Pattern Recognition 68 (2017) 158–174

Contents lists available at ScienceDirect

Pattern Recognition

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

# Fractals based multi-oriented text detection system for recognition in mobile video images

Palaiahnakote Shivakumara[a,*], Liang Wu[b], Tong Lu[b], Chew Lim Tan[c], Michael Blumenstein[d], Basavaraj S. Anami[e]

[a] Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia
[b] National Key Lab for Novel Software Technology, Nanjing University, Nanjing, China
[c] School of Computing, National University of Singapore
[d] Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia
[e] KLE Institute of Technology, Hubli, India

ARTICLE INFO

*Article history:*
Received 21 June 2016
Revised 23 January 2017
Accepted 11 March 2017
Available online 14 March 2017

*Keywords:*
Fractal theory
Fractal gradient
Wavelet decomposition
Optical flow
Multi-oriented text detection
Mobile video text detection

ABSTRACT

Text detection in mobile video is challenging due to poor quality, complex background, arbitrary orientation and text movement. In this work, we introduce fractals for text detection in video captured by mobile cameras. We first use fractal properties such as self-similarity in a novel way in the gradient domain for enhancing low resolution mobile video. We then propose to use k-means clustering for separating text components from non-text ones. To make the method font size independent, fractal expansion is further explored in the wavelet domain in a pyramid structure for text components in text cluster to identify text candidates. Next, potential text candidates are obtained by studying the optical flow property of text candidates. Direction guided boundary growing is finally proposed to extract multi-oriented texts. The method is tested on different datasets, which include low resolution video captured by mobile, benchmark ICDAR 2013 video, YouTube Video Text (YVT) data, ICDAR 2013, Microsoft, and MSRA arbitrary orientation natural scene datasets, to evaluate the performance of the proposed method in terms of recall, precision, F-measure and misdetection rate. To show the effectiveness of the proposed method, the results are compared with the state of the art methods.



## 1. Introduction

Text detection, tracking and recognition in video through mobile devices is an active field of research as it facilitates human and machine interaction to retrieve the desired information instantly in real-time environments [1]. For example, a blind person can walk freely on a road with his/her mobile device to reach a destination without any assistance by retrieving information about their surroundings, which include street names, building names, park names, traffic symbols, etc. Apart from these human-machine applications, we can see other potential real-time applications where text detection and recognition plays a vital role, such as multimedia database indexing and retrieval tasks, and text removal in video sequences [2–4]. For example, in natural scene video and images, texts or strings usually appear in nearby sign boards and hand-held objects, which provide significant knowledge of the surrounding environment and objects [5]. Besides, as noted in [6], text detection in video can be used for generating relevant databases, where subtitles in video reduces human effort in creating databases for face recognition systems. The main reason to draw the attention of researchers is that texts in video, such as graphics/caption/artificial/superimposed text that is edited manually, and scene text that naturally exists, play a key role in bridging the gap between low and high level features. This is actually the main difficulty of content-based methods for real-time robotic applications [5–7].

In general, human-machine based applications often use mobile devices that have low resolution cameras for retrieving information from images or videos [1]. Therefore, in order to meet this requirement, we use low resolution cameras for experimentation in this work. Actually, text detection from low resolution videos is still an open problem in the field of computer vision and robotic applications [2–4]. To the best of our knowledge, text detection from videos captured by mobile devices is still at the infancy stage. This is because such videos usually suffer from limited computing power, sensor, display resolution, or memory con-

* Corresponding author.
*E-mail addresses:* shiva@um.edu.my, hudempsk@yahoo.com (P. Shivakumara), wuliang0301@hotmail.com (L. Wu), lutong@nju.edu.cn (T. Lu), tancl@comp.nus.edu.sg (C.L. Tan), Michael.Blumenstein@uts.edu.au (M. Blumenstein), anami_basu@hotmail.com (B.S. Anami).

http://dx.doi.org/10.1016/j.patcog.2017.03.018
0031-3203/

ditions in robot applications [2–4]. Therefore, the primary objective of the proposed work is to develop an approach which can withstand adverse factors caused by mobile images such as low resolution, low contrast, small font, multi-oriented texts, different scripts and complex background for robust text detection.

According to the literature on document analysis [5,6], text detection from scanned document images is not a new problem. However, the approaches developed for document image analysis including handwritten, degraded, historical document image analysis may not be used directly for text detection in video and natural scene images. The reason is that those approaches work based on the assumption that the images have either plain or homogeneous backgrounds with high resolution. Therefore, we can conclude that these approaches are sensitive to complex backgrounds.

Several methods have been developed [8–10] for text detection in natural scene images captured by high resolution cameras, where images usually contain high contrast texts with complex backgrounds. Since the images have high contrast texts, the methods use characteristics of character shapes. Therefore, the methods rely on connected component analysis to achieve good accuracies. However, due to the low resolution of mobile video images with complex backgrounds, it is hard to preserve character shapes and hence disconnections or the loss of shape information are often caused. As a result, the methods may not be used for text detection in video directly [5,6].

To overcome the problem of low resolution and complex backgrounds of video, some existing methods have been proposed for text detection in video. These methods can be classified broadly as connected component-based, texture-based and, edge and gradient-based methods [5,6]. Since connected component-based methods [11,12] require character shapes, they may not give good accuracies for low resolution texts with complex backgrounds. To alleviate this problem of complex backgrounds, texture feature-based methods have been developed [13]. These methods are computationally expensive and their performance rests on classifier training and the number of samples. To achieve efficiency, edge and gradient information-based methods are also developed [14]. These methods work well with less computation but are sensitive to the background and hence they give more false positives. In addition, most of the methods of each category only target graphic text but not both graphics and scene texts.

To find solution to the problems of video, such as the presence of both graphics and scene texts in different orientations, new methods have been developed recently [15,16]. These methods basically explore the contrast information of texts rather than the use of text characteristics. Though the methods solve the problem of video, they do not utilize temporal information of video, but rather they rely on individual frames. Therefore, the methods can detect only static texts but not moving texts in video. From the above discussions, we note that none of the existing methods give a perfect solution to the video text detection problem. Besides, none of the methods use video or images captured by mobile devices for text detection and text tracking. Therefore, there is an urgent need for developing a new approach, which is capable of handling the limitations of mobile video images with a good accuracy irrespective of orientations.

## 2. Related work

This section provides a literature review of the methods that use temporal information for text detection in video. There are a few methods [13,17–27] exploring temporal information for video text detection in the literature as follows, which are unlike the methods mentioned in the introduction section.

Li et al. [13] proposed a method for video text tracking based on wavelet and moments features. Huang et al. [17] proposed a method for scrolling text detection in video using temporal frames. However, it is limited to only scrolling texts but not the texts of other directions. Zhou et al. [18] exploit edge information and geometrical constraints to form a coarse-to-fine methodology to define text regions. However, it is not clear as to how the method works for video captured by mobile devices. Mi et al. [19] proposed a text extraction method based on multiple frames. Edge features are explored with a similarity measure for identifying text candidates. Wang and Chen's method [20] uses the spatio-temporal wavelet transform to extract text objects in video documents. Huang [21] detected video scene texts based on video temporal redundancy. The method performs motion detection in 30 consecutive frames to synthesize a motion image. Further, video scene text detection is implemented in each single frame to retrieve candidate text regions. Huang et al. [22] proposed a method for video text detection using temporal frames based on motion features by integrating multiple frames, which give text regions. However, this method focuses on horizontal graphics texts but not multi-oriented scene texts. Zhao et al. [23] proposed an approach for text detection using corners in video. This method proposes to use dense corners for identifying text candidates. Liu et al. [24] proposed a method for video caption text detection using stroke like edges and spatio-temporal information. Li et al. [25] proposed a method for video text detection using multiple frame integration. This method uses edge information to extract text candidates. Moseleh et al. [26] proposed an automatic inpainting scheme for video text detection and removal based on a stroke width transform to identify text objects.

The above literature review reveals that most of the methods focus on caption text and horizontal text detection and tracking, but not both caption and scene texts with movements. Wu et al. [27] proposed a method for detecting both caption and scene texts in video using temporal information and Delaunay Triangulation. However, the scope of the method is limited to static texts having the same direction. Similarly, optical flow based properties have been proposed by Shivakumara et al. [28] for dynamic curved text detection in video. The method is good when text is moving with a static background but not with a moving background. In the same way, Wu et al. [29] proposed multi-oriented scene text line detection and tracking in video based on gradient directional symmetry, the spatial study of Delaunay triangulation and multi-scale integration. The focus of this method is text detection and tracking in video that are not captured by mobile cameras. Khare et al. [30] proposed a new Histogram Oriented Moments descriptor for multi-oriented moving text detection in video. Though the methods [29,30] address multi-oriented issue and use temporal frames for text detection and tracking in video, they have not been tested on mobile video. Therefore, we can infer that the existing methods are not sufficiently equipped to tackle the limitations of mobile video to achieve good results.

Recently, Khare et al. [31] and Roy et al. [32] respectively proposed methods for addressing a single adverse factor such as noisy images generated by Laplacian operation and blur images generated by camera or text movements by proposing specific models to improve text detection and recognition accuracies. However, the scope of the methods is limited to video images but not images captured by mobile cameras with 2 Mega Pixels (MP). In addition, the methods do not focus on utilizing temporal information. In the same way, Zhu et al. [33] proposed a new idea for improving text detection accuracies in natural scene images by introducing context given by background of text information. However, the scope of the method is limited to images captured by high resolution cameras but not mobile cameras or video.

Hence, in this paper, we introduce a new method based on fractals for detecting multi-oriented texts in video using temporal information. As we are inspired by the self-similarity property

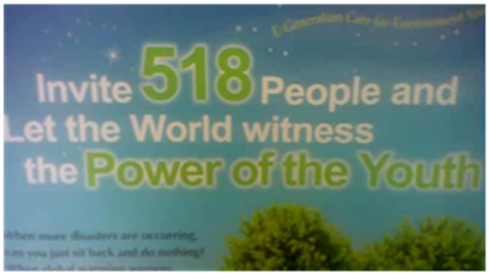



**Fig. 1.** Input mobile video frame containing graphics and scene text multi-fonts, font size and colors with a complex background (tree structure).

of fractals [34,35], where it is shown that fractals are used for enhancing object details in images, we explore fractals for video image enhancement in the gradient domain in this work. Furthermore, fractal expansion is used in the wavelet domain in a pyramid structure to identify text candidates of different font sizes. Then the identified text candidates are verified by optical flow properties such as constant velocity and uniform magnitude to identify potential text candidates. Finally, direction guided boundary growing is proposed for the detection of multi-oriented texts in video. The main contributions of the method are two-fold: (1) Exploring fractals in a novel way for video enhancement, which further leads to finding text components, and (2) The use of fractal expansion for text components in a different way for identifying text candidates in the wavelet domain. To the best of our knowledge, this is the first attempt to explore fractals for text detection in low resolution video captured by mobile cameras.

