation motivated us to propose a new method for the classification of 2D and 3D texts in both video and natural scene images so that we can choose an appropriate method or modify the existing methods for achieving better results for 2D and 3D text recognition. It can be verified from the results shown in Fig. 1(c), where the same methods report correct recognition results for both 2D and 3D text after classification. Therefore, it is expected the recognition methods should score better results after classification compared to before classification.

2 Related Work

As mentioned in the previous section, several methods are proposed for recognizing text in video and natural scene images in the literature. Cheng et al. [9] proposed a method for arbitrarily oriented text recognition in natural scene images based on deep learning. The method proposes Arbitrary Orientation Network (AON) to capture deep features of irregular texts directly, which generate character sequences using an attention based decoder. Tian et al. [5] proposed a framework for text recognition in web videos based on text tracking. The method combines the information of text detection and tracking for recognizing texts. Luo et al. [7] proposed a multi-object rectified attention network for scene text recognition. The method explores a deep learning model, which is invariant to geometric transformation. The method works well for images affected by rotation, scaling and some extent to distortion. Raghunandan et al. [1] proposed multi-script-oriented text detection and recognition in video, natural scene and born digital images. The method extracts features based on wavelet transform for detecting characters with the help of an SVM classifier. Next, the method explores Hidden Markov Model (HMM) for recognizing characters and words in images.

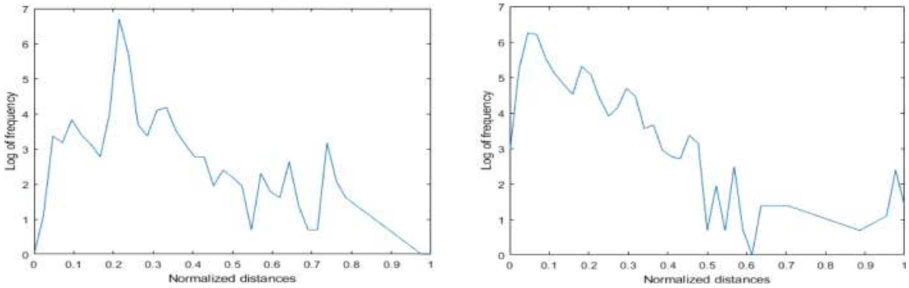
Qi et al. [10] proposed a novel joint character categorization and localization approach for character level scene text recognition. The idea of the method is to categorize characters by a joint learning strategy such that recognition performance improves. Shi et al. [6] proposed an attentional scene text recognizer with flexible rectification. The method uses thin plate spline transformation to handle a variety of text irregularities. The idea behind the method is to avoid pre-processing before recognition such that errors can be reduced to improve recognition rate. Rong et al. [11] proposed unambiguous scene text segmentation with referring expression comprehension. The method proposed a unified deep network to jointly model visual and linguistic information on both region and pixels levels for understanding texts. Villamizar et al. [12] proposed a multi-scale sequential network for semantic text segmentation and localization. The method explores fully convolutional neural networks that apply to particular cases of slide analysis to understand texts. Feng et al. [13] proposed an end-to-end framework for arbitrary shaped text spotting. The method proposes a new differentiable operator named RoISlide, which detects and recognize texts in images.

In the light of above review on text recognition methods, it is noticed that the methods find solutions to several challenges. However, the main aim of the methods is to recognize 2D text but not 3D text. As shown in Fig. 1, when we run the existing methods with a pre-defined network on 3D text, the recognition performance degrades. This is due to the effect of 3D, where we can expect depth and shadow information, which makes the problem more complex compared to 2D. Therefore, the scope of the above methods is limited to 2D images. Hence, this work aims at proposing a new method for classification of 2D and 3D texts in video and natural scene images such that the same methods can be modified to obtain better results for 3D text. However, there is an attempt to solve this classification problem due to, Xu et al. [14] where a method for multi-oriented graphics-scene 3D text classification in video is proposed. The method explores medial axis points of a character for classifying 2D and 3D texts in video. This method is sensitive to images with complex background and shadow because gradient directional symmetry depends on edge image. Zhong et al. [15] proposed to use shadow detection for 3D text classification. However, the method is not robust to complex background images because the threshold used in the method does not work well for different images. In addition, if an image does not contain enough shadow information, the method may not work well. Therefore, there is a need for developing a new method, which can overcome the above problems to improve recognition performance.

3 Proposed Method

The work described here takes as its input text line images that are produced by existing text detection methods for classification 2D and 3D text. We use the method called CRAFT [8] for text detection as it uses a powerful deep learning model and is robust to several challenges, which is evident from the results shown in Fig. 1(a) and (b).

