Piece-Wise Linearity Based Method for Text Frame Classification in Video

Nabin Sharma^{a,*}, Palaiahnakote Shivakumara^b, Umapada Pal^c, Michael Blumenstein^a, Chew Lim Tan^d

^aGriffith University, Gold Coast, Queensland 4222, Australia.

^bUniversity of Malaya (UM), Kuala Lumpur-50603, Malaysia

^cComputer Vision and Pattern Recognition Unit, Indian Statistical Institute, India

^dSchool of Computing, National University of Singapore, Singapore

Abstract

The aim of text frame classification technique is to label a video frame as text or non-text before text detection and recognition. It is an essential step prior to text detection because text detection methods assume the input to be a text frame. Consequently, when a non-text frame is subjected to text detection, the precision of the text detection method decreases because of false positives. In this paper a new text frame classification approach based on component linearity is proposed. The method firstly obtains probable text clusters from the gradient values of the RGB images of an input video frame. The Sobel edges corresponding to the text cluster are then extracted and used for further processing. Next, the method proposes to eliminate false text components before undertaking a linearity check where the linearity of the text components is determined using their centroids in a piece-wise manner. If the components in a frame satisfy the defined linearity condition, then the frame is considered as a text frame; otherwise it is considered as a non-text frame. The proposed method has been tested on standard text and non-text datasets of different orientations to demonstrate that it is independent of orientation. A comparative study with the existing method shows that the proposed method is superior in terms of classification rate and processing time.

Keywords: Video Document Processing, Component Centroid Linearity, Text Frame Classification, Video Text Processing, Video Text Detection

1. Introduction

Text detection and recognition in both video and natural scene images is a popular research topic because of several associated useful applications such as exciting events identification in sports video, accurate indexing, and retrieval of events. Additionally, text detection with an Optical Character Recognizer helps in bridging the gap between low level and high level features (semantic features), resulting in a close representation of the video's content [1-4]. As a result, numerous methods for text detection have been proposed

*Corresponding author

Email addresses: nabin.sharma@griffithuni.edu.au (Nabin Sharma), shiva@um.edu.my (Palaiahnakote Shivakumara), umapada@isical.ac.in (Umapada Pal), m.blumenstein@griffith.edu.au (Michael Blumenstein), tanc1@comp.nus.edu.sg (Chew Lim Tan)

to date in the literature with most of the text detection methods considering frames containing at least a portion of text (text frame) as input rather than considering the whole video containing both text and non-text frames. In other words, for these methods, only selected text frames from the video are to be input for text detection.

Selecting a text frame manually is problematic since videos contain many duplicate frames resulting in a large amount of data, containing both text and non-text frames. In the case, for example, that the text detection methods are used for text frame classification by finding text appearance/disappearance in video, most of the time, the methods classify non-text frames as text frames because the method identifies non-text regions as text regions for the non-text frames. This results in poor precision and low recall. The main reason for the poor performance is that these methods assume that the text frames are available and are input for text detection. Consequently, it is difficult to directly use text detection methods for text frame classification. This is illustrated in Figure 1 where a text detection method identifies text lines accurately for text frames as shown in Figure 1(a) and (c), but the same method also detects non-text as text for non-text frames as shown in Figure 1(b) and (d). Work proposed in [5] for text detection and text frame classification, also clearly shows that these text detection methods give poor accuracy for text frame classification. In addition, the presence of both graphics text (which is edited text) and scene text (which exists naturally) makes the text frame classification problem more complex and challenging. Since graphics are considered as edited text, they are easy to detect, whereas scene text is part of the image, and characteristics such as contrast, fonts, font size, orientation, background and text movements are unpredictable when compared to graphics text. This paper aims to address this complex issue of classifying text and non-text frames to enhance the performance of text detection and recognition methods in video.

Figure 1: Samples of text and non-text frames

It may be noted that text detection in document images and natural scene images is not a new problem

[1-4]. However, document image text detection methods cannot be used for text detection in video directly because these methods are developed based on the fact that text in a document image usually has a high resolution and it preserves the shape of the characters [6-8]. These constraints are not necessarily true for text in video because video frequently has a low resolution and complex background. As a result, there is a possibility of losing the shape of the characters, and single characters may be split into several sub-components.

Similarly text detection methods in video can be classified into three categories, namely, connected components [6], texture [9-13], and gradient-based methods [14-18]. Since connected component-based methods work as per the methods used for document analysis, the methods are not appropriate for complex backgrounds. Conversely, texture-based methods are good for complex backgrounds but are sensitive to fonts, and font size in addition to having considerable computational expense. On the other hand, gradient-based methods are fast compared to texture-based methods, but these methods are sensitive to background. And therefore, produce more false positives resulting in low precision. Additionally, most of the text detection methods focus on horizontal graphics text detection, but not on multi-oriented scene text detection and arbitrarily-oriented text detection in video. Multi-oriented and arbitrary scene text detection methods [13, 14, 17, 18] which detect both graphics and scene texts with reasonably good accuracy are therefore rarely presented in the literature. In summary, though text detection methods work well irrespective of orientation, script type, contrast etc., when the same methods are run on non-text frames, the methods misidentify non-text regions as text regions though there is no text in the frame. Hence, the precision becomes low due to false positives identified by the method.

In order to improve precision or remove false positives given by the text detection methods, some methods [19, 20] have been developed with the intention of text block verification and by considering text blocks detected by text detection methods as input. However, since these methods require text blocks as inputs, they cannot be used for text frame classification.

