Expert Systems with Applications 41 (2014) 8027–8048

Contents lists available at ScienceDirect

Expert Systems with Applications

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

# A robust arbitrary text detection system for natural scene images

Anhar Risnumawan [a], Palaiahankote Shivakumara [a,*], Chee Seng Chan [a], Chew Lim Tan [b]

[a] Centre of Image and Signal Processing, Faculty of Computer Science and Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia
[b] School of Computing, National University of Singapore, Singapore

ARTICLE INFO

*Article history:*
Available online 17 July 2014

*Keywords:*
Arbitrary text detection
Invariant properties
Symmetry features
Ellipse growing
Text verification
Text restoration

ABSTRACT

Text detection in the real world images captured in unconstrained environment is an important yet challenging computer vision problem due to a great variety of appearances, cluttered background, and character orientations. In this paper, we present a robust system based on the concepts of Mutual Direction Symmetry (MDS), Mutual Magnitude Symmetry (MMS) and Gradient Vector Symmetry (GVS) properties to identify text pixel candidates regardless of any orientations including curves (e.g. circles, arc shaped) from natural scene images. The method works based on the fact that the text patterns in both Sobel and Canny edge maps of the input images exhibit a similar behavior. For each text pixel candidate, the method proposes to explore SIFT features to refine the text pixel candidates, which results in text representatives. Next an ellipse growing process is introduced based on a nearest neighbor criterion to extract the text components. The text is verified and restored based on text direction and spatial study of pixel distribution of components to filter out non-text components. The proposed method is evaluated on three benchmark datasets, namely, ICDAR2005 and ICDAR2011 for horizontal text evaluation, MSRA-TD500 for non-horizontal straight text evaluation and on our own dataset (CUTE80) that consists of 80 images for curved text evaluation to show its effectiveness and superiority over existing methods.



## 1. Introduction

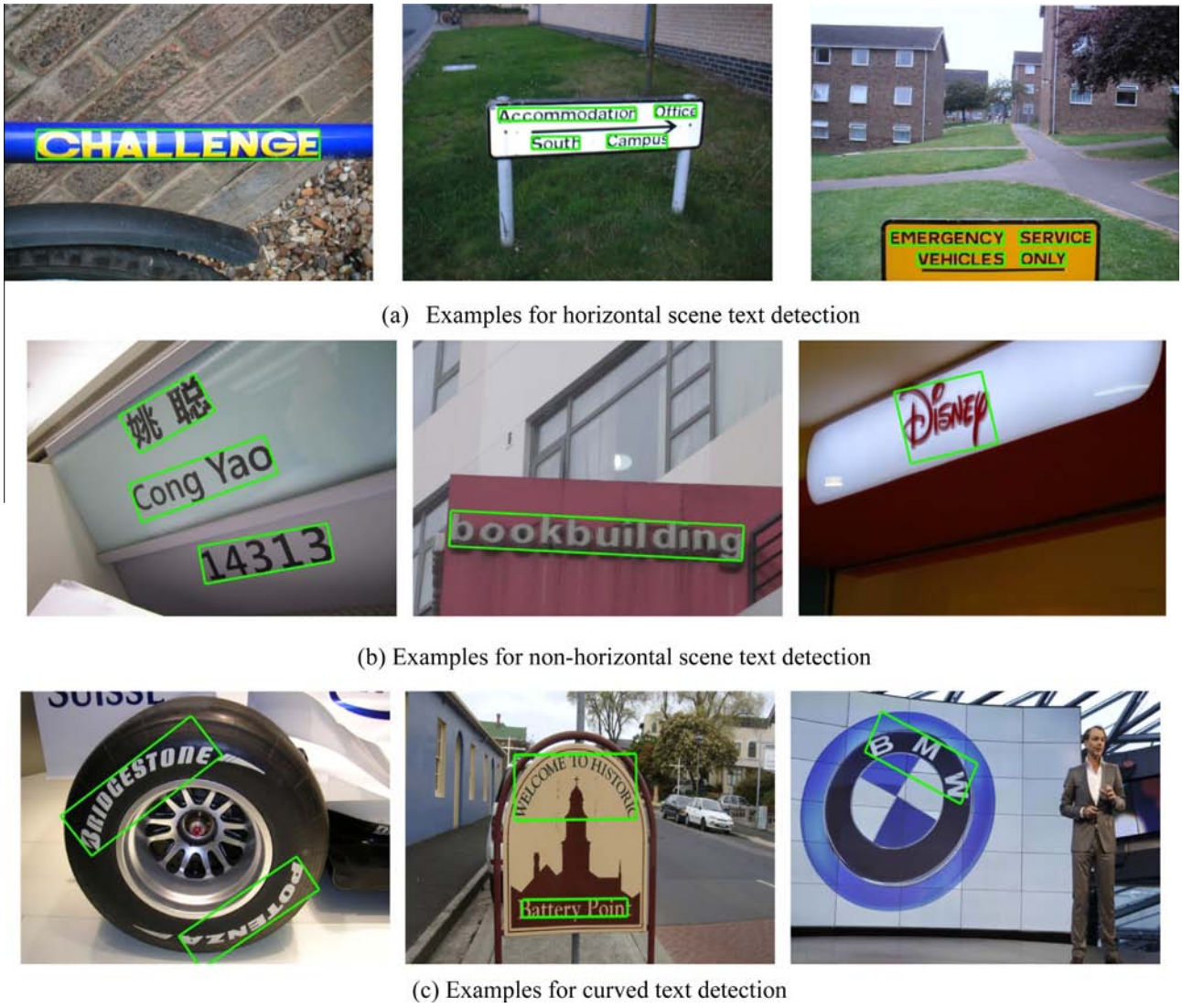
In recent years, text detection and recognition from natural scene images has gained much research attention aiming to achieve comparable accuracy to that in document analysis (González & Bergasa, 2013). This is mainly due to its usefulness for many real world applications, such as assisting visually impaired people, enhancing safe vehicle driving, helping tourists in navigation through reading road signs, visual inspection tasks and so on (Grafmuller & Beyerer, 2013; Jung, Kim, & Jain, 2004; Liang, Doermann, & Li, 2005; Park & Kim, 2013). In addition, it enhances the capability of content-based image retrieval models in order to retrieve meaningful events from the event database because it provides semantics to the content of images and videos (Mishra, Alahari, & Jawahar, 2012; Smith, Field, & Learned-Miller, 2011; Wang, Babenko, & Belongie, 2011; Wei & Lin, 2012). This shows that the proposed robust text detection system is like an expert system which is useful in several real time applications. The same conclusions can be drawn from Wei and Lin (2012) where robust video text detection system is proposed.

Despite many efforts (Mishra et al., 2012; Smith et al., 2011; Wang et al., 2011; Wei & Lin, 2012; Weinman, Learned-Miller, & Hanson, 2009) for text detection from natural scene images, it is still considered a challenging and unsolved problem. This is because text in natural scene images suffers from complex background, varieties of appearance, color variations, font size variations, font variation, arbitrary orientation, contrast variations, perspective effects and occlusion as illustrated in Fig. 1 where one can see that there is a large variety of text with complex background. Therefore, there is still much room for further research and improvements.

Generally, most existing methods (Chen & Yuille, 2004; Fernandez-Caballero, Lopez, & Castillo, 2012; Minetto, Thome, Cord, Fabrizio, & Marcotegui, 2010; Neumann & Matas, 2012; Pan, Hou, & Liu, 2011; Shivakumara, Phan, & Tan, 2011; Yao, Bai, Liu, Ma, & Tu, 2012; Yi & Tian, 2011; Yun, Jing, Yu, & Huang, 2010) focus on the use of classifier with training samples for classification of text pixels and text components based on the characteristics of character shapes. However, the use of classifier with training samples restricts generalization capability such as multilingual text detection, while the characteristics based on character shapes are only good for text that preserves the shapes of the characters. Such constraints do not necessarily hold for arbitrary text embedded in the background cluttered with grass, leaves and

* Corresponding author.
*E-mail addresses:* anh.risn@um.edu.my (A. Risnumawan), shiva@um.edu.my (P. Shivakumara), cs.chan@um.edu.my (C.S. Chan), tancl@comp.nus.edu.sg (C.L. Tan).

http://dx.doi.org/10.1016/j.eswa.2014.07.008
0957-4174/




(a) Examples for horizontal scene text detection

(b) Examples for non-horizontal scene text detection

(c) Examples for curved text detection

**Fig. 1.** Sample results of text detection on different types of texts.

buildings. In addition, determining the number of training samples for both text and non-text is non-trivial because of the unpredictable background in natural scene images. To the best of our knowledge, almost all existing methods focus on horizontal and non-horizontal straight texts but none on arbitrary (curved) texts aligned in such formations as circles, arcs, S and Z shapes. Examples of these texts are shown in Fig. 1 where the first, second and third rows show horizontal, non-horizontal and curved texts, respectively. One can understand that scene text in the real world can have any shapes and orientations. As a result of this, conventional solutions in the literature will fail to detect curved text accurately leading to a large number of false positives and hence poor accuracy. This shows that there is a great demand for robust systems that work for text in any orientation.

Hence, in this paper, we introduce a novel method based on the concepts of Mutual Direction Symmetry (MDS), Mutual Magnitude Symmetry (MMS) and Gradient Vector Symmetry (GVS) for each edge pixel in both the Sobel and Canny images of the input image. The method essentially chooses pairs of pixels that satisfy the above symmetry properties resulting in text pixel candidates. Then, local descriptors (SIFT) are employed to refine the text pixel candidates, resulting in text representatives for each text line. The combination of symmetry properties at the pixel level and local features at the text pixel candidate level proposed in this paper contributes to the detection of arbitrary texts while at the same time achieves better accuracy for horizontal and non-horizontal straight texts compared to the state of the art method. Besides, we introduce the first curved text dataset, namely CUTE80 that consists of 80 curved text images. With the publication of this paper, we intend to release this dataset with its ground truth to the public.

The remainder of this paper is structured as follows. The literature review is provided in Section 2. The novel combination of invariant features and local features for text detection is proposed in Section 3. Section 4 provides experimental results and performance evaluation. Lastly, conclusion and future works are discussed in Section 5.

## 2. Related work

Comprehensive surveys on text detection in scene images and video can be found in Jung et al. (2004) and Liang et al. (2005). Most existing methods of text detection in natural scene images and video can be classified roughly into three categories: texture based methods, region based methods and hybrid methods.

Texture based methods (Chen & Yuille, 2004; Fernandez-Caballero et al., 2012; Shivakumara, Dutta, Tan, & Pal, 2013; Shivakumara et al., 2011; Yao et al., 2012; Yi & Tian, 2011) usually treat the pattern of text appearance as a special texture. Techniques used in these methods include Fourier transform, wavelet decomposition, combination of wavelet and moments with the help of a classifier to classify text and non-text candidates. Chen and Yuille (2004) extracted 79 features for the text region given by a classifier and the method uses an adaptive binarization method to classify text and non-text pixels. Yi and Tian (2011) proposed a partitioning method using gradient and color information of pixels. It then uses features at the character level to study the regularity of texts to locate text in the images. The main problems of texture based methods lie in the large number of features that heavily depend on the classifier in use and the number of training samples. In addition, most of the methods focus on horizontal and non-horizontal straight lines but not curved lines. Shivakumara et al. (2013) proposed a combination of wavelet and median moments to identify text candidates at the block level followed by an angle projection boundary growing method to deal with multi-oriented text problem. The method is shown to work well for text detection in video as well as natural scene images. However, angle projection boundary growing assumes that text components in text lines are in one direction. As a result, the method is good for non-horizontal text lines but not text lines appearing in arc and ellipse shapes. In addition, the primary focus of that method is to detect text in video but not in scene images.

The next category is region-based methods (Neumann & Matas, 2012; Phan, Shivakumara, & Tan, 2012; Shivakumara et al., 2013; Yao et al., 2012). These methods first identify text regions through edge detection or clustering followed by some heuristics to classify text components followed by a process to eliminate false positives. However, these methods heavily depend on heuristics and parameters setting. Epshtein, Ofek, and Wexler (2010) proposed an image operator called the Stroke Width Transform (SWT) on the Canny edge image to detect texts. This method looks for similar stroke widths to group text components and it studies the component properties to classify text components. Neumann and Matas (2012) exploited Maximally Stable Extremal Regions (MSER) to extract text components. The method uses geometrical properties of the components and a classifier to detect the text. Yao et al. (2012) proposed a method for arbitrary (non-horizontal straight lines but not curved text lines) text line detection. This method uses the SWT for extracting components and it studies the various features based on color, geometrical properties at the component level for classification of text and non-text components. To handle multi-oriented texts, this method uses the linearity of the text components (considering that characters in a text line have uniform orientation).

Phan et al. (2012) proposed the idea of symmetry using gradient vector flow (GVF) computed from the common information of the Sobel and the Canny edge images. Grouping of text components is done based on the components' geometrical properties and text verification is done using a classifier with training samples to achieve good accuracy. Du, Duan, and Ai (2012) developed a method based on context information of the text pixels to detect text in natural scene images. The context is studied using stroke properties and spatial distribution of the text line. The method uses an SVM classifier to learn the context and hence the performance of the method depends on the classifier and training samples. Yi and Tian (2013) proposed a method based on character appearance and structure modeling to detect texts in natural scene images. The method proposes a model by making use of corner and interest points detected by the Harris corner detector. This method also depends on the classifier in use and training samples to achieve good accuracy. Shi, Wang, Xiao, Zhang, and Gao (2013) proposed a graph model based on MSER and then used color and geometric features to classify text and non-text by minimizing a cost function. Meng and Song (2012) proposed two steps for text detection based on salient regions. However, the method has a problem when the text size is small, or the difference between background and text is small. González and Bergasa (2013) proposed a text reading algorithm for natural scene images using a set of gradient and geometric features of text and dynamic programming for characters recognition. Though the method does not depend on any classifier, it heavily depends on the shapes of the characters since the proposed features require the complete shape of each text component. Koo and Kim (2013) proposed two classifiers utilizing MSER. The first classifier is used to classify connected components based on adjacency relationship, and the second classifier classifies text or non-text. Though they consider skew feature to determine adjacency relationship, it may not work for curved text. Sharma, Shivakumara, Pal, Blumenstein, and Tan (2012) and Shivakumara, Phan, Shijian, and Tan (2013) proposed a method by exploring gradient and gradient vector flow, respectively for arbitrary text detection from video without classifier's help. It is seen from Shivakumara et al. (2013) that the method gives poor accuracy for ICDAR data when the method uses strict measures at the word level for text detection from natural scene images. On the other hand, Sharma et al. (2012) requires a classification algorithm for separating horizontal text from other data and the proposed growing method is sensitive to character touching, small fonts, noise etc. In addition, the objective of the method is to detect text in video but not in scene images. The third category, namely hybrid methods, (Pan et al., 2011) proposes both texture and region based methods for text detection. Recently, Opitz, Diem, Fiel, Kleber, and Sablatnig (2014) proposed end-to-end text recognition using local ternary patterns, MSER and deep convolutional nets. The text detection system consists of a sliding window classifier which creates confidence maps. Then MSERs are extracted and classified as text or non-text. The remaining candidate characters are grouped based on the classifier's results to obtain words. The method is limited to horizontal text detection and the extracted features are sensitive to rotation and scaling. Bai, Yin, and Liu (2014) proposed a two level algorithm for text detection in natural scene images using connected component analysis and an SVM classifier. The method classifies components into four classes, namely, text, non-text, probable text and undetermined connected components. The conditional random field is used for the final decision. The method may not be robust to noise and disconnections. Rong, Suyu, and Shi (2014) proposed a method for scene text extraction based on seed-based segmentation. This method explores stroke width and gradient information to detect text polarity and then based on polarity the method extracts foreground and background seeds for text detection. However, the method is tested only on a small set of images.

