Contour Restoration of Text Components for Recognition in Video/Scene Images

Yirui Wu, Palaiahnakote Shivakumara, Tong Lu, *Member, IEEE*, Chew Lim Tan, Michael Blumenstein, *Senior Member, IEEE*, and Govindaraj Hemantha Kumar

***Abstract*— Text recognition in video/natural scene images has gained significant attention in the field of image processing in many computer vision applications, which is much more challenging than recognition in plain background images. In this paper, we aim to restore complete character contours in video/scene images from gray values, in contrast to the conventional techniques that consider edge images/binary information as inputs for text detection and recognition. We explore and utilize the strengths of zero crossing points given by the Laplacian to identify stroke candidate pixels (SPC). For each SPC pair, we propose new symmetry features based on gradient magnitude and Fourier phase angles to identify probable stroke candidate pairs (PSCP). The same symmetry properties are proposed at the PSCP level to choose seed stroke candidate pairs (SSCP). Finally, an iterative algorithm is proposed for SSCP to restore complete character contours. Experimental results on benchmark databases, namely, the ICDAR family of video and natural scenes, Street View Data, and MSRA data sets, show that the proposed technique outperforms the existing techniques in terms of both quality measures and recognition rate. We also show that character contour restoration is effective for text detection in video and natural scene images.**

***Index Terms*— Laplacian, zero crossing points, gradient magnitude, Fourier phase angle, character reconstruction, video text recognition, object recognition.**

I. Introduction

RECENTLY, text detection and recognition has received a significant amount of attention in real-life applications such as iTown and many other smart city developments [1], [2]. These applications require robust text detection and recognition approaches due to the large variations in text fonts or font sizes embedded in complex backgrounds with buildings, trees, etc. Many techniques have thus been developed for improving the accuracy of text detection and recognition in video/scene images [1], [2]. It can be classified broadly as connected component, texture and edge/gradient-based techniques [1], [2]. Connected component-based techniques have inherent limitations such as the need for proper sizes of character components and homogenous backgrounds. Therefore, these techniques may not be suitable for scene images or videos, where we can expect high variations in background, foreground, contrast, font size, font and orientation, or severe illumination and blurring effects, etc. [2]. Texture-based techniques are good for complex background images, but sensitive to font variations. In addition, extracted text features may overlap with features of background objects because these techniques consider the appearance of characters as a special kind of texture [2]. Finally, texture-based techniques are often computationally expensive due to the involvement of a large number of features and classifiers.

With this context, edge/gradient-based techniques [3]–[8] have drawn the attention of researchers in recent years. It is noted from the literature that most of the techniques use stroke width distance and gradient direction for extracting features independent of scripts, orientations or text types. Thus they are computationally inexpensive compared to texture-based techniques. However, the main issue with these techniques is that they produce more false positives for complex background images. Therefore, although a large number of techniques have been published for scene text detection and recognition in the past decade [9], [10], the performance of these techniques are still not satisfactory because text recognition in video and natural scene images is essentially an ill-posed problem. It is evident from [11] that the recognition accuracy of Optical Character Recognition (OCR) engines on words cropped from street view images is as low as 35% [11]. For video text recognition, OCR gives a recognition rate typically from 0% to 45% due to low resolution and complex backgrounds [12]. This accuracy is far from the typical OCR accuracy on scanned documents, which generally reaches more than 90%. The main reason for the poor accuracies of the existing techniques is that most of them fail to extract features which preserve the shapes of characters in video and natural scene images. This is due to the limitation that edge detectors give fine edges for high

Manuscript received January 14, 2016; revised August 7, 2016; accepted August 29, 2016. Date of publication September 8, 2016; date of current version October 7, 2016. This work was supported in part by the Natural Science Foundation of China under Grant 61672273, Grant 61272218, and Grant 61321491, in part by the Science Foundation for Distinguished Young Scholars of Jiangsu under Grant BK20160021, and in part by the University of Malaya HIR under Grant M.C/625/1/HIR/210. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Lei Zhang. *(Corresponding author: Tong Lu.)*

Y. Wu and T. Lu are with the National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China (e-mail: wuyirui1989@163.com; lutong@nju.edu.cn).

P. Shivakumara is with the Department of Computer System and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia (e-mail: hudempsk@yahoo.com).

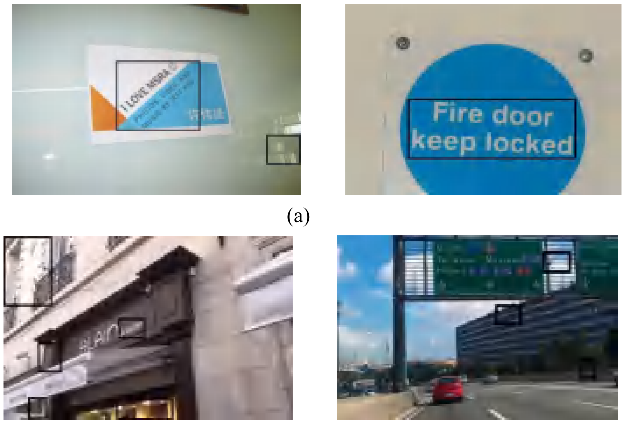
C. L. Tan is with the Department of Computer Science, School of Computing, National University of Singapore, Singapore 119077 (e-mail: tancl@comp.nus.edu.sg).

M. Blumenstein is with the Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, NSW 2007, Australia (e-mail: michael.blumenstein@uts.edu.my).

G. H. Kumar is with the Department of Studies in Computer Science, University of Mysore, Mysore 570006, India (e-mail: ghk.2007@yahoo.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2607426



(a)

(b)

Fig. 1. Illustration for text detection on high and low contrast images. (a) Text detection by SWT for high contrast images (scene image). (b) Text detection by SWT for low contrast images (from video).

contrast texts with less complex backgrounds, but not for low contrast texts with complex backgrounds in video.

One such example for text detection by the technique in [3], which is a state-of-the-art technique that finds stroke widths through the Stoke Width Transform (SWT) and the Canny edges of the input image, is shown in Fig. 1. We can see that the technique successfully detects texts for high contrast images chosen from a natural scene image dataset as shown in Fig. 1(a), but the same technique fails to detect texts for video images containing low contrast texts with perspective distortion as shown in Fig. 1(b). In the same way, the existing technique that heavily depends on edge images [13]–[16] fails to recognize characters correctly due to the loss of shapes or contours during binarization. Hence, it is a significant challenge to restore character shapes to improve recognition rates for texts in both video and scene images. In this work, we propose a novel technique to restore character shapes to achieve good recognition rates with the available OCR [17], rather than developing separate video OCR which is expensive and not practical.

II. Related Work

There are three ways to improve the recognition rate for video/scene texts according to the literature [13]–[16]. First, binarization techniques are proposed, which are similar to thresholding techniques, to preserve the shapes of characters during binarization. The features for text binarization mainly include intensity, color and stroke, which are simple and efficient but often produce poor results due to background variations. Recently, to overcome the problems faced by the above existing techniques, Zhang and Wang [18] proposed the binarization of overlaid texts. This technique finds text polarity and then applies k-means clustering in the RGB color space. An MRF model is exploited to get binarization results. However, the scope of the technique is limited to superimposed text but not scene text.