A few methods have been proposed for text frame classification to improve the performance of text detection methods by classifying text frames from video automatically before applying text detection methods. For instance, Shivakumara et al. [21] proposed a method for text frame classification at the block level and for each block. The method tests the mutual nearest neighbour criterion based on wavelet and moments to identify the presence of text. If any one of the blocks out of 16 satisfies the mutual nearest neighbour symmetry criterion, then the whole frame is said to be a text frame; otherwise it is considered as a non-text frame. Though this works irrespective of orientation, scripts etc., the method gives poor accuracy for text frame classification because of the strict conditions imposed for identifying the presence of text in the blocks. In another work, Shivakumara et al. [22] proposed a method which also works at the block level and in which edge-based features such as the proximity of edges, height and straightness of the edges are explored to classify text and non-text blocks. The division of fixed size blocks for frames can also slice the text

portion into arbitrary fragments, leading to the loss of text properties in the blocks. This may result in text fragments at the block level tending to become non-text like structures. Therefore, the accuracy for text frame classification is low compared to non-text frame classification. In view of the previously discussed limitations with current methods, in the current work, we propose effective and efficient methods based on studying the linearity of the text components in a text line for video text frame classification. The main contribution of the proposed method is obtaining text candidates for testing linearity, as well as the novel idea in which linearity testing of the text components is performed in a text line.

2. Proposed Method

Generally, video frames contain text with low contrast due to low resolution as compared to camera-based video, which has better resolution. Additionally, although text does not occupy the whole frame, the location of the text is not fixed and can appear at any position in a video frame. Hence, the proposed method considers examining the whole frame instead of dividing it into smaller blocks. Although the whole frame is considered for text frame classification in the present work, it differs from text detection strategy because text frame classification methods try to search only one text block/region rather than detecting all the text regions.

A high level overview of the proposed method is shown in Figure 2. In order to overcome the limitations of video frames, such as low resolution, or blur, the RGB color sub-bands are considered to enhance the text components. Gradient information from each of the R, G and B color sub-bands and the gray scale image of the input frame are extracted. The gradient information extracted from the three color sub-bands are processed to form an enhanced gradient image (Indirect gradient image), which in combination with the gradient values of the gray scale (Direct gradient image) input frame, produces the text region candidates.

Presence of false text regions/candidates in the text region candidates are quite obvious and needs to be eliminated. Connected component analysis (CCA) based techniques and Histogram of Gradient (HoG) features are used, therefore, to filter the non-text components, which result in the potential text candidates. The potential text candidates are then tested by our proposed piece-wise linearity checking algorithm, which classifies a frame as text or non-text. This algorithm is capable of handling text that may have arbitrary orientations, which makes it superior to state-of-the-art techniques.

The key ideas for achieving good accuracy are the following. (1) The way in which color and gradient information are combined, especially direct gradient and indirect gradient values for identifying text region candidates with the help of k-means clustering algorithm. (2) Combining shape features and geometrical properties of text in different way for eliminating false text candidates. (3) Introducing piece wise linearity check based on text properties for handling arbitrary orientations.

The proposed method is divided into three sections. First, Section 2.1 describes how to obtain text

Figure 2: Logical flow of the proposed method

region candidates from the input frames by exploring a new way of combining color sub-bands and gradient information. Connected component analysis-based features and histograms of gradient features are proposed in Section 2.2 for identifying potential text candidates.. Finally, Section 2.3 presents a new piece-wise linearity check for classifying text and non-text frames.

2.1. Text region candidates extraction

Low contrast, low resolution and blur form the major challenges in processing video for text detection. We propose, therefore, the use of color information of the video frame to enhance the text pixel contrast. The method obtains the three color sub-bands i.e. R, G and B images of the input video frame for enhancement [24] and clustering text and non-text pixels.

Figure 3 analyses the intensity values of text and non-text regions in the three color sub-bands. The text and non-text regions marked in the input frame shown in Figure 3(a) are examined in detail. The intensity values are plotted for the R, G and B sub-band images of the marked text and non-text regions.

Figure 3: Analysis of intensity values in R, G and B sub bands of text and non-text regions in video frames

The R,G and B sub-band images are shown in Figure 3(b), (f) and (j), respectively, and their corresponding intensity graphs are shown in Figure 3(c), (g), and (k). Similarly, the R, G and B sub-band images of the non-text region are shown in Figure 3(e), (i) and (m), respectively, and their corresponding intensity graphs

are shown in Figure 3(d), (h), and (i). Sharp and higher peaks at nearly regular intervals can be noticed in the three intensity graphs of the text region, whereas, the peaks of the intensity graphs are irregular for non-text regions. The intensity graphs reveal that for same pixels color sub-bands finer details are provided at different levels. However, the information content is higher and more regular in the case of a text region and less and irregular in case of non-text regions. This clue helps us to assimilate finer details for the text pixels using the information from the three color sub-bands.

Based on the illustration shown in Figure 3, a gradient operation is performed on the color sub-band images using a Sobel mask [14] to enhance the text pixels and suppress the non-text pixels. In order to integrate the gradient information from three color sub-bands, a new operation is proposed which selects maximum gradient value among the three values of each pixel. This results in a new enhanced gradient image which we call the Indirect Gradient Image (IGI). Figure 4(a) and (d) presents the Indirect Gradient Images for the text and non-text regions shown in Figure 3, respectively. The intensity graphs for the IGI of text and non-text regions are shown in Figure 4(b) and (c). It can be noticed from the graphs that the intensity values in the IGI of the text region now have high gradient values, and this is not true for non-text regions.

Figure 4: Analysis of direct gradient and indirect gradient values for text and non-text regions

The background of a text frame can be of any kind ranging from simple to complex, and from very low to high contrast/resolution. Hence instead of just using IGI as the basis for text candidate extraction, we

propose a new way to separate text and non-text and reduce false positives from non-text frames. Therefore, the gradient operation was performed on the gray scale image of the input video frame, which results in an image which we call the Direct Gradient Image (DGI). The DGIs corresponding to the text and non-text region marked in Figure 3 are shown in Figure 4(e) and (h), respectively. The respective intensity graphs are shown in Figure 4(f) and (g). Intensity graphs of both IGI and DGI of text region have high gradient values, but there are more variations in the peaks of DGI as they are unprocessed gradient values, whereas, the intensity graphs of both IGI and DGI of non-text regions looks nearly same. This observation reveals that when an input image has low contrast, IGI helps in increasing the difference between the gradient values of text and non-text pixels. Conversely, DGI is useful when the input image has high contrast. Combining these two observations can help to overcome the inherent problems with video. Therefore, a k-means clustering algorithm with $k=2$ was applied on both IGI and DGI for clustering the high gradient (text) and low gradient (non-text) values. The cluster with the greatest centroid value is considered as a text cluster. The pixels which are present in the text clusters of both IGI and DGI are considered as text candidates. Thus, color and gradient information are utilized here in a novel way for text candidate extraction.