Motivated by the method [14] where it is stated that stroke width distance, which is thickness of the stroke, is almost the same for each whole character, the proposed method exploits the same property for extracting features that can discriminate 2D and 3D texts in this work. As a result, it is expected that stroke width distance for 2D text exhibits regular patterns while does not for 3D text. It is illustrated in Fig. 2 for 2D and 3D text line images in Fig. 1(b), where it can be seen that the histogram which is drawn for stroke width distance *vs* frequencies appear like a normal distribution but from the histogram of 3D text, we cannot predict the behavior. This is true because for 2D text, the number of pixels which satisfy stroke width is larger than the other distances, while for 3D, it does not. The proposed method finds the pixels that have opposite directions with a certain degree in 3 × 3 window of every edge pixel in the image as dominant points. The special property of Gradient Vector Flow (GVF) is that the GVF arrow pointing towards edges [16] due to force at edges. Therefore, for the pixels that represent text edges, one can expect opposite GVF direction symmetry with a certain degree. This works well for text with and without shadows. This step results in dominant points, which generally represent edge pixels including edges of shadow. However, we explore Generalized Gradient Vector Flow (GGVF) [17] for obtaining GGVF direction for pair pixels unlike traditional GVF, which is not robust to the pixels at corners and low resolution [16].



(a) Stroke width distances for 2D text

(b) Stroke width distances for 3D text

Fig. 2. Histogram for the stroke width distances of 2D and 3D texts.

The proposed method finds the distance between each dominant point and its eight neighbors. We perform histogram operation on distance value for choosing the pixel which contributes to the highest peak as candidate points. In other words, the proposed method chooses a pair of pixels which satisfy the distance symmetry. This step eliminates false dominant points. If it is 3D text, one can expect a shadow. If there is a shadow, we can expect candidate points which share the stroke width of shadow information, resulting in common points which are visited by twice while checking distance symmetry. This is the cue for extracting statistical features for using common point as well candidate points. The extracted features are passed to a Neural Network (NN) classifier for the classification of 2D and 3D texts.

3.1 Dominant Point Detection

Inspired by the work [17] where GGVF is proposed for medical image segmentation, we explore GGVF for finding common points detection in this work. According to the method [17], a GGVF arrow usually points towards edges because of high force at edges. As a result, every edge pixel can have opposite GGVF arrow directions in this work. If a pixel represents text, it is expected that the two opposite arrows have almost the same angle, else the two opposite arrows have different angles. The general equation for GGVF is defined in Eq. (1), which is an energy function of the GGVF snake model. We use the same equation to obtain GGVF for the input image. This GGVF accepts edge images as the input for finding GGVF, and the proposed method obtains Canny edge image for the input image to obtain GGVF arrows as shown in Fig. 4(a), where we can see GGVF arrows for 2D and 3D text images. The energy function of GGVF field $z(x, y) = (u(x, y), v(x, y))$ is defined as,

$$E = \iint g(|\nabla f|)(\emptyset(|\nabla u|) + \emptyset|\nabla v|) + h(|\nabla f|)(z - \nabla f) dx\, dy \tag{1}$$

where $\emptyset(|\nabla v|) = \sqrt{1 + |\nabla v|^2}$ and $\emptyset(|\nabla u|) = \sqrt{1 + |\nabla u|^2}$ with

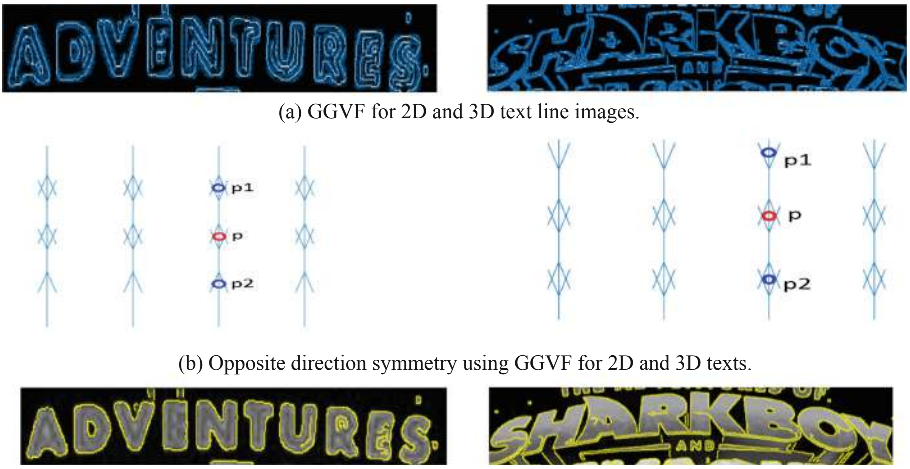
$$g(|\nabla f|) = e^{\frac{-\nabla f}{k}} \quad \text{and} \quad h(|\nabla f|) = 1 - g(|\nabla f|)$$

In Eq. (1), the first term denotes smoothing, which produce a vector field. The second term is the data fidelity that drives the vector field z close to the gradient of the image i.e. ∇f. Also parameter k acts as a weighing parameter that balance the smoothing and data fidelity term. Here, higher the value of noise indicates a larger value of k.