## 3. Proposed approach

The proposed approach extracts key frames containing texts from video for text detection. It is noted from [6] that video is generally captured with low resolution cameras especially in mobile devices. Therefore, to achieve good results for such video, there is a need to enhance low contrast text pixels. Inspired by the work presented in [34] where fractals are used for enhancing edge details of objects, we explore fractals for enhancing low contrast texts in video for text detection in the gradient domain with the help of interpolation techniques in this work. For a given input image, the proposed approach produces enhanced images based on fractals properties, where one can expect a wide gap between text and non-text pixels. Since our intention is to separate text components from non-text ones, we propose to employ k-means clustering with k= 2 on an enhanced image for classifying text components. This is valid because usually text pixels have high contrast compared to their background. As a result, Fractals give high values for text pixels. Therefore, k-means clustering classifies high values into one cluster and low values into another cluster. The cluster which gives the highest mean is considered as the text cluster. The reason to choose k-means clustering is that it is an unsupervised algorithm and hence it does not require predefined labeled samples, which is unlike supervised methods that require the number of samples. In addition, choosing the number of samples especially for classifying non-text pixels has no limit and boundary. This limits the ability to work on a general task such as text detection. It is true that one of the big challenges in text detection in video is font size, color or contrast of word variation in a text line apart from background variations [5]. One such example is shown in Fig. 1, where we can see multi-colored words, multi-font sizes and complex background. To tackle such issues, we explore fractals, further called fractal expansion in the gradient domain, to study the characteristics of text components such as intra- and inter-symmetry of character components to eliminate misclassified non-text components, which results in text candidates. To handle multi-fonts and multi-font-size texts in video images, the proposed approach obtains an enhanced gray image using the above fractal enhancement, and then proposes wavelet decomposition in a pyramid structure for the enhanced gray image to sharpen edge details. It employs k-means clustering with k= 2 on the output of wavelet decomposition to obtain text components, which are similar to the text components extracted from the above enhanced gradient image. For the text components given by wavelets, the proposed approach uses the above fractals expansion in the gradient domain with intra- and inter-symmetry properties of character components to identify text candidates. Due to complex backgrounds, sometimes, the above step may still misclassify a non-text candidate as a text candidate. Therefore, we propose optical flow properties for text candidates given by the above steps with the help of temporal information based on the fact that text pixels move in a particular direction with constant speeds and have almost uniform values to eliminate false text candidates. This results in Potential Text Candidates (PTCs). To group potential text candidates into text lines of any direction, we propose direction-guided boundary growing, which traverses along the text direction to extract text lines. This outputs text detection.

In summary, the proposed approach consists of five steps, namely, Fractals in the Gradient domain for Text Components (F-TC), which involves Image Enhancement (IE-Gradient) and k-means clustering, Fractals Expansion in the Gradient domain for Text Candidates (FE-TC), which exploits the self-similarity property of fractals to extract intra- and inter-symmetry of character components to eliminate misclassified non-text components, Wavelet decomposition for text candidates detection (WD-TC) in multi-font-size text images, which involves Image Enhancement in the Gray domain (IE-Gray), FE-TC, Optical flow for Potential Text Candidates (PTC), which explores optical flow properties such as velocity and the direction for eliminating non-text candidates, Direction Guided Boundary Growing (DGBG) for text line extraction, which uses the direction of PTC and the spacing between the PTC for text line extraction of any direction. This outputs text detection with a bounding box for text lines in the image. The flow of the proposed approach can be seen in Fig. 2.

### 3.1. Fractals in gradient for text components through image enhancement

It is a fact that fractal geometry was developed to provide the means to study irregular objects compared to conventional classical geometry [35] because fractals are capable of representing natural scenes. It measures the changes in details at different scales, and gives an estimation of the self-similarity of the fractal object, i.e., the ability of the object to keep the same detail when scaled up in contrast to classical geometry where the fractal dimension is an integer. This observation motivated us to propose fractals for enhancement and then text candidate selection because in the case of text components, the values of text pixels are almost the same at the component level and hence they can be extracted by estimating the self-similarity at different scales with fractals. The key factor of fractals is to find the fractal dimension [34]. The principle of fractal dimension is based on the fact that the number of segments $N$ is proportionally related to the scale factor $S$, i.e.:

$$N \propto s^{-D} \tag{1}$$

where $D$ is the corresponding fractal dimension. By rearranging Eq. (1), the value of the fractal dimension is calculated as follows:

$$log_s N = -D = \frac{\log N}{\log S} \tag{2}$$

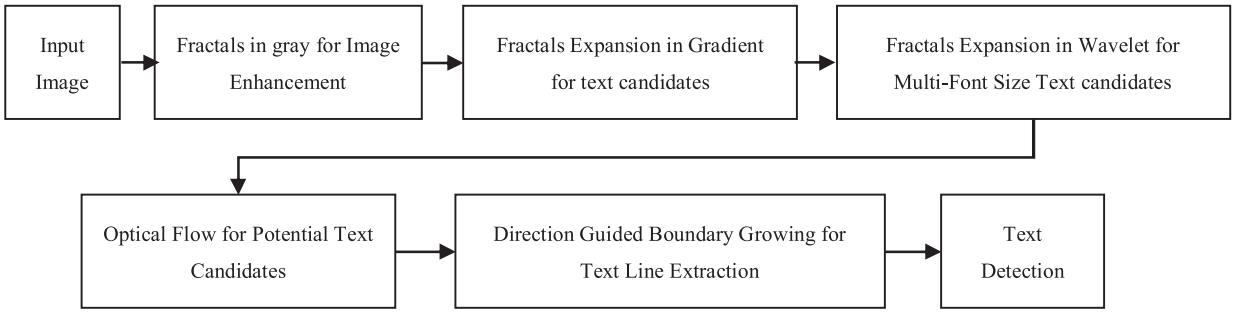



**Fig. 2.** Pipe-line of the proposed approach.


(a) (b)

**Fig. 3.** Enhancement by super-resolution using the scaling factor $n = 2$: (a) Enhanced gray image for the image (IE-Gray) in Fig. 1 and (b) Gradient image of the enhanced gray image (IE-Gradient) in Fig. 3(a).

To calculate fractals, the dimension for an image is as follows. Let an image $f(x)$: $x \in R^2$ be modeled as the pair $(X, \mu)$ where $X$ is the fractal set, and $\mu$ is the corresponding measure of the set. In this work $X$ is the union of fractal sets, each of which is composed of all the points that have the same gradient value, and $\mu$ is the number of those points. We consider the fractal dimensions of all the fractal sets in $X$ together to form a multi-fractal spectrum (MFS) vector. This vector gives a global fractal analysis of the image and reflects the irregularity of the image. However, this work requires local fractal analysis but not global fractal analysis to study the image.

To determine the local fractal dimension of a pixel $x \in R^2$, a formula similar to Eq. (1) is used such as in the following:

$$\mu(B_r(x)) \propto (2r)^D \tag{3}$$

Where $B_r(x)$ is a square of length $r$ centered at pixel $x$. $\mu(B_r(x))$ is the measure supported on $B_r(x)$. Similar to Eq. (2), the local fractal dimension is calculated as follows:

$$D(x) = \lim_{r \to 0} \frac{\log \mu(B_r(x))}{\log 2r} \tag{4}$$

Theoretically, when $r \to 0$ the local fractal dimension $D(x)$ will converge and will be the same in [34].

In order to enhance image details using super-resolution, the fractal dimension of an image should be scale-invariant [34]. The following formula is used to enhance the interpolated gradient image based on the invariance assumption of local fractal dimension and length.

$$grad(f_h(y))_s = \beta_e \frac{\left\| grad\left(\hat{f}_h(y)\right) \right\|}{\left\| grad\left(\hat{f}_h(y)\right)^{\alpha} \right\| + 0.01} \left( grad\left(\hat{f}_h(y)\right) \right)^{\alpha} \tag{5}$$

where $\hat{f}_h$ is the interpolated up-sampled image, $f_h$ is the enhanced up-sampled image. $\beta_e, \alpha$ are the enlarging and sharpening parameters, respectively, which are calculated as per the instructions given in [34]. $grad(.)$ is the gradient of the respective image.

The estimation of the high resolution enhanced image is performed by solving the following optimization problem with variable $f$:

$$f_h = \arg\min_f \left\| G * f - \hat{f}_h \right\|_2^2 + \lambda \| (\nabla_x f_h)_s \|_2^2 + \lambda \left\| (\nabla_y f_h)_s \right\|_2^2 \tag{6}$$

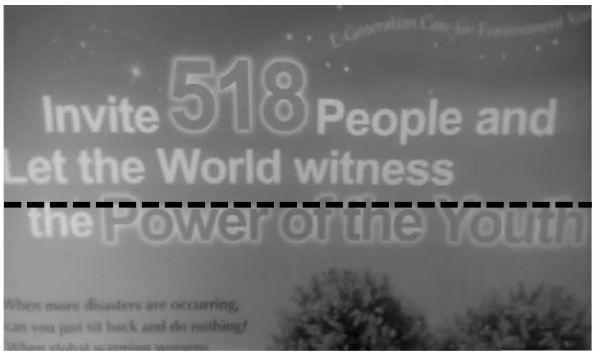
where $G$ is a Gaussian smoothing kernel, and $\hat{f}_h$ is the estimated image using bi-cubic interpolation. $(\nabla_x f_h)_s$ and $(\nabla_y f_h)_s$ are the directional differentials extracted from $grad(f_h(y))_s$. The operation $*$ is a conventional 2D convolution operator. $\lambda$ is a regularization term.

In this work, we use the scaling factor $n = 2$ to enlarge the first frame using super-resolution as shown in Fig. 3, where we can see the enhanced gray image and its gradient image. This step enhances the contrast of text pixels considerably.

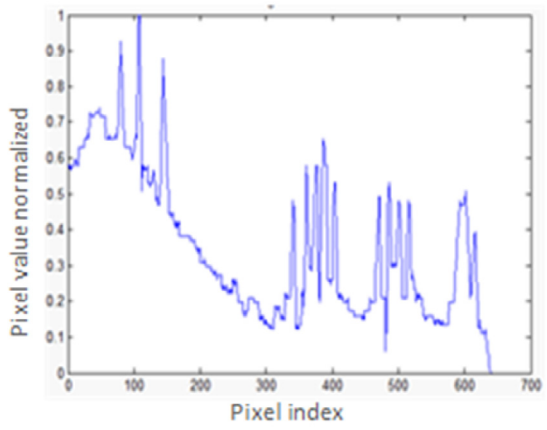
The effect of enhancement for video frames can be seen in Fig. 4, where we plot line graphs for the red line shown in Figs. 4(a) and (b), respectilvey (see Figs. 4(c) and (d)). It is observed from Figs. 4(c) and (d) that the line graphs of the enhanced image give sharp peaks compared to the line graphs in Fig. 4(c) because of the enhancement of low contrast text pixels by the fractal enhancement process. In this way, the proposed method enhances low contrast text information in mobile video frames. However, gray values are sensitive to backround variations. Therefore, we consider a gradient image of the enhanced gray image (EI-Gradient) for the separation of text and non-text pixels. This is because the gradient operation involves the first order derivative which gives high positive values for high contrast pixels near or at the edges by supressing background pixels as shown in Fig. 3(b), where one can see that the edge pixels are sharpened as compared to the pixels in Fig. 3(a).