Figure 5 illustrates the text candidate extraction steps using text and non-text frames. The text frame shown in Figure 5(a) is an example of arbitrary oriented text present on a complex and noisy background, whereas, the non-text frame shown in Figure 5(a) has sharp edges and is also quite complex. The R, G and B sub-band images of the corresponding text and non-text frames shown in Figure 5(b), reveal the high contrast pixels. The gradient operation is performed on the sub-band images shown in Figure 5(b) and the corresponding results are shown in Figure 5(c). It can be seen from Figure 5(c) that the gradient operation enhances text pixels and suppresses non-text pixels. Some of the high contrast pixels in the non-text frame are also enhanced, as shown in Figure 5(c). The IGI obtained using the R, G and B gradient images are shown in Figure 5(d). DGIs from both text and non-text frames are shown in Figure 5(d) and (e) respectively. Figure 5(f) shows the text candidates extracted from the text cluster of both IGI and DGI, for the text and non-text frame.

In this work, we have used the k-means algorithm with $k=2$ for text pixels (high gradient) and non-text pixels (low gradient) clustering. To validate $k=2$ is the correct choice for a text cluster, we tested k-means clustering with $k=2$ and $k=3$ and the results are reported in Figure 6. According to the results in Figure 6, $k=3$ produces more noisy pixels and loses some text pixels as shown in Figure 6(b) compared to results with $k=2$ shown in Figure 6(a). We could have used classifiers such as SVM with different kernels instead of k-means clustering for classification of text and non-text pixels in this work. However, as text candidate selection is a pre-processing step for text frame classification, we preferred unsupervised k-means clustering rather than classifiers

Figure 5: Intermediate results of text and non-text frame processing for finding text region candidates

2.2. Potential text candidates selection

It is noted from the results shown in Figure 5(g) that text candidate images still contain some false text candidates for both text and non-text frames. Therefore, we propose a connected component analysis-based technique to eliminate false text candidates in this section. Firstly, small components (one or two-pixel components) are removed as they do not constitute text. Secondly, the method removes components such as straight lines by checking whether the centroid of the component falls on itself as straight line components may not represent text components. Let $\text{MPT}(x_m, y_m)$ be the mid-point and $\text{CEN}(x_c, y_c)$ be the centroid

Figure 6: Validating the parameter value of 'k' in k-means clustering algorithm for obtaining text candidates

of the component. If the mid-point and the centroid nearly coincide or overlap, the component is considered as straight [24] and is removed. Thirdly, a technique based on the minor axis of the component, which considers the length of the minor axis, is proposed to remove false text candidates as shown in Figure 7(a) which illustrates that this rule is effective in removing false text candidates.

Figure 7: Potential text candidates selection after filtering false text candidates

The heuristic rule is defined as the ratio between the distance (D) of the two furthest points of the components and the minor axis length (Min_L) of the component. If $D > Min_L$, the component is removed. Next, we propose a Histogram Oriented Gradient (HoG) [23] feature to identify false text candidates as it is noted from [23] that HoG gives a unique gradient angle distribution for text components. The method finds 16 directions using HoG for each of the components. Text components are then verified based on the

fact that text edges usually have very high variance in the edge orientation. The components that have low variation are removed. This is evidenced in Figure 7(b), where the method has eliminated most of the non-text components from the non-text frame and retained most of the text components in the text frame. As a result, potential text candidates are obtained. Figure 7(b) shows false candidate elimination rules, which also sometimes eliminate text components. Since the aim of this work is to classify text and non-text frames, the method does not require full text information as in text detection. Therefore, removal of some text components does not have an effect on classification. These potential text candidates are used for linearity checking to classify text and non-text frames. This will be discussed in a subsequent section.

Figure 8: Classification of text and non-text frames using linearity property of potential text candidates

2.3. Piece-Wise Linearity (PWL) for text frame classification

For the potential text candidates of both text and non-text frames, we propose a new method to find linearity and non-linearity based on the fact that text components share regular patterns such as uniform size, spacing between components, and direction of the components in contrast to non-text ones in non-text frames. We combine size of the components, distance of the components and direction of the components to define linearity and non-linearity in order to classify text and non-text frames. This work proposes a simple linearity check rather than using standard methods such as the Hough transform for line detection or circle detection etc. This is because the main objective of the method is to classify a given frame as text or non-text with a minimum number of computations. Therefore, we preferred to use a simple criterion, which checks linearity based on the fact that character components are usually arranged in a regular fashion with uniform spacing between the characters in a text line. The criterion should also work for arbitrary text lines (e.g. circle-shaped text lines) and should get character components of the same line. Though the Hough transform (HT) is good for checking linearity of straight lines and circles, it requires more computations and storage. Moreover, we were not sure how HT works for a frame with multiple arbitrary oriented text lines as HT may not consider proximity between the components for estimating angle of the components. Since the objective of this work is to classify text frames where the text lines can be of any orientation, we found centroids for each component and then estimated a proximity matrix. The proximity matrix is the distance between all centroids. The method traces the minimum distance and next minimum distance and so on for at least three components. Then the method checks the size of the components, proximity between the components and orientation of the components to define linearity. If any one of the group of components satisfy the linearity property, then the frame can be considered as a text frame; otherwise it is a non-text frame. In the case of components in non-text frames, the above linearity may not exist due to irregular shapes of the components, non-uniform spacing between the components and different directions of the components. Thus, the proposed method is said to be a piece-wise linearity check, which works well for the frames containing arbitrary texts. This is illustrated in Figure 8 where (a)-(d) shows the piece-wise linearity check for the frame containing arbitrary text, horizontal text, non-horizontal text and no text, respectively. Figure 8(a) shows that for complex background text frames, the methods find centroids for the potential text candidates as illustrated in the second results in Figure 8(a) and the linearity check using proximity and size can be seen in graphs in Figure 8(a). The graph in Figure 8(a) shows clearly where linearity exists for the component in the arbitrary text frame for both proximity as well as size. Similarly, for the frame containing horizontal text and non-horizontal text, the piece-wise linearity exists as shown in Figure 8(b) and (c), respectively. It is observed from Figure 8(d) that for the frame containing no text, the method does not find linearity as is evident from the graph in Figure 8(d) where both proximity and size do not satisfy the linearity property. This is because of irregular spacing between the potential components and irregular size of the components as shown in the second results in Figure 8(d). Therefore, it is considered