For each edge pixel in the image, the proposed method defines a window of 3 × 3 dimension over the input image. The proposed method checks GGVF arrow directions of vertical, horizontal and diagonal pixels pairs in 3 × 3 window as shown in Fig. 3(b). If any pair satisfies the opposite arrow direction symmetry as defined in Eq. (2), the pair of pixels are considered as dominant points as shown in Fig. 3(c), where one can see almost all the edge pixels of text are detected irrespective of 2D and 3D texts. It is noted that the results in Fig. 3(c) contain still edges of background information. This is due to complex background and shadow of images.

$$GGVF(p1) - GGVF(p2) = \pi \tag{2}$$

where $p1$ and $p2$ are the pair of pixels in the 3×3 window, which can represent vertical, horizontal and diagonal pixels.



(a) GGVF for 2D and 3D text line images.

(b) Opposite direction symmetry using GGVF for 2D and 3D texts.

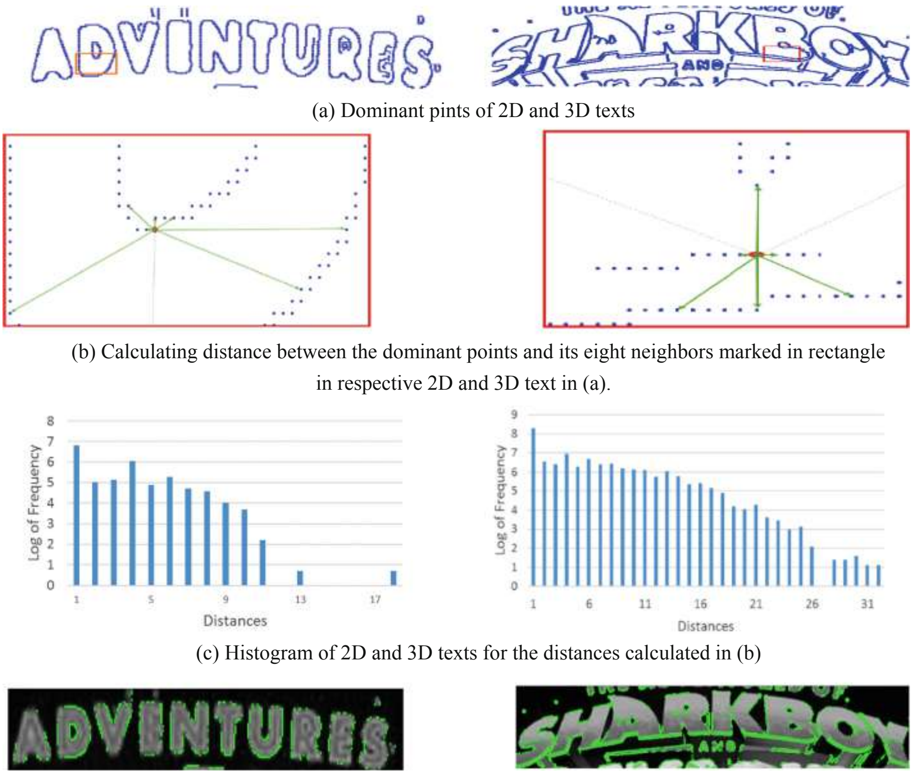
(c) Detecting dominant points for 2D and 3D text line images.

Fig. 3. Dominant points detection for 2D and 3D using GGVF opposite direction symmetry.

3.2 Candidate Points Detection

In order to eliminate false dominant points, inspired by the statement that the pixels which represent edges of characters have almost the same stroke width distance [16], the proposed method finds the distance between the eight neighbor points of each dominant point as defined in Eq. (3). It is illustrated in Fig. 4(a) and (b), where (a) shows the results of dominant points detection for 2D and 3D texts, and Fig. 4(b) gives directions of eight neighbor points to find the distances between the center and its eight neighbors. This process continues for all the dominant pints in the image in Fig. 4(a).

To extract the point which represents stroke width of characters, the proposed method performs histogram operation on distance values, and considers the pixels that contributes to the highest peak in the histogram as candidate points as shown in Fig. 4(c) and (d), respectively. Note: before performing histogram operation, we normalize the distance values using a log function to balance distance values, and ignore the first highest peak in the histogram because it considers adjacent dominant point of the center point, which does not give stroke width of the text. Usually the distance between adjacent candidate points is one or two and hence this value corresponds to the highest peak in the histogram. Therefore, the proposed method considers the second highest peak for detecting candidate points, which is considered as distance symmetry. It is observed from the results in Fig. 4(d) that some of the dominant points which represent background as well as text are removed, especially for 2D text. Removing a few text pixels does not affect for the classification of 2D and 3D texts in this work. It is also noted from the results of 3D text in Fig. 4(d) that the detected candidates represent



(a) Dominant pints of 2D and 3D texts

(b) Calculating distance between the dominant points and its eight neighbors marked in rectangle in respective 2D and 3D text in (a).