Second, classifiers for training features are used to recognize characters. Wang et al. [19] proposed an End-to-End scene text recognition technique, which involves both character detection and recognition of full words. This technique uses HOG features and semantics to improve the recognition rate of scene text in natural scene images. Smith et al. [20] proposed enforcing similarity constraints with integer programming for better scene text recognition. Mishra et al. [21] proposed a framework that exploits both bottom-up and top-down cues. The bottom-up cues are derived from individual character detections from an image. Based on the detections, the technique builds a Conditional Random Field model and imposes top-down cues obtained from lexicon-based priors. Recently, Phan et al. [11] proposed a technique for recognizing texts with perspective distortions in natural images. This technique uses a Scale Invariant Features Transform (SIFT) using a pre-trained vocabulary. Context information is utilized through lexicons. Then the technique formulates word recognition as finding the optimal alignment between the set of characters and the list of lexicon words. However, its major weakness is that the technique is expensive as it involves classifiers or semantics with a large number of samples. In addition, the use of a large number of samples for classifier training may restrict the ability to handle multilingual scripts and hence lose generality. Therefore, this technique achieves good recognition rates for only one type of data and reports inconsistent results for other data types.

Third, stroke width distance is also explored to reconstruct the shape of a character to improve the recognition rate with the help of available OCR techniques [17]. For example, Shivakumara et al. [13] proposed a new Ring Radius Transform (RRT) for character shape reconstruction to fill the gaps on character contours. This technique explores medial axis points that remain constant throughout a character to define an interpolation criterion. Experimental results show that if the technique fills the gaps on contours, the recognition accuracy will be improved significantly. This provides clues to our work in enhancing character recognition. Tian et al. [14], [15] proposed techniques to reconstruct the shapes based on medial axis information in natural scene images. However, these techniques require high contrast character images in order to achieve better results because the gradient histogram operation is involved in selecting the text candidates. Shivakumara et al. [16] proposed an iterative midpoint technique to overcome the problems of RRT, which works well only for horizontal and vertical gap filling. This technique explores the directions of contours and the mutual nearest neighbor criterion for filling up text pixels. However, this technique produces a poor accuracy when multiple gaps appear on the same contour.

From the above discussion, we notice that although the techniques improve recognition accuracy, their performance often decreases when video/scene images have background and foreground variations. The main reason is that the steps proposed in these techniques require an edge image detected by Canny from the input image. However Canny produces erratic and spurious edges when complex backgrounds exist and furthermore it may lose pixels due to low resolution. Therefore, the solution to To overcome this problem these shortcomings is to explore gray information to produce the contours for character components, which can preserve character shapes for more accurate recognition.

The same conclusions can be drawn from the literature on text detection in video as well as natural scene images [22]–[26]. Most of the existing text detection techniques expect shapes to be preserved because they explore Maximally Stable Extremal Regions (MSER) followed by HOG and SIFT features along with a classifier, which works well for high contrast uniform color character components. Due to low resolution and complex backgrounds, the techniques that use MSER may fail to obtain complete character components. Instead, they often generate many sub-components for one character component. This leads to confusion for classifiers or recognizers. However, recently, a few techniques have been developed for text detection in video and natural scene images without depending much on MSER [27]–[31]. These techniques are better than the existing techniques, but the performance of these approaches is not consistent for both video and natural scene images because the main goal of these approaches is to achieve good results for video but not natural scene images. The inconsistent results are caused by the fact that the approaches fail to retain significant information of the text components due to the different nature of video and natural scene images.

In light of the above discussions, one can conclude that the techniques proposed for both text recognition and detection utilize character shapes directly and indirectly in order to achieve better results. Besides, due to the different nature of video and natural scene images in terms of contrast, background complexity and text appearance, the performance of the existing techniques are not consistent for both text detection and recognition. Therefore, to overcome the above shortcomings, there is a great need for restoring shapes of characters for both video and natural scene images as it assists in achieving good results for both text detection as well as recognition. Thus, we propose a novel technique to restore character shapes in video and natural scene images by exploring the strengths of zero crossing points in the gray domain given by the Laplacian operator and a spatial study between zero crossing points. The main contribution of the proposed work is to explore Laplacian zero crossing points in different directions for predicting contour pixels of character components in both video and natural scene images without the help of an edge detector. In addition, we will also use stroke width distances and the spatial study of zero crossing points in a new way for verifying paired pixels which represent the strokes of character contours. The advantage of the proposed technique is that it is script and contrast independent, and invariant to rotation or scaling. Furthermore, the proposed technique has an ability to extract contours from general objects.

III. Proposed Technique

In this work, we consider segmented characters based on the method described in [32] from text lines of video and natural scene images as the input for contour reconstruction. The method in [32] explores wavelet decomposition to segment words first from arbitrarily-oriented text lines based on the fact that blur increases as decomposition level increases, which results in the merging of character components into a word component. For each segmented word, the method proposes horizontal sampling to find the spacing between characters, which works based on the percentage of zero crossing points over characters and in spacing between characters. In the same way, the method proposes vertical sampling to verify the spacing between characters found by horizontal sampling, which results in character segmentation. The main advantage of this method is that it works well for video and natural scene images of arbitrarily-oriented text lines and different fonts or font sizes as required in the proposed work.

In order to restore the contours of character components directly from gray values, we need to focus on identifying edge pixels, which represent contour pixels of characters. There are methods in the literature [3], [13]–[16] as mentioned in the previous section, which use stroke width distance as medial axis for restoring character shapes. It is also noted that these methods work well for both video and natural scene images. However, the main weakness of these methods is that their performances depend on how well an edge operator determines the edges. It is obvious that since edge operators are sensitive to complex backgrounds and contrast variations, the existing methods do not produce consistent results for video and natural scene images. This observation motivates us to explore properties which help to restore contours in the gray domain without the use of edges detected by edge operators.

Inspired by the work in [33] for text detection, which uses positive and negative peaks given by Laplacian operators to identify the presence of text in video images, we further explore the identification of Stroke Pixel Candidates (SPCs) along the same lines. Then we introduce new features that utilize the inherent symmetry property of edge pixels to help identify Probable Stroke Candidate Pairs (PSCPs) from SPCs. Due to complex backgrounds and low resolution, the above features may still produce false PSCPs for non-text pixels. Therefore, we explore the spatial and intensity relationships among PSCPs to filter out false ones, which results in Seed Stroke Candidate Pairs (SSCPs). Finally, we propose a greedy growing algorithm, which extends these key SSCPs to restore complete contours of character components based on neighborhood information. More details can be found in the subsequent sections.

A. Directional Laplacian Cue for SPC Detection

In this section, we first use the Laplacian operator to generate Laplacian images in different directions, because the scope of the proposed work is to address differently oriented characters. Next, we found the pixels which give high positive and negative peaks in the Laplacian domain. The positive and negative peaks are used for identifying zero crossing points that represent SPCs of character contours.

For the input text components shown in Fig. 2(a), we can see that character “A” has a low contrast, while the second one has a complex background. We can visualize the complexity of the contour restoration problem of video text components by the Canny edges of the input images as shown in Fig. 2(b). In that figure, we can see that the first edge image loses text pixels due to low contrast, while the second one generates many spurious edges due to complex background. Since our target

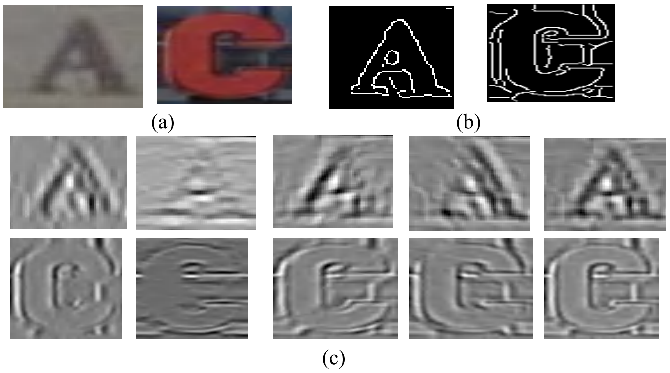


Fig. 2. Multi-directional Laplacian images and their average for low contrast and complex background text components. (a) Input text components. (b) Canny edge images. (c) Multi-directional and Mean Laplacian images for “A” and “C” in (a).