It is noted from the above process that the gap between text and non-text pixels is widened in the IE-Gradient image. Since our objective is to separate text pixels from non-text ones, we propose k-means clustering with k=2 for IE-Gradient, which outputs two clusters. We consider the cluster that gives the high mean as a text cluster as shown in Fig. 5(a), where one can notice that almost

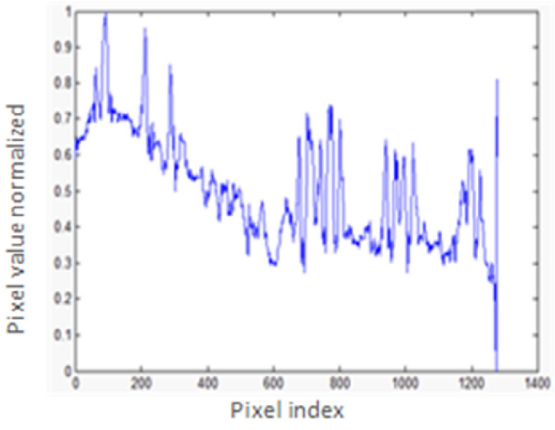

(a) Input image




(b) Enhanced Gray image (IE-Gray)

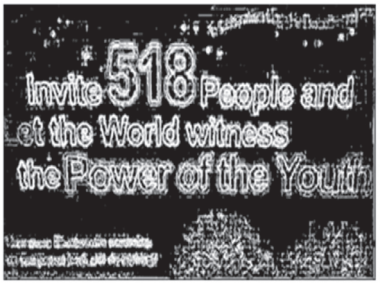


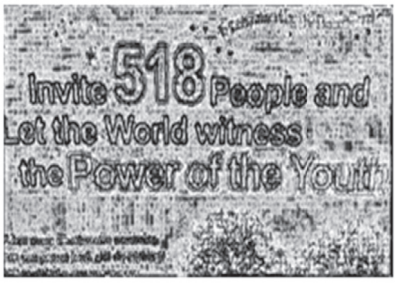

(c) Graph for the dashed line in (a)




(d) Graph for the dashed line in (b)

**Fig. 4.** Illustrating the effect of enhancement by super-resolution: (a) Dimension of the image in (a) is (640 × 352) and (b) Dimension of the image in (b) is (1280 × 704). Line graphs in (c) and (d) represent pixel index vs values of the dashed line over (a) and (b), respectively.




(a). High contrast cluster (text cluster)(b) Low contrast cluster (non-text cluster)

**Fig. 5.** Text components using k-means clustering on the IE-Gradient image in Fig. 3(b).

all the text pixels are classified into the text cluster, and the other cluster is considered as a non-text one as shown in Fig. 5(b), where it is noted that non-text pixels are classified into this cluster. It is expected that a text cluster should contain text pixels. Due to complex backgrounds, non-text pixels are also classified as text pixels as shown in Fig. 5(a). This shows that the pixels which represent noise created by the background may not be classified into non-text clusters, rather they are classified as text clusters. Therefore, if we use k = 3, there is a high chance of losing text pixels because text pixels may have neither high values nor low values. Hence, we prefer k= 2 for text pixel classification from the EI-Gradient image, which we call text components. To remove non-text candidates in Fig. 5(a), we propose fractal expansion to extract intra- and inter-symmetry properties of character components for text components in the text cluster, which results in text candidates. This will be discussed in the next section.

### 3.2. Fractal expansion in the gradient domain for text candidates

As noted from the previous section, for the input image, the proposed approach detects text components from the IE-Gradient image as shown in Fig. 5(a). Fig. 5(a) shows that a few non-text components are misclassified as text components due to background variations in the video image. To remove such false text components, we propose fractal expansion to study the characteristics of text components such that non-text components can be reduced. As mentioned in the Proposed Approach Section, the self-similarity of fractals can be explored to extract features, which represent character components, such as intra- and inter-symmetry

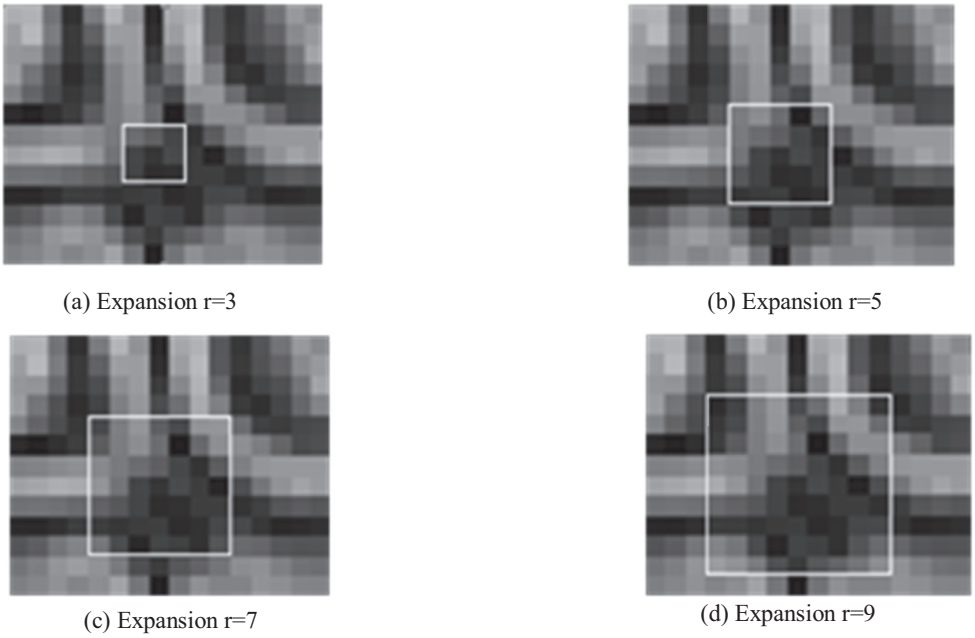



**Fig. 6.** Fractal expansion for one pixel in the gradient domain. The value for a specific square length is obtained for the center pixel by multiplying the values of pixels inside the square with corresponding Gaussian smoothing kernel values and then taking the sum.

properties. Therefore, the output of this step is considered as the text candidates (FE-TC).

The idea of the expanding algorithm is to study the effects of a pixel on neighboring pixels when enhancing the image. In other words, this algorithm gives an insight of how a single pixel is repeated in the gradient domain at different scales.

We consider a square window of a constant length $r \in \mathbb{Z}$ centered at the pixel in the enhanced gradient image. Then we compute the weighted sum $g_r$ of all the values of the pixels inside this window. Thus for different square lengths, the method produces different outputs for each given pixel. Each output corresponding to a specific square length makes a sample point as in the following:

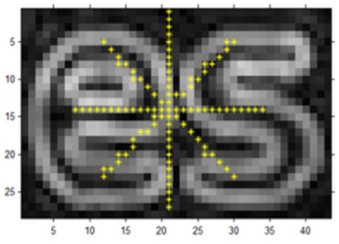
$$S_x = \{(r,\ g_r)\}_{r \in \mathbb{Z}} \tag{7}$$

where $S_x$ is the set of the sample point for pixel $x$, and $r$ is the length of the square. For example, Fig. 6(a)–(d) illustrate how a set of sample points are calculated for the center of the pixel of the image at different r values. This process continues till it reaches the boundary of the image. This process is called fractal expansion. The weighted sum is calculated by multiplying the gradient values inside the window of size $r$ with values of a symmetric Gaussian smoothing kernel $G$ of the same size as in Eq. (8):

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{8}$$

where $\sigma$ is the standard deviation of the Gaussian distribution. $x$ and $y$ are the distances from the origin of the window in the horizontal axis and the vertical axis, respectively.

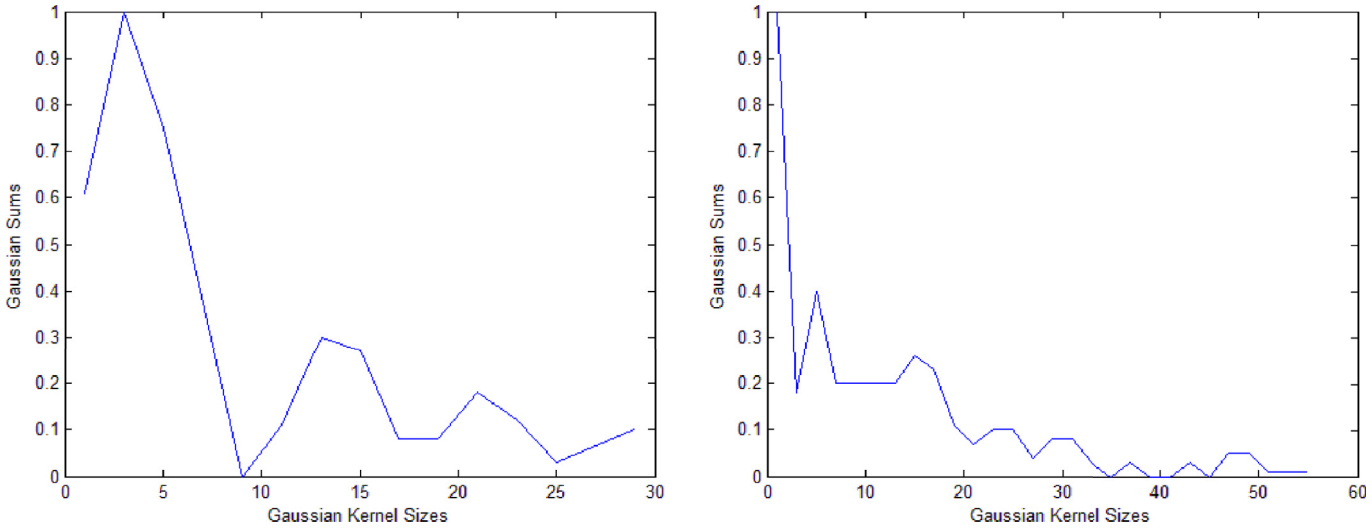
The proposed method employs the expanding process on each text component in the text cluster image in Fig. 5(a) to study the characteristics of text components. The sample text components of "es" in the gradient domain are shown in Fig. 7. It is a fact that a text component exhibits uniform spacing between character components (inter symmetry), constant stroke width distance (intra symmetry) throughout each character component [8,26], and a uniform value for the whole character component. To extract such observations, we consider the center pixel of a text component as the starting pixel for applying an expansion algorithm,




8 directions from the centroid

**Fig. 7.** Studying the characteristics of the text components and the background by the fractal expansion Algorithm 1 in different directions.

which computes the radius and weighted sum as described above for every pixel in the 8 directions as shown Fig. 7, respectively. To find the weighted sum of the expansion of each pixel in the respective directions, we consider only the difference between the current weighted sum and the previous one (which corresponds to the weighted sum at the current length - 1) as the weighted sum for the current length. This is because the expansion algorithm includes the previous pixel value given by the expansion algorithm for computing the weighted sum for the current pixel. As expansion grows, the expansion algorithm computes the weighted sums in a cumulative manner. To realize the significance of a set of weighted sums of the pixels of respective directions, we plot line graph pixels vs weighted sums for the horizontal direction as shown in Figs. 8(a) and (b), where one can notice that Fig. 8(a) gives peaks which exhibit a periodic property for the consecutive pixels, while Fig. 8(b) does not. Since character components have intra- and inter-symmetry, where the edge appears, the fractals expansion algorithm gives high peaks and where there is a space or background, fractal expansion gives low peaks. When we study such peaks in all the 8 directions, if a component is a text one, we can see periodic peaks with constant distances and almost the same height, as opposed to non-periodic peaks as shown in Figs. 8(a) and (b), for the text component, "es", respectively. The proposed approach checks this periodic property in all the 8 directions. If all the directions exhibit at least one periodic property,




(a) Text pixel's chart (b) Non-text pixel's chart

**Fig. 8.** Periodic property by fractal expansion in the gradient domain for text and non-text components in different directions shown in Fig. 7.