(c) Histogram of 2D and 3D texts for the distances calculated in (b)

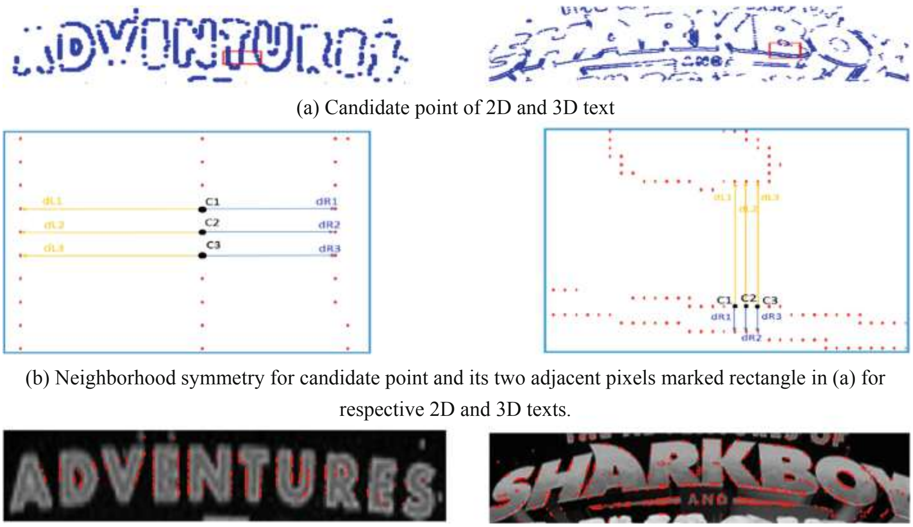
(d) Candidate points detection for 2D and 3D texts.

Fig. 4. Candidate points detection for 2D and 3D text using distance symmetry.

both text and shadow pixels. This shows that stroke width of the pixel that represents shadow have almost the same distance as text pixels. However, still we can see some points which represent background information.

$$CN(x,y) = \{a_i(x,y), b_i(x,y)\} \ \forall \ \{a(x,y), b(x,y)\} \in DH \leftrightarrow H_i = h2 \qquad (3)$$

where $h2$ denotes the value of distance at second highest peak, H be the set of all distance between the eight neighbor points of each dominant point and set DH contains set of pair points $\{a_i(x,y), b_i(x,y)\}$ corresponding to distance $H_i, i \in length(H)$ and $CN(x,y)$ denote candidate points.



(a) Candidate point of 2D and 3D text

(b) Neighborhood symmetry for candidate point and its two adjacent pixels marked rectangle in (a) for respective 2D and 3D texts.

(c) Common points detection for 2D and 3D texts using neighborhood symmetry.

Fig. 5. Common points detection for 2D and 3D texts using neighborhood symmetry.

3.3 Common Points Detection

It is noted from the results of candidate point detection shown in Fig. 5(a) for 2D and 3D text that candidate points are detected irrespective of 2D and 3D. It is also noted that due to shadow, one can expect more candidate points which share the stroke width of the shadow as well as characters in the case of 3D text. But for 2D, we cannot expect as many as candidate points because the points share the stroke width only at character boundaries. To extract the above observation, the proposed method finds stroke widths for each candidate points and its two nearest neighbor candidate points based on gradient direction and opposite gradient direction of the points as shown in Fig. 5(b), where we can see stroke width for three candidate points for both 2D and 3D texts. When the proposed method finds stroke width distances, it is noted that some of the points are hit twice; these are common points to the text and shadow in the case of 3D, and character boundary points in the case of 2D as shown in Fig. 5(c). The common point detection is defined in Eq. (4). In Fig. 5(c), one can notice a few pixels for 2D and as many as for 3D. If the points share the stroke width of character and shadow in case of 3D and character boundaries in case of 2D, the points considered as common points and the condition is called neighborhood symmetry.

$$CP(x,y) = (x_i, y_i) \leftrightarrow (dR1_i = dR2_i = dR3_i)\ \forall\ (\mathrm{x_i}, \mathrm{y_i}) \in \mathrm{CN}(\mathrm{x_i}, \mathrm{y_i}) \tag{4}$$

where $\{dL1, dL2, dL3\}$ denote the nearest pixel distance in gradient direction, $\{dR1, dR2, dR3\}$ denote the nearest pixel distance in the opposite of gradient direction, and $CP(x,y)$ denote the common points.