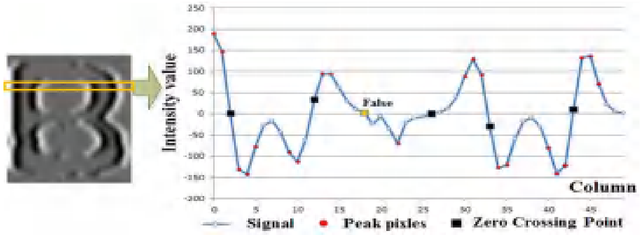


Fig. 3. We represent one row in the horizontal Laplacian image of Character ‘B’ as a directional signal, implied by the blue line. Pixels colored in red are peak pixels. Correct and false zero crossing points are marked by black and yellow rectangles, respectively.

is to handle arbitrarily-oriented text components, we propose Laplacian operations in multi-directions to collect the details of text pixels. More precisely, the proposed technique performs Laplacian operations in four directions and calculates their average for each input image as respectively shown in Fig. 2(c) and Fig. 2(d).

More formally, let I be the input image and M_k be the one-dimensional Laplacian masks to be convolved with I. The multi-directional Laplacian images can be calculated by

$$I_k = I * M_k, \quad k \in \{1, 2, 3, 4\} \tag{1}$$

where $k = 1, 2, 3, 4$ denote the horizontal, the vertical, the diagonal and the secondary diagonal Laplacian images, respectively.

From the view of signal processing, the pixel intensity along direction k can be expressed as a one dimensional signal $g_u(k)$ for a specific pixel u. That is, we can regard one row, column, the diagonal and the secondary diagonal as an independent directional signal. Considering Taylor signal expansion theory, as mentioned in [34] for making LBP invariant features for rotation and scaling, the proposed multi-direction Laplacian operator encodes the first order directional derivatives of the directional signals $g_u(k)$, and also contributes significant and fundamental parts of the directional signals. For text images, the Laplacian operator could obtain the illumination variance information from directional signals, which results in high positive and negative values for high contrast pixels, namely, candidate edge pixels, and low values for non-edge pixels. Furthermore, we apply k-means clustering with $K = 2$ in each directional signal of Laplacian multi-direction images I_k to find the peaks that represent edge pixels. The cluster with the high centroid value is regarded as a cluster of pixels with peak Laplacian values, named as peak pixels. This can be denoted mathematically as

$$P_u(k) = 1, \quad if \;\; u \in Max_C \, f_m \left(g_u \left(I_k\right)\right) \tag{2}$$

where P denotes the binary peak pixel image for direction k, C represents the value of the cluster centroid, $f_m()$ represents the the k-means method, and $g_u\left(I_k\right)$ represents the dimensional signal of the Laplacian image I_k. However, we could get false peak pixels representing small Laplacian values because their corresponding signals are too weak in the overall intensity. Therefore, we propose to use non-maximum suppression to eliminate false ones, which could be represented as u

$$P_u^{'}(k) = 1, \quad if \, I_k\left(u\right) > \epsilon \cdot f_\mu\left(g_u\left(I_k\right)\right) \tag{3}$$

where ϵ is a preset threshold and $f_\mu()$ calculates the mean value of the corresponding directional signals of pixels.

Next, we propose to search zero crossing points between nearby peak pixels, which belong to the same directional signal and are opposite in signs. However, searching zero crossing points is not easy because the Laplacian operation is sensitive to background or color changes [33]. To overcome this problem, we propose an algorithm to identify stable zero crossing points by analyzing the sequence of the Laplacian values between peak pixels. Specifically, the sequence of Laplacian values between peak pixels must have transitions from either decreasing to increasing or vice versa. In Fig. 3, we show some correct zero crossing points and a false one, which doesn’t satisfy the transition rule, by black and yellow rectangles, respectively. Representing this transition rule as a binary function Φ_{tra}, we define the following equation to identify zero crossing points from two nearby peak pixels m and n:

$$Z_k\left(u\right) = 1, \quad if \;\; I_k\left(u\right) = Min\left(g_{mn}\left(I_k\right)\right) \wedge \Phi_{tra}\left(g_{mn}\left(I_k\right)\right) = 1 \tag{4}$$

where Z_k denotes the binary zero crossing image for direction k and $g_{mn}\left(I_k\right)$ represents a signal in the Laplacian image I_k, consisting of sequential pixels from m to n.

Finally, we congregate all zero crossing points in Z_k to consititute a set of candidate edge pixels, called Stroke Pixel Candidates (SPCs). The effect of SPC for the four directional images, say the average of $Z_k, k\in \{1, 2, 3, 4\}$, are shown in Fig. 4, respectively. It is observed from Fig. 4 (the average of four Laplacian directional images in the last column in Fig. 4) that the proposed SPC detection, misclassified non-text pixels as SPC due to the complexity of the images.

B. Probable Stroke Candidate Pair Detection

It is true that character images exhibit double and parallel edges with uniform distances between the pixels in the parallel edges as shown in Fig. 4 (the last column of “A”) [3].

Algorithm 1 Detecting PSCPs

```
Input: Stroke Pixel Candidates image $Z$, Laplacian average image $I_a$
Output: Probable Stroke Candidate Pair (PSCPs)
For $m$ = each pixel in $Z$ do
        $r = m + q \cdot d_m, \quad q > 0$
until $r$ comes across another pixel n in $Z$
set $p_{mn} = (m, n)$;
end for
{$d_m$ is defined as the gradient direction at pixel $q$; $\{m, n\}$ represents a pair of pixels represented by $m$ and $n$}
For each pixel pair $p_{mn}$ do the following filters
    $f_1(p_{mn}) = 1, if \left|V_p(m, I_k) - V_p(n, I_k)\right| \le \alpha \cdot V_p(m, I_k)$
    $f_2(p_{mn}) = 1, if \left|G_m(m, I_a) - G_m(n, I_a)\right| \le \beta \cdot G_m(m, I_a)$
    $f_3(p_{mn}) = 1, if \left|cos(< G_o(m, I_a), G_o(n, I_a) >) - 1\right| \le \gamma$
    $f_4(p_{mn}) = 1, if \left|Sw(g_{mn}(I_a)) - Sw(g_{no}(I_a))\right|$
        $\le \delta \cdot Sw(g_{mn}(I_a))$
    $f_5(p_{mn}) = 1, if \left|f_\varepsilon(g_{mn}(I_a)) - f_\varepsilon(g_{no}(I_a))\right|$
        $\le \varepsilon \cdot f_\varepsilon(g_{mn}(I_a))$
where <> denotes an angle between two vectors; $k = 1,2,3,4$.
if $f_1 \wedge f_2 \wedge f_3 \wedge f_4 \wedge f_5 = 1$
$f_{PSCP}(p_{mn}) = 1$;
    end if
end for
set $\{m, n\} = (m, n)\ if\ f_{PSCP}(p_{mn}) = 1$
```

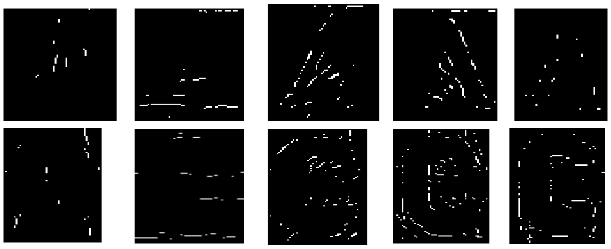


Fig. 4. Binary zero crossing point images and Stroke Pixel Candidates detection results of character “A” and “C”.