**Algorithm 1** Fractal expansion process for 8 directions.

Input: Text component detected (F-TC) by the step presented in Section 3.1, where k-means clustering has been employed on the IE-Gradient image to obtain a text cluster as shown in Fig. 5(a).
1. For every component in the high cluster of the enhanced gradient $H_0$ do the following:
   a. Apply the expanding algorithm horizontally.
   b. If the component passes (a) then go to (c), else remove the component.
   c. For each of the 8 possible directions in Fig. 7:
      (i) For every pixel $x$ located at the straight line of the current direction, use the expanding algorithm to find $S_x = \{(r,\ g_r - g_{r-1})\}_{r \in \mathbb{Z}}$ for all the discrete values $r = 1$ *to* $R$ with a step of 2 where $R$ is limited by the borders of the bounding box of the component.
      (ii) Find all the sequences of consecutive pixels with a given length (ex. two consecutive pixels) that have the same peaks and the same distances between peaks.
   d. If the number of the directions that have at least one sequence is sufficient enough, then consider the component as a possible text candidate. Otherwise remove the component.
2. Produce the final output of the expanding tests, which represent fractal analysis of the components.

Output: Text candidates that satisfy the periodic property (FE-TC).

then the proposed approach considers it as a text candidate, or else it considers it a false text candidate. In this way, fractal expansion helps to extract characteristics, which represent the local structure of character components to identify text candidates. Algorithmic steps for obtaining a periodic property for the text component in 8 directions can be seen in Algorithm 1.

### 3.3. Fractal expansion in the wavelet domain for multi-font-size text candidates

Section 3.1 presents fractals in the gradient domain for image enhancement (IE-Gradient), which gives text components (F-TC). For each text component in text clusters, fractal expansion in the gradient domain presented in Section 3.2 gives text candidates, which are called FE-TCs. These two steps work well for the images that have a uniform font and font size. This is because Fractals expansion is explored to study local structure of the character components by extracting inter and inter symmetry properties to identify the text candidates. However, this process is not robust to multi-font, multi-font size. In general, video frames and natural scene images contain texts of different fonts and font sizes. To strengthen the proposed approach, we propose wavelet decomposition at different levels to identify text candidates, which is called WD-TC. To achieve this, the proposed approach employs wavelet decomposition on IE-Gray images and then uses the inverse wavelet transform to reconstruct IE-Gray images at different levels. Note that an IE-Gray image is obtained by the algorithm presented in Section 3.1. It employs k-means clustering with k=2 as discussed in Section 3.1 on the reconstructed images to obtain text components. For each text component, the proposed approach invokes the same FE-TC algorithm which uses fractal enhancement presented in Section 3.1 and fractal expansion presented in Section 3.2 to identify text candidates, which is called WD-TC. In this way, the approach identifies text candidates for the images of different fonts and font sizes. In other words, the proposed wavelet decomposition extracts the global structure of components, while Fractal expansion extracts the local structure of components. As a result, we integrate both the advantages to identify text candidates. Based on the experimental results, we noticed that two levels are enough to achieve good results. Note that we have used the Haar wavelet in this work as it is good at analyzing the textual properties of images [13,15].

For the orthogonal wavelet representation for two dimensions of an image, $A_1^d f$ in the pyramid structure, there is a set of $3J+1$ discrete images as in Eq. (9):

$$\left(A^d_{2^{-j}} f,\ \left(D^1_{2^j} f\right)_{--J\le j\le -1},\ \left(D^2_{2^j} f\right)_{--J\le j\le -1}, \left(D^3_{2^j} f\right)_{--J\le j\le -1}\right) \quad (9)$$

*For any* $J > 0$

where:

$$D^1_{2^j} f = \left(\left(f(x,y) *\ \psi^j_1(x,y)\right)\ \left(2^{-j}n,\ 2^{-j}m\right)\right)_{(n,m)\in\ \mathbb{Z}^2} \quad (10)$$


(a) Level-0 Level-1 Level-2

(b) Text candidates (WT-TC) for Level-0, Level-1 and Level-2

(c) Merging text candidates of different levels

**Fig. 9.** WT-TC for multi-font, multi-font-size text in a video image.

$$D^2_{2^j} f = \left( \left( f(x,y) * \psi^j_2(x,y) \right) \left( 2^{-j}n,\ 2^{-j}m \right) \right)_{(n,m)\in\ \mathbb{Z}^2} \tag{11}$$

$$D^3_{2^j} f = \left( \left( f(x,y) * \psi^j_3(x,y) \right) \left( 2^{-j}n,\ 2^{-j}m \right) \right)_{(n,m)\in\ \mathbb{Z}^2} \tag{12}$$

where image $A^d_{2^{-j}} f$ is the coarse approximation at resolution $2^{-j}$, and $D^k_{2^j} f$ gives the detailed signals for different orientations and resolutions (i.e. their number is 3*J*: three images for each resolution). Thus we can say that an image $A^d_{2^{j+1}} f$ is decomposed into $A^d_{2^j} f$, $D^1_{2^j} f$, $D^2_{2^j} f$, and $D^3_{2^j} f$, where these four images correspond to the lowest frequencies in both directions, vertical high frequencies, horizontal high frequencies, and high frequencies in both directions, respectively. Those four components are used to reconstruct image $A^d_{2^{j+1}} f$ using a pyramidal algorithm similar to the reconstruction in one dimensional wavelet transforms.

Detecting text candidates using wavelet decomposition (WD-TC) is illustrated in Fig. 9, where (a) denotes different levels of IE-Gray images, (b) gives text candidates (WD-TC) corresponding to the levels in (a) after applying FE-TC on text components given by k-means clustering on reconstructed images, and (c) shows the final merged text candidates from different levels. Interestingly, we note from Fig. 9(b) that the numeral "518" is missing on Level-0 and Level-1, but it is detected on Level-2 because this numeral has a different contrast and color compared to the other components. This is the advantage of wavelet decomposition, which helps in extracting text candidates of different font sizes, colors and contrasts. The text candidate image shown in Fig. 9(c) is the input for the next step to eliminate false text candidates.

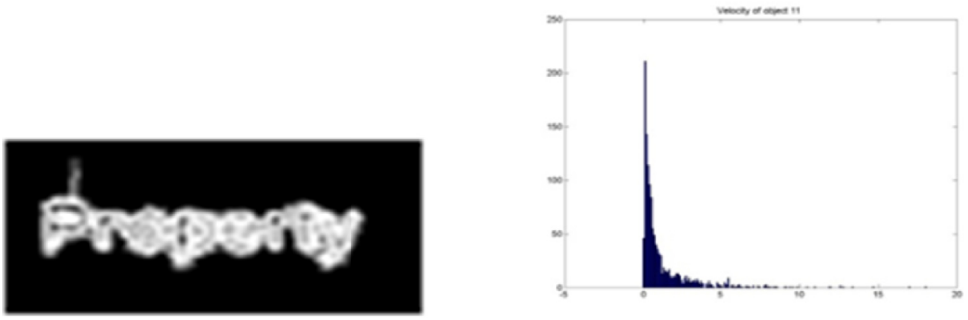
### 3.4. Optical flow for potential text candidates

Fig. 9(c) shows that preventing false text candidates completely is hard due to the unpredictable nature of backgrounds and text appearance in video. Besides, since the Fractals expansion extract features based on self-similarity pattern, the objects like tree structure in background may be misclassified as text candidates. It is also true that video provides temporal information. Furthermore, each text component in video generally moves with a constant velocity and uniform direction for a few frames because it has to be legible and visible to readers [28,29]. Therefore, we propose to explore optical flow-based properties for text candidate verification with the help of temporal information because optical flow helps us to extract velocity and direction of each text candidate regardless of shapes and patterns of the components. It is noted that originally optical flow has been used for measuring velocities of movements of brightness patterns in video [36]. Inspired by this, we propose an iterative algorithm to determine optical flow which we call an HS algorithm. This algorithm considers text candidates as brightness patterns in video frames. The proposed method finds the rate of change of each text candidate and further summarizes the total error for the rate of changes. The total error value is used for finding the optical flow between text candidates in different temporal frames. To make optical flow robust to noisy text candidates, which may exist due to background variations, we introduce a special weighting factor, say, $\alpha$ to control error values between text candidates. It can be formulated as follows. Let u, v be the two optical flow velocities of text candidates where $u = dx_i/dt$, and $v = dy_i/dt$ for all the pixels $(x_i, y_i)$ of the text candidates in the image. The total error function to be minimized is given in Eq. (13):

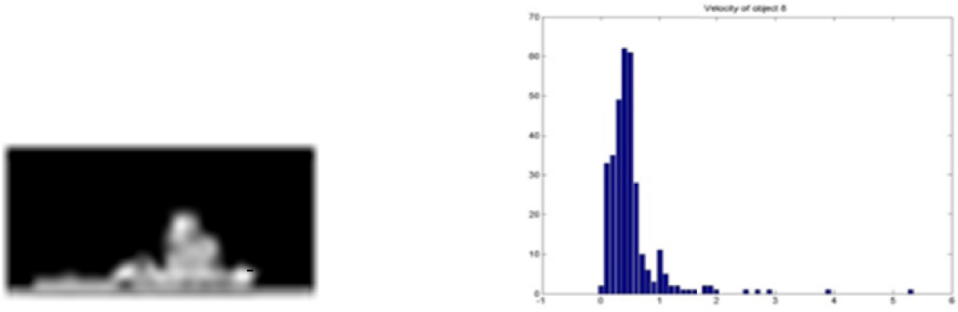
$$\Phi^2 = \iint (\alpha^2 \Phi_c^2 + \Phi_b^2) dx\ dy \tag{13}$$

$$\Phi_c^2 = \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial y} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 \tag{14}$$

$$\Phi_b = E_x u + E_y v + E_t \tag{15}$$



(a) Text candidate and its velocity magnitude histogram



(b) Non text candidate and its velocity magnitude histogram

**Fig. 10.** Optical flow properties of moving text and non-text candidates.

where $\Phi_c^2$ is the measure of the departure from smoothness in the velocity flow, $\Phi_b$ is the sum of the errors for the rate of brightness change. $E_x$, $E_y$ and $E_t$ are the partial derivatives of the text candidates in the image with respect to (w.r.t.) x, y and time t in the sequence, respectively, and $\alpha$ is the weighting factor to control the error.