as a non-text frame. In this way, the proposed method classifies text and non-text frames irrespective of the orientation of the text. The algorithmic steps for the linearity check are given below, which we call Arbitrary Linearity Estimation (ALE).

ALE Algorithm

1. Proximity matrix (PM) of the centroids of all components is formed using the distance between the centroids is formed.

Let $CT = \{CT_1(x_1, y_1), CT_2(x_2, y_2), \dots, CT_n(x_n, y_n)\}$ be the set of centroids of all the components. The proximity matrix (PM) is defined by:

$$PM(r, c) = \sqrt{(x_r - x_c)^2 + (y_r - y_c)^2}, \quad (1)$$

Where, $1 \leq r \leq n, 1 \leq c \leq n$, and n is the total number of the components left in the image after false positive elimination.

2. Use the proximity matrix and group the nearest neighbour component based on the following criteria, Let $Ct_1(x_1, y_1)$ and $Ct_2(x_2, y_2)$ be the centroids and Ht_1 and Ht_2 be the heights of the components C_1 and C_2 , respectively. Let $S(C_1)$ be the characteristic scale of the component C_1 . $S(C_1)$ is defined as, $S(C_1) = L_{Maj}(C_1) + L_{Min}(C_1)$, where, $L_{Maj}(C_1)$ and $L_{Min}(C_1)$ are the major and minor axis lengths of the component C_1 .

- a. Proximity Check: $PM(C_1, C_2) \leq \delta \cdot S(C_1)$
- b. Size Consistency Check: $S(C_1) \approx S(C_2)$
- c. If the components satisfies the above conditions, form a Group G_1 (say)

3. Linearity Check: If a group is formed based the mentioned conditions, then the linearity of the group is checked if there are at least 3 components in the group, in the following way,

- a. The 1st and 2nd Components in the group G_1 are used to find the orientation, say O_1 . The 3rd component in the group is include to find the orientation, i.e 1st, 2nd, and 3rd component are together used to find the orientation, say O_2 . In the similar way all the component in the group are used find the orientation, and a set of orientation values $Ori = \{O_1, O_2, O_{n-1}\}$ is formed.
- b. If the variance of the orientation values present in the set Ori is very less or $< T$, the group is considered to be a text patch as the component are nearly linear, otherwise all the other groups are examined for linearity.
- c. If none of the groups satisfy the linearity check, the frame is considered as Non-text.

The threshold values of δ for proximity and T for variance in orientation were fixed based on an experimental analysis. The details of the experimental analysis and the threshold values selected are given in Section 3.1.

The main advantage of the step (ii) mentioned in ALE is that since it works based on a nearest neighbour criterion, which finds the nearest component based on distance between the centroids, this step helps in finding successive components in the same line to check linearity property. The linearity property looks for at least three such components for classification. Ordering of components or sequencing of components does not arise here because our purpose is to check linearity rather than character/word recognition. Therefore, any combination of three successive character components in a text line is enough for our linearity check algorithm. If a frame contains two or more lines with different orientations and with little space in between, ordering of character components is important as it may consider one component from each text line. In this case, the linearity check may fail. However, this situation seldom arises because the space between the characters in a text line is generally less than the space between the text lines. The nearest neighbour criterion used for grouping also takes care of finding components from the same text line as this criterion checks not only the distance but also properties of text, such as size and orientation.

(a) Text frame with multiple text lines

(b) Selected potential text candidate group marked in red

(c) Graph for linearity check properties of the text candidate shown in (b)

Figure 9: Illustration for classification of text frames having multiple text lines with different orientations

Another advantage of the proposed linearity check is that when the frame contains multiple text lines with different orientations, the method classifies the text frames successfully irrespective of orientations. This is because the method considers any three successive components of any line regardless of orientation to check linearity. Once the method finds a text component group which satisfies the linearity check, the input frame is labelled as text without verifying other components in other lines. This helps us to save time. Therefore, multiple lines with different orientations do not affect the classification accuracy. An example of a text frame having multiple lines with different orientations is shown in Figure 9(a). This Figure illustrates the scenario of how the linearity check works when multiple text lines with different orientations are present in a frame. Figure 9(b) shows the potential text candidates; the candidate which satisfied the linearity check

is marked in red. The linearity check graph corresponding to the marked text component group is shown in Figure 9(c).