Besides, the intensity values of each pair of SPCs, which are on the parallel strokes, tend to be similar due to homogenous backgrounds [3], [6]. The two SPCs are considered as a pair when the distance between two zero crossing points represents a stroke width. The same observation is valid for different domains, such as gray, gradient and frequency [27]. This cue leads to a proposed symmetry property to identify pairs of SPC, which represent actual parallel edges and results in Probable Stroke Candidate Pairs (PSCPs). We propose different filters to extract different symmetry between SPC pairs as presented in Algorithm-1. For these filters, a pair of SPC are the inputs and the SPC pair is identified by stroke width distance between two zero crossing points as calculated by the method presented in III.A.

In Method-1, Filter-1 checks the color symmetry between the SPC pair by calculating the mean values of positive and negative Laplacian peaks V_p of the SPC pair based on the fact that a pair of SPC share the same color value. Filter-2 and Filter-3 are proposed to check the gradient and distance symmetry between the SPC pair using the gradient magnitude G_m and orientation G_o in the average Laplacian image I_a, which is computed by $I_a = \sum_{k=1}^{4} |I_k/4$. Filter-3 examines whether the gradient directions of the SPC pair are in the same or opposite directions. Filter-4 checks distance symmetry between stroke width Sw of the SPC pair. The distance symmetry is determined as follows. We draw a ray by starting from one SPC and moving in the gradient direction until it touches the other SPC as shown in Fig. 5, which gives the stroke width. Due to low contrast and low resolution of video, there is a chance of losing significant information which represents pairs of SPC. Therefore, we propose Filter-5 to check the symmetry property in the Fourier domain because it provides high frequency coefficients for the edge pixels and low frequency coefficients for non-text pixels [33]. The proposed technique compares the stroke width of an end-to-end pixel pair, i.e. (m, n) and (n, o), to identify the actual pair. In this work, we prefer to use the phase spectrum that constitutes high and low frequency coefficients to check the symmetry between a SPC pair by assigning higher weights for high frequency coefficients that represent edge pixels. We compute a weighted value of the phase spectrum for each signal as follows:

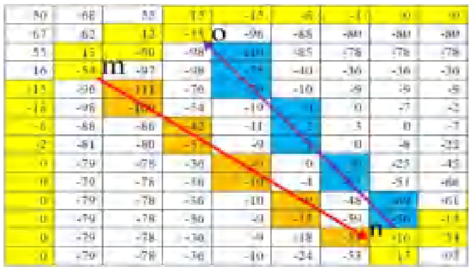


Fig. 5. One example of Filter 4 where m and n, n and o are end-to-end pairs confirmed by the Stroke Width Transform. The detailed value of each pixel refers to the average Laplacian value. Yellow, orange and blue pixels represent SPCs, Ray_{mn} and Ray_{no}, respectively.

$$f_\varepsilon(g_{mn}(I_a)) = f_\theta(f_{FFT}(g_{mn}(I_a))) \cdot \omega \tag{5}$$

where ω is a pre-defined weight vector, $f_{FFT}()$ represents the Fast Fourier transform which converts a signal to its real and imaginary parts, and $f_\theta()$ computes the phase angle between the real and imaginary parts computed by $f_{FFT}()$. Filter-5 checks both distance symmetry as in Filter-4, and the phase angle symmetry pixels of the SPC pair. If any SPC pair satisfies the above filters, then the pair is considered as a Probable Stroke Candidate Pair (PSCP).

The parameters mentioned in Method-1, namely, $\alpha, \beta, \gamma, \delta, \varepsilon$, are determined empirically in this work. For this purpose, we randomly choose 100 characters from different datasets, which in total gives 500 characters for experimentation. We record the values of α while calculating PSCP on the 500 characters manually. With these values, the proposed method considers the values which are near to the maximal bin with the range w (predefined as 0.3). Further, the value of the parameter α is defined as $\alpha = S/$w, where S is the sum of the values of α of the range chosen

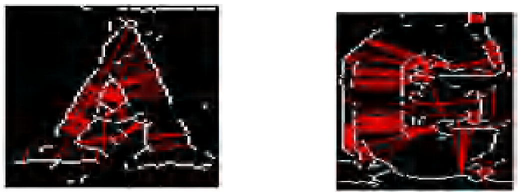


Fig. 6. Examples of Probable Stroke Candidate Pairs. We show one PSCP by connecting both pixels with red lines.

manually by experimentation. By the same way, the proposed method determines the values for the other parameters consisting of $\beta, \gamma, \delta, \varepsilon$.

The effect of the PSCP algorithm can be seen in Fig. 6, where it is noted that most PSCPs represent edge pixels. However, it is also observed that there may be false PSCPs, which represent the background due to the background's complexity. Therefore, to eliminate such false PSCPs, we further explore their symmetry properties at the component level.

C. Seed Stroke Candidate Pair Detection

For Seed Stroke Candidate Pair (SSCP) detection, the PSCP identified in the previous section is the input. The symmetry properties proposed in the previous sections are checked at the SPC pair level. As a result, it considers just paired pixel information of the SPC. Therefore, to remove false PSCPs which represent non-text, we explore symmetry properties at the character component level. We perform histogram operations on the mean intensity values of PSCPs over the whole component to find the variance between them. Due to background variations, the PSCP which represents the background has different mean values, while a genuine pair of PSCP represents character edge pixels that have almost uniform mean values. As a result, the proposed histogram operation removes any pair which gives high variance as a false PSCP pair. Specifically, we consider the PSCP that contributes to the highest peak in the histogram as a correct one, which can be defined as

$$C_g = Max_C\, f_{hist}\left(\bigcup_{\{m,n\}} f_\mu\left(g_{mn}\left(I\right)\right)\right) \tag{6}$$

where $f_\mu()$ and $f_{hist}()$ represent the mean and histogram operations, respectively. Any PSCP whose mean intensity value lies in a fixed region centered at C_g is regarded as a correct pair as shown in Fig.7(a), where we can see the effect of the histogram operation on the mean values. Since the problem being considered is complex, a histogram operation alone is insufficient to identify the correct pair.

Therefore, for each PSCP pair, we obtain regions given by MSER using the input image. If the pair represents boundary pixels of the region given by MSER, then the pair is considered as a correct pair. The reason for using MSER regions here is that MSER is robust to blur and viewpoint or light changes. The details of the steps are as follows. The effect of MSER regions for identifying correct PSCPs can be seen in Fig. 7(b), where one can notice that this criterion eliminates a few false PSCPs compared to the results shown in Fig. 7(a):

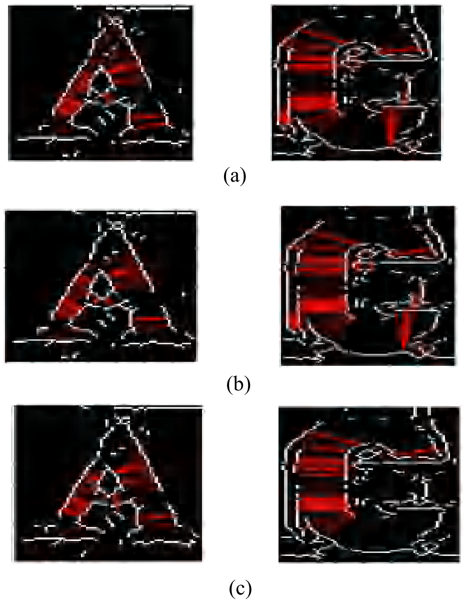
$$f_{PSC}\left(\{m,n\}\right) = 1, \quad if\ \{m,n\} \in f_b\left(f_M\left(I\right)\right) \tag{7}$$

where function $f_M()$ extracts stable extremal regions from the character image. We define the boundary region as $f_b(R) = M \bigoplus R - R$, where $\bigoplus$ is an expansion operation and M is a template of size 5×5.