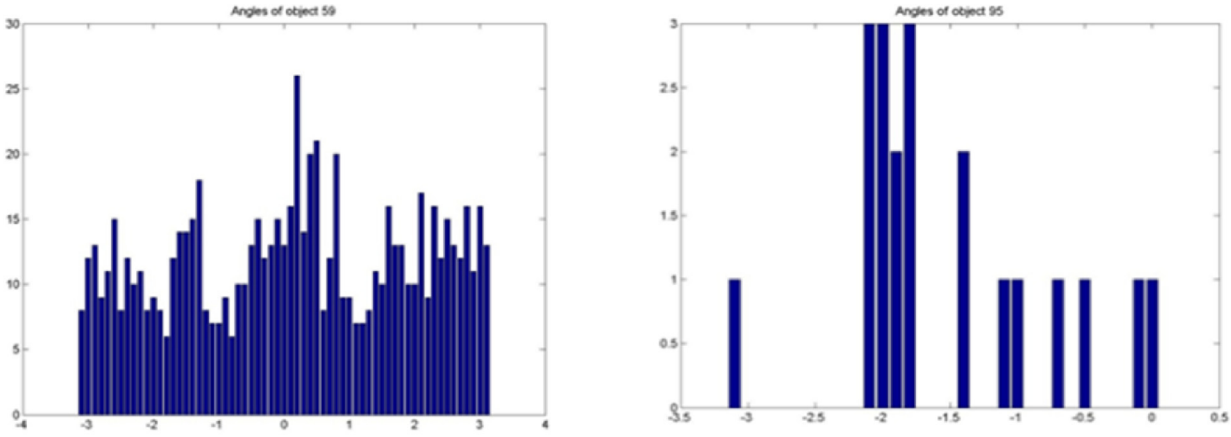
In this way, the proposed method computes optical flow of text candidates in temporal frames to eliminate false text candidates. First, the proposed method computes the optical flow between the first frame and the second frame. Then it considers the second and the third frames and finds another optical flow. After that, the algorithm checks the difference between the two computed optical flows. This process continues between every two successive frames until the difference in the optical flows between the frames is sufficiently small (converges). The text candidates that satisfy optical flow convergence are considered for extracting textual properties for verification. For these text candidates, the proposed method plots histogram pixels vs velocity magnitude on the velocity of the pixels in the text candidates as shown in Fig. 10, where we can see Fig. 10(a) shows the velocity of most of the pixels of a text candidate is close to zero, while Fig. 10(b) shows the velocity of the pixels of a non-text candidate varies as the candidate moves. This is one property used for eliminating false text candidates. Similarly, we also use the angle histogram of the pixels of each text candidate using the formula $\theta = atan2\ (u, v)$, where $u$, $v$ are velocity components for pixels as shown in Fig. 11. Fig. 11(a) shows that dense distribution tends to be a normal distribution because every pixel in a text candidate gets almost the same angle. Since the pixels of a text candidate give almost a uniform velocity, the distribution of angles of the pixels tends to be a normal distribution, while for a non-text candidate, a scattered angle histogram and distribution can have any shape as shown in Fig. 11(b). If a text candidate in the image satisfies these two optical flow-based properties then it is considered as a potential text candidate, else a false text candidate. The effect of false text candidate removal can be seen in Fig. 12, where it can be noted that non-text candidates are removed compared to Fig. 9(c).

In summary, using optical flow, the proposed approach extracts features that represent text-based, text pixel direction and movement in temporal frames to eliminate false text candidates.

### 3.5. Direction guided boundary growing for text detection

Potential Text Candidates (PTCs) are detected by the previous step. Next, with these PTCs, we need to extract text lines of any orientation. When we look at the directions of two to three component groups in a text line, the groups exhibit almost the same direction. However, when we take the whole text line, the direction may differ much substantially as compared to the groups. This observation is true for any orientation including curved text lines such as “STARBUCKS” . This clue leads to the proposal of a new Direction Guided Boundary Growing (DGBG) scheme for multi-oriented text line extraction from video.

The proposed DGBG fixes a bounding box for each PTC in the image and then it grows pixel by pixel until it finds the nearest white pixel. Next, the proposed DGBG merges two components as one component and then finds the angle for this merged component. Note that before calculating the angle, DGBG checks whether a merged component is enough to compute an angle or not based on an iterative method, which calculates angle differences of components before terminating [15]. Once DGBG gets the angle of the merged component, it grows the boundary of the second component in the angle direction to find the nearest neighbor component rather than growing in all the directions. This process continues until the end of the text line. The end of the text line is determined based on the space between words and text lines. The threshold




(a) Text component (b) Non-text component

**Fig. 11.** The distribution of angle histogram for text and non-text candidates.

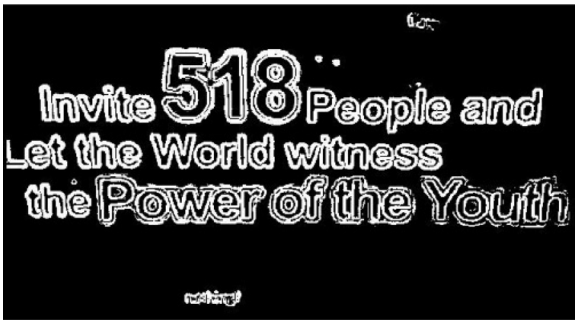



**Fig. 12.** The final text candidates after removing false positives using optical flow.

value is calculated based on the fact that the space between two text lines is larger than the space between words and characters [15,16]. The advantage of this DGBG is that it extracts words as well as text lines of any orientation. Fig. 13(a) shows one example for word extraction, where DGBG fixes the bounding boxes for the words in it. Similarly, Fig. 13(b) shows examples of text line extraction by fixing the bounding boxes for the text lines.

The effect of direction guided boundary growing can be seen in Fig. 14, where one can notice the two horizontal text lines are very close to each other. The proposed DGBG moves along the text line direction as shown in Fig. 14(a) without touching another text line since it involves the direction followed by growing. The final result can be seen in Fig. 14(b), where two text lines are extracted correctly. For the same image, when we use DGBG without direction, it fixes one bounding box for the two text lines as shown in Fig. 14(c). This is the advantage of the proposed DGBG.

In summary, since video or natural scene images may contain texts of any direction, after finding PTC, we need some criteria to group PTC to extract text lines from the image. Therefore, DGBG is proposed to traverse along text directions to fix the bounding box for the text line of any direction as shown in Fig. 14(b). The main advantage of DGBG is that when two text lines are close to each other and if both the text lines are touching at some point, the algorithm fixes bounding boxes correctly for each text line.

## 4. Experimental results

We consider three types of datasets for evaluating the proposed method, namely, (1) Video captured by a mobile camera (2 mega pixels) created by us to test the effectiveness on poor quality frames as in robotic applications, (2) Benchmark video from the ICDAR 2013 dataset [37] and benchmark YouTube Video Text data [38] to validate the proposed method on both low and high resolution frames, and (3) Benchmark datasets from the ICDAR 2013 natural scene dataset [37], Microsoft [8] and MSRA [39] to test the effectiveness of the proposed method on high resolution camera-based images. The datasets that fall in the first category include different contrasts, backgrounds, fonts, font sizes and orientations. The datasets that fall in the second category comprise benchmark video from ICDAR 2013 which involves video captured by different devices and most of the texts are in linear directions and benchmark video from YouTube Text Data (YVT) which involves scene texts of different background complexities. The datasets that fall in the third category comprise benchmark ICDAR 2013 scenes data which contains high resolution images with different background complexities of large font size variations; the benchmark Microsoft dataset that contains street view text images with complex backgrounds of trees, buildings, roads, skies, etc., and the benchmark MSRATD-500 dataset that consists of arbitrarily-oriented text lines of different scripts. In this way, we considered diverse datasets to test the proposed method's ability on poor quality low resolution, and high resolution frames, and the frames with arbitrarily-oriented text lines or with multi-lingual texts.

We consider four standard measures, namely, recall, precision, F-measure, midsection rate (MDR) and Average Processing Time (APT) which is the mean processing time taken for each image for measuring the performance of the proposed method. Since our mobile video dataset and three different video datasets do not have ground truth, we manually count actual text blocks (ground truth), the number of text blocks detected by the text detection method, and false positives. We follow the definitions given in [9,13,14] for calculating the above measures.

The Actual Number of Text Blocks (ATB) is the ground truth which gives the total number of text lines in frames. Truly Detected Block (TDB) contains texts, while the Falsely Detected Block (FDB) contains non-texts. A Text block that misses texts of 20% is considered as a Misdetection Blocks (MDB). Based on these definitions of text blocks detected by the proposed method, the performance measures are defined as follows:

Recall (R) = TDB/ ATB,
Precision (P) = TDB/ (TDB+FDB),
F-measure (F) = 2 P R/ (P+R), and
Misdetection Rate (MDR) = MDB/TDB.


(a) Word detection (b) Text line detection

**Fig. 13.** Text detection using DGBG on PTC.

(a) DGBG for the first text line extraction

(b) Extraction with DGBG (c) Extraction without DGBG

**Fig. 14.** Effect of direction guided growing for text line detection.

For natural scene datasets, since the ground truth and evaluation scheme are available for calculating measures, we follow the standard evaluation scheme given in the ICDAR 2013 robust reading competition [37] and the ground truth for calculations. However, for the MSRA dataset, we follow the instructions given in [39] for calculating the measures because the results are reported at the text line level with the standard evaluation scheme but not at the word level as in the ICDAR 2013 and Microsoft datasets.

In order to show the effectiveness of the proposed technique, we compared it with the state-of-the-art methods, namely, Li et al.'s method [13], which uses moments and wavelet combination for text detection in video, Zhao et al.'s method [23] which uses corner-based features and optical flow for caption text detection in video, Mosleh et al.'s method [26] which uses stroke width transform and inpainting for text detection and text removal, Epshtein et al.'s method [8] which proposes stroke width transform for text detection from natural scene images, Wu et al.'s method [29] which is developed for text detection tracking in video using Delaunay triangulation, and Yao et al.'s method [39] which uses an improved version of the stroke width transform for non-horizontal text detection from natural scene images. Since the first three methods use key frames and temporal frames for text detection, we compared these methods for all the experiments. Despite the fact that Epshtein et al.'s method is developed for text detection in natural scene images, we compared this method for text detection in video, as well as natural scene images as it is considered to be the state-of-the-art method for both video as well as natural scene images. We report the results of Wu et al.'s method for ICDAR video and natural scene data, and Yao et al's method for MSRATD-500 data to compare them with the proposed method because we follow the same evaluation criteria and dataset as in [29,39] for calculating the measures.