With this, it is observed that most of the existing methods focus on horizontal text detection and a few of them focus on non-horizontal straight text lines. In all these methods, the components in a text line have the same orientations (maintaining a linear correlation between components). However, on any arbitrary text lines (such as arc, circular, S, or Z shaped text lines) such a characteristics is no more true as character components on a curved line have different orientations. This difference in orientation makes the text detection problem more challenging and demands a new robust text detection technique. We also note that scene text data do not necessarily appear in horizontal and non-horizontal straight lines, and they can have texts in circular or other curved alignment. The clear differences among horizontal, non-horizontal and curved text is reported in Sharma et al. (2012). For instance, in "STARBUCKS" – the name of a coffee house chain, one can see that the text is always in a circular form, embedded in a complex

background. Therefore, arbitrary text detection in natural scene images is much more complex than horizontal and non-horizontal straight text detection. The methods presented in Pan et al. (2011) and Yao et al. (2012) exploit linearity (regularity in direction of each components) in arbitrary text lines using a large number of heuristics with parameters at the component level which may limit its ability to work with arbitrary texts. Hence, our proposed method which falls in the region based category aims to detect arbitrary text lines in natural scene images without sacrificing the horizontal and non-horizontal text accuracy.

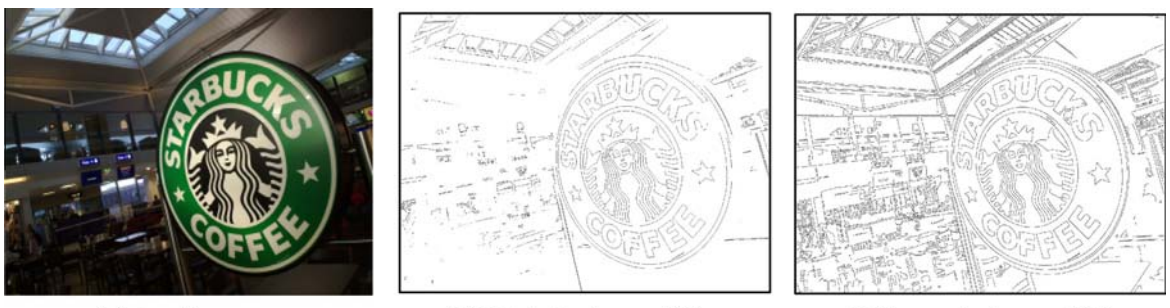
In summary, the proposed robust system has the following advantages. The proposed method works based on the fact that the text patterns in both Sobel and Canny images share the same properties, while non-text patterns do not. This is the main basis to propose three novel features in this work, namely, Mutual Direction Symmetry (MDS), Mutual Magnitude Symmetry (MMS) and Gradient Vector Symmetry (GVS) to extract the common patterns of the text. This results in text candidates. The way we integrate these three features in a novel way eliminates most of non-text components despite complex background. The use of local descriptors (SIFT) in a different way to eliminate false text candidates given by the integrated features provides at least one representative for each line in the images. The growing methods based on ellipse properties of text components extracts a full text line of any orientation because this growing process works based on a nearest neighbor criterion. The main advantage of this growing is that it restores the full text line with one representative of the text line because while growing it makes reference to the Sobel edge image of the input image. Since growing works based on the nearest neighbor criterion, it extracts arbitrary text line, which is hardly addressed in the literature. Further, the method is capable of detecting multi-lingual text because the method does not involve any language specific features and classifiers.

## 3. Proposed method

The work presented in Phan et al. (2012) shows that the text pattern in both the Sobel and Canny edge images of the input image exhibit the same properties (e.g. symmetry) while non-text in the image does not exhibit the same properties due to background complexity where the Sobel and Canny operators give different edge patterns. For example, for the input image shown in Fig. 2(a), both the Sobel and Canny edge images shown respectively in Fig. 2(b) and (c) give uniform edge pattern for text components (foreground) but non-uniform edge pattern for non-text components (background). It is also observed that the Sobel operator gives few pixels for the background while the Canny operator gives a lot of edges with different patterns for the same background. Hence the combination of Sobel and Canny operations aid in detecting both low and high contrast text without losing much text information. Therefore, the method in Phan et al. (2012) achieves the best recall compared to the state-of-the-art methods. However, it is noted from the experimental results of method (Phan et al., 2012), the precision is lower than the existing methods because the proposed GVF symmetry alone is not sufficient to eliminate non-text pixels as non-text. Besides, the scope of the method is limited to horizontal text detection but not arbitrary-oriented or curved text detection. Hence this work aims to overcome these drawbacks by making use of the same basis that was proposed in Phan et al. (2012) and proposing new features that can work regardless of the text orientation to achieve better accuracy.

Inspired by the Stroke Width Transform (SWT) Epshtein et al., 2010 which selects text components based on the gradient direction of each pixel, we extend the same idea in a novel way to propose three different symmetry properties to separate text pixels from the non-text pixels in edge images of the input image. In particular, the main idea of SWT is to identify text components but not text pixels by transforming input image to stroke width image. Though SWT is invariant to the orientation of the text, it requires additional features to handle curved text as stated in Yao et al. (2012). Therefore, the SWT is extended to non-horizontal straight text lines detection in Yao et al. (2012) where the authors proposed additional features based on linearity of text components in text lines. Unfortunately, this method is limited to non-horizontal straight text detection but not curved text. This is because the linearity of text components cannot be maintained due to varying orientation of character components in a curved text line. It has been shown that the combination of stroke width with gradient magnitude proposed in Pan et al. (2011) for text detection is good for classifying text and non-text components. However, experimental results have shown that the proposed features are not sufficient to overcome the problems of scene text detection especially the problems of curved text detection despite the feature are invariant to rotation.

In this work, the method finds pairs of pixels by making use the concept of SWT and then checks whether these two pixels satisfy MDS property. For instance, let $p_i$ be an edge pixel on stroke as shown in Fig. 3(a) with green color, from which, the method traverses in perpendicular direction to stroke direction until it reaches edge pixel, say q as shown in Fig. 3(a) with green color on another stroke. The distance between pi and q gives stroke width distance as suggested in Epshtein et al. (2010), which we called pair of pixels in the proposed work. With this, to define MDS, the method considers location of q and then it traverses in perpendicular direction to stroke direction of pixel q until it reaches a white pixel, say $p_j$ as shown in Fig. 3(a) with orange color. If the distance between pi and $p_j$ is less than three then the pair of pixels (p and q) is said to be satisfied MDS. Thus, the pair of pixels is considered as text pixel candidates. Note that, this is a departure from existing methods to address the curved scene text problem (more details can be found in Section 3.1).




(a) Input image (b) Sobel edge image of (a) (c) Canny edge image of (a)

**Fig. 2.** Sobel and Canny edge maps for the input image.

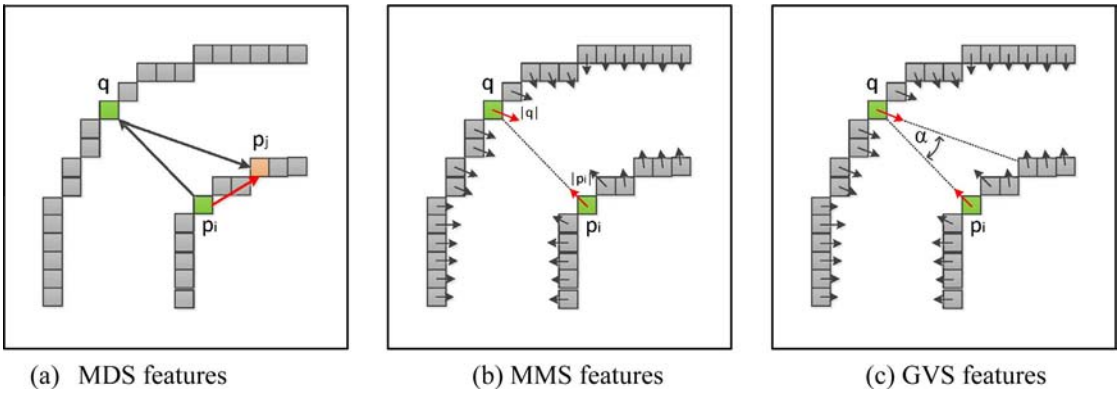



**Fig. 3.** Illustration for three symmetric features.

Motivated from the work in Pan et al. (2011) where gradient was explored along with Stroke Width Transform for text detection, we propose a new Mutual Magnitude Symmetry (MMS) property for the selected pairs of pixels (more details in Section 3.1). Thus, the objective of MMS is to check whether the gradient magnitude of a pair of pixels satisfies Mutual Magnitude Symmetry property or not to separate text pixels from non-text pixels. For instance, the same pair of pixels ($p_i$ and $q$) as shown in Fig. 3(b) with green color as discussed above for MDS, the proposed method checks whether gradient magnitude of these two pixels are same or not with the margin of threshold 0.15 magnitude difference. If the magnitude difference is less than 0.15 then pixels are said to satisfy MMS and hence classified considered as text pixel candidates.

As we observed from Shivakumara et al. (2013) for arbitrary video text detection where they exploit GVF for identifying dominant text pixels, we are inspired by this observation to propose a new Gradient Vector Symmetry (GVS) property to separate text pixels from non-text pixels. This property checks whether each selected pair of pixels satisfies the gradient vector flow symmetry or not. For example, for the same pair of pixels ($p_i$ and q), the method finds gradient angle of $p_i$ and q pixel using gradient vector flow operation as shown in Fig. 3(c) where $\alpha$ denotes angle difference. More specifically, if the angle difference is less than $\pi/9$ then it is said that the pair of pixels satisfies gradient vector flow symmetry. Thus, the pair of pixels is declared as text pixel candidates. We will elaborate in more details for the MDS, MMS and GVS features in Section 3.1. Since the arbitrary text detection from natural scene images is a complex problem, we integrate the above three symmetry properties to classify text pixels accurately from the Sobel and Canny edge images of the input image, which results in text pixel candidates.

In order to refine the text candidates, we propose SIFT features in a different way. The SIFT features are used in several papers (Guo, Gurrin, Lao, Foley, & Smeaton, 2011; Smith et al., 2011; Zheng, Chen, Zhou, Gu, & Guan, 2011) for text recognition in scene images and they have shown that SIFT is useful for finding matches between the target and reference character images because of its invariance to scale, rotation, illumination, and viewpoint. On the other hand, it is known that applying SIFT for all pixels is expensive (Guo & J., 2011). Therefore, we propose SIFT features for the text candidates to identify genuine text and to remove false text candidates in this work. We called these features Local Descriptors (SIFT). The notion for applying SIFT is that the value of the descriptors for the text candidates is almost uniform while for non-text candidates the descriptors are different due to the presence of uniform background and neighboring information for text candidates and vice versa for non-text candidates. This observation motivates us to perform a variance operation on descriptors of each text candidate to remove false text candidates. We propose a grouping approach based on characteristics of ellipse, which grows along text direction in the Canny edge image of the input image to extract text lines. The overview of the proposed method is illustrated in Fig. 4.