3. Experimental Results

We conducted experiments on a variety of datasets comprising video frames containing arbitrary text, non-horizontal text, horizontal text, a publicly-available video dataset (Hua’s) and publicly available camera-based images (ICDAR 2003 competition data) to evaluate the performance of the proposed method in terms of classification rate. To study the effectiveness of the proposed method, we compared it with an existing method [21]. Note that the existing method was tested on text frames containing horizontal and non-horizontal text but not curved text and non-text frames. The method [21] divides the given input frame into equal sized sub-blocks. The probable text blocks are identified based on wavelet and moment combination features with a k-means clustering algorithm. The combined features are used for each probable text block to identify the text candidates which represent text. Subsequently, the mutual nearest neighbour criterion has been proposed for identifying the presence of text, which in turn is used for classification of text frames. As the method applies an expensive transform such as wavelets, the moments combination for each sliding window of the probable text block and connected component analysis to check the mutual nearest neighbour criterion, it is expensive compared to the proposed method. The reason for the poor accuracy is the division of the video frames into fixed size blocks, which slices the text portion and hence, leads to the loss of text properties in the block. In contrast, the proposed method uses the information from the whole frame instead of dividing it into blocks, which helps to preserve the text edges and avoids slicing. The proposed method is also less time consuming because it does not involve expensive transforms and moments computations. There is no comparative study for frames containing curved text. Our database includes 1220 text frames and 800 non-text frames, 45 frames of Hua’s dataset [25], 140 frames containing curved text lines and 215 camera based images. In total, 2405 (1220+800+140+45) video frames and 251 camera based images are used for the purpose of experimentation. All the experiments are conducted on a PC with a Core i5 2.60 Ghz Processor having 4 GB RAM running on the Windows 7 operating system.

We have used Recall (R), Precision (P), F-measure (F), False Positive rate (FP) and Average Time Processing (AVT) as measures to show that text detection methods may not be suitable for text frame classification by testing on both text and non-text frames and have followed the instructions stated in [21]. More details about the three measures can be found in [21]. For the purpose of this study, we have tested our text detection method in [14] which works well for any orientation of text and is a state-of-the-art method as it may receive both text and non-text frames.

The proposed HoG features in Section 2.2 may eliminate some of the text components because they may share the same shape properties with the text-like components created by background complexity/noise.

Since our method requires three components for classification, if HoG eliminates some of the text components, it does not substantially affect the overall performance of the method. To validate this assertion, we conducted a new experiment on our video dataset to calculate recall, precision and F-measure for potential text candidate selection. In this experiment, if any of three successive components in a text line are selected as potential text candidates, we consider that the text line is detected correctly for calculating the above measures. The results are reported in Table 1. According to Table 1, the method gives promising results for 1200 video frames. It is also observed from Table 1 that the proposed text candidate selection method gives almost consistent results for all three measures. This shows that discarding a few text components as non-text components by HoG does not substantially affect the overall accuracy of the system.

Table 1: Validating HoG features for potential text candidate selection

Dataset	Recall	Precision	F-measure
Text Frame (1200 frames)	0.85	0.86	0.85

3.1. Threshold selection for ALE algorithm

The threshold δ for proximity and T for variance in orientation used in ALE algorithm were fixed based on experimental study. Three different experiments were conducted to estimate the threshold values. In the first two experiments the possible ranges δ and T were found. In the third experiment, the optimal values of δ and T were estimated based on the classification results using the possible range of δ and T.

It was found that the space between the characters has some relationship to the characteristic scale (Major axis length + Minor axis length of a character component) of the characters. In order to validate this hypothesis, an experiment was conducted on the dataset consisting of 1542 pairs of character components taken from 300 text frames. The text frames were randomly taken from the text frame datasets used for testing the proposed method. In this first experiment, the ratio (δ) of the distance between character pairs and their characteristic scale was examined. A cumulative frequency distribution of the ratio was plotted as shown in Figure 10 (a). From the figure it was noted that more than 90% of the character pairs can be obtained if the range of δ between 0.3 to 1.5 is considered. However, to allow some tolerance the range of δ was considered from 0.3 to 1.7.

Similarly, the range of threshold value T was estimated based on the second experimental study. The variance of angles in a text component group is supposed to be zero for the ideal case of a text line (e.g. straight line). However, estimating orientation angle for arbitrary orientation text line, especially circle-shaped text line, is challenging due to the different angles of each character pair in the line. Hence, in order to understand how the difference in orientation of character components in a text line varies, this experiment was conducted on a dataset of 540 text lines taken from the 300 text frames which were also used for the

(a) Cumulative frequency distribution of text components based on Character distance and characteristic scale ratio

(b) Cumulative frequency distribution of the variances in the orientation of the text component groups

(c) Classification accuracy obtained at various combinations of thresholds δ and T

Figure 10: Experimental study for selection of δ and T

estimation of δ . In this experiment, the variance of the orientation values of different text component group in a line is examined and the frequencies of the variance values are plotted in Figure 10 (b). From the figure we noted that more than 90% of the text lines can be obtained if we consider T greater than 8. Hence, the values of T starting from 8 were considered in the third experiment.

Figure 10 (c) shows the graph of text frame classification accuracy obtained using various combinations of values of δ and T. Using the range for δ (0.3, 0.5, ..., 1.7) obtained from first experiment, and T (8, 10, 12, ...) obtained from the second experiment, the optimal values for both the thresholds were found from a classification experiment. This third experiment was conducted on a dataset containing 600 video frames (300 text frames used in the first two experiments and 300 non-text frames). The graph shows that better accuracy was obtained at T=10, whereas, the accuracy decreased when T was increased to 12. Hence the value of T was fixed to 10. From the experiment it was also noted that classification accuracy of 85.81% was achieved when T=10 and $\delta=1.5$ were considered. Although the accuracy increases slightly at $\delta=1.7$ and T=10, however to reduce false positives and allow some tolerance to handle arbitrary components, fonts

and font size, the value of δ was fixed as 1.5. Thus, the thresholds for T and δ were decided as 10 and 1.5, respectively.