### 4.1. Analyzing the contribution of each step of the approach

We created our dataset using a mobile camera which captured 1000 different videos of about one second with different text types. The mobile camera is of 2 MP, 30 fps video recording with dimensions ($640 \times 352$). In summary, our dataset includes 233 static graphic text videos, 338 dynamic graphic text videos, 257 static camera scene text videos, and 172 dynamic camera scene text videos. The major steps of the proposed method are implementing fractals in the gradient domain for frame enhancement and text component detection (IE-Gradient), fractal expansion in the gradient domain to extract intra- and inter-symmetry properties, which represent character components to identify text candidates from text components (FE-TC). The purpose of using the same image enhancement approach and fractal expansion in the wavelet domain is to identify potential text candidates, which are independent of fonts, font size, and optical flow-based textual properties.

To understand the contributions of the above steps, we conducted experiments on our dataset (1000 videos) separately by calculating recall, precision, F-measure and Misdetection rate as reported in Table 1. It is observed from Table 1 that the proposed approach gives good results compared to those of the other steps.


(a) Input image (b) Proposed approach (c) With intensity (d) Without wavelet (e) Without optical flow

**Fig. 15.** Qualitative text detection results to show the effectiveness of each experiment listed in Table 1.

**Table 1**
Analyzing each step of the proposed method on our 1000 mobile video data.

| Method | Recall | Precision | F-Measure | MDR | APT |
|---|---|---|---|---|---|
| **Proposed method** | 0.70 | 0.81 | 0.75 | 0.18 | 1.4 |
| **With Intensity** | 0.45 | 0.71 | 0.54 | 0.18 | 1.2 |
| **Without Wavelet** | 0.56 | 0.84 | 0.67 | 0.21 | 1.2 |
| **Without Optical Flow** | 0.71 | 0.74 | 0.72 | 0.20 | 1.3 |

**Table 2**
Performance of the proposed and existing methods on our 1000 mobile video data.

| Method | Recall | Precision | F-Measure | MDR | APT |
|---|---|---|---|---|---|
| **Proposed method** | 0.70 | **0.81** | 0.75 | 0.18 | 1.2 |
| **Zhao et al. [23]** | 0.49 | 0.72 | 0.50 | 0.30 | 3.4 |
| **Li et al. [13]** | **0.77** | 0.80 | **0.78** | 0.18 | **0.8** |
| **Epshtein et al. [8]** | 0.68 | 0.78 | 0.71 | **0.10** | 1.0 |
| **Mosleh et al. [26]** | 0.70 | 0.73 | 0.71 | 0.13 | 1.3 |

The proposed approach with intensity values scored low recalls compared to those of the proposed approach with gradient because intensity values are sensitive to color, while the gradient is not. In this experiment, the proposed approach uses only the enhanced gray image (IE-Gray) instead of the gradient image for text candidate detection using fractal expansion and wavelet decomposition. Similarly, the proposed approach without wavelets scored low recalls compared to those of the proposed approach because it loses text candidates when the frame contains different text font sizes. In this experiment, the proposed approach uses only IE-Gradient images and fractal expansion FE-TC for text candidate detection without wavelet decomposition (WD-TC). In the same way, the proposed approach without optical flow-based properties gives low recall, precision and F-measure compared to the proposed approaches. This is due to false positives created by complex backgrounds. In this experiment, the proposed approach finds text candidates (WD-TC) from the input image and sends them to DGBG to extract text lines without eliminating false positives. In this way, the proposed method is insensitive to multiple colors, fonts, font sizes, contrasts and orientations. It may be noted from Table 1 that APT in seconds is almost the same for all the steps. To visualize the effectiveness of each experiment listed in Table 1, we present sample qualitative results in Fig. 15, where we can see that the proposed approach detects text well in the input image in Fig. 15(a) as shown in Fig. 15(b); the proposed approach with intensity and without wavelet decomposition misses text compared to the results of the proposed approach as shown in Figs. 15(c) and (d), respectively, and the proposed approach without optical flow gives more false positives as shown in Fig. 15(e). This shows that each step makes a contribution to achieve better results for the complex text detection problem.

### 4.2. Experiments on our mobile data

Sample qualitative results of the proposed method are shown in Fig. 16, where one can see different inputs for different situations. The results of the proposed method are also shown. It is noted from the experimental results in Fig. 14 that the proposed method detects texts well for all the situations. The quantitative results of the proposed and existing methods are reported in Table 2, where it can be seen that the proposed method is the best at precision, but low at recall and F-measure compared to Li et al.'s method. This is because Li et al.'s method is capable of handling arbitrary movements of texts in video, while the proposed method sometimes loses text components when the text has large movements. Therefore, there is a scope for the improvement of the proposed method in the future. However, misdetection rates of the proposed and Li et al.'s method are the same. The proposed method ranks in the third position in terms of APT because of Fractal expansion and the direction of the guided steps. The other existing methods report low results compared to the proposed and Li et al.'s methods because Zhao et al.'s and Mosleh et al.'s methods focus on caption texts but not scene texts in video, while Epshtein et al.'s method is developed for high resolution camera-based scene images but not video.

### 4.3. Experiments on benchmark video datasets

We test the ability of the proposed method for text detection in video, where we can expect low resolution with complex backgrounds in contrast to mobile video where we can expect poor quality, low resolution and complex backgrounds. Therefore, as discussed in Section 4, we consider two benchmark video datasets, namely, the ICDAR 2013 video which contains both caption and scene texts in video, and the YVT video dataset which contains only scene texts in video. The ICDAR2013 dataset [37] includes 15 test videos captured by special cameras, where the texts are of different languages such as Spanish, French, and English. The videos are captured by different camera devices to get a diversified dataset. The duration of the videos varies from 5 seconds as the minimum to 1 minute and six seconds as the maximum with the number of frames as 162 and 1980, respectively. Similarly, the YouTube Video Text (YVT) dataset [38] contains 30 videos of 15 seconds length with 30 frames per second and HD 720P quality. We can conclude that the ICDAR 2013 video dataset provides substantial variation in contrast and resolution, while the YVT provides high resolution videos.

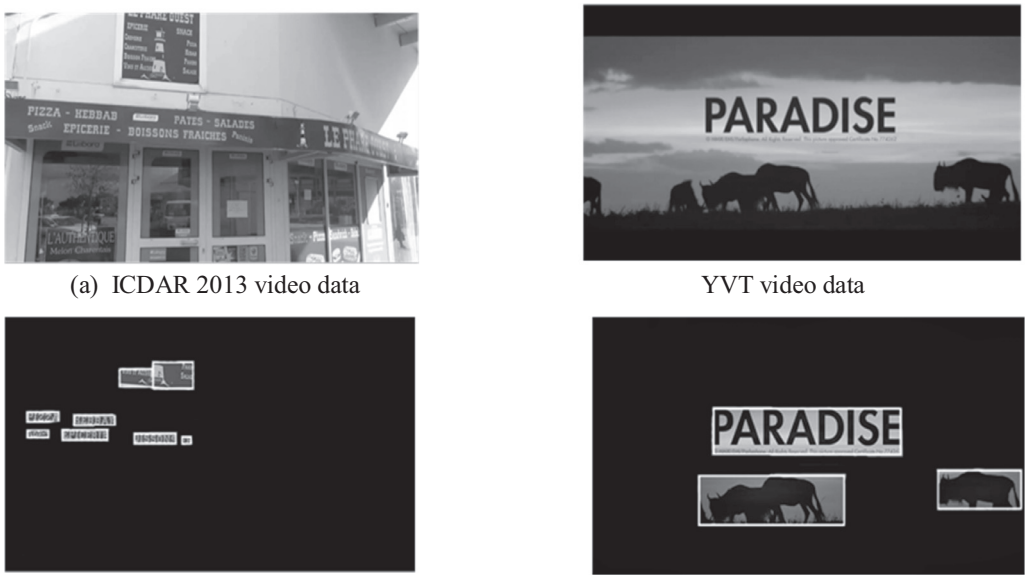
#### 4.3.1. Experiments on ICDAR 2013 video data

Sample qualitative results of the proposed method are shown in Fig. 17 for the frames of small font texts with different backgrounds. It can be seen in Fig. 17 that the proposed method misses several examples of text in the frames. Since videos are captured by different camera devices in different situations, the data contains more contrast variations compared to the Mobile video dataset. In addition, the proposed method is developed to handle the videos captured by mobile cameras. Therefore, the pro-


(a) Input mobile video frames from our dataset


(b) Text detection of the proposed approach

**Fig. 16.** Sample qualitative results of the proposed approach on our dataset.




(a) ICDAR 2013 video data YVT video data

(b) Text detection results of the proposed method for ICDAR 2013 and YVT video frames

**Fig. 17.** Sample qualitative results of the proposed method on ICDAR 2013 and YVT video datasets.

**Table 3**
Performance of the proposed and existing methods on ICDAR2013 Video dataset.

| Method | Recall | Precision | F-Measure | MDR | APT |
|---|---|---|---|---|---|
| **Proposed method** | 0.57 | 0.61 | 0.59 | 0.17 | 1.3 |
| **Wu et al. [29]** | **0.68** | **0.63** | **0.65** | –- | 1.3 |
| **Zhao et al. [23]** | 0.32 | 0.23 | 0.27 | 0.23 | 3.4 |
| **Li et al. [13]** | 0.67 | 0.25 | 0.36 | 0.20 | **0.8** |
| **Epshtein et al. [8]** | 0.47 | 0.48 | 0.47 | **0.14** | 0.9 |
| **Mosleh et al. [26]** | 0.49 | 0.50 | 0.49 | 0.16 | 1.2 |

posed method misses some text and hence poor results are reported in Table 3 compared to Table 2. Table 3 shows that the proposed method achieves the best results for recall, precision and F-measure compared to the existing methods except Wu et al.'s method. Li et al.'s method scores low results compared to the Mobile video data because the method is developed and trained with a classifier for English texts but not different language texts. Since Zhao et al.'s and Mosleh et al.'s methods focused on caption texts in English, the methods report low results compared to the proposed method. In the same way, since Epshtein et al.'s method is developed for natural scene text detection, the method reports low results. However, Epshtein et al.'s method is the best at MDR compared to all the methods including the proposed method as it does not involve any expensive steps and works at the edge component level. Wu et al.'s method is the best at recall, precision and F-measure compared to all the methods because it has the ability to find motion status of text in video. On the other hand, since the proposed method is developed for mobile video, it gives slightly poor results compared to Wu et al.'s method. However, the proposed method is the best in terms of recall, precision and F-measure compared to other methods. The proposed method ranks fourth when we consider the time parameter.