The results in Fig. 5(c) show that spatial distribution of candidate and common points indicates clear difference between 2D and 3D texts. In addition, it is also true that the points which represent shadow have low intensity values compared to character pixels. Based on these observations, the proposed method extracts the following statistical features for candidate and common points. Since every common points associated with two neighbor points, the proposed method consider the point detected using gradient direction to the common point as Neighbor point-1 and the point detected using opposite gradient direction to the common point as Neighbor point-2 as shown in Fig. 5(b). The proposed method considers pixels between the common point to Neighbor point-1 and the common point to Neighbor point-2 as two separate groups for feature extraction, say, Group-1 and Group-2. Group-1 and Group-2 include the pixels of all the common points in the image. The proposed method calculates the mean of intensity values of the pixels in the respective groups, which gives two features as defined in Eq. (5). The proposed method also calculates standard deviation for the intensity values of the pixels in the respective groups, which gives two more features as defined in Eq. (6). In the same way, instead of intensity values, the proposed method considers distance between the common point to Neighbour points to calculate mean and standard deviation for the respective groups, which gives four features as defined in Eq. (7) and Eq. (8).

In total, 8 features are extracted from common points. In addition, the proposed method calculates the mean and standard deviation for the distances of candidate points of respective two groups, which gives four features. Overall, the proposed method extracts 12 features using common points and candidate points, which represent spatial distribution and difference between intensity values of points for classification of 2D and 3D texts. The distribution of 12 features are shown for the input 2D and 3D texts in Fig. 7, where one can see smooth variation for 2D and large variations for 3D. This is expected because the space between the characters does not vary much in the case of 2D text, while we can expect large variations in the case of 3D text due to presence of shadows. This shows that the proposed feature extracts distinct property of 2D and 3D texts for classification. Note that the values are normalized to the range of 0 and 1 before plotting the graphs in Fig. 6.

$$MI = \frac{\sum m}{n} \tag{5}$$

$$SI = \sqrt{\frac{\sum (m - \bar{m})^2}{n}} \tag{6}$$

$$MD = \frac{\sum d}{n} \tag{7}$$

$$SD = \sqrt{\frac{\sum (d - \bar{d})^2}{n}} \tag{8}$$

where *MI* denotes the mean, *m* represents the intensity values, *n* is the total number of pixels, *SI* denotes the standard deviation, *MD, SD* denote the mean and standard deviation of distance values and d is the distance between the common point and its neighbor points.

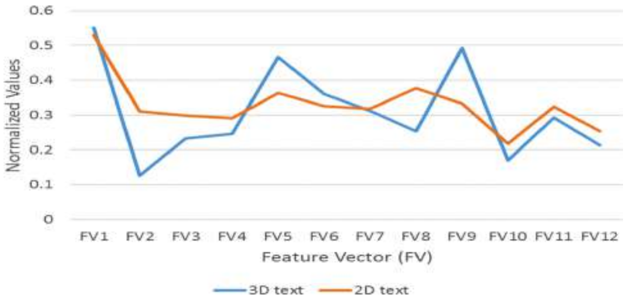


Fig. 6. Statistical Feature Vector for 2D and 3D text classification

Motivated by the ability of neural networks [18] for classification, we propose such a network for classifying 2D and 3D text in this work by feeding 12 features into the network. The structure of proposed network is as follows. It has 5 intermediate dense layers with 50, 100, 50, 25 and 10 units/features each. Input layer has 12 features and final output layer has 1 feature {0 for 2D and 1 for 3D}. We use Rectified linear activation function (ReLU) for all intermediate layers and Sigmoid [19] activation function for the final layer. Dropout with drop rate of 20% between the intermediate layers is applied to reduce the overfitting problem. Binary cross entropy loss is estimated [20] as defined in Eq. (9), where y is the label (Ground Truth), and p(y) is the predicted probability for the total number of N samples. Adam [21] optimizer with learning rate of 0.01 is used during the training process and the batch size used is 8 during training. Each model is trained for 200 epochs with model checkpoint to store the best trained model using Keras framework [22]. For training and testing, we use 80% samples for training and 20% for testing in this work. The details of proposed deep neural network are listed in Table 1.

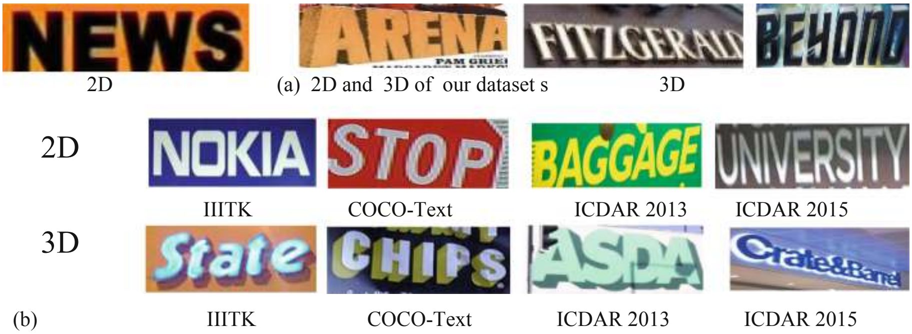
$$BCE(q) = -\frac{1}{N}\sum_{i=1}^{N}(y_i.\log(p(y_i)) + (1 - y_i).\log(1 - p(y_i))) \tag{9}$$