### 3.1. Integration of symmetry properties

For the input image shown in Fig. 2(a), the method obtains Sobel and Canny edge images as shown in Fig. 2(b) and (c), respectively. From the Sobel edge image, pairs of text pixels are determined using the gradient direction of edge pixels as stated in Epshtein et al. (2010). For each pair of pixels, we test three symmetry properties to identify the text pixel candidates as illustrated in the following. The first property called Mutual Direction Symmetry (MDS) is illustrated in Fig. 5 where (a) shows a text region on the right side and a non-text region on the left side selected from the input image to demonstrate the usefulness of the MDS, (b) illustrates MDS symmetry property for the pair of text pixels marked in green color in the magnified text region as we discussed in the previous section. The method estimates the distance between $p_i$ and $p_j$. Empirically, if the distance is less than three pixels then it is said to be the pair of pixels that satisfies the MDS property. The effect of MDS on both selected text and non-text region is shown in Fig. 5(c) and (d) where one can notice that there are more text pixels (represented in green color) in (c) and fewer text pixels

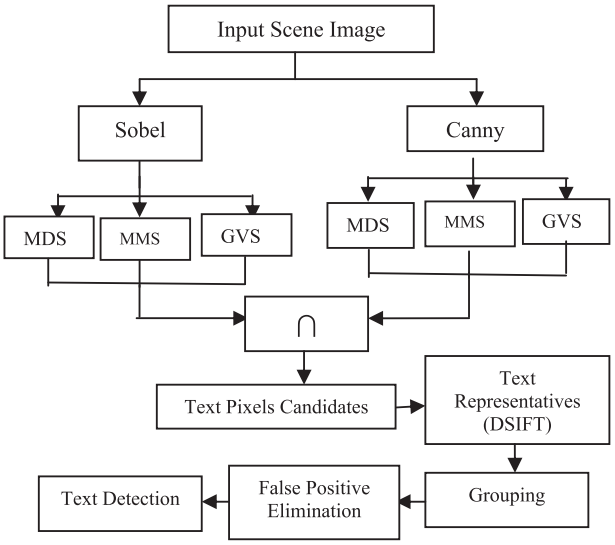



**Fig. 4.** Block diagram of the proposed method.

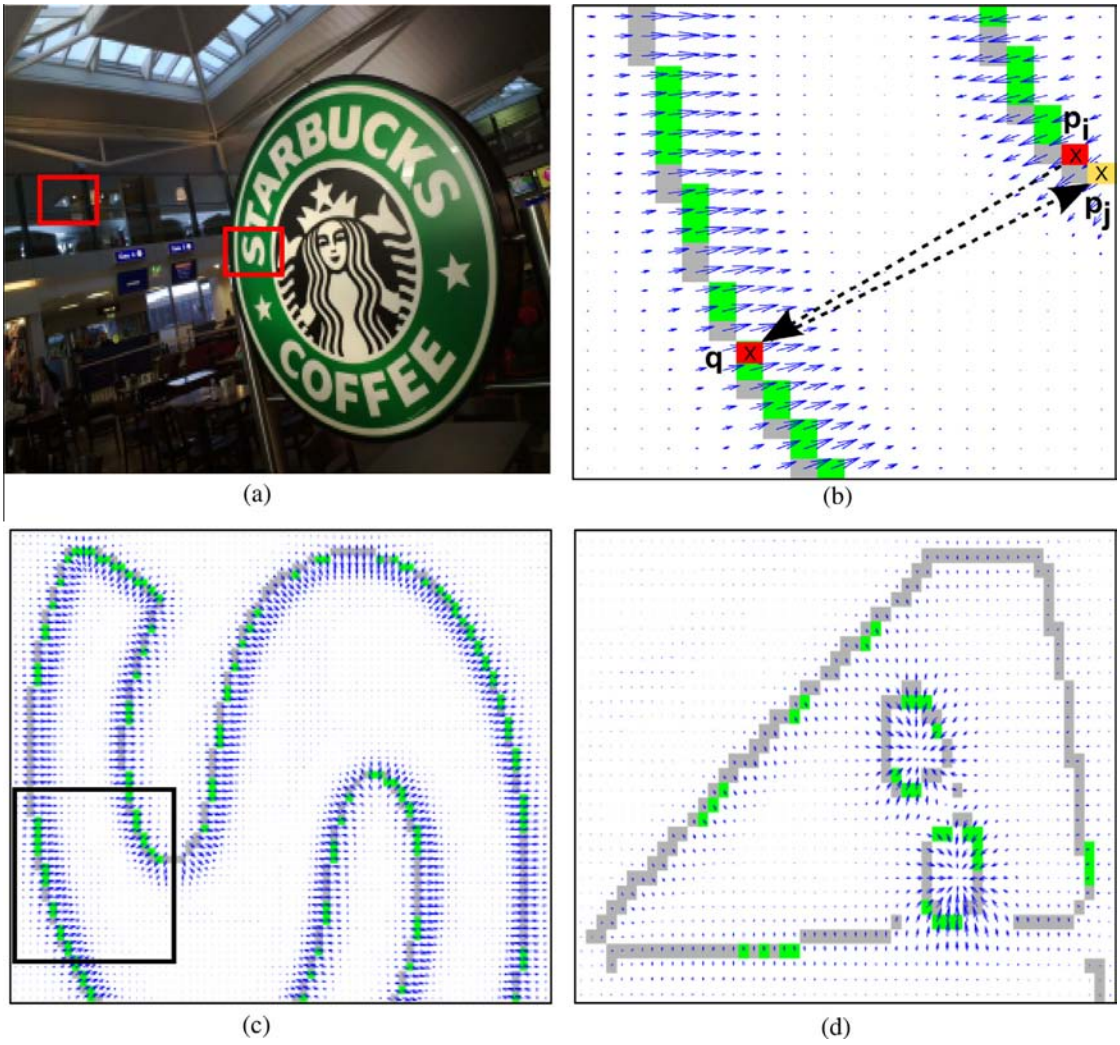



**Fig. 5.** Illustration of Mutual Direction Symmetry (MDS) property. (a) Sample text region marked by red color rectangle on the right (S) and non-text region on the left (Triangle shape) are Chosen, (b) MDS for selected pair of text pixels, (c) Magnified text region corresponding to text region in (a) where it shows more pairs of text pixels in green color, (d) Magnified non-text region where it shows few pairs of non-text pixels (false positives) in green color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(represented in green color) in (d). In this way, MDS helps in selecting text pixels which represent text information.

For the purpose of illustration of how the second property called Mutual Magnitude Symmetry (MMS) selects text pixels, Fig. 6(a) and (b) show the gradient magnitude values for the text region and non-text region (each marked by a rectangle), respectively. The gradient magnitude values in the rectangular region in Fig. 6(a) and (b) are magnified in Fig. 6(c) and (d), respectively. One can notice from Fig. 6(e) and (f) the magnitude differences among most of the text pixels (green color) are less than a threshold (set at 0.15) while the magnitude differences among most of the non-text pixels are greater than the 0.15 threshold. This is valid because when a pair of text pixels represents text pixels, then their magnitude must be close to each other. Please note that we have normalized the pixels gradient magnitude before taking the difference and the threshold value 0.15 is selected empirically and it is close to 0.

Both MDS and MMS use normal gradient to process, but GVS uses gradient vector flow (GVF) field. The GVF can be calculated as defined in Eq. (1) by minimizing the following energy function (Xu & Prince, 1998),

$$E = \int\int \mu(\nabla^2 g) + |\nabla f|^2 |g - \nabla f|^2 dxdy \quad (1)$$

where $g(x,y) = [u(x,y), v(x,y)]$ is the GVF field, while $\nabla f$ is the gradient of the edge map. The above equation smoothens the gradient $g$ while $\nabla f$ is small but makes $g$ equal to $\nabla f$ otherwise. The intuition is that we will get smooth gradient value compared to the normal gradient which has problem at corner points. However, we can get smooth gradient values using GVF even at the corners also. The Gradient Vector Symmetry (GVS) property selects pixels that satisfy symmetry using the gradient vector flow of each pair of pixels. The direction is roughly opposite if $d_p = -d_q \pm \pi/9$. Fig. 7(a) and (b) show text pixels classified by GVS property for both text and non-text region, respectively, where we can see more green color pixels denoting text in the text region while a few are misclassified as text (green color) in non-text pixels (green color). This shows that GVS helps in selecting text pixels correctly.

We integrate the above three symmetry properties to obtain text candidates that satisfy these three properties on each common pixel in the Sobel and Canny edge images. The integrated result is shown in Fig. 8 where it can be noticed that most of the non-text pixels are removed at the cost of few text pixels compared to the Sobel and Canny images shown in Fig. 4(b) and (c), respectively. This outputs the text candidates. In addition, the integration of MDS and MMS help to improve precision while GVS helps to improve recall, as we will show in the experiment results. Then,

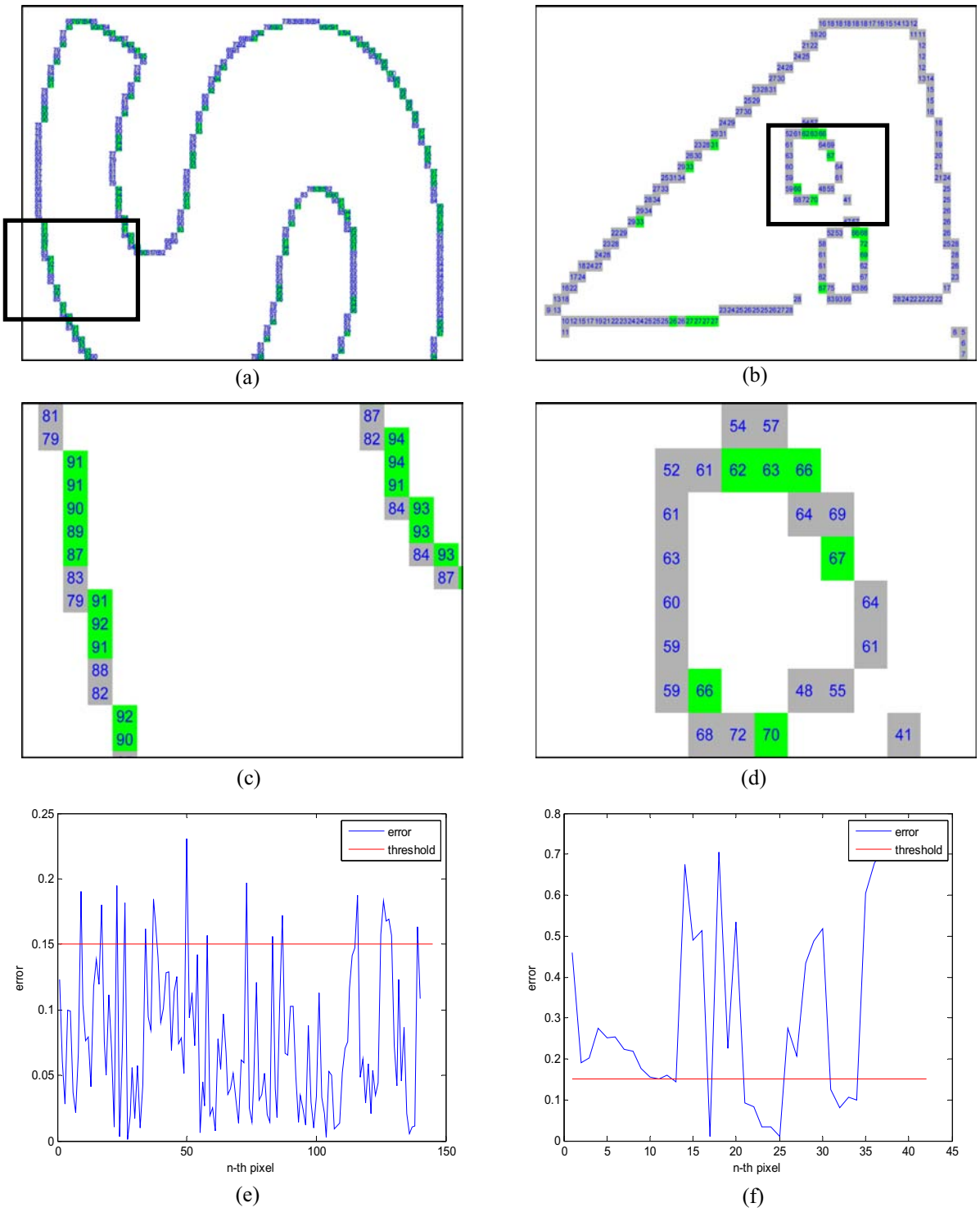
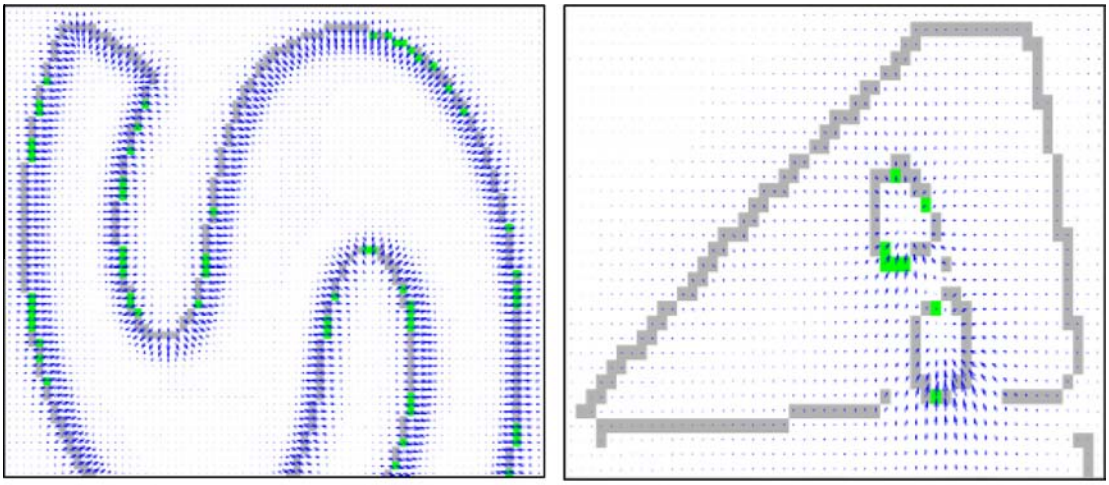



**Fig. 6.** Illustration of Mutual Magnitude Symmetry (MMS) property. (a) Small portion of text region marked by rectangle region is chosen, (b) Small portion of non-text region marked by rectangle is chosen, (c) Gradient magnitude for the pair of text pixels shown in green color in (a) and (d) Gradient magnitude for the pair of non-text pixels (false positives) shown in green color in (b) and (e) The magnitude difference of the pair of text pixels with 0.15 threshold and (f) The magnitude difference of non-text pixels (false positives) with threshold 0.15. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

we next propose a new feature based on SIFT to remove false text pixel candidates.