3.2. Suitability of text detection methods for text and non-text frames classification

To study the suitability of the text detection methods for text frame classification, we conducted two kinds of experiments (EXP1 and EXP2): Calculating recall, precision and F-measure for both text and non-text frames before classification and after classification. Both text frame and non-text frames were given as inputs to text detection methods before classification in EXP1 and only text frames were given as input to text detection methods after classification in EXP2. For both the experiments, the 1220 text and 800 non-text frame datasets were used. Sample qualitative results of the text detection method on non-text frames are shown in Figure 11 where (a) shows an input frame having different backgrounds and (b) shows the output of the text detection method. It is observed from Figure 11(b) that the text detection method detects non-text as text for the non-text frames and, as a result, it misclassifies non-text frames as text frames. The quantitative results of the text detection method are reported in Table 2, where it can be seen that the recall, precision, F-measure and false positive rate before classification are higher than the results after classification. It can also be noted that precision is very low and the false positive rate is very high compared to when it is processed after classification. This shows that the text detection method detects non-text as text for the non-text frames leading to the false positives that are high and to precision that is low. On the other hand, the same text detection method gives good results after classification because it considers only text frames for text detection. The average processing time is also high before classification compared to after classification. Furthermore, when the image contains multiple text lines with arbitrary orientations as shown in Figure 12(a), the text detection methods require more time (8.58 seconds) because the method has to detect all lines without missing any text components as shown in Figure 12(b). On the other hand, the proposed method requires any three successive components as shown in Figure 12(c) for classification and is therefore faster (1.3 seconds). It can be inferred from the two experiments, therefore, that the text detection methods are not suitable for text frame classification on video data containing large numbers of text and non-text frames.

Table 2: Results of text detection before and after text frame classification

Experiment	P	R	F	FP	APT
EXP1 (before classification)	0.27	0.76	0.39	0.73	15.52
EXP2 (after classification)	0.84	0.81	0.83	0.16	9.32

(a) Sample non-text frames chosen from our database

(b) False positives detected by the text detection method [14] for non-text frames in (a)

Figure 11: Misclassification of non-text frames as text frames by the existing text detection method [14]

(a) Frame with multiple text lines

(b) Result of text detection method [14]

(c) Proposed method

Figure 12: Illustration of design objectives of text detection and text frame classification method: The proposed method takes 1.3 seconds for classification of the frame in (a) and the text detection method [14] takes 8.58 seconds for text detection as shown in (b)

Table 3: Classification rate (in %) of the proposed text candidates + Linearity and existing MSER + Linearity check for text frame classification on Video data of 1200 text frames and 800 non-text frames

Method	Text frame data			Non-text frame data		
	Text	Non-text	APT	Text	Non-text	APT
Proposed text candidate + Linearity	87.13	12.87	1.02	3.75	96.25	0.92
MSER + Linearity	71.22	28.78	6.54	7.35	92.65	6.10

According to the literature, the use of Maximally Stable External Region (MSER) [26] for identifying text candidates is common for text detection and recognition. To show the usefulness of the proposed

Figure 13: Text candidates obtained from MSER regions and the proposed technique

classification method, we replaced the text region candidate extraction step of the proposed method with MSER for classification. In other words, we used text candidates given by MSER as shown in Figure 13(b) for the input images in Figure 13(a) as input to the potential text candidate selection step and linearity check. The results reported in Table 3 show that MSER + the Linearity check requires more time compared to the proposed text region candidates + Linearity check. In addition, the MSER + Linearity check reports low accuracy compared to the proposed method. One of the major reasons behind the low performance of MSER is its sensitivity to blur, which is a common problem with video text. Another reason is discretization of the similar stable regions, which results in detection of multiple smaller blobs than actually expected, as well as over lapping blobs. An example of text candidates obtained using our proposed method and from the MSER regions are shown in Figure 13. The results show that a greater number of text regions/components are detected by the proposed method than that of MSER as shown Figure 13(c). It can be seen clearly in Figure 13(d) that characters 'L' and 'A' in the words 'CLOWN' and 'CHAMP' are missed by the MSER algorithm. On the other hand, most of the text components are extracted correctly by our proposed technique. This

clearly shows that the proposed method is less sensitive to background noise and blur compared to MSER. MSER + Linearity check requires more time for processing because MSER involves a substantial number of computations to determine the maximally stable boundary using grey values.

3.3. Experiments on text frames containing curved text

To demonstrate that the proposed method classifies text frames containing arbitrary oriented text, we evaluated the proposed method in terms of classification rate by conducting experiments on frames having curved text. Sample results of the proposed method are shown in Figure 14 where (a) shows input frames having different orientations of texts, (b) shows potential text candidates detected by the proposed method and (c) shows the piece-wise linearity for the potential text candidates given by the proximity and size of the components to classify them as text frames. The quantitative results of the proposed method are reported in Table 4 which evidences that classification rate for text frames is promising and encouraging but classification rates for non-text frames is high. This is because detecting potential text candidates in text frames containing curved text is not as easy as non-horizontal text frames. There is scope, therefore, for improvement. However, the average processing time is not high for finding potential text candidates.

Table 4: Classification rate of the proposed method for curved text frames

Method	Classification Rate (in %)		APT
	Text	Non-text	
Proposed method	67.95	32.05	1.07

3.4. Experiments on text frames containing horizontal/non-horizontal text and non-text frames

The qualitative results of the proposed method are shown in Figure 15 where (a) and (d) show input text frames of different text lines with different backgrounds, (b) and (e) show the potential text candidates detected by the proposed method, and (c) and (f) show the graphs of proximity between the components and size of the components to check linearity. It can be observed from Figure 15(c) and (f) that the proposed method finds linearity for the curved potential text candidates and hence those frames are classified as text frames. Similarly, for the frames containing no text shown in Figure 16(a), the potential text candidates detected by the proposed method shown in Figure 16(b) do not satisfy the linearity property as shown in Figure 16(c). Therefore, those frames are classified as non-text frames. The quantitative results of the proposed and existing methods for both text frame and non-text frame data are reported in Table 5 which illustrates that the proposed method is good for text frame classification as the classification rate is

(a) Sample curved/arbitrary text frames

(b) Potential text candidates and the candidate which passed linearity check is marked in red

(c) Linearity property for arbitrary text based on distance between potential text components and size of the components

Figure 14: Classification of arbitrary text frame as a text frame based on linearity property

higher than the existing method, but for non-text frame data, the classification rate is slightly lower than the existing method. However, average processing time in seconds of the proposed methods is lower than the existing method. This is because of the linearity property defined by the proposed method and strict measures based on mutual nearest neighbour criteria used by the existing method for classification. The strict measure defined by the existing method is good for classification of non-text frames but not classification of text frames. In this sense, the proposed method is better than the existing one if we consider the overall accuracy.