We propose one more symmetry property based on the distance between the PSCPs of the character components. As mentioned above, the pair of PSCPs should have uniform distances throughout the character [6]–[9]. We thus plot a histogram for distances vs. PSCP pairs. The PSCP that gives a dominant peak is considered as the actual PSCP pair. It is formulated as follows:

$$C_d = Max_C\, f_{hist}\left(\bigcup_{\{m,n\}} Sw\left(g_{mn}\left(I\right)\right)\right) \tag{8}$$

The PSCP pair that satisfies the above three symmetries is considered as the correct one, called a seed stroke candidate pair as shown in Fig. 7(c), where we can see actual PSCPs which represent strokes of character components. We can also see that the proposed technique misses several pairs that represent stroke pixels. Therefore, we consider the output of this step as Seed Stroke Candidate Pairs (SSCP) to restore the missing pairs. It will be discussed in the next section. Note that since the proposed symmetry properties are on SPC pair and PSCP levels, and are based on the characteristics of text components, therefore the considered pairs are invariant to rotation, scaling and scripts.



(a)

(b)

(c)

Fig. 7. Seed stroke candidate pair detection. (a) The effect of the dominant peak of the histogram using intensity values. (b) The effect of MSER boundary regions. (c) The effect of the dominant peak histogram using stroke width distances.

D. Contour Reconstruction

The objective of this section is to reconstruct the complete contours using SSCP based on neighboring pixels of SSCP. For each SSCP, the proposed technique considers the combination

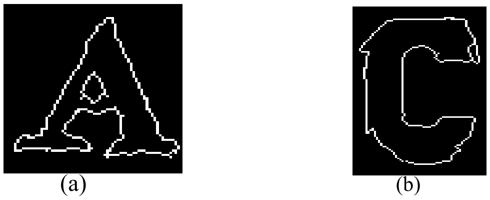


Fig. 8. Restored contours by the proposed technique.

Algorithm 2 Iterative Contour Restoration

```
Input: SSCP set $\{m,n\}$, normalized tangent field of input image $I^t$. For each pixel $u$ in $I^t$, its tangent vector is computed as $\{ I^t_{u,x} = G_{u,y}/\sqrt{{G_{u,x}}^2 + {G_{u,y}}^2}, I^t_{u,y} = -G_{u,x}/\sqrt{{G_{u,x}}^2 + {G_{u,y}}^2} \}$, where $I^t_{u,x}$ represents the x-coordinate of tangent filed value at pixel $n_i$.
Output: character shape contour
Set $Seed = \{m,n\}$
For each pair of seed point $\{m_i, n_i\}$
Fixed $p_m = m_i$, then Set $p_n = n_i + I^t_{n_i}$
while $\| p_m, p_n \| < C_d$
  If $p_n \in Z_k$ or $|G_m(p_m, I^t) - G_m(p_n, I^t)| \leq \beta \cdot G_m(p_m, I^t)$ && $|Cos(< G_o(p_m, I^t), G_o(p_n, I^t) >) - 1| \leq \gamma$ && $|Sw(p_m) - Sw(p_n)| \leq \delta \cdot Sw(p_m)$
    Add $\{p_m, p_n\}$ into $Contour$
      $p_n = p_n + I^t_{p_n}$
else
        break
end
Fixed $p'_n = n_i$, Set $p'_m = m_i + I^t_{m_i}$ and do the same loop above for $\{p'_m, p'_n\}$. Note that $C_d$ is defined as the stroke width value in Eq. 8.

{Any point, which directly stops the loop will be named as the end point}
For each end point $e_i$, Set $p_e = e_i$
While $judge(p_e, I^t) == 1$
                  $p_e = p_e + I^t_{p_e}$
Add $p_e$ into $Contour$
{$judge(p_e, I^t)$ will determine whether a pixel is convincing with respect to its tangent field vector. We give a conclusion based on the distance with the nearest SSCP pixel or MSER boundary pixel. If this value gets smaller by running along the tangent field direction of $p_e$ in 5 steps, we assign $p_e$ as a pixel with a reliable tangent field direction.}
end if
end for
```

of all the pairs formed by 8 neighbor pixels of SSCP. Since the neighboring pair combinations are derived based on the seed stroke candidate pair, the proposed technique employs the filters proposed in Section III.B to identify valid neighbor pairs out of many combinations of pairs iteratively along the tangent direction of SSCP pixels. As a result, this process continues along the character contour until it visits all SSCPs. The steps of the iterative algorithm for restoring the complete contour are presented in Algorithm 2. The effect of the iterative algorithm on two example characters “A” and “C” can be seen in Fig. 8, where the proposed technique restores the contours successfully without disconnections and losing character shapes. This is the advantage of the proposed technique.

As discussed in Section I and Section II, most text detection techniques require characters for text line extraction in video and natural scene images. For example, the text method proposed in [7] explores the characteristics of character components for detecting texts in natural scene images. For an input image, the method extracts components using the MSER concept. For each component given by MSER, it extracts the properties which represent characters such as stroke width information, perceptual divergence and a histogram of gradients at the edges, and then passes the extracted features to a Bayesian classifier to obtain probable character components. For training the Bayesian classifier, the method uses labelled data in the ICDAR 2013 dataset [35]. Next, the method constructs graphs for probable character components and then checks the properties, namely, stroke width divergence and color divergence, to remove non-text components, which results in character components. Further, the method uses mean shift based clustering for grouping characters into text lines. By this way, the text detection method finds characters and then uses character information to extract text lines in the image.

Since the text detection method was developed for natural scene images, when we apply the same method on video images, there are chances of losing character shapes due to contrast variations in the foreground and background. Therefore, it affects the overall performance of text detection in video images. This motivates us to apply the proposed contour reconstruction for character components given by the text detection method to restore character shapes such that the text detection results can be improved for both video and natural scene images. In this way, we combine the proposed contour reconstruction with the text detection method for better text detection. The effect of the proposed contour reconstruction in improving text detection results is illustrated in Fig. 9, where we can see that for the input image in (a), character components given by the text detection method in [7] lose shapes, especially for small fonts below and on the right side of the “JUNKE JOINT” text as shown in Fig. 9(b). This affects text detection as shown in Fig. 9(c), where it misses small font texts in the image. To overcome this problem, the proposed method modifies the step of extracting character components in [7] such that it gives better character components than in Fig. 9(b) as shown in Fig. 9(d), where one can see that the modification step does not miss small font texts compared to Fig. 9(b). The modification is that obtaining new parameter values with predefined labelled video and natural scene character components. Then, for each character component candidate in Fig. 9(d), the proposed contour reconstruction approach has been applied to restore the shapes of character components as shown in Fig. 9(e), where it can be seen that the characters preserve shapes compared to Fig. 9(b). This effect can be seen in Fig. 9(f), where we can notice