Table 1. The details of the deep neural network classifier

Layer number	Out size
Input layer	12 × 1, ReLU
Dense layer 1	50 × 1, Dropout = 20%, ReLU
Dense layer 2	100 × 1, Dropout = 20%, ReLU
Dense layer 3	50 × 1, Dropout = 20%, ReLU
Dense layer 4	30 × 1, Dropout = 20%, ReLU
Dense layer 5	10 × 1, ReLU
Output layer	1 × 1, Sigmoid

4 Experimental Results

For 2D and 3D text classification, we create our own dataset assembled from different sources, including YouTube, scene images and movie posters that are fond online. Our dataset includes text line images with complex backgrounds, low resolution, low contrast, font font-size variations, arbitrary shaped characters and some extent to distortion as shown sample images in Fig. 7(a), which gives 513 2D and 505 3D text line images. For objective evaluation of the proposed method, we also collect 2D and 3D text line images from the benchmark datasets of natural scene images, namely, IIIT5K, COCO-Text, ICDAR 2013 and ICDAR 2015, which have a few 3D text line images as shown sample images in Fig. 7(b). We collect 317 2D, 305 3D from IIIT5K dataset, 472 2D, 530 3D from COCO-Text, 123 2D, 74 3D from ICDAR 2013 and 111 2D, 90 3D from ICDAR 2015 datasets, which gives total 1023 2D text line images and 999 3D text line images. This dataset is considered as the standard dataset for experimentation in this work.



(a) 2D and 3D of our dataset s

(b)

Fig. 7. Sample images of 2D and 3D text from our dataset and standard dataset

In total, 3040 text line images, which include 1536 2D images and 1504 3D images are considered for evaluating the proposed and existing methods.

To show effectiveness of the proposed method, we implement two existing methods which have the same objective of 2D and 3D text classification. Xu et al. [14] method employs gradient inward and outward directions for 2D and 3D text image classification. Zhong et al.'s [15] method proposes shadow detection for 2D and 3D text image classification. The motivation to choose these two existing methods is to show that stroke width features used in Xu et al. for classifying 2D and 3D text image is not enough to handle the images considered in this work. Similarly, the features used for shadow detection in Zhong et al. are not adequate. Similarly, to validate the proposed classification, we implement two recognition methods, namely, ASTER [6] and MORAN [7], which explore powerful deep learning models for recognizing text of different complexities in natural scene images. The codes of these methods are available to the public.

For measuring performance, we use the standard metrics, namely, a confusion matrix and average classification rates. The classification rate is defined as the number of images classified correctly by the proposed method divided by the actual number of images. The average classification rate is defined as the mean of diagonal elements of the confusion matrix. For recognition experiments, we use recognition rate, which is defined as the number of characters recognized correctly divided by the actual number of characters. To test the utility of the proposed classification, we calculate the recognition rate of the methods before after classification. Before classification includes text lines of both 2D and 3D for calculating recognition rate, while after classification includes text lines of 2D and 3D separately. It is expected that the recognition rate will be higher after classification compared to before classification. This is because after classification the complexity of the problem is reduced and the method can be modified/trained according to the complexity of individual classes. In this work, we use pre-trained model with the same parameters and values, while for after classification, we tune the parameters, namely, "Epochs" and "Batch size" according to complexity of individual classes. In case of ASTER method, 200 epochs with batch size of 16 and in case of MORAN method, 120 epochs with batch size of 32 after classification are used.

The proposed method involves a few key steps, namely, Canny edge pixels, Dominant, Candidate and Common points detection. To assess the contribution of each step, we calculate classification rate using our dataset and the results are reported in Table 2. It is noted from Table 2 that when we consider all the pixels in the edge image for the classification of 2D and 3D text lines, the proposed method does not score as high as other steps and the proposed method with common points. This is valid because all edge pixels involve background information which does not contribute for classification of 2D and 3D text. However, for Dominant and Candidate points, the proposed method scores almost the same. This shows that both the steps contribute equally for achieving better results. It is also observed from Table 2 that the individual steps or feature points do not have the ability to achieve results as achieved by the proposed method. This is due to the inclusion of many background pixels, while common points does not.

Table 2. Confusion matrix and average classification rate of the different steps for 2D and 3D texts classification on our dataset.