3.5. Experiments on the Hua dataset

We conducted experiments on this publicly-available dataset providing independent evaluation of the proposed method. Sample results of the proposed method are shown in Figure 17 where (a) shows the input

(a) Sample horizontal text frames taken from the text frame dataset

(b) Potential text candidates and the candidate which passed linearity check is marked in red

(c) Distribution of distance and size of the selected text components

(d) Sample non-horizontal text frames taken from the text frame dataset

(e) Potential text candidates and candidate which passed linearity check is marked in red

(f) Distribution of distance and size of the selected text components

Figure 15: Classification of horizontal and non-horizontal text frames as text frames using linearity property.

(c) Linearity property does not satisfy for the potential text candidates in (b) for the non-text frames shown in (a)

Figure 16: Frames with no text classified as non-text frames

Table 5: Classification rate of the proposed and existing methods for text and non-text frame data

Method	Text frames data			Non-text frames data		
	Text	Non-text	APT	Text	Non-text	APT
Proposed method	87.13	12.87	1.02	3.75	96.25	0.92
Shivakumara et al. [21]	74.13	25.87	2.04	2.14	97.86	2.04

frames, (b) shows potential text candidates given by the proposed method and (c) shows linearity graphs of the proximity between the components and the size of the components. Figure 17(c) shows that the potential text candidates of the input frames satisfy the linearity property and that therefore, frames can be classified as text frames correctly. The quantitative results of the proposed and existing methods are reported in Table 6 which illustrates that the proposed method is better than the existing method for the same reasons as discussed in the previous section.

(a) Input text frames

(b) Potential text candidates

(c) Linearity property of the potential text candidates using distance and size of the components

Figure 17: Classification of sample frames from Hua's data as text frames

Table 6: Classification rate of the proposed and existing methods for Hua's dataset

Method	Classification Rate (in %)		APT
	Text	Non-text	
Proposed method	97.62	100	1.05
Shivakumara et al. [21]	75.54	24.46	2.04

3.6. Experiments on ICDAR 2003 dataset

This sub-section discusses experiments on one more publicly available camera-based images dataset. These images have high resolution compared to video frames since these images are captured by high resolution cameras. To show that the proposed method works as well for high resolution images as for low resolution video frames, we have tested it on ICDAR 2003 [27], which contains 251 camera images. This dataset contains only scene text, but not as in the video where both scene and graphics text are present. This is a major difference between this dataset and the video data. Sample results of the proposed method are shown in Figure 18 where (a) shows input frames having different backgrounds and text, (b) shows

potential text candidates detected by the proposed method and (c) shows linearity graphs of the proximity between the components and size of the components. It can be seen from Figure 18(c) that the proposed method finds linearity for the potential text candidates of the input images, and that, therefore, the method classifies them as text frames correctly. The quantitative results of the proposed and existing methods are reported in Table 7 which evidences that the proposed method is comparable to the existing method as the classification rate of the text frames is slightly lower than the existing method. However, the misclassification rate of the proposed method is lower than the existing method. Therefore, it can be concluded that the proposed method is comparable and works well for high resolution images.

Figure 18: Classification of frames from ICDAR 2003 dataset as text frames

Table 7: Classification rate of the proposed and existing methods on ICDAR 2003 dataset

Method	Text (%)	Error (%)	APT
Proposed method	80.97	19.03	1.23
Shivakumara et al. [21]	81.12	18.88	N.A.

(a) Input text frames with different texture background

(b) Text region candidates

(c) Potential Text candidates

(d) Potential text candidate group which passed the linearity check

(e) Linearity check graphs based on distance, size and orientation

Figure 19: Classification results obtained for the different texture background frames using proposed method

3.7. Error Analysis and discussion

In order to show robustness of the proposed classification method, we tested the method on different texture background frames as shown in Figure 19 (a). The reason for choosing such images is that, texture may sometimes produce text like features. For the frames shown in Figure 19 (a), the proposed method classifies the frames as text frames correctly. This is because of the way the method obtains text candidate regions as shown in Figure 19(b) by the combination of color and gradient information. The potential text candidates as shown in Figure 19(c) obtained by studying shapes and geometrical properties of text, eliminate almost all non-text components in spite of the existence of texture in the frames. As the proposed method requires just three components to check the linearity property, if the proposed features of false positive elimination remove some of the text and non-text components as shown in Figure 19(c), it does not have a substantial effect on the overall classification. Figure 19(d) shows the text components which satisfied the linearity check. Therefore, the features which we propose to eliminate false positives and the linearity check property help us to classify the text frames correctly irrespective of background complexity.

Though the proposed method performed well for both text and non-text frames, there are cases where erroneous results have occurred as shown in Figure 20. Sometimes, if the image contains text of a very low resolution then the method may classify it as a text frame by incorrectly identifying non-text components as potential text candidates. One such example is shown in Figure 20(a) where in spite of the presence of text, the proposed method identifies a repeated pattern in the building as text, and hence it is classified as a text frame. Figures 20(b) and (c) demonstrate that sometimes text frames are misclassified as non-text frames due to a loss of text components at the potential text candidate selection stage. The rules based on connected component analysis proposed in Section 2.2 eliminate actual text components as shown in the resultant images in Figures 20(b) and (c). This is because of very low resolution, fonts that are too small, and blurring in the image. Hence, there is scope to investigate new features which can work well in diverse cases.