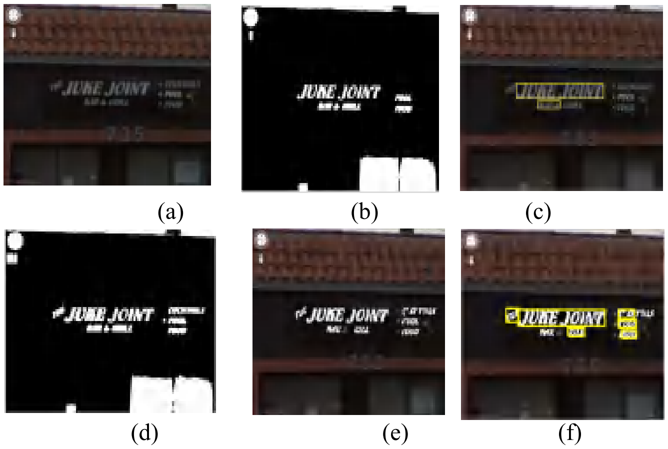


Fig. 9. Illustration to show the effect of the proposed contour reconstruction for text detection with the existing text detection method [7]. (a) Input image. (b) Character components. (c) Text detection. (d) Modified components. (e) reconstruction. (f) Text detection.

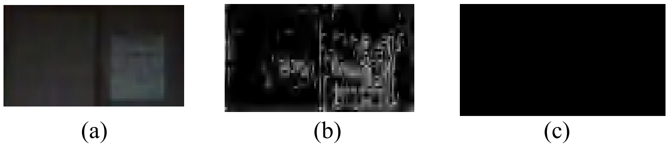


Fig. 10. The proposed contour reconstruction for non-text components. (a) Non-text. (b) SPC. (c) Reconstruction.

that the text detection results are substantially improved when compared to Fig. 9(c). This is the advantage of the proposed contour reconstruction.

In the case when the text detection method misclassifies non-text components as character components as shown in Fig. 9(b), which shows that the character component steps in [7] misclassify non-text components as character components. In this situation, when we pass a false character component to the proposed contour reconstruction, the proposed symmetrical features classify it as a non-text component and output nothing because the proposed symmetrical features exist only in character components. Suppose the proposed contour reconstruction misclassifies a non-text component as a character, the step proposed in [7] (after character component extraction) classifies it as a non-text component because the mean shift clustering used for text line extraction considers only character components. One such example is shown in Fig. 10, where for the non-text component in Fig. 10(a), the proposed method gives Stroke Pixel Candidates (SPC) as shown in Fig. 10(b). However, the final result of contour reconstruction outputs nothing as shown in Fig. 10(c) because the proposed symmetrical features for the SPC do not satisfy the symmetry property to restore shapes. Therefore, we can conclude that the combination of the proposed contour reconstruction and the text detection method is effective and useful for improving the performance of text detection methods.

IV. Experimental Results

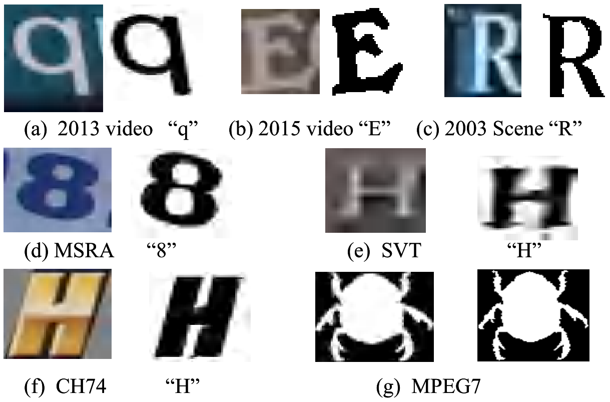
To evaluate the proposed technique for reconstructing contours, we consider characters from standard datasets, namely, ICDAR 2015 [36], ICDAR 2013 video [35], ICDAR 2003 scene [37], Char74 scene data [38], Street View Data (SVT) [39] and MSRA-TD500 data [4], which results in 2408 character images. The character dataset includes 300 characters from ICDAR 2015 video, 300 from ICDAR 2013 video, 413 from ICDAR 2003, 463 characters from Chars74, 432 from SVT, and 500 from MSRA. We also consider 170 objects from the MPEG7 dataset [40] to test our technique for object contour reconstruction. It is noted from the above standard datasets that video characters suffer from low resolution, low contrast, multi-scripts and complex backgrounds. Also, the characters of natural scene data suffer from large font size variations or complex backgrounds, the characters of SVT suffer from severe complex backgrounds containing greenery, buildings, sky, etc., the characters of CH74 data suffer from background variations, and the characters of MSRA data suffer from arbitrary orientations with complex backgrounds. As a result, the collected character dataset covers most of the challenges to validate the performance of the proposed technique. Furthermore, the objects from the MPEG7 dataset are used to validate the generic properties of the proposed technique.

We use standard quality measures such as Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR) and Structural Index Measurement (SSIM) for measuring the quality of contour reconstruction given by the proposed technique. To validate the effectiveness and usefulness of the proposed technique, we conducted experiments on the recognition of the same number of characters with the help of a publicly available OCR [17]. We use recognition rates at the character level for evaluating recognition results for different datasets.

For evaluating the proposed text detection method which combines the existing text detection method in [7] and contour reconstruction on the video datasets, we consider ICDAR 2015 [36] and ICDAR 2013 [35] video frames. Similarly, for evaluating the performance of the text detection method on natural scene images, we consider SVT [39] and MSRA datasets [4] because these datasets are much more complex than the ICDAR natural scene dataset [35] as mentioned in Section IV. Since ICDAR 2015 and ICDAR 2013 video do not provide the number of frames as natural scene image datasets, we extract keyframes which include 549 from ICDAR 2015 and 340 from ICDAR 2013. For natural scene datasets, namely, SVT and MSRA, we respectively use 250 and 200 testing samples as mentioned in [4] and [39] for experimentation. In total, we use 889 video frames and 450 natural scene images for evaluation. To show that the combination of the existing text detection method and the proposed contour reconstruction is effective for text detection, we conduct experiments using only Li et al.'s [7] approach without contour reconstruction and the combined method with the existing method. To measure the performance of the methods, we calculate standard measures, namely, recall, precision and f-measure as per the instructions given in the ICDAR 2013 robust competition [35].

A. Experiments on Quality Measures and Recognition

We define the following quality measures as discussed in the previous section to evaluate the contours reconstructed by



(a) 2013 video "q" (b) 2015 video "E" (c) 2003 Scene "R"

(d) MSRA "8" (e) SVT "H"

(f) CH74 "H" (g) MPEG7

Fig. 11. Sample contour-restored results of the proposed method and respective recognition results for the different datasets. Note: Since there is no OCR available for MPEG data, recognition results are not provided. (a) 2013 video "q". (b) 2015 video "E". (c) 2003 Scene "R". (d) MSRA "8". (e) SVT "H". (f) CH74 "H". (g) MPEG7.