### 3.2. Refinement based on local features

For each pixel candidate in Fig. 8, we extract SIFT descriptors from every local fixed patch of size $31 \times 31$ pixels. Local patches are the areas of grey input image centered at text pixel candidates. With this, every pixel of the patch will be represented by a 128 dimensional descriptors. We ignore zero values to estimate the variance as defined in Eq. (2) for each candidate. It is observed that the value of the text-patch descriptors do not have much variations while the value of the non-text-patch descriptors have large variations due to homogeneous background of text pixel candidates and



(a) Pair of text pixels that satisfies GVS (b) Pair of non-text pixels (false positives) that satisfies GVS

**Fig. 7.** Illustration of Gradient Vector Symmetry (GVS) property.

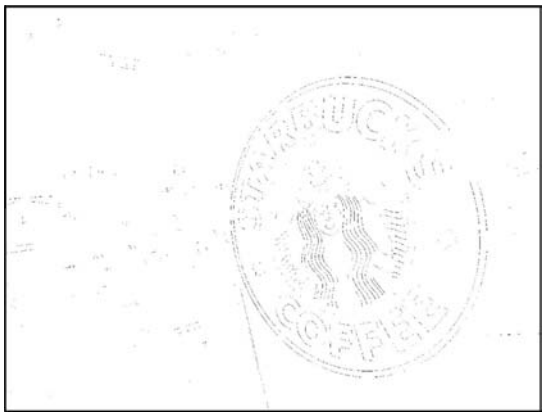


**Fig. 8.** Effect of integration of three symmetry properties for text pixel candidates.

vice versa for non-text pixel candidates. Therefore, we calculate variance for the descriptors of both text and non-text patch. More specifically, to classify text patch from non-text patch, we employ k-means clustering algorithm with $k = 2$ as defined in Eq. (3). Since it is an unsupervised clustering algorithm, we consider the cluster which gives the lower mean variance between the two clusters as the text cluster. In other words, variance of descriptors of text patch and non-text patch are the input to k-means clustering algorithm to identify accurate text pixel candidates. The result of k-means clustering algorithm to separate the text candidates is shown in Fig. 9(a) where green color pixels denote text candidates and red color pixels denote false text candidates (non-text candidates).

$$\sigma_i^2 = \frac{\sum_{j=1}^{m}(d_j - \mu_i)^2}{m} \tag{2}$$

$$\min \sum_{k=1}^{K}\sum_{i=1}^{n}\|\sigma_i^2 - \mu_k\|^2 \tag{3}$$

where $d_j$ is the descriptors value, $m$ is the number of descriptors after zero removal, $\sigma_i^2$ is the variance of the $i$th text pixels candidate. Fig. 9(b) shows text and non-text pixel candidates marked by rectangles which are denoted by (t) and (nt), respectively. For the descriptors of text and non-text pixel candidates in rectangle region shown in Fig. 9(b), variance is calculated. The variance values for text and non-text region are plotted in Fig. 9(c) and (d), respectively, where we can see that the $y$ axis scale for text pixel candidates varies from 1400 to 2800 and for non-text pixel candidates it $y$ axis scale varies from 1500 to 3800. This infers that SIFT descriptors give low variance values for text pixel candidates and high variance values for non-text pixel candidates. Note that the $x$-axis and $y$-axis of the results in Fig. 9(a) and (b) are similar with the pixel position representing column and row. For each candidate after refining by SIFT, in order to restore edges of components which are required to study geometrical properties of characters, we extract edge components in the Canny edge image corresponding to pixel text candidates as shown in Fig. 9(e). Note that false text components in Fig. 9(e) are reduced compared to the Canny image shown in Fig. 2(c). We call the results in Fig. 9(e) as text representatives for text lines.

The main basis for proposing three proposed symmetry features for text pixel candidates identification is the inadequacy of the Stroke Width Transform (SWT) which explores gradient direction to identify text components (Epshtein et al., 2010). To show that SWT alone is not sufficient for text detection especially text like curved shape, we compare the SWT results with the proposed features results for the image shown in Fig. 2(a). The results are shown in Fig. 10 where (a) shows the results of the proposed features after refinement method and (b) shows the results of SWT. For a fair comparison, the proposed method is computed using the three features (MDS, MMS, and GVS) and refinement using SIFT and k-means, but without false positives elimination. On the other hand, the SWT connected components was computed from SWT image using a simple rule that the ratio of SWT values of neighboring pixels is less than 3.0 which is suggested by Yao et al. (2012), with the additional component filtering, that is, width variation (the ratio of standard deviation and mean of stroke width), aspect ratio (the minimum ratio of the component width/height or height/width), and occupation ratio (the ratio of the number of component pixels and bounding box area). As we can see in Fig. 10, although the proposed method has few missing text characters, it is able to remove most of the non-text components as compared with SWT. Thus, we can infer that SWT is alone is not sufficient for curved text detection.

### 3.3. Ellipse growing for grouping text components

For every text representative found in Fig. 9(e), the method next constructs an ellipse to group the text representative. This differs from existing methods which fix a rectangular bounding box to study the characteristics of text components. For each constructed

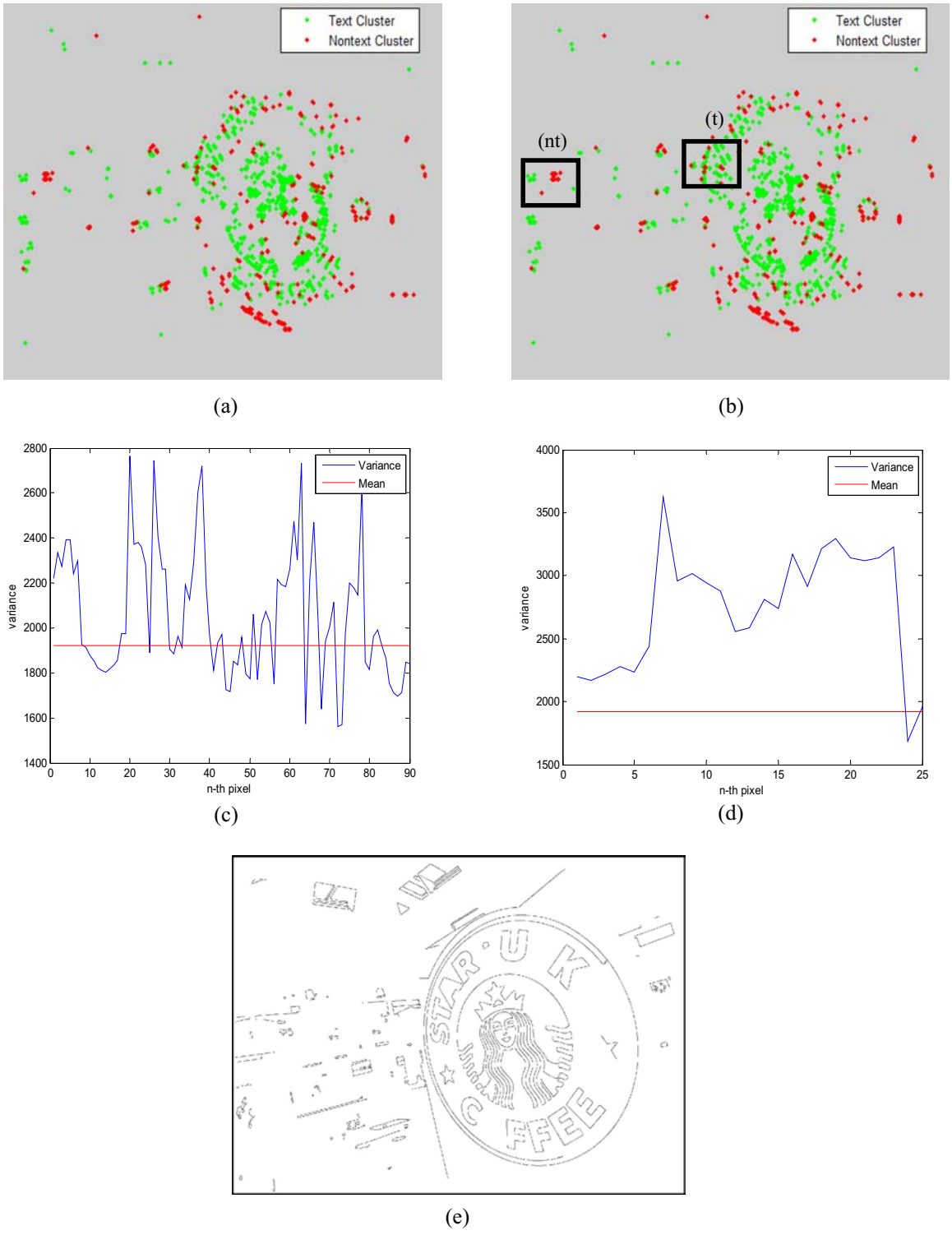
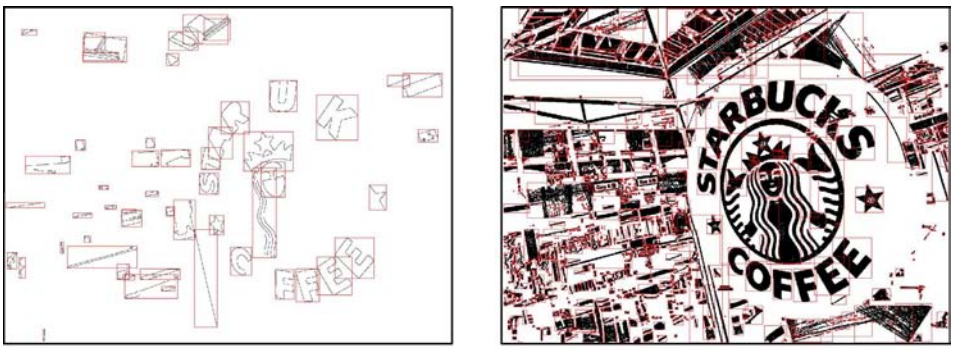



**Fig. 9.** Text representatives selection using SIFT descriptors (SIFT). (a) Result of K-means clustering on variance of SIFT, (b). Sample text region (t) on the right side and non-text region (nt) on the left side marked by rectangle is chosen, (c) Variance of SIFT for the text region in (b), (d) Variance of SIFT for non-text region (b) and (e) Text representatives.

ellipse, we find its major (A) and minor axis (a) in order to study its geometrical properties. We construct an ellipse by incrementally growing its major or minor axis pixel by pixel until it finds the nearest neighbor component along the text direction in the Canny edge image. The main advantage of this grouping is that it is not constrained by the direction of the text line, therefore and hence it is able to handle arbitrary text lines. Essentially, it works based on the fact that the proximity between character components is closer than the proximity between words and text lines. Initially, a random text component $c_i$ will be chosen, and we grow the ellipse by a unit pixel, on both the $A(c_i)$ denotes major axis and $a(c_i)$ denotes minor axis at a time until it touches another text component $c_j$. This process will continue till the end of the text. For instance, for the character component "S" in Fig. 11(a), the growing algorithm fixes an ellipse and then the ellipse grows until it reaches the character "T" since character "T" is next to the

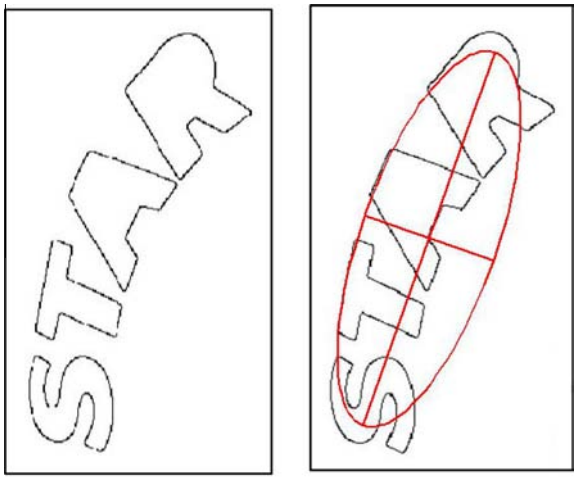



(a) Proposed features (b) SWT [23]

**Fig. 10.** Comparison of the proposed method with Stroke Width Transform (SWT).

character "S". This process continues till it reaches the character "R". Then it merges ellipses of all the character components to obtain one ellipse for the word "STAR" as shown in Fig. 11(b). The result of grouping component ellipses can be seen in Fig. 11(b) where we can see both text and non-text components are grouped with their ellipses.

In Fig. 12, we give example for the growing process with the two types of starting points, that are, at the starting and middle of a text-line. For the first type (Fig. 12(a)), it will create an ellipse of a component and grow its major and minor axis till it finds nearest components (indicated as green character) and marked it as traversed as shown in first row of Fig. 12(a). The terminating condition is defined as the maximum number of iterations must be less than or equal to median of minimum bounding-box's width and height of the components. This condition is set based on the fact that the gap between characters is not more than the width of character. Then, it finds all the components along the text direction and stop when the maximum number of growing is reached and there is no nearby un-traversed components at the end of text line as shown in second row of Fig. 12(a). For the second type, the growing process starts traversing towards the left direction and then stop as shown in first row of Fig. 12(b) then it traverse towards right direction based on the nearest neighbor criterion along text direction from the middle component till it reaches the maximum number of iterations as shown in second row of Fig. 12(b).




(a) (b)

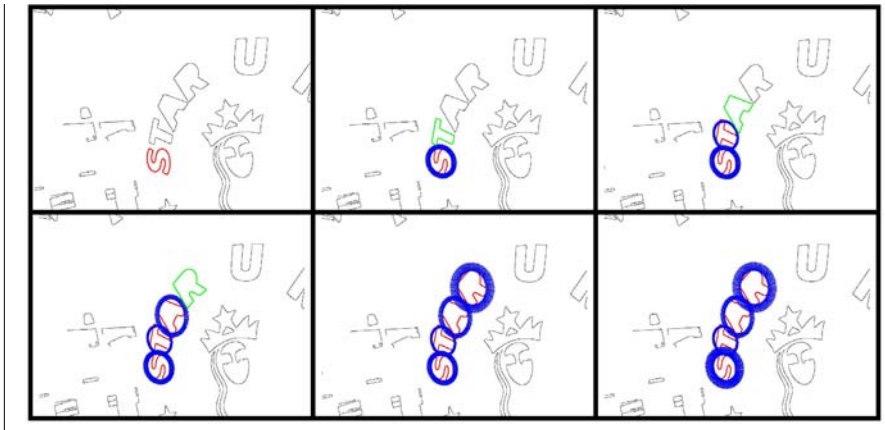
**Fig. 11.** Ellipse growing to group the text component. (a) Sample text components of the word "STAR" and (b) Ellipse for the whole word.