4. Conclusion and future work

This paper presents a new method based on linearity and non-linearity of the text components for text frame classification from the pool of a large number of text and non-text frames. The method identifies text candidates by combining color and gradient information in a novel way. The potential text candidates are identified by exploring geometrical properties of text components. The linearity and non-linearity is defined based on the fact that potential text candidates have uniform spacing between the components and the size of the components while non-text components have non-uniform spacing between the components and size of the components. The main advantage of this approach is that the proposed method works well irrespective of the orientation of the texts, background and contrast. Experimental results demonstrate

(a) Text frame misclassified as non-text frame due to small font and too low contrast text

(b) Text frame misclassified as non-text frames due to loss of text information

(c) Text frame misclassified as non-text frame due to loss of text property

Figure 20: Samples of erroneous results

that the proposed method performs better than the existing method in this area, but that the proposed method attains a low accuracy for curved text and also for any images that contain low resolution text and repeated patterns. Therefore, we plan to extend the method to improve the accuracy by exploring temporal information, which is not currently investigated in the present work.

Acknowledgement

This work was conducted jointly by Griffith University, Australia, University of Malaya, Malaysia, National University of Singapore, Singapore and the Indian Statistical Institute, Kolkata, India. The work is also partly supported by the University of Malaya HIR under Grant No. M.C/625/1/HIR/210.

The authors would like to thank the reviewers and the editor for their useful comments which helped us to improve the quality and clarity of the work in a great way.

References

- [1] J.Zang, R.Kasturi, Extraction of text objects in video documents: recent progress, in: Proceedings of the DAS, 2008, pp. 5-17.

- [2] N. Sharma, U. Pal, M. Blumenstein, Recent Advances in Video Based Document Processing: A Review, in: Proceedings of DAS, 2012, pp. 63-68.
- [3] K.Jung, K.I.Kim, A.K.Jain., Text in formation extraction in images and video: a survey, Pattern Recognition, 37 (2004) 977-997.
- [4] D.Doremann, J.Liang, H.Li, Progress in camera-based document image analysis, in: Proceedings of the ICDAR, 2003, pp.606-616.
- [5] H.Li, D.Doermann, O.Kia, Automatic text detection and tracking in digital video, IEEE Trans. Image Process. 9 (2000) 147-156.
- [6] A.K.Jain, B.Yu, Automatic text location in images and video frames, Pattern Recognition, 31 (12)(1998) 2055-2076.
- [7] H. Zhang, K. Zhao, Y. Z. Song and J. Gao, Text extraction from natural scene image: A survey, Neurocomputing, (2013) 310-323.
- [8] S. H. Lee and J. H. Kim, Integrating Multiple Character Proposals for Robust Scene Text Extraction, Image and Vision Computing, (2013) 823-840.
- [9] K.Jung, Neural network-based text location in color images, Pattern Recognition Lett. 22 (2001) 1503-1515.
- [10] V.Wu, R.Manmatha, E.Riseman, TextFinder:an automatic systems to detect and recognize text in images, IEEE Trans. PAMI, 21 (1999) 1224-1229.
- [11] K.L Kim, K.Jung, J.H.Kim., Texture-based approach for text detection in images using support vector machines and continuous adaptive mean shift algorithm, IEEE Trans. PAMI, 25 (2003) 1631-1639.
- [12] P. Shivakumara, T. Q. Phan and C. L. Tan, New Fourier-Statistical Features in RGB Space for Video Text Detection, IEEE Trans. CSVT, 20 (2010) 1520-1532.
- [13] P. Shivakumara, T. Q. Phan, and C. L. Tan, A laplacian approach to multi-oriented text detection in video, IEEE Trans. PAMI, 33 (2011) 412-419.
- [14] N. Sharma, P. Shivakumara, U. Pal, M. Blumenstein, C. L. Tan, A New Method for Arbitrarily-Oriented Text Detection in Video, in: Proceedings of DAS, 2012, pp. 74-78.
- [15] Q. Ye, Q. Huang, W. Gao, D. Zhao, Fast and robust text detection in images and video frames, Image and Vision Computing, 23 (2005) 565-576.
- [16] X. Tang, X. Gao, J. Liu, H. Zhang, A spatial temporal approach for video caption detection and recognition, IEEE Trans. Neural Networks, 13 (2002) 961-971.
- [17] P. Shivakumara, R. P. Sreedhar, T. Q. Phan, S. Lu, and C. L. Tan, Multi-oriented video scene text detection through Bayesian classification and boundary growing, IEEE Trans. CSVT, 22 (2012) 1227-1235.
- [18] P. Shivakumara, H. T. Basavaraj, D. S. Guru and C. L. Tan, Detection of Curved Text in Video: Quad Tree based Method, In Proc. ICDAR, 2013, pp. 594-598.
- [19] J. Zhang, D. Goldgof and R. Kasturi, A new edge-based text verification approach for video, In Proc. ICPR, 2008, pp. 1-4.
- [20] X. Wang, L. Huang and C. Liu, A new block partitioned for text verification, In Proc. ICDAR, 2009, pp. 366-370.
- [21] P. Shivakumara, A. Dutta, T. Q. Phan, C. L. Tan, and U. Pal, A novel mutual nearest neighbor based symmetry for text frame classification in video, Pattern Recognition, 44 (2011) 1671-1683.
- [22] P. Shivakumara, and C. L. Tan, Novel Edge Features for Text Frame Classification in Video, in: Proceedings of the ICPR, 2010, pp. 3191-3194.
- [23] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the CVPR, 2005, vol.1, pp. 886-893.
- [24] P. Shivakumara, T. Q. Phan and C. L. Tan, New Fourier-Statistical Features in RGB Space for Video Text Detection, IEEE Trans. CSVT, 20 (2010) 1520-1532.

- [25] X. S. Hua, L. Wenyin and H. J. Zhang, An automatic performance evaluation protocol for video text detection algorithms, IEEE Trans. on CSVT, 14 (2004) 498-507.
- [26] H. Chen, S. S. Tsai, G. Schroth, D. M. Chen, R. Grzeszczuk, B. Girod, Robust text detection in natural images with edge-enhanced Maximally Stable Extremal Regions, IEEE International Conference on Image Processing (ICIP), 2011, pp. 2609-2612.
- [27] S. M. Lucas, ICDAR 2005 text locating competition results, In Proc. of ICDAR, 2005, pp.80-84.