Features	All edge pixels		Dominant points		Candidate points		Proposed (common points)	
Classes	2D	3D	2D	3D	2D	3D	2D	3D
2D	66.86	33.14	69.78	30.22	62.76	37.24	92.62	7.38
3D	28.32	71.68	18.02	81.98	8.52	91.48	6.93	93.07
Average	69.27		75.88		72.37		**92.8**	

4.1 Evaluating the 2D and 3D Text Classification

For testing the proposed classification on each dataset and the full dataset, we calculate average classification rate as reported in Table 3, where one can see as sample size increases, the results also increase. This shows that each dataset has its own complexity. For the full dataset, which is the collection from all the dataset, the proposed method scores promising results. Quantitative results of the proposed and existing methods for our dataset and the standard full-dataset are reported in Table 3, where it is noted that the proposed method is the best at average classification rate for both our and the standard full dataset compared to the existing methods. This is justifiable because the features used in the methods are not adequate to cope with the challenges of our dataset and the standard datasets. In addition, the features are sensitive to complex background. When we compare the results of two existing methods, Zhong et al. achieve poor results compared to Xu et al. This is due to constant thresholds used for shadow detection of 3D text in the method. Since the dataset contains a large variation, constant thresholds do not work, while Xu et al. use dynamic rules for classification, which is better than fixing constant threshold values. On the other hand, the proposed method explores Generalized Gradient Vector Flow (GGVF) for finding directional and distance symmetry, which is effective compare to the features used in the existing methods. It is observed from Table 4 that the proposed method scores low results for the standard full-dataset compared to our dataset. This shows that the standard dataset is much more complex than our dataset.

Table 3. Confusion matrix and average classification rate of the proposed method on different standard natural scene datasets and full dataset.

Datasets	IIIT5K		COCO-Text		ICDAR 2013		ICDAR 2015		Standard full-dataset	
Classes	2D	3D	2D	3D	2D	3D	2D	3D	2D	3D
Size	317	305	472	530	123	74	111	90	1023	999
2D	85.61	14.38	91.79	8.21	73.61	26.38	70.45	29.54	86.33	13.6
3D	13.27	86.72	7.92	92.07	31.42	68.58	20	80	12.6	87.3
Average	86.15		91.93		71.1		75.27		**86.83**	

Table 4. Confusion matrix and average classification rate of the proposed and existing methods on both our and standard datasets (in %)

Dataset	Our dataset						Standard full-dataset					
Methods	Zhong et al.		Xu et al.		Proposed		Zhong et al.		Xu et al.		Proposed	
Classes	2D	3D	2D	3D	2D	3D	2D	3D	2D	3D	2D	3D
2D	32.74	67.25	66.28	33.72	92.62	7.38	53.86	46.13	46.6	53.4	86.33	13.67
3D	46.33	53.67	28.31	71.69	6.93	93.07	32.8	67.2	31.2	68.8	12.66	87.34
Average	43.2		70.48		**92.8**		60.53		57.7		**86.83**	

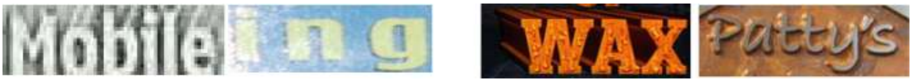
4.2 Recognition Experiments for Validating the Proposed Classification

As mentioned earlier, to show the usefulness of the proposed classification, we conduct recognition experiments before and after classification using different recognition methods on both our own and the standard datasets. The recognition methods run on all the images regardless of 2D and 3D for before classification, while the methods run on each class individually after classification. The results of recognition methods on the above mentioned datasets are reported in Table 5, where it is noticed clearly that the recognition rates given by two methods after classification are improved significantly for both the datasets compared to the recognition rates before classification. This shows that the proposed classification is useful for improving recognition performance on text detection in 3D video and natural scene images. It is observed from Table 5 that both the recognition methods achieve low results for 3D and compared to 2D after classification. This indicates that the recognition methods are not capable of handling 3D texts. This is true because in this work, we train recognition methods on 2D and 3D classes individually after classification step. We modify the architecture using transfer learning to achieve better results according to complexity. However, the proposed classification improves the overall performance of the recognition methods compared to before classification.

Table 5. Recognition performance of different methods before and after classification on our and standard datasets (in%)

Methods	Our dataset				Standard dataset-FULL			
	Before classification	After classification			Before classification	After classification		
	2D + 3D	2D	3D	Average	2D + 3D	2D	3D	Average
ASTER [6]	78.15	97.0	85.7	91.35	88.5	96.1	85.4	90.75
MORAN[7]	85.5	94.1	87.6	90.85	89.78	96.09	86.4	91.25

Sometimes, when the images affected by multiple adverse effects as shown in Fig. 8, there are chances of misclassification. When the features overlap with the background information, it is hard to find common points for differentiating 2D and 3D texts, which shows there is a scope for improvement in future work.