**Table 4**
Performance of the proposed and existing methods on the YVT scene text video dataset.

| Method | Recall | Precision | F-Measure | MDR | APT |
|---|---|---|---|---|---|
| **Proposed method** | 0.73 | 0.79 | 0.76 | 0.20 | 1.2 |
| **Wu et al. [29]** | 0.73 | **0.81** | **0.77** | – | 1.3 |
| **Zhao et al. [23]** | 0.41 | 0.34 | 0.37 | 0.28 | 3.1 |
| **Li et al. [13]** | 0.57 | 0.32 | 0.41 | 0.17 | **0.7** |
| **Epshtein et al. [8]** | 0.76 | 0.68 | 0.72 | **0.11** | 1.0 |
| **Mosleh et al. [26]** | **0.79** | 0.72 | 0.75 | 0.12 | 1.1 |

*4.3.2. Experiments on YVT scene text video data*

Sample qualitative results of the proposed method are given in Fig. 17, where it is noted that the proposed method detects text well for those texts with complex backgrounds, but misses some text with perspective distortions. The quantitative results of the proposed and existing methods are reported in Table 4, which shows that all the methods including the proposed method report improved results compared to Table 3 because the YVT video dataset provides high contrast texts compared to the ICDAR 2013 video dataset. In addition, the YVT data does not include texts of different scripts as in the ICDAR 2013 videos. It is also observed from Table 4 that the proposed method is the second best at precision and F-measure, while Epshtein et al.'s method is the best at MDR, and Mosleh et al.'s method is the best at recall. Epshtein et al.'s and Mosleh et al.'s methods well compared to the proposed method because these methods are good for high contrast images as in caption texts in the ICDAR 2013 video dataset and texts in natural scene images. In addition, both the methods use a stroke width transform for text detection. However, Li et al.'s method reports poor results because the method is trained for both caption and scene texts in video but not scene texts only as in the YVT dataset. Zhao et al.'s method reports poor results because the corner detection used in the method is sensitive to complex backgrounds. Wu et al.'s method is the best at precision and F-measure because of the advantage of tracking. On the other hand, the proposed method gives good results compared to the existing methods except for Wu et al.'s method because of the advantage of fractals, fractal expansion and optical flow-based textual properties. The proposed method is fourth best at APT.

*4.4. Experiments on benchmark natural scene datasets*

We also tested the proposed method's ability for text detection in natural scene images, where we can see only scene texts with high contrast and complex backgrounds. Since the proposed method uses key frames for text candidate detection, the same steps are used for text detection in natural scene images. In other words, the proposed method does not use optical flow-based properties for text detection in natural scene images because optical flow-based properties require temporal frames. The proposed method uses text candidate detection and then DGBG for text detection on natural scene datasets. For experimentation, we respectively used 233, 307 and 200 test sample images reported from the ICDAR 2013 scene natural scene data, Microsoft natural scene data and MSRA natural scene dataset.

*4.4.1. Experiments on ICDAR 2013 data*

Sample qualitative results of the proposed method are given in Fig. 18(a), where the proposed method detects texts well in the images of different cluttered backgrounds and fonts. The quantitative results of the proposed and existing methods are reported in Table 5, where it is seen that the proposed method achieves the best precision, while Wu et al.'s method achieves the best recall and F-measure compared to the proposed method and other existing methods. Mosleh et al.'s and Zhao et al.'s methods report poor results because that the features are sensitive to cluttered background, Li et al.'s method is developed for video but not scene texts in natural scene images, and Epshtein et al.'s method is the fourth in the top list on recall and F-measure and is the best at MDR. Mosleh et al.'s method is better than Epshtein et al.'s because the former is the extension of the latter one. On the other hand, the proposed method is good because of the advantages of using fractals. The proposed method is third best in terms of APT.

**Table 5**
Performance of the proposed and existing methods on ICDAR 2013 scene text data.

| Method | Recall | Precision | F-Measure | MDR | APT |
|---|---|---|---|---|---|
| **Proposed method** | 0.65 | **0.79** | 0.71 | 0.12 | 1.2 |
| **Wu et al. [29]** | **0.70** | 0.76 | **0.73** | – | 1.2 |
| **Zhao et al. [23]** | 0.20 | 0.18 | 0.19 | 0.29 | 3.5 |
| **Li et al. [13]** | 0.61 | 0.21 | 0.31 | 0.20 | **0.9** |
| **Epshtein et al. [8]** | 0.60 | 0.73 | 0.66 | **0.09** | **0.9** |
| **Mosleh et al. [26]** | 0.66 | 0.76 | 0.71 | 0.10 | 1.0 |

**Table 6**
Performance of the proposed and existing methods on Microsoft scene data.

| Method | Recall | Precision | F-Measure | MDR | APT |
|---|---|---|---|---|---|
| **Proposed method** | 0.49 | **0.59** | **0.54** | **0.13** | 1.2 |
| **Wu et al. [29]** | 0.71 | 0.38 | 0.50 | **–** | 1.2 |
| **Zhao et al. [23]** | **0.76** | 0.33 | 0.50 | 0.26 | 3.3 |
| **Li et al. [13]** | 0.35 | 0.23 | 0.28 | 0.19 | **1.0** |
| **Epshtein et al. [8]** | 0.50 | 0.51 | 0.51 | **0.13** | **1.0** |
| **Mosleh et al. [26]** | 0.51 | 0.48 | 0.49 | 0.15 | 1.2 |

*4.4.2. Experiments on Microsoft scene data*

Sample qualitative results of the proposed method are shown in Fig. 18(b), where it is seen that the proposed method misses texts as well as gives more false positives compared to the results of ICDAR 2013 scene dataset. This data is much harder than ICDAR 2013 scene dataset due the presence of vegetation and repeated patterns such as windows, trees, and greenery along with small sized font texts. Therefore, the proposed method reports poor results for recall, precision and F-measure compared to the ICDAR 2013 scene dataset results, according to the results reported in Table 6. However, when we compare the results of the proposed method with the existing methods, the proposed method outperforms the existing methods in terms of precision and F-measure. Interestingly, Zhao et al.'s method gives the best recall compared to the other methods including the proposed method because corner detection can work for small font size texts. However, its precision is the worst in comparison with the other methods, due to more false positives. Wu et al.'s method scores poor results because it fails extracts dominant points for the small size fonts with clutter background. Li et al.'s method fails because of small size font texts and cluttered backgrounds. Epshtein et al.'s method is good at MDR and rates at third from the top in terms of recall. The proposed method is the second best at APT.

*4.4.3. Experiments on MSRA-TD500 data*

Sample qualitative results of the proposed method are shown in Fig. 18(c), where it is noted that the proposed method detects text lines well for different orientations. The quantitative results of the proposed and existing methods are listed in Table 7, which shows that the proposed method is better than the existing methods in terms of precision. Wu et al.'s method scores the best recall and F-measure because it has the ability to handle multi-fonts and multi-font sizes. Zhao et al.'s method gives poor results for precision compared to the other methods. When we look at the results of Tables 6 and 7, the results of the proposed and existing methods slightly increase for the MSRA dataset in terms of recall. However,


(a) ICDAR 2013 (b) Microsoft (c) MSRA

**Fig. 18.** Qualitative results of the proposed method for ICDAR 2013, Microsoft and MSRA natural scene datasets.


(a) Sample input video frames

| | | |
|---|---|---|
| The wheel was opened in March 2000to mark the end of the 20th century | KERETA | ECKER CHLEOKER |

(b) Ground truth for recognition

| | | |
|---|---|---|
| The wheel was eed MEIRSEDto mark the end eentglil/I | KERETA | acxlan Z: GHLECNKEIR |

(c) Recognition result by Tesseract OCR

**Fig. 19.** Sample recognition results by Tesseract OCR on our data, ICDAR2013 and the YVT video dataset.

**Table 7**
Performance of the proposed and existing methods on MSRA-TD500 arbitrary-oriented scene data.

| Method | Recall | Precision | F-Measure | MDR | APT |
|---|---|---|---|---|---|
| **Proposed method** | 0.54 | **0.68** | 0.60 | 0.10 | 1.2 |
| **Wu et al. [29]** | **0.70** | 0.63 | **0.66** | – | 1.2 |
| **Zhao et al. [23]** | 0.69 | 0.34 | 0.46 | 0.19 | 3.4 |
| **Li et al. [13]** | 0.65 | 0.26 | 0.37 | 0.17 | **0.9** |
| **Epshtein et al. [8]** | 0.50 | 0.52 | 0.51 | **0.09** | 1.0 |
| **Mosleh et al. [29]** | 0.53 | 0.56 | 0.55 | 0.11 | 1.2 |
| **Yao et al. [39]** | 0.63 | 0.63 | 0.60 | – | 2.3 |

the precisions drop reasonably, especially for Zhao et al.'s method and Li et al.'s method, because these methods do not work well for arbitrary orientations. Since Yao et al.'s and Mosleh et al.'s methods are the extensions of Epshtein et al.'s method, they give good results for arbitrary oriented texts compared to Epshtein et al.'s method. In the same way, the proposed method is invariant to rotation of text lines, and gives good results for arbitrarily oriented texts in scene images. The proposed method ranks in third position in terms of APT.

In summary, from the above discussions on the experimental results, we can conclude that the proposed method is a robust and generalized approach as it gives consistent results for diverse datasets such as mobile video data, video data, natural scene data, multi-lingual data and multi-oriented data, while the existing methods do not give consistent results for different situations because they are developed for specific datasets and objectives.

To show the usefulness of text detection, we report sample recognition results given by Tesseract OCR [40] which is available publicly for our dataset, ICDAR 2013 video and YVT video datasets in Fig. 19, where (a) shows the input frames, (b) shows the ground truth for texts, and (c) shows the recognition results by Tesseract OCR. These results show that if we feed text lines detected by the text detection method to an available OCR system, we can get better recognition results as opposed to feeding the whole image with text lines for which OCR often returns unintelligible results due to complex backgrounds.

## 5. Conclusions and future work

We have proposed a new approach for text detection in low resolution video captured by mobile cameras by exploring fractals and fractal expansion for the first time in this work. The proposed approach makes use of the self-similarity property of fractals in a new way for achieving good text detection results for different types of data, which include different contrasts, font sizes or background variations, multi-script text lines and arbitrarily-oriented texts. Temporal frames are used for optical flow estimation to improve text detection results. Experimental results on different datasets show that the proposed approach outperforms the existing approaches in terms of consistency, multi-lingual ability and generalization. However, the reported results appear low compared to those attained in standard document analysis work. Therefore, we are planning to extend the approach for achieving comparable results as in the area of document analysis. Next, our plan is to de-

velop a text detection and recognition method for robots that track plants based on labels to pour water in farms.

## Acknowledgement

The work described in this paper was supported by the Science Foundation for Distinguished Young Scholars of Jiangsu under Grant No. BK20160021, the Natural Science Foundation of China under Grant No. 61672273 and No. 61272218, and partly supported by the University of Malaya HIR under Grant No. UM.C/625/1/HIR/210. The authors would like to thank Mohamed Lubani, Research Assistant, University of Malaya for his support in implementing the initial steps.