### 3.4. False positive elimination

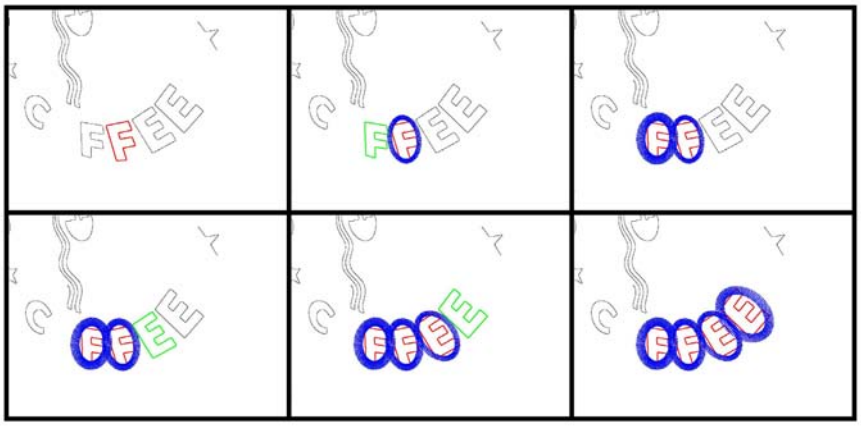
It is known that eliminating false positives completely is hard in the case of scene text images. Therefore, we propose the following objective heuristics for the purpose of eliminating false positives. (i) For each group of components, we compare the angle of the major axis given by the ellipse and the angle computed by the Principal Component Analysis (PCA). For a component, if the angles computed for both its ellipse's major axis and PCA are the same, then we will confirm that it is a text component or else discard it as a non-text component. It is illustrated in Fig. 13 where we can see in the first row the angle of the major axis and PCA are not the same and hence it is eliminated. In the second row of Fig. 13 where the angles of PCA and the major axis of the components are the same and hence it is considered as text. (ii) We divide the group of components into two equal sub-groups if the ellipse covers a word with greater than or equal to four components as shown in the first row in Fig. 14. Thus we see two subgroups having separate ellipses with different colors as shown in the first row of Fig. 14. We check whether the minor axis, pixel distribution, direction, and standard deviation of the proximity matrix of each sub-group are similar or not. Proximity is the distance between pixels. Table 1 shows the definition of each condition. If the component subgroups satisfy $minorLen(c_i) < 0.3$, $numPix(c_i) < 0.4$, $dir(c_i) < \cos(50^\circ)$, and $std(c_i) < 0.15$ then we consider them as text components or else non-text components. In Fig. 14, the first row shows the two sub groups of ellipses satisfy the above rules and hence it is considered as text component. On the other hand, in Fig. 14 the second row shows that the two sub-groups do not satisfy the rules defined in Table 1 and hence it is discarded as false positives.

where $c_i$ is i-th group of components, $a_i^1$, $a_i^2$ are the minor axis lengths of subgroups 1 and 2, respectively, the number of pixels in subgroups 1 and 2 are denoted as $np_i^1$, $np_i^2$, the major axis directions of subgroup 1 and 2 are denoted as $dirA_i^1$, $dirA_i^2$, and the standard deviation of proximity matrix of subgroups 1 and 2 are denoted as $stdProx_i^1$, $stdProx_i^2$. Standard deviation proximity matrix is computed by taking the matrix into one column and computed the standard deviation.

We also defined a condition using the number of end points to eliminate false positives such as tree and grass structure components. Firstly, we perform morphological thinning and remove isolated pixels (pixels that have only 1 neighborhood pixel) in the group of components. Then, secondly, we compute the number of endpoints (e.g., a line has two endpoints). If the number of components minus the number of endpoints is less than −15 then we remove this group of components. It is illustrated in Fig. 15 where one can notice that the first component contains a large number of end points but text components have less number of end points.



(a) Ellipse growing for grouping with the starting component at the start of text line



(b) Ellipse growing for grouping with the starting component at the middle of text line

**Fig. 12.** Ellipse growing with different starting points.

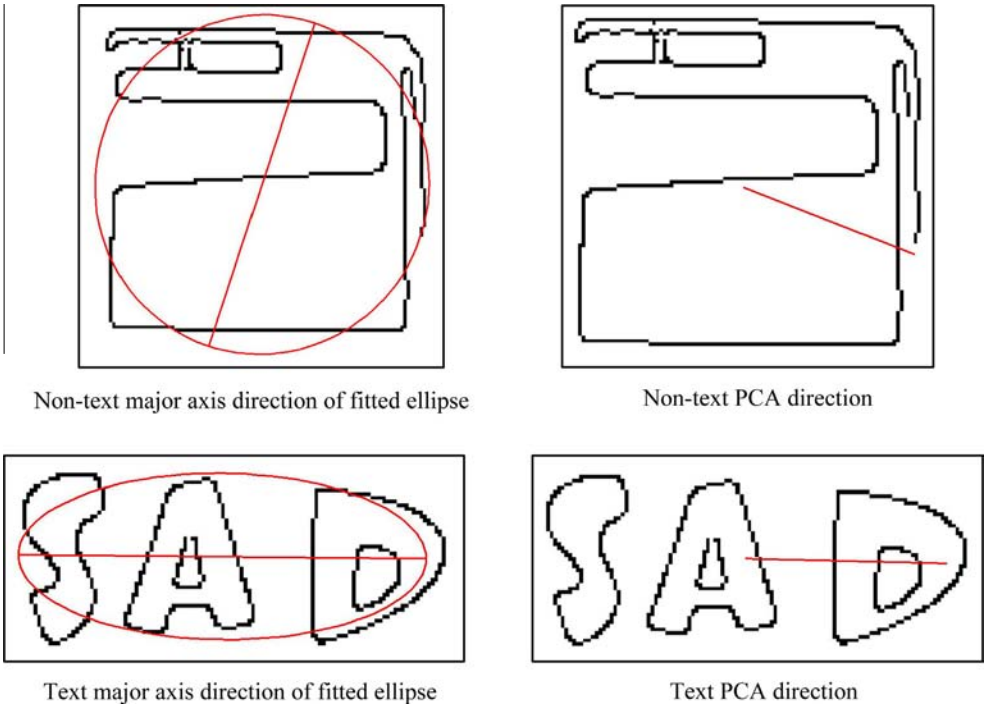
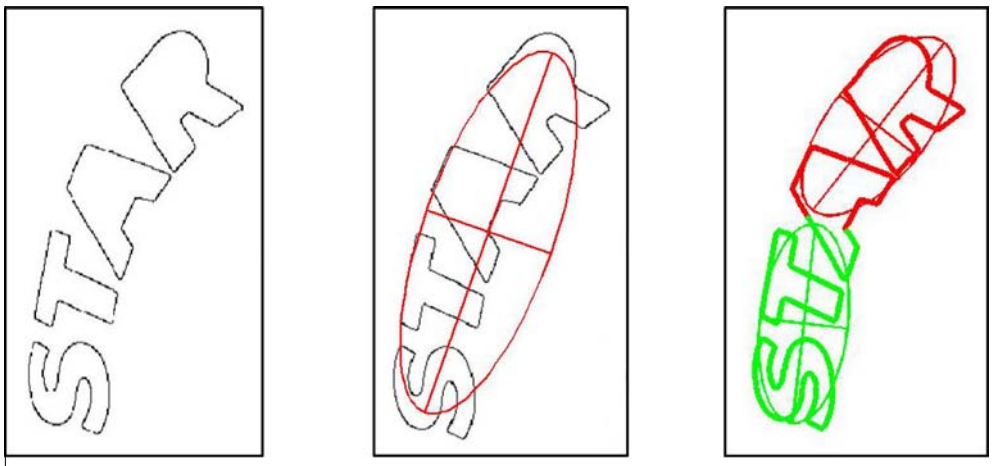



**Fig. 13.** PCA and major axis direction for text and non-text components.



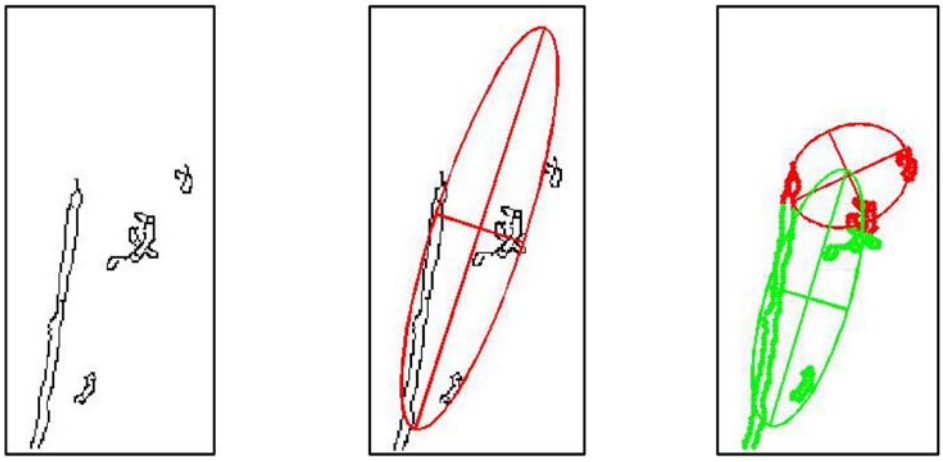




**Fig. 14.** Sub groups for text and non-text components.

**Table 1**
Rules for false positive elimination.

| Features | Definition |
|---|---|
| Minor axis length | $minorLen(c_i) = \lvert a_i^1 - a_i^2 \rvert / \max(a_i^1, a_i^2)$ |
| The number of pixels | $numPix(c_i) = \lvert np_i^1 - np_i^2 \rvert / \max(np_i^1, np_i^2)$ |
| Direction | $dir(c_i) = 1 - \frac{\lvert dirA_i^1 . dirA_i^2 \rvert}{\lvert dirA_i^1 \rvert \lvert dirA_i^2 \rvert}$ |
| Standard deviation of proximity matrix | $std(c_i) = \frac{\lvert stdProx_i^1 - stdProx_i^2 \rvert}{\max(stdProx_i^1, stdProx_i^1)}$ |

If a group contains less than four components then we check whether these components have close contour or not for eliminating false positives. Closed contour components can be checked using the number of endpoints of a contour, ensuring that it has less than two end points. This condition is based on the observation that characters usually have closed contours. It is illustrated in Fig. 16 where the left result shows non-text components as they contain more end points and the right result shows text components as they do not contain more than two end points.

Finally, the result of ellipse grouping and the result of false positive elimination are shown in Fig. 17 where (a) shows how grouping is done by ellipse growing and (b) shows the final text components after eliminating false positives using the above objective heuristics.

### 3.5. Text restoration and verification

At this point, it is noted from Fig. 17(b) that some text components are missing in each word. To restore the missing text components, we grow ellipses of the text components again along the text direction in the Canny edge image of the input image. Before restoring, the method finds the direction of the components using the major axis of the ellipse and then it grows along the major axis direction to group the components. While grouping, the method checks direction, color similarity, direction and mean stroke width as defined in Table 2 before restoring the missing text components and merging them. The direction and its threshold are defined in Fig. 18 to restore missing text components. As a result, Fig. 19(a) shows how the missing characters "B and C" are restored and the word "BUCKS" is merged with "STAR" after verification with the properties of the word "STAR" as shown in Fig. 19(b). The same procedure is used to restore the missing character "O" in the word "COFFEE" in Fig. 19(a) as shown in Fig. 19(b). The final output of curved text detection is shown in Fig. 19(c) where two bounding boxes are constructed for the two words.

where $\overline{g\_im}(\boldsymbol{c}_{cur})$ and $\overline{g_{im}}(c')$ are the mean from the gray image of the current group of components $\boldsymbol{c}_{cur}$ and a new component $c'$, respectively. The example of $\boldsymbol{c}_{cur}$ is depicted in Fig. 18 as a group of characters 'DPO' and a new component $c'$ is depicted as the

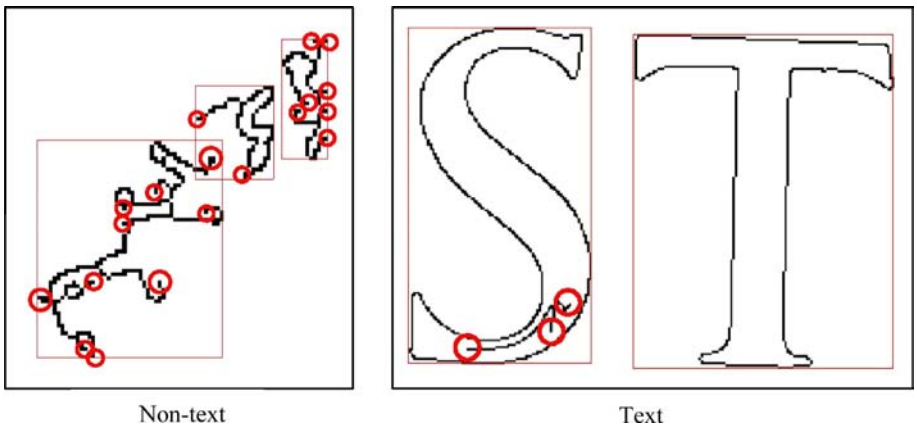



**Fig. 15.** Removing non-text group that have many endpoints. Red rectangles indicate component's bounding box, and red circles indicate endpoints. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

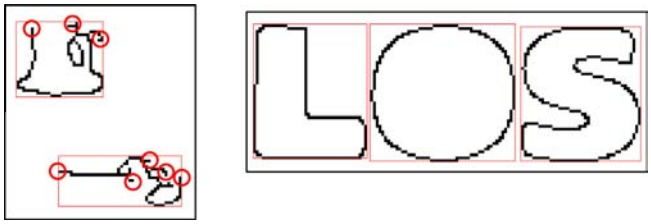


**Fig. 16.** Removing non-text group if the number of components in a group is less than 4 and it is not a closed component.

**Table 2**
Rules for words verification while restoring and merging.

| Features | Definition |
|---|---|
| Color similarity | $\overline{c\_im}_{eg}(\boldsymbol{c}_{cur}, c') = \lvert\overline{g_{im}}(\boldsymbol{c}_{cur}) - \overline{g_{im}}(c')\rvert/255$ |
| Direction | $dir_{eg}(c_{cur}, c') = 1 - \frac{\vec{a}.\vec{b}}{\lvert\vec{a}\rvert\lvert\vec{b}\rvert}$ |
| Mean stroke width | $\overline{sw}_{eg}(\boldsymbol{c}_{cur}, c') = \lvert\overline{sw}_{\boldsymbol{c}_{cur}} - \overline{sw}_{c'}\rvert/\overline{sw}_{\boldsymbol{c}_{cur}}$ |

character 'G'. Fig. 18 also illustrates how to compute the direction. Similarly, the mean stroke width is computed using the mean of stroke width of the current group of components $\boldsymbol{c}_{cur}$ and the new component $c'$. Any components which are not satisfying the conditions, $\overline{c\_im}_{eg}(\boldsymbol{c}_{cur}, c') < 0.2,\ 0 \leqslant dir_{eg}(\boldsymbol{c}_{cur}, c') \leqslant (Th = \cos 50^{\circ})$, and $\overline{sw}_{eg}(\boldsymbol{c}_{cur}, c') < 0.5$, are discarded.