(a) Our dataset Standard (b) Our dataset Standard

Fig. 8. Example of unsuccessful classification of the proposed method on our and standard datasets. (a) 2D misclassified as 3D and vice versa for the images in (b)

5 Conclusion and Future Work

In this paper, we have proposed a new method for the classification of 2D and 3D text in video images which includes natural scene images such that recognition performance can be improved significantly. We explore generalized gradient vector flow for directional and distance symmetry checking, unlike conventional gradient vector flow which is not robust for low contrast and corners. Based on the property of stroke width distance, our approach defines directional and distance symmetry for detecting dominant and candidate points. The proposed method defines neighborhood symmetry for finding common points which share the stroke width of characters as well as shadow information. For the common points, our approach extracts statistical features, and the features are further passed to a deep neural network for the classification of 2D and 3D texts in video and natural scene images. Experimental results on different datasets show that the proposed classification outperforms the existing classification methods. The recognition results before and after classification show that recognition performance improves significantly after classification compared to before classification. However, there are still some limitations as discussed in the Experimental Section, thus we plan to improve the method in an attempt to address such limitations in the future.

References

1. Raghunandan, K.S., Shivakumara, P., Roy, S., Kumar, G.H., Pal, U., Lu, T.: Multi-script-oriented text detection and recognition in video/scene/born digital images. IEEE Trans. CSVT **29**, 1145–1162 (2019)
2. Roy, S., Shivakumara, P., Pal, U., Lu, T., Kumar, G.H.: Delaunay triangulation based text detection from multi-view images of natural scene. PRL **129**, 92–100 (2020)
3. Shivakumara, P., Raghavendra, R., Qin, L., Raja, K.B., Lu, T., Pal, U.: A new multi-modal approach to bib number/text detection and recognition in Marathon images. PR **61**, 479–491 (2017)
4. Xue, M., Shivakumara, P., Zhang, C., Lu, T., Pal, U.: Curved text detection in blurred/non-blurred video/scene images. MTAP **78**(18), 25629–25653 (2019). https://doi.org/10.1007/s11042-019-7721-2
5. Tian, S., Yin, X.C., Su, Y., Hao, H.W.: A unified framework for tracking based text detection and recognition from web videos. IEEE Trans. PAMI **40**, 542–554 (2018)
6. Shi, B., Yang, M., Wang, X., Luy, P., Yao, C., Bai, X.: ASTER: an attentional scene text recognizer with flexible rectification. IEEE Trans. PAMI **41**, 2035–2048 (2019)
7. Luo, C., Jin, L., Sun, Z.: MORAN: a multi-object rectified attention network for scene text recognition. PR **90**, 109–118 (2019)
8. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: Proceedings of CVPR (2019)
9. Cheng, Z., Xu, Y., Bai, F., Niu, Y.: AON: towards arbitrarily-oriented text recognition. In: Proceedings of CVPR, pp. 5571–5579 (2018)
10. Qi, X., Chen, Y., Xiao, R., Li, C.G., Zou, Q., Cui, S.: A novel joint character categorization and localization approach for character level scene text recognition. In: Proceedings of ICDARW, pp. 83–90 (2019)
11. Rong, X., Yi, C., Tian, Y.: Unambiguous scene text segmentation with referring expression comprehension. IEEE Trans. IP **29**, 591–601 (2020)

12. Villamizar, M., Canevert, O., Odobez, J.M.: Multi-scale sequential network for semantic text segmentation and localization. PRL **129**, 63–69 (2020)
13. Feng, W., He, W., Yin, F., Zhang, X.Y., Liu, C.L.: TextDragon: an end-to-end framework for arbitrary shaped text spotting. In: Proceedings of ICCV, pp. 9076–9085 (2019)
14. Xu, J., Shivakumara, P., Lu, T., Tan, C.L., Uchida, S.: A new method for multi-oriented graphics-scene-3D text classification in video. PR **49**, 19–42 (2016)
15. Zhong, W., Raj, A.N.J., Shivakumara, P., Zhuang, Z., Lu, T., Pal, U.: A new shadow detection and depth removal method for 3d text recognition in scene images. In: Proceedings of ICIMT, pp. 277–281 (2018)
16. Khare, V., Shivakumara, P., Chan, C.S., Lu, T., Meng, L.K., Woon, H.H., Blumenstein, M.: A novel character segmentation-reconstruction approach for license plate recognition. ESWA **131**, 219–239 (2019)
17. Zhu, S., Gao, R.: A novel generalized gradient vector flow snake model using minimal surface and component-normalized method for medical image segmentation. BSPC **26**, 1–10 (2016)
18. Silva, I.N.D., Spatti, D.H., Flauzino, R.A., Liboni, L.H.B., Reis Alves, S.F.D.: Artificial Neural Networks, vol. 39. Springer, Heidelberg (2017). https://doi.org/10.1007/978-3-319-43162-8
19. Narayan, S.: The generalized sigmoid activation function: competitive supervised learning. IS **99**, 69–82 (1997)
20. Nasr, G.E., Badr, E.A., Joun, C.: Cross entropy error function in neural networks: forecasting gasoline demand. In: Proceedings of FLAIRS, pp. 381–384 (2002)
21. Kingma, P.D., Bai, J.L.: Adam: a method for stochastic optimization. In: Proceedings of ICLR, pp. 1–15 (2015)
22. Keras: Deep learning library for theano and tensorflow (2015). https://keras.io/