## References

[1] C. Zhang, X. Xu, M.L. Shyu, Q. Peng, Integration of visual temporal information and textual distribution information for news web video event mining, IEEE Trans. HMS (2016) 124–135.
[2] K.L. Bouman, G. Abdollahaian, M. Boutic, E.J. Delp, A low complexity sign detection and text localization method for mobile applications, IEEE Trans. Multimedia 13 (2011) 922–934.
[3] C. Yi, Y. Tian, Scene text recognition in mobile applications by character descriptor and structure configuration, IEEE Trans. IP (2014) 2972–2982.
[4] A. Hartl, G. Reitmayr, Rectangular target extraction for mobile augmented reality applications, in: Proc. ICPR, 2012, pp. 881–884.
[5] K. Jung, K.I. Kim, A.K. Jain, Text information extraction in images and video: a survey, Pattern Recognit. (2004) 977–997.
[6] Q. Ye, D. Doermann, Text detection and recognition in imagery: a survey, IEEE Trans. PAMI (2015) 1480–15000.
[7] D. Zeng, Y. Bao, K. Liu, F. Zhao, Q. Tian, Face database generation based on text-video correlation, Pattern Recognit. (2016) 240–249.
[8] B. Epshtein, E. Ofek, Y. Wexler, Detecting text in natural scenes with stroke width transform, in: Proc. CVPR, 2010, pp. 2963–2970.
[9] Y.F. Pan, X. Hou, C.L. Liu, A hybrid approach to detect and localize texts in natural scene images, IEEE Trans. IP (2011) 800–813.
[10] T.Q. Phan, P. Shivakumara, S. Tian, C.L. Tan, Recognizing text with perspective distortion in natural scenes, in: Proc. ICCV, 2013, pp. 569–576.
[11] K. Jung, K.I. Kim, A.K. Jain, Text information extraction in images and video, Pattern Recognit. (2004) 977–997.
[12] D. Chen, J.M. Odobez, Video text recognition using sequential Monte Carlo and error voting methods, Pattern Recognit. Lett. (2005) 386–1403.
[13] H. Li, D. Doermann, O. Kia, Automatic text detection and tracking in digital video, IEEE Trans. IP (2000) 147–156.
[14] M.R. Lyu, J. Song, M. Cai, A comprehensive method for multi-lingual video text detection, localization and extraction, IEEE Trans. CSVT (2005) 243–255.
[15] P. Shivakumara, A. Dutta, C.L. Tan, U. Pal, Multi-Oriented Scene Text Detection in Video based on Wavelet and Angle Projection Boundary Growing, MTA, Springer-Verlag, 2013.
[16] P. Shivakumara, T.Q. Phan, S. Lu, C.L. Tan, Gradient vector flow and grouping based method for arbitrarily-oriented scene text detection in video images, IEEE Trans. CSVT (2013) 1729–1739.
[17] W. Huang, P. Shivakumara, C.L. Tan, Detecting moving text in video using temporal information, in: Proc. ICPR, 2008.
[18] J. Zhou, A robust system for text extraction in video, in: Proc. ICMV, 2007, pp. 119–124.
[19] C. Mi, Y. Xu, H. Lu, X. Xue, A novel video text extraction approach based on multiple frames, in: Proc. ICICSP, 2005, pp. 678–682.
[20] Y.K. Wang, J.M. Chen, Detection video texts using spatial-temporal wavelet transform, in: Proc. ICPR, 2006, pp. 754–757.
[21] X. Huang, A novel approach to detecting scene text in video, in: Proc. CISP, 2011, pp. 469–473.
[22] X. Huang, H. Ma, H. Yuan, A novel video text detection and localization approach, in: Proc. PCM, 2008, pp. 525–534.
[23] X. Zhao, K.H. Lin, Y. Fu, Y. Hu, Y. Liu, T.S. Huang, Text from corners: a novel approach to detect text and caption in videos, IEEE Trans. IP (2011) 790–799.
[24] X. Liu, W. Wang, Robustly extracting captions in videos based on stroke-line edges and spatio-temporal analysis, IEEE Trans. MM (2012) 482–489.
[25] L. Li, J. Li, Y. Song, L. Wang, A multiple frame integration and mathematical morphology based technique for video text extraction, in: Proc. ICCIA, 2010, pp. 434–437.
[26] A. Mosleh, N. Bouguila, A.B. Hamza, Automatic inpainting scheme for video text detection and removal, IEEE Trans. IP (2013) 4460–4472.
[27] L. Wu, P. Shivakumara, T. Lu, C.L. Tan, Text detection using Delaunay triangulation in video sequences, in: Proc. DAS, 2014, pp. 41–45.
[28] P. Shivakumara, M. Lubani, K.S. Wong, T. Lu, Optical flow based dynamic curved video text detection, in: Proc. ICIP, 2014, pp. 1668–1672.
[29] L. Wu, P. Shivakumara, T. Lu, C.L. Tan, A new technique for multi-oriented scene text line detection and tracking in video, IEEE Trans. MM (2015) 137–1152.
[30] V. Khare, P. Shivakumara, P. Raveendran, A new histogram oriented moments descriptor for multi-oriented moving text detection in video, ESWA (2015) 7627–7640.
[31] V. Khare, P. Shivakumara, P. Raveendran, A blind deconvolution model for scene text detection and recognition in video, Pattern Recognit. (2016) 128–148.
[32] S. Roy, P. Shivakumara, H.A. Jalab, R.W. Ibrahim, U. Pal, T. Lu, Fractional Poisson enhancement model for text detection and recognition in video frames, Pattern Recognit. (2016) 433–447.
[33] A. Zhu, R. Gao, S. Uchida, Could scene context be beneficial for scene text detection, Pattern Recognit. (2016) 204–215.
[34] H. Xu, G. Zhai, X. Yang, Single image super-resolution with detail enhancement based on local fractal analysis of gradient, IEEE Trans. CSVT (2013) 1740–1754.
[35] Yong Xu, Hui Ji, Cornelia Fermüller, Viewpoint invariant texture description using fractal analysis, IJCV (2009) 85–100.
[36] B.K.P. Horn, B.G. Schunk, in: Determining Optical Flow, 17, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, U.S.A. Artificial Intelligence, 1981, pp. 185–203.
[37] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L.G.I. Boorda, S.R. Mestre, J. Mas, D.F. Mota, J.A. Almazan and L.P. De las Heras, "ICDAR 2013 robust reading competition", In Proc. ICDAR, 2013, 1115–1124.
[38] P.X. Nguyen, K. Wang, S. Belongie, Video text detection and recognition: dataset and benchmark, in: Proc. WACV, 2014.
[39] C. Yao, X. Bai, W. Liu, Y. Ma, Z. Tu, Detecting text of arbitrary orientations in natural scene images, in: Proc. CVPR, 2012, pp. 1083–1090.
[40] Tesseract. http://code.google.com/p/tesseract-ocr/.

**Shivakumara P** received B.Sc., M.Sc., M.Sc Technology by research and Ph.D degrees from the University of Mysore, Mysore, Karnataka, India in 1995, 1999, 2001 and 2005, all in computer science. Currently, he is working as a Senior Lecturer at University of Malaya (UM), Kuala Lumpur, Malaysia. From 1999 to 2005, he was Project Associate at the University of Mysore, where he obtained M.Sc Technology by research degree and Ph.D degree in computer science and he conducted research on pattern recognition, image processing, document image processing. He worked as a Research Fellow at National University of Singapore, Singapore from 2005–2007. He worked as Research Consultant at Nanyang Technological University, Singapore. He joined back to National University of Singapore, Singapore as a Research Fellow for the period of five years from 2008–2013. Based on his work, he has published more than 160 research papers in national, international conferences and journals. He has been serving as Associate Editor for Transactions on Asian Language Information Processing (TALIP). He got Top Reviewer Recognition award from Pattern Recognition Letter Journal. He has been serving as a Program Committee Member (PCM) for the several International Conferences. His area of research includes video text understanding, document analysis and image processing related.

**Liang Wu** is currently working toward the M.S. degree in computer science and technology at Nanjing University, Nanjing, China. His current research interests include media data analysis, computer vision, and pattern recognition algorithms.

**Tong Lu** received the PhD degree in computer science from Nanjing University in 2005. He received his M.Sc. and B.Sc. degree from the same university in 2002 and 1993, respectively. He served as Associate Professor and Assistant Professor in the Department of Computer Science and Technology at Nanjing University from 2007 and 2005. He is now a full Professor at the same university. He also has served as Visiting Scholar at National University of Singapore and Department of Computer Science and Engineering, Hong Kong University of Science and Technology, respectively. He is also a member of the National Key Laboratory of Novel Software Technology in China. He has published over 60 papers and authored 2 books in his area of interest, and issued more than 20 international or Chinese invention patents. His current interests are in the areas of multimedia, computer vision and pattern recognition algorithms/systems.

**Chew Lim Tan** (M'03–SM'03) received the B.Sc. (Hons.) degree in physics from the University of Singapore, Singapore, in 1971, the M.*I.* degree in radiation studies from the University of Surrey, Surrey, U.K., in 1973, and the Ph.D. degree in computer science from the University of Virginia, Charlottesville, VA, USA, in 1986. He is currently a Professor with the Department of Computer Science, School of Computing, National University of Singapore, Singapore. He has authored or coauthored more than 360 research publications. His current research interests include document image analysis, text and natural language processing, neural networks, and genetic programming. Dr. Tan is an Associate Editor of *Pattern Recognition* and the *ACM Transactions on Asian Language Information Processing*, and is a member of the editorial board of the *International Journal on Document Analysis and Recognition*. He is a member of the Governing Board of the International Association of Pattern Recognition.

**Michael Blumenstein** is a Professor and Head of the School of Software at the University of Technology Sydney, Australia. He was previously the Head of the School of Information and Communication Technology, and also the Dean (Research) of the Science, Environment, Engineering and Technology Group at Griffith University in Queensland, Australia. Professor Blumenstein is an internationally and nationally renowned expert in the areas of Pattern Recognition and Artificial Intelligence (specifically Machine learning and Neural Networks). He has published over 150 papers in refereed conferences, journals and books in these areas. His research and consultancy projects span numerous fields of engineering (e.g. Artificial Intelligence-based long-term bridge performance models for the Queensland bridge network), environmental science (e.g. application of artificial neural networks to a flood emergency decision support system) neurobiology (e.g. automated analysis of multidimensional brain imagery) and coastal management (e.g. a predictive assessment tool for beach conditions using video imaging and neural network analysis).

**Dr. Basavaraj S. Anami** is currently the Principal of K.L.E. Institute of Technology, Hubli, Karnataka. He is one amongst the few who propagated Computer Science education in North Karnataka and is a BoS member in Computer Science in Visvesvaraya Technological University, Belgaum, Karnataka. He has served as a Visiting Professor at Saginaw Valley State University, Michigan, USA during fall of 2006. Having 28 years of experience in teaching computer science both at undergraduate and postgraduate levels, he is presently guiding five members for their Ph.D. His research interests include image processing, natural language processing, speech processing, knowledge based systems and has to his credit around 75 papers published in journals and conference proceedings.