TABLE I

MEASURES AND OCR RESULTS OF THE PROPOSED AND THE EXISTING TECHNIQUES FOR ICDAR2015 VIDEO DATA

Methods	OCR	PSNR	MSE	SSIM
Otsu et al. [41]	0.41	**6.11**	**15941**	0.24
Niblack [42]	0.29	5.29	19217	0.27
Savoula [43]	0.31	5.59	17953	0.01
Shivakumara et al. [13]	0.42	2.30	38320	0.14
Shivakumara et al. [16]	0.35	4.29	24219	0.25
Epshtein et al. [3]	0.39	4.70	22014	0.18
Matas et al.[44]	0.31	5.36	18943	0.25
Proposed	**0.43**	5.01	20498	**0.31**

TABLE II

MEASURES AND OCR RESULTS OF THE PROPOSED AND THE EXISTING TECHNIQUES FOR ICDAR2013 VIDEO DATA

Methods	OCR	PSNR	MSE	SSIM
Otsu et al. [41]	0.37	**5.52**	**18246**	0.21
Niblack [42]	0.28	4.84	21356	0.25
Savoula [43]	0.30	5.11	20037	0.007
Shivakumara et al. [13]	0.40	2.04	40675	0.09
Shivakumara et al. [16]	0.35	3.93	26321	0.24
Epshtein et al. [3]	0.35	4.32	24076	0.14
Matas et al.[44]	0.31	4.95	20784	0.23
Proposed	**0.41**	4.34	23931	**0.28**

the proposed technique:

$$MSE = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(I(i,j)-P(i,j))^2}{m*n} \tag{9}$$

$$PSNR = 10 \times \log\left(\frac{255^2}{MSE}\right) \tag{10}$$

$$SSIM\,(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \tag{11}$$

where m and n refer to the width and the height of the image; I and P represent the contour reconstruction image and the noisy original image; μ_x and μ_y represent the means of x and y; σ_x^2 and σ_y^2 represent the variances of x and y; and σ_{xy} is the covariance of x and y. c_1 and c_2 are two preset variables. Note that we use character images that are created manually as the ground truth for calculating the above three measures.

Examples of qualitative results of the proposed technique on sample characters chosen from different datasets are shown in Fig. 11. One can notice that the proposed technique reconstructs contours well without losing the shapes of components, including objects of different situations, such as low contrast images, complex background images, different oriented character images, and the images with different font sizes. It is evident by the recognition results given by OCR for the reconstructed characters shown in Fig. 11 in double quotes, that the OCR recognizes the restored characters correctly. This shows that the proposed technique works well for different character images.

To show the superiority of the proposed technique, as compared to existing techniques, we implemented baseline thresholding techniques, namely, Otsu [41], which uses global thresholding for binarization, Niblack [42] and Sauvola and Pietikainen [43] which are well known binarization techniques for both video as well as natural scene text images, RRT [13] and IMM [16], which are the recent techniques developed for character shape reconstruction, and SWT [3] which is the state-of-the-art technique for text detection in natural scene images. For an objective comparative study, we chose the baseline character reconstruction and contour reconstruction techniques by SWT, which use publicly available OCR [17] as with the proposed technique. There are other techniques in the literature which recognize characters of the same datasets [11], [19]–[24]. Since these techniques use classifiers and lexicons, it is not fair to conduct a comparison with them. However, most of these methods consider the output of Maximally Stable Extremal Regions (MSER) as the first step and the basis for feature extraction and recognition with a classifier because it is expected that MSER outputs character components. Therefore, we implement MSER as suggested in [44], where it is shown that it is robust to multiple fonts, font sizes, orientations and objects as well. Thus, we consider the output of MSER proposed by [44] for input characters as reconstruction results to show that MSER alone is not sufficient for contour reconstruction.

The quantitative results of the proposed technique and the existing techniques are reported for different datasets, namely, ICDAR 2015 video, ICDAR 2013 video, CH74, ICDAR 2003, SVT, MSRA and MPEG data in Table 1-Table 7, respectively. It is observed from Table 1-Table 7 that the proposed technique is the best for OCR and SSIM across all the experiments except for MSE and PSNR as compared to the existing techniques. This shows that the proposed technique preserves the topology of character components well. The main reason for the poor accuracies obtained from the existing techniques is that they use thresholds and edge images for binarization. We can also observe from Table 1-Table 6 that the OCR accuracies of scene datasets are on average higher when compared to video datasets. This is because video suffers from low contrast, which makes the problem more challenging than for scene images that have high contrast texts. However, the OCR accuracy of the proposed technique is lower compared to the other existing methods that use classifiers

TABLE III

MEASURES AND OCR RESULTS OF THE PROPOSED AND THE EXISTING TECHNIQUES FOR CH74 DATA

Methods	OCR	PSNR	MSE	SSIM
Otsu et al. [41]	0.34	6.21	15555	0.32
Niblack [42]	0.47	4.92	20938	0.28
Savoula [43]	0.29	2.36	37724	0.004
Shivakumara et al. [13]	0.49	4.27	24326	0.17
Shivakumara et al. [16]	0.39	4.30	24168	0.19
Epshtein et al. [3]	0.37	4.66	22253	0.26
Matas et al.[44]	0.46	6.39	14921	0.30
Proposed	**0.53**	**6.69**	**13944**	**0.33**

TABLE IV

MEASURES AND OCR RESULTS OF THE PROPOSED AND THE EXISTING TECHNIQUES FOR ICDAR 2003 DATA

Methods	OCR	PSNR	MSE	SSIM
Otsu et al. [41]	0.45	4.48	23165	0.22
Niblack [42]	0.31	5.65	17720	0.27
Savoula [43]	0.27	2.03	40709	0.004
Shivakumara et al. [13]	0.41	5.05	20338	0.18
Shivakumara et al. [16]	0.36	4.22	24607	0.23
Epshtein et al. [3]	0.30	3.98	26026	0.12
Matas et al.[44]	0.44	5.39	18781	0.25
Proposed	**0.55**	**6.19**	**15620**	**0.28**

TABLE V

MEASURES AND OCR RESULTS OF THE PROPOSED AND THE EXISTING TECHNIQUES FOR SVT DATA

Methods	OCR	PSNR	MSE	SSIM
Otsu et al. [41]	0.47	5.05	20348	0.24
Niblack [42]	0.28	5.37	18877	0.21
Savoula [43]	0.27	2.95	32984	0.003
Shivakumara et al. [13]	0.48	4.18	24822	0.06
Shivakumara et al. [16]	0.38	5.06	20273	0.28
Epshtein et al. [3]	0.33	4.64	22339	0.19
Matas et al.[44]	0.37	4.90	21035	0.20
Proposed	**0.50**	**5.86**	**16883**	**0.29**

TABLE VI

MEASURES AND OCR RESULTS OF THE PROPOSED AND THE EXISTING TECHNIQUES FOR MSRA DATA

Methods	OCR	PSNR	MSE	SSIM
Otsu et al. [41]	0.45	3.65	28058	0.13
Niblack [42]	0.31	5.48	18401	0.22
Savoula [43]	0.32	1.50	46018	0.004
Shivakumara et al. [13]	0.43	4.91	20999	0.15
Shivakumara et al. [16]	0.31	3.31	30344	0.19
Epshtein et al. [3]	0.32	3.48	29188	0.10
Matas et al.[44]	0.44	5.73	17386	0.24
Proposed	**0.51**	**6.52**	**14480**	**0.28**

and lexicons [11], [19]–[24]. It is valid because although the proposed technique constructs contours well, sometimes OCR may not provide correct recognition results due to its inherent limitations such as font or font size variations and language models.

B. Experiments for Text Detection

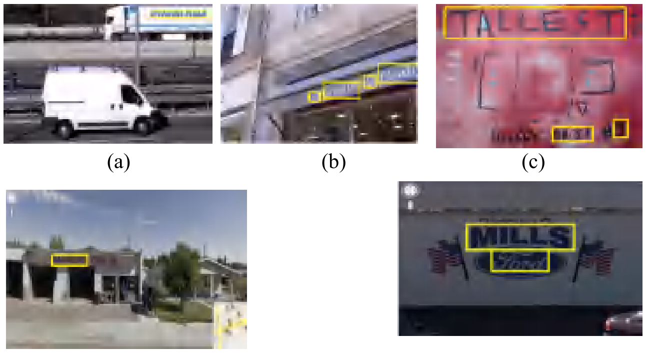


Fig. 12. Text detection results of the proposed method for different ICDAR and other standard datasets. (a) 2013 Video. (b) 2015 video. (c) 2003 scene data. (d) SVT data. (e) MSRA data.