## 4. Experimental results

There are several standard datasets for scene text detection available publicly, namely, ICDAR2003 (Lucas et al., 2003), ICDAR2005 (Lucas, 2005), ICDAR2011 (Shahab, Shafait, & Dengel, 2011) robust reading competition data, Street View Text data (SVT) Wang & Belongie, 2010, KAIST scene text data (Lee, Cho, Jung, & Kim, 2010), Microsoft data (Epshtein et al., 2010), Oriented Scene Text Database (OSTD) Yi & Tian, 2011 and MSRA Text Detection 500 data (MSRA-TD500) (Yao et al., 2012). Out of these, we use ICDAR2005, ICDAR2011 and MSRA-TD500 data for experimentation and evaluation because ICDAR2005 and ICDAR2011 data are widely used for text detection compared to the other data while SVT data is a more specific dataset mostly containing street view images and it does not expect any algorithm to detect all the texts in the image. KAIST and OSTD data provide incomplete ground truth (e.g. for small text) though it includes non-horizontal text lines. On the other hand, MSRA-TD500 provides data with bounding box ground truth for non-horizontal text detection. However, none of the above datasets contains curved text lines. Therefore, we introduce the first curved text dataset to be made public, namely CUTE80 that consists of 80 curved text line images with complex background, perspective distortion effect and poor resolution effect (in circle, S, Z shaped text lines). CUTE80 is necessary in order to show the capability of the proposed method in handling curved texts. With the publication of this paper, we would make the dataset and ground truth publicly available. In summary, we use ICDAR2005, ICDAR2011, MSRA-TD500 and CUTE80 to evaluate the proposed method performance for horizontal, non-horizontal and curved text line detection, respectively. We believe that if

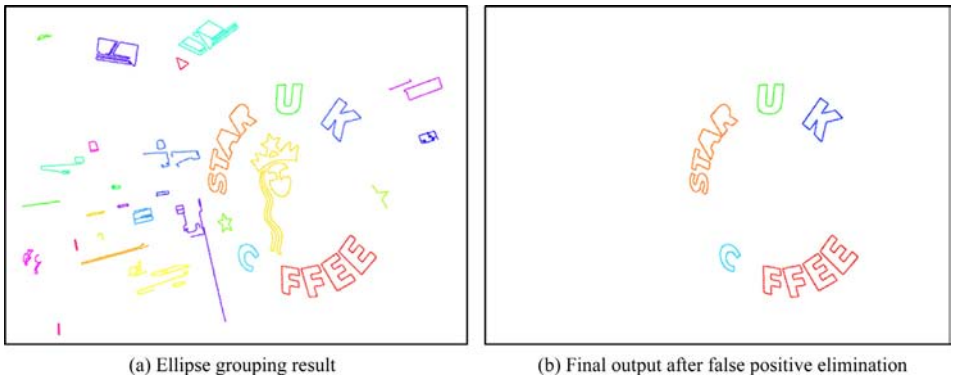



**Fig. 17.** Potential text components.

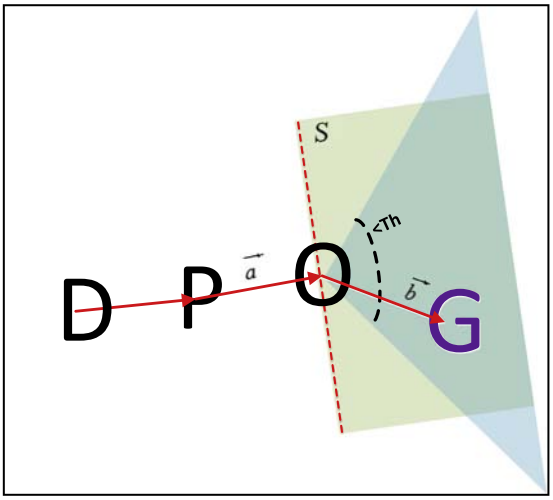



**Fig. 18.** Checking direction during ellipse growing is done utilizing the nearest component 'PO' with the vector $\vec{a}$. In this example, the current group of components $\boldsymbol{c}_{ref}$ is represented by characters 'DPO' and a new component $c$'is character 'G' (purple color). The new character $c'$ is examined with the vector $\vec{b}$. The searching space is indicated by $S$ following the direction of vector $\vec{a}$. Thus any components that are not in the space and within threshold < Th are discarded. More specifically, the above procedure can be written as $0 \leqslant dir_{eg}(\boldsymbol{c}_{ref}, c') \leqslant Th$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the proposed method works for these complex datasets then the same method should work for other datasets with little modifications.

We conduct experiments to find optimal threshold values for the three proposed symmetric features (MDS, MMS and GVS) to find text pixel candidates which are illustrated in Fig. 20. We choose 100 images randomly from the training dataset and manually labeled every component that represents text from the extracted Canny edges. Then we run the proposed features separately to calculate recall, precision by varying their threshold values. We plot a graph for the recall and precision values vs. different threshold values for each feature as shown in Fig. 20(a)–(c), respectively. When recall and precision curve intersects, the corresponding threshold value is considered as optimal threshold values. For example, to find recall and precision of the MDS feature, we vary the MDS threshold without using MMS and GVS features, and similarly with MMS and GVS features.

Fig. 20 shows when recall and precision curves intersect while varying their threshold values that threshold value is considered as optimal threshold values since it provides a good balance between these two curves. As the results showed, we get MDS threshold = 3, MMS threshold = 0.15, and GVS threshold = $\pi/9$. We can also infer from Fig. 20 that MDS and MMS provide good precision while GVS provides good recall. Thus by combining the three features together, we are able to achieve good precision and recall.

### 4.1. Dataset and evaluation

In this section, we introduce our dataset and evaluation method. We name our dataset CUTE80 since it contains 80 images of curved text. These images are either indoor or outdoor images captured with a digital camera or retrieved from the Internet. Example images are shown in Fig. 21. The ground truth is manually annotated containing a set of polygon points of the bounding box for each curved text line as shown in Fig. 21. The evaluation of each curved text line is performed by finding the minimum intersection area between the ground truth and the estimated polygon points of the curved text line. For example, we have a set of polygon points of the $i$th text line from the ground truth $\boldsymbol{p}_i^g = \{p_1, p_2, \ldots, p_N\}$ and a set of estimated polygon points of the same text line computed by our proposed method $\boldsymbol{p}_i^e = \{p_1, p_2, \ldots, p_M\}$. The minimum intersection area $a_i$ is defined as follows,

$$a_i \triangleq \frac{area(\boldsymbol{p}_i^e)}{(area(\boldsymbol{p}_i^e) \cup area(\boldsymbol{p}_i^g)) - (area(\boldsymbol{p}_i^e) \cap area(\boldsymbol{p}_i^g))} \tag{4}$$

The definitions of precision, recall, and F-score are as follows,

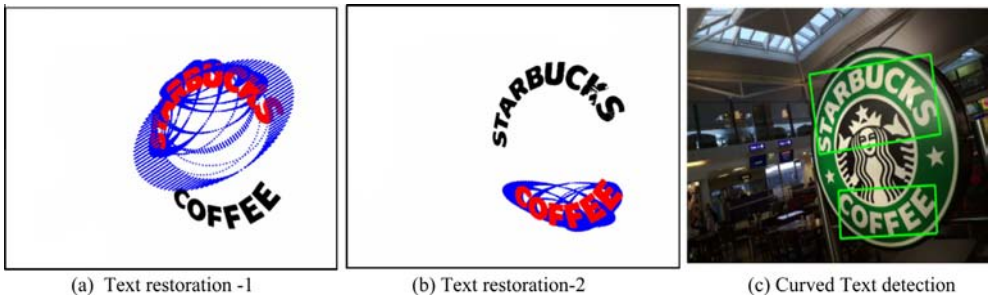
$$precision \triangleq \frac{\sum_i a_i}{|\boldsymbol{P}^e|} \tag{5}$$

$$recall \triangleq \frac{\sum_i a_i}{|\boldsymbol{P}^g|} \tag{6}$$

$$Fscore \triangleq \frac{2 \cdot precision \cdot recall}{(precision + recall)} \tag{7}$$

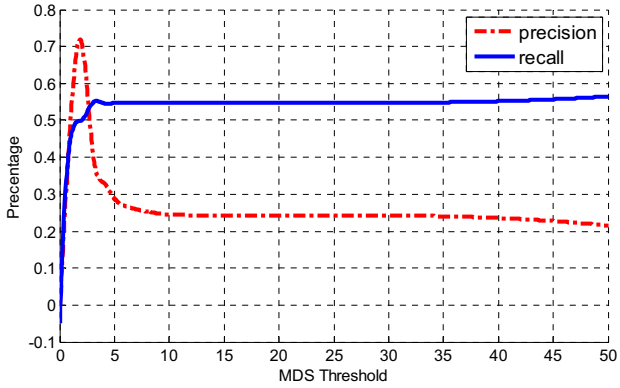
In ICDAR data, there are 251 images for testing and 258 images for training. We use the 258 training data for testing conditions and rules of the proposed algorithm and we use the 251 testing data for calculating recall, precision and f-measure. For evaluation on horizontal text detection, we follow the instructions given in Lucas (2005). Similarly, there are 300 images for training and 200 images for testing in the case of MSRA-TD500 data. Here, we use 200 images for evaluating the proposed method performance on non-horizontal text detection without using the training images. The definitions suggested in Yao et al. (2012) are used for calculating recall, precision and f-measure. For curved text detection, we use the 80 images from CUTE80 and Eqs. (5)–(7) for calculating recall, precision and f-measure. Note that since ICDAR performance measures are based on word level evaluation, we modify our method such that it segments text lines into words with the help of distances between words and characters during ellipse growing. For MSRA-TD500 and our data, we use the text line evaluation method suggested in Yao et al. (2012).

In order to find optical thresholds to define overlapping region, we conduct experiments on 50 samples chosen randomly from

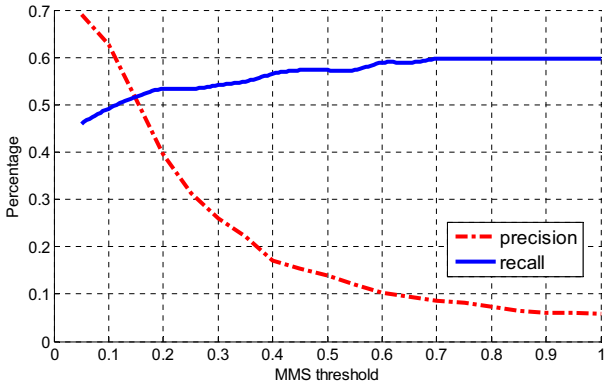



(a) Text restoration -1 (b) Text restoration-2 (c) Curved Text detection

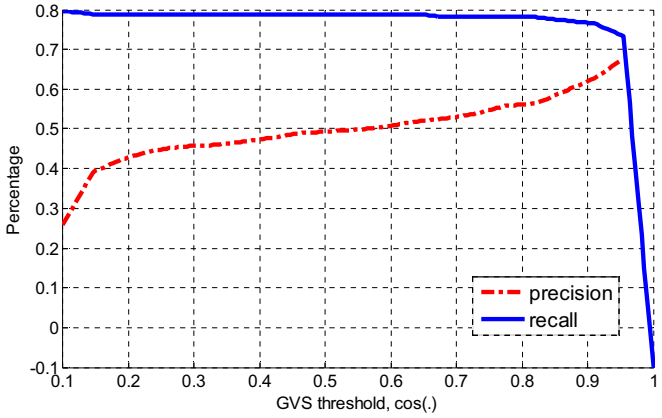
**Fig. 19.** Text restoration and verification using angle and spatial information.




(a) Recall and precision for MDS




(b) Recall and precision for MMS




(c) Recall and precision for GVS

**Fig. 20.** Individual MDS, MMS and GVS contribution by varying threshold values. *x* axis denotes threshold values and y axis denotes recall and precision values.


**Fig. 21.** Ground truth from our proposed dataset.

each database and the ROC curve for false positive rate vs recall rate are shown in Fig. 22 where varying the thresholds of $t_p$ and $t_r$ between range $[0.5, 0.7]$ suggests optimal results, while for other range, there is a significant drop in performance. Note that the different threshold values of precision $t_p \in [0, 1]$ and area recall $t_r \in [0, 1]$ are chosen according to the description in Lucas (2005). The same values are used for all experiments to calculate recall, precision and F-measure in this work.

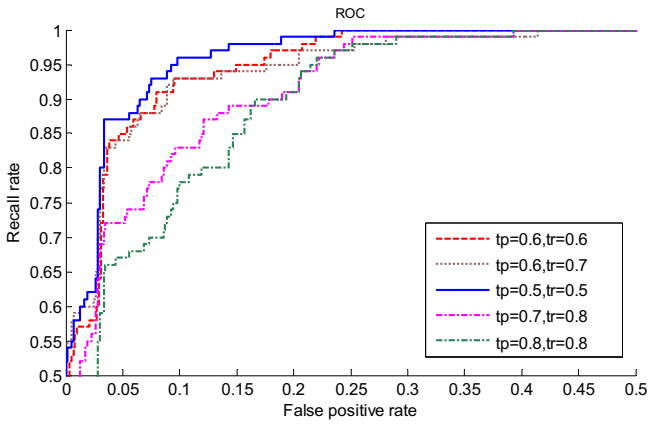



**Fig. 22.** Optimal threshold selection to define overlapping region.

### 4.2. ICDAR 2005

The qualitative results of the proposed method on ICDAR 2005 data are shown in Fig. 23 where one can see that the proposed method detects text well for the variety of images of different orientations, fonts and background. The quantitative results according to ICDAR 2005 instructions are reported in Table 3. Table 3 shows that the proposed method gives better F-score than existing methods. González and Bergasa (2013) gives the highest precision but the worst recall compared to the other methods. Phan et al. (2012) gives the highest recall compared to the other methods including our method. However, when we compare the F-score, our proposed method achieves the best F-measure, similar to Phan et al. (2012) and Neumann and Matas's method (2012). It is noted that all existing methods use a classifier and a large number of training samples to achieve better accuracy but the proposed method does not use any classifier and training samples to achieve the F-score. Thus, as a trade-off for not using classifier, the average time of the proposed method is lower than the state-of-the-art methods. However, it is worth noting that the complexity of the integration of the 3 proposed features (MDS, MMS, and GVS) is O(NM) which is linear on the number of edge pixels (N) and the distance (M) from one pixel to another pixel in perpendicular direction. In addition, the proposed method is able to detect horizontal, non-horizontal, and curved text lines.

### 4.3. ICDAR 2011

The quantitative results according to ICDAR 2011 instructions are reported in Table 4. In Table 4, there is a noticeable recall improvement in the proposed method and a comparable F-score as compared to the state-of-the-art methods. Though the overall performance improvement is small, this can be attributed to the generalization of the proposed method to account arbitrary orientation of text lines as compared to the other methods that is limited to a certain orientation. The average time of the proposed method is lower than the state-of-the-art methods, since the proposed method is not utilizing any classifier. Moreover, it has the advantage of not using overwhelming number of training samples. In addition, the proposed method is able to detect horizontal, non-horizontal, and curved text lines. Fig. 24 qualitatively demonstrates that the proposed method is able to detect horizontal text lines in scene images.

### 4.4. MSRA-TD500

The qualitative results of the proposed method on MSRA-TD500 data are shown in Fig. 25 where it can be seen that the proposed

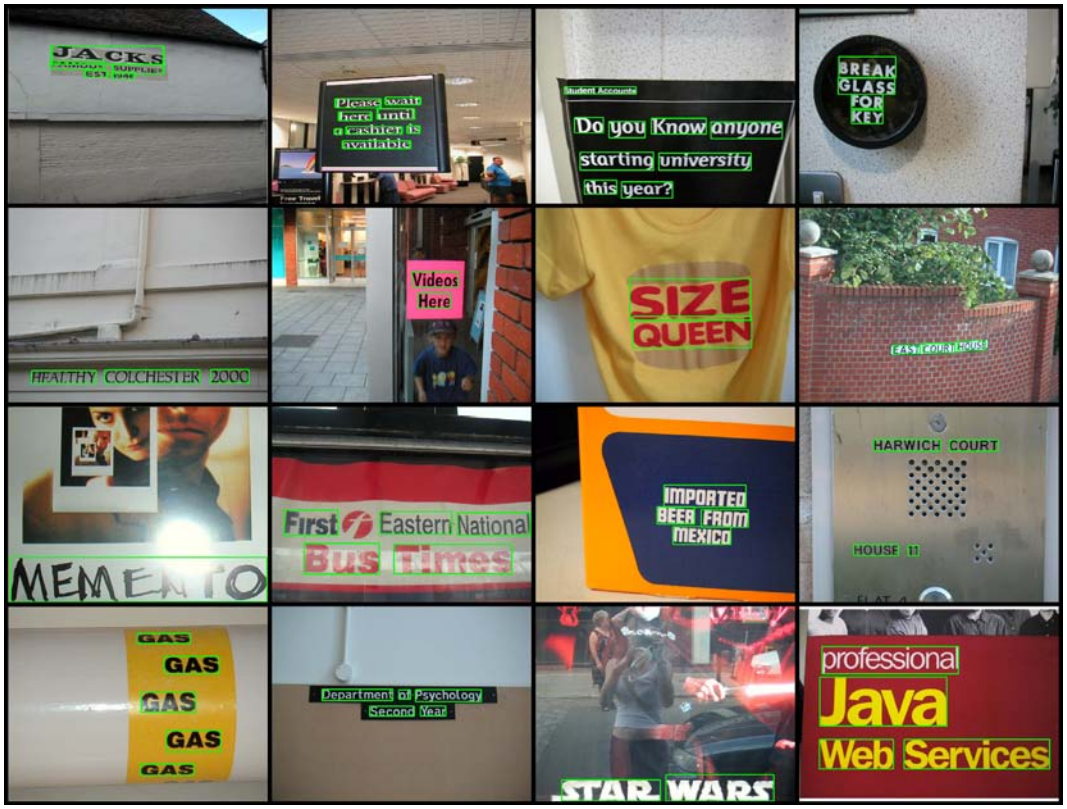



**Fig. 23.** Sample results of the proposed method for horizontal text detection (ICDAR 2005 dataset).

**Table 3**
Performance of the proposed and existing methods on ICDAR 2005 dataset.

| Methods | Precision | Recall | F-score | Aveg. time (s) |
|---|---|---|---|---|
| Proposed method | 0.76 | 0.63 | **0.69** | 15.8 |
| Shivakumara et al. (2013) | 0.74 | 0.62 | 0.68 | 12.7 |
| Du et al. (2012) | 0.74 | 0.61 | 0.67 | 0.55 |
| Yi and Tian (2013) | 0.71 | 0.62 | 0.63 | 16.2 |
| González and Bergasa (2013) | **0.81** | 0.57 | 0.67 | – |
| Phan et al. (2012) – HOG | 0.70 | 0.69 | **0.69** | – |
| Phan et al. (2012) – without HOG | 0.63 | 0.69 | 0.66 | – |
| Yao et al. (2012) – mixture | 0.69 | 0.66 | 0.67 | – |
| Yao et al. (2012) – ICDAR | 0.68 | 0.66 | 0.66 | – |
| Yi and Tian (2011) | 0.73 | 0.60 | 0.66 | 0.94 |
| Yi and Tian (2011) | 0.71 | 0.62 | 0.66 | – |
| Pan et al. (2011) | 0.674 | **0.697** | 0.685 | 2.43 |

The bold values indicate that highest accuracy of the method.

**Table 4**
Performance of the proposed and existing methods on ICDAR 2011 dataset.

| ICDAR 2011 | Precision | Recall | F-score | Aveg. time (s) |
|---|---|---|---|---|
| Proposed method | 0.83 | **0.71** | 0.77 | 13.9 |
| González and Bergasa (2013) | **0.892** | 0.701 | **0.785** | – |
| Neumann and Matas (2013) | 0.854 | 0.675 | 0.754 | 0.6 |
| Neumann and Matas (2012) | 0.73 | 0.65 | **0**.69 | 1.8 |
| Shi et al.'s method (2013) | 0.833 | 0.631 | 0.718 | 1.5 |
| Kim's method | 0.83 | 0.625 | 0.713 | – |
| Koo (2013) | 0.791 | 0.62 | 0.695 | – |

The bold values indicate that highest accuracy of the method.

method detects text well for images of different orientations and background complexity. The reported quantitative result in Table 5 shows that the proposed method achieves better recall, precision and F-measure compared to the method (Yao et al., 2012). This shows that the proposed method is good for arbitrary orientation text detection because of the advantage in integrating the three symmetry properties and SIFT.

### 4.5. Our CUTE80 data

To give an idea of the qualitative results of the proposed method, we show sample results in Fig. 26. Fig. 26 shows that the proposed method detects well for curved text lines having different backgrounds, font size, fonts and contrast. Since to our knowledge, none of the existing methods is able to handle curved text lines, it is not possible to compare the proposed method with any existing method. We only report the quantitative results of the proposed method reported in Table 6 which shows that the method gives promising results for the curved text lines. The method achieves encouraging results without sacrificing horizontal, non-horizontal data accuracy. This is the main contribution of our method.

Though the main scope of this work is to detect text in natural scene images, we provide some sample recognition results for each database to show end results of text detection as shown in Fig. 27 where (a)–(e) show sample recognition results of ICDAR 2005,


**Fig. 24.** Sample results of the proposed method for horizontal text detection (ICDAR 2011 dataset).


**Fig. 25.** Sample results of the proposed method for arbitrary orientation text detection (MSRA-TD500 dataset).

**Table 5**
Performance of the proposed and existing methods on MSRA-TD500 dataset.

| Methods | Precision | Recall | F-score |
|---|---|---|---|
| Proposed method | **0.70** | **0.68** | **0.69** |
| Yao et al. (2012) – mixture | 0.63 | 0.63 | 0.63 |
| Yao et al. (2012) – ICDAR | 0.53 | 0.52 | 0.53 |
| Epshtein et al. (2010) | 0.25 | 0.25 | 0.25 |
| Chen and Yuille (2004) | 0.05 | 0.05 | 0.05 |

The bold values indicate that highest accuracy of the method.

**Table 6**
Performance of the proposed method on curved text dataset.

| Method | Precision | Recall | F-score | Aveg. time (s) |
|---|---|---|---|---|
| Proposed method | 0.65 | 0.68 | 0.61 | 16.1 |

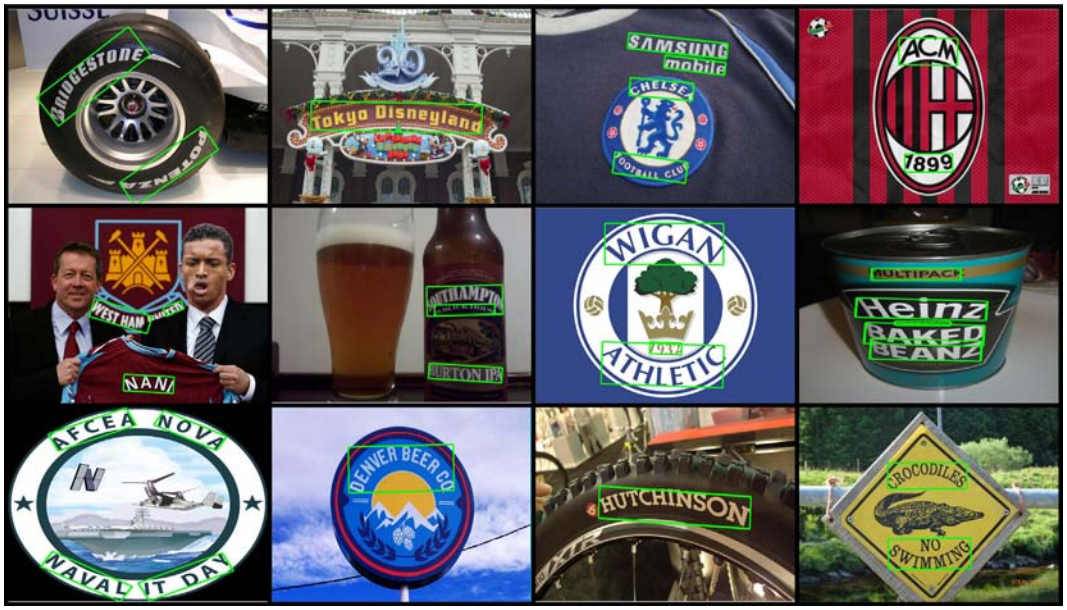
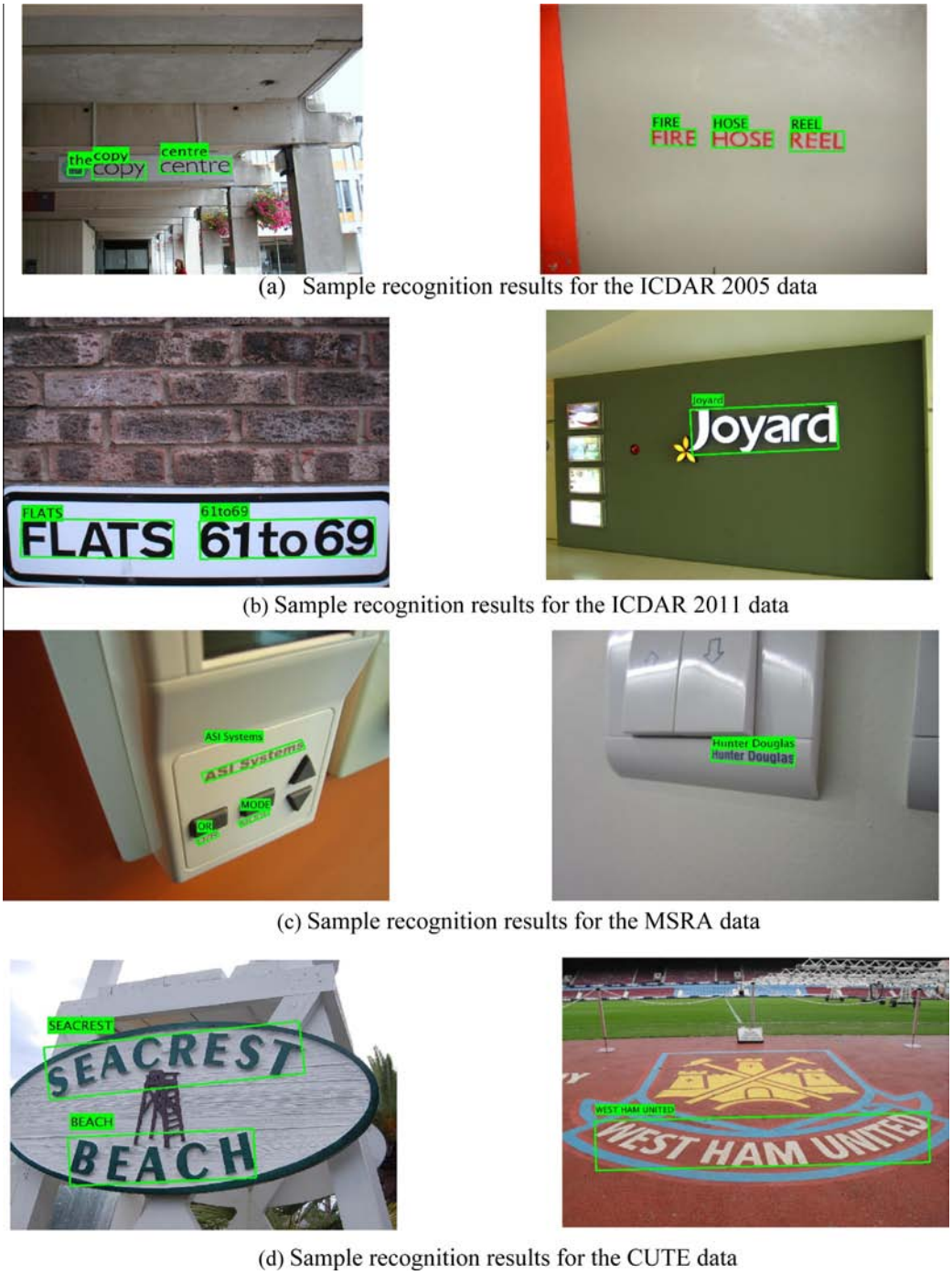



**Fig. 26.** Sample results of the proposed method for curved text detection (CUTE80 dataset).




(a) Sample recognition results for the ICDAR 2005 data

(b) Sample recognition results for the ICDAR 2011 data

(c) Sample recognition results for the MSRA data

(d) Sample recognition results for the CUTE data

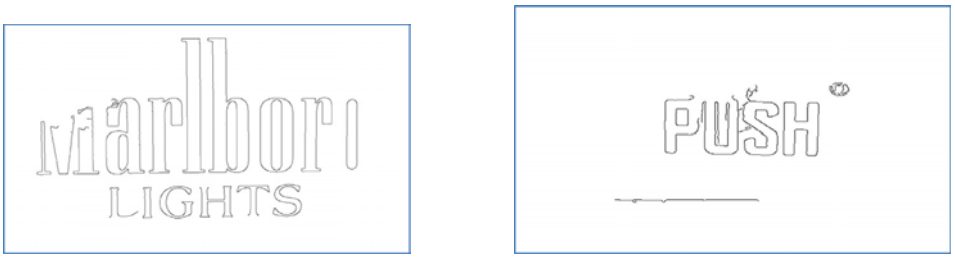
**Fig. 27.** Sample recognition results of the different databases given by the Tessearact OCR.

ICDAR 2011, MSRA and CUTE databases, respectively. We pass the segmented words detected by the text detection method to Tesseract OCR (Google OCR) for recognizing words (Tesseract). The Google OCR is available publicly. Since the OCR accepts only binary image, we manually convert grey images to binary images. The recognition rates at the word level given by the OCR are as follows: 46%, 50%, 39% and 47% word recognition rates for ICDAR 2005, ICDAR 2011, MSRA and CUTE data, respectively. The method reports low accuracy rates for all the datasets because of the limitations of the current OCR, such as working only for plane background images and selected fonts. This shows that there is a scope for improving recognition rate for natural scene characters. This will be another challenging issue for our future work.

### 4.6. Discussions

Since the proposed method has ability to restore a full text line from just one component, the method works well even if we lose some text components due to illumination effect or any distortion effects as shown in Fig. 28 where the method restores all the components though there is severe illumination effect in the input images. In other words, if any text information is missed out due


(a) Input images affected by severe illumination




(b) Recovered components by growing method

**Fig. 28.** Sample results for the illumination affected images.


**Fig. 29.** Failure cases of the proposed method.

to illumination effect, it does not affect much for our method. However, if illumination effect causes to the loss of the entire text line then our method fails to detect text line in the image.

As the gradient and the SIFT features require high contrast information to identify the text candidates, the proposed method fails sometimes to detect text with too low contrast and low resolution images as shown in Fig. 29. In this case, the proposed method gives low recall and high precision.

It is observed from Tables 3–6 that for the ICDAR2005 and ICDAR2011 data, the proposed method gives low accuracy compared to the results of MSRA-TD500 data. This is because ICDAR 2005 and ICDAR2011 data requires word segmentation to calculate the measures while for MSRA-TD500 data requires line segmentation to calculate the measures. On the other hand, the proposed method gives promising accuracy for curved text data.

The proposed novel integrated method and the several objective heuristics generally eliminate most of the false positives. However, when the space between two text lines is too little or any characters are touching between two text lines, the method fixes one bounding box for the two lines. In addition, if text is of too small fonts, the method does not detect text candidates due to low resolution. The sample cases are shown in Fig. 29 where we can see that the method fixes improper bounding boxes for the text lines and gives false positives.

## 5. Conclusion and future work

In this paper, we have proposed a new robust system that integrates Mutual Direction Symmetry, Mutual Magnitude Symmetry and Gradient Vector Symmetry properties for identifying text candidates from the common information in Sobel and Canny edge images of the input image regardless of the orientation of the text line. The local features are introduced and used in a new way for identifying false text candidates, which outputs text representative. We have proposed an ellipse growing process for each representative based on the nearest neighbor concept for grouping the text components. This is followed by false positive elimination based on some heuristics. The experimental results on ICDAR and MSRA-TD500 data show that the proposed method outperforms the existing methods in terms of precision and F-measure.

Experimental result on curved text data shows that the proposed method is promising.

The following are the main contributions of the proposed work: (1) The basis for identifying text candidates from the natural scene images is that text pattern in Sobel and Canny edge image of the input images share the same text properties. This is a new observation and motivation for proposing new three symmetrical features. (2) The proposed new symmetrical features, such as Mutual Direction Symmetry (MDS), Mutual Magnitude Symmetry (MMS) and Gradient Vector Flow Symmetry (GVS) are the insights of the proposed work. (3) The way the method integrates these three features is also another feather of the proposed work to identify the text candidates. (4) Exploring local descriptor (SIFT) for the purpose of refining text candidates by eliminating false text candidates is another contribution of the proposed work. (5) The proposed Ellipse growing based on text properties which works for any orientation of text is one more contribution as it restores full text information from one text component (representative) without connecting non-text components. (6) The objective heuristics proposed for false positive elimination is one more contribution as these heuristics do not use constant threshold values. (7) The proposed method outperforms the existing methods in terms of measures for all types of datasets. (8) The proposed method is independent of scripts and orientations, contrast, resolution, fonts and font size etc.

The following are possible future directions of the proposed work. (1) The proposed method works based on the common information in Sobel and Canny edge images of the input images. It is observed that Sobel operation gives fine details for high contrast text and loses information for low contrast texts. In other words, if Sobel operation loses text pixels of full text line due to severe illumination effects, low resolution and perspective distortions then the method fails to detect text in the images. Therefore, we need to investigate a method which does not use the Sobel edge images for text detection. (2) Sometimes, due to complex background, the proposed three symmetry features may identify non-text candidate as text candidates and hence the method gets poor precision rate. We need to strengthen the features to avoid false text candidate selection. (3) Since SIFT is invariant to rotation, scaling, different views, we need to explore these properties completely to refine the text candidates because we use SIFT here in a simple way to refine the text candidates. (4) The proposed growing method is slightly expensive since it involves connected component analysis. Therefore, there is a scope for reducing number of computations while growing. (5) Due to background variations, when the space between text lines is too small then the growing method fixes one bounding box for two lines. Therefore, there is a scope for improving the growing process such that it grows along the text direction irrespective of background variations and space between the lines. (6) Fixing exact bounding box for the words by growing sometimes fails due to background complexity. This leads to get poor recall. Therefore, there is scope for developing a new word segmentation method. (7) Experimental results show that the accuracy is not high as in document analysis for both text detection and recognition due to variations of fonts, fonts size, contrast etc. There is a great need for improving text detection accuracy, as well as recognition accuracy.

## Acknowledgments

This work is done jointly by University of Malaya (UM), Malaysia supported by the University of Malaya HIR under Grant Nos. UM.C/625/1/HIR/037, J0000073579 and National University of Singapore (NUS), Singapore.

The work is also partly supported by the University of Malaya HIR under Grant No. UM.C/625/1/HIR/210.

We thank anonymous reviewers and editor for their constructive comments to improve the quality, as well as clarity of the paper.

## References

Bai, B., Yin, F., & Liu, C. L. (2014). A seed-based segmentation method for scene text extraction. In *Proceedings of the DAS* (pp. 262–266).

Chen, X., & Yuille, A. (2004). Detecting and reading text in natural scenes. In *Proceedings of the CVPR* (pp. 366–373).

Du, Y., Duan, G., & Ai, H. (2012). Context-based text detection in natural scenes. In *Proceedings of the ICIP* (pp. 1857–1860).

Epshtein, B., Ofek, E., & Wexler, Y. (2010). Detecting text in natural scenes with stroke width transform. In *Proceedings of the CVPR* (pp. 2963–2970).

Fernandez-Caballero, A., Lopez, M. T., & Castillo, J. C. (2012). Display text segmentation after learning best-fitted OCR binarization parameters. *Expert Systems with Applications*, 4032–4043.

González, Á., & Bergasa, L. M. (2013). A text reading algorithm for natural images. *Image and Vision Computing, 31*(3), 255–274.

Grafmuller, M., & Beyerer, J. (2013). Performance improvement of character recognition in industrial applications using prior knowledge for more reliable segmentation. *Expert Systems with Applications*, 6955–6963.

Guo, J., Gurrin, C., Lao, S., Foley, C., & Smeaton, A. (2011). Localization and recognition of the scoreboard in sports video based on SIFT point matching. In *Proceedings of the advances in multimedia modelling* (pp. 337–347).

Jung, K., Kim, K. I., & Jain, A. K. (2004). Text information extraction in images and video: A survey. *Pattern Recognition*, 977–997.

Koo, H. I. (2013). Scene text detection via connected component clustering and nontext filtering. *IEEE TIP*, 2296–2305.

Koo, Hyung Il, & Kim, Duck Hoon (2013). Scene text detection via connected component clustering and non text filtering. *IEEE Transactions on Image Processing*, 2296–2305.

Lee, S., Cho, M. S., Jung, K., & Kim, J. H. (2010). Scene text extraction with edge constraint and text collinearity. In *Proceedings of the ICPR* (pp. 3983–3986).

Liang, J., Doermann, D., & Li, H. (2005). Camera-based analysis of text and documents: A survey. *International Journal of Document Analysis and Recognition, IJDAR*, 84–104.

Lucas, S. (2005). ICDAR 2005 text locating competition results. In *Proceedings of the ICDAR* (pp. 80–84).

Lucas, S. M., Panaretos, A., Sosa, L., Tang, A., Wong, S., & Young, R. (2003). ICDAR2003 robust reading competitions. In *Proceedings of the ICDAR* (p. 682).

Meng, Q., & Song, Y. (2012). Text detection in natural scenes with salient region. In *Proceedings of the DAS* (pp. 384–388).

Minetto, R., Thome, N., Cord, M., Fabrizio, J., & Marcotegui, B. (2010). Snooper text: A multiresolution system for text detection in complex visual scenes. In *Proceedings of the ICIP* (pp. 3861–3864).

Mishra, A., Alahari, K., & Jawahar, C. (2012). Top–down and bottom–up cues for scene text recognition. In *Proceeding of the CVPR* (pp. 2687–2694).

Neumann, L., & Matas, J. (2012). Real-time scene text localization and recognition. In *Proceedings of the CVPR* (pp. 3538–3545).

Neumann, L., & Matas, J. (2013). On combining multiple segmentation in scene text recognition. In *ICDAR 2013* (pp. 523–527).

Opitz, M., Diem, M., Fiel, S., Kleber, F., & Sablatnig, R. (2014). End-to-end text recognition using local ternary patterns, MSER and deep convolutional nets. In *Proceedings of the DAS* (pp. 186–190).

Pan, Y.-F., Hou, X., & Liu, C.-L. (2011). A hybrid approach to detect and localize texts in natural scene images. *IEEE Transactions on Image Processing*, 800–813.

Park, J.-G., & Kim, K.-J. (2013). Design of a visual perception model with edge-adaptive Gabor filter and support vector machine for traffic sign detection. *Expert Systems with Applications*, 3679–3687.

Phan, T. Q., Shivakumara, P., & Tan, C. L. (2012). Detecting text in the real world. In *Proceedings of the ACM multimedia* (pp. 765–768).

Rong, L., Suyu, W., & Shi, Z. X. (2014). A two level algorithm for text detection in natural scene images. In *Proceedings of the DAS* (pp. 329–333).

Shahab, A., Shafait, F., & Dengel, A. (2011). ICDAR 2011 robust reading competition challenge. 2: Reading text in scene images. In *Proceedings of the ICDAR* (pp. 1491–1496).

Sharma, N., Shivakumara, P., Pal, U., Blumenstein, M., & Tan, C. L. (2012). A new method for arbitrarily-oriented text detection in video. In *Proceedings of the DAS* (pp. 74–78).

Shi, C., Wang, C., Xiao, B., Zhang, Y., & Gao, S. (2013). Scene text detection using graph model built upon maximally stable extremal regions. *Pattern Recognition Letters*, 107–116.

Shivakumara, P., Dutta, A., Tan, C. L., & Pal, U. (2013). Multi-oriented scene text detection in video based on wavelet and angle projection boundary growing. *Multimedia Tools and Applications*, 1–25.

Shivakumara, P., Phan, T. Q., Shijian, L., & Tan, C. L. (2013). Gradient vector flow and grouping based for arbitrarily-oriented scene text detection in video images. *IEEE Transactions on Circuits and Systems for Video Technology*, 1729–1739.

Shivakumara, P., Phan, T. Q., & Tan, C. L. (2011). A Laplacian approach to multi-oriented text detection in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 412–419.

Smith, D., Field, J., & Learned-Miller, E. (2011). Enforcing similarity constraints with integer programming for better scene text recognition. In *Proceedings of the CVPR* (pp. 73–80).

Tesseract. Available from: <http://code.google.com/p/tesseract-ocr/>.

Wang, K., & Belongie, S. (2010). Word spotting in the wild. In *Proceedings of the ECCV* (pp. 591–604).

Wang, K., Babenko, B., & Belongie, S. (2011). End-to-end scene text recognition. In *Proceedings of the ICCV* (pp. 1457–1464).

Wei, Y. C., & Lin, C. H. (2012). A robust video text detection approach using SVM. *Expert Systems with Applications*, 10832–10840.

Weinman, J., Learned-Miller, E., & Hanson, A. (2009). Scene text recognition using similarity and a lexicon with sparse belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1733–1746.

Xu, C., & Prince, J. L. (1998). Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, 359–369.

Yao, C., Bai, X., Liu, W., Ma, Y., & Tu, Z. (2012). Detecting texts of arbitrary orientations in natural images. In *Proceedings of the CVPR* (pp. 1083–1090).

Yi, C., & Tian, Y. (2011). Text string detection from natural scenes by structure-based partition and grouping. *IEEE Transactions on Image Processing*, 2594–2605.

Yi, C., & Tian, Y. (2013). Text extraction from scene images by character appearance and structure modelling. *Computer Vision and Image Understanding*, 182–194.

Yun, J., Jing, L., Yu, J., & Huang, H. (2010). A multi-layer text classification framework based on two-level. *Expert Systems with Applications*, 2035–2046.

Zheng, Q., Chen, K., Zhou, Y., Gu, C., & Guan, H. (2011). Text localization and recognition in complex scenes using local features. In *Proceedings of the ACCV* (pp.121–132).