TABLE VII

MEASURES OF THE PROPOSED AND THE EXISTING TECHNIQUES FOR MPEG7 OBJECT DATA

Methods	PSNR	MSE	SSIM
Otsu et al. [41]	**24.8**	**216**	0.93
Niblack [42]	21.6	446	0.84
Savoula [43]	5.24	19478	0.45
Shivakumara et al. [13]	0.88	53042	0.08
Shivakumara et al. [16]	13.8	2716	0.78
Epshtein et al. [3]	8.30	9623	0.55
Matas et al.[44]	21.0	517	0.91
Proposed	14.7	2190	**0.95**

As discussed in Section IV, the combination of the proposed reconstruction technique and Li et al.'s method is considered as the text detection technique for extracting texts from video and natural scene images to show its significance. To show the effectiveness of the proposed technique, we undertake a comparative study with the state-of-the-art techniques, namely, Li et al.'s method [7] which uses character detection and a Bayesian classifier for text detection in natural scene images, Moselh et al.'s method [5] which explores the stroke width transform for text detection in video images, Epshtein et al.'s method [3], which proposes the stroke width transform for text detection in natural scene images, Li et al.'s method [27] which uses moments and wavelets for text detection in video, Zhao et al.'s method [28] which explores dense corners for text detection in video, Yin et al.'s method [25] which explores MSER for text detection in natural scene images, Khare et al.'s method [29] which explores moments as a descriptor for text detection in video images, Wu et al.'s method [31] which proposes gradient symmetry for text detection in video and Liang et al.'s method [3] which proposes the combination of the Laplacian and wavelets for text detection in video. Among these existing techniques, we re-implement some of the techniques [3], [5], [7], [27]–[31] according to the details in the respective papers and use the code available for Yin et al.'s method [25] for the purpose of a comparative study. Since we consider both video and natural scene image data for evaluating the proposed text detection technique, we use both video and natural scene image text detection techniques for comparative studies in this work.

Examples of qualitative results of the proposed technique for different datasets are shown in Fig. 12, where it can be seen that texts are detected from video, scene images,

TABLE VIII
PERFORMANCE OF TEXT DETECTION ON ICDAR 2015 VIDEO

Methods	precision	recall	f-measure
Proposed+ Li et al. [7]	0.48	**0.72**	0.58
Li et al. [7]	0.42	0.64	0.51
Mosleh et al. [5]	0.46	0.44	0.45
Epshtein et al. [3]	0.43	0.42	0.42
Li et al. [27]	0.22	0.64	0.33
Zhao et al. [28]	0.20	0.30	0.24
Yin et al. [25]	0.59	0.54	0.56
Khare et al. [29]	0.49	0.48	0.48
Wu et al. [31]	**0.69**	0.62	**0.65**

TABLE IX
PERFORMANCE OF TEXT DETECTION ON ICDAR 2013 VIDEO

Methods	precision	recall	f-measure
Proposed+ Li et al. [7]	0.51	**0.75**	0.61
Li et al. [7]	0.46	0.70	0.56
Mosleh et al. [5]	0.50	0.49	0.49
Epshtein et al. [3]	0.48	0.47	0.47
Li et al. [27]	0.25	0.67	0.36
Zhao et al. [28]	0.23	0.32	0.27
Yin et al. [25]	0.64	0.57	0.60
Khare et al. [29]	0.55	0.53	0.54
Wu et al. [31]	**0.81**	0.73	**0.77**

TABLE X
PERFORMANCE OF TEXT DETECTION ON SVT

Methods	precision	recall	f-measure
Proposed+ Li et al. [7]	**0.77**	0.65	0.70
Li et al. [7]	0.74	0.60	0.66
Mosleh et al. [5]	0.76	**0.66**	**0.71**
Epshtein et al. [3]	0.73	0.60	0.66
Li et al. [27]	0.21	0.61	0.31
Zhao et al. [28]	0.18	0.20	0.19
Yin et al. [25]	0.50	0.38	0.43
Khare et al. [29]	0.41	0.44	0.42

TABLE XI
PERFORMANCE OF TEXT DETECTION ON MSRA

Methods	precision	recall	f-measure
Proposed+ Li et al. [7]	0.69	0.64	0.66
Li et al. [7]	0.67	0.61	0.64
Mosleh et al. [5]	0.56	0.53	0.55
Epshtein et al. [3]	0.52	0.50	0.51
Li et al. [27]	0.26	0.65	0.37
Zhao et al. [28]	0.34	0.69	0.46
Yin et al. [25]	0.61	0.71	0.66
Yao et al. [4]	0.63	0.63	0.60
Yin et al. [26]	0.63	**0.81**	**0.71**
Khare et al. [29]	0.45	0.53	0.48
Liang et al. [30]	**0.74**	0.66	0.70
Wu et al. [31]	0.63	0.70	0.66

street view images and arbitrary text images successfully. This shows that the proposed technique helps in achieving good text detection results. The quantitative results of the proposed and the existing techniques are respectively reported in Table 8-Table 11 for ICDAR 2015 video, ICDAR 2013 video, SVT and MSRA datasets. The proposed technique is the best at recall and f-measure for ICDAR 2015 video, the best at recall for ICDAR 2013 video, the best at precision for the SVT data and the second at precision for the MSRA data. One can notice from Table 8-Table 11 that the proposed technique scores better on recall, precision and f-measure than Li et al.'s method without reconstruction. This shows that the proposed shape restoration contributes significantly to text detection, separate from recognition. Table 8 and Table 9 show that the proposed technique is the best at recall compared to the other existing techniques. However, Wu et al.'s method achieves the best precision and f-measure. This is because the technique is developed for video text detection and uses temporal frames for improving text detection results. It is observed from Table 8 and Table 9 that the proposed technique scores the best precision compared to the other existing techniques. It is observed from Table 10 that Mosleh et al.'s method scores a better recall and f-measure compared to the other techniques. Since Mosleh et al.'s method involves stroke width transform, which is proposed for text detection in natural scene images, it scores a better recall and f-measure. For MSRA data as reported in Table 11, the proposed technique reports poor results compared to the other existing techniques. The reason is that since Li et al.'s method [7] is good for horizontal and slightly non-horizontal texts, it does not cope well with the challenges of arbitrary orientation of texts in the MSRA data. However, Yin et al.'s method achieves the best recall and f-measure and Liang et al.'s method is the best at precision compared to the other techniques because these techniques are invariant to different orientations. It is noted from Table 11 that the proposed technique is the second at precision compared to the other techniques for MSRA data.

In summary, we can conclude that the proposed technique is significant in two ways as it contributes to good recognition rates through character shape restoration for video and natural scene character images. It also improves text detection results both video and natural scene images. Sometimes, the proposed technique may not perform well for blurred and very small fonts for both character shape restoration and text detection. This is mainly because the proposed filters in Section III.A and III.B may fail to identify correct pairs when there is severe blurring. Therefore, there is further work required for extracting features which are invariant to blur.

V. CONCLUSIONS AND FUTURE WORK

This paper proposes a new technique to restore character contours in video and scene images. We have explored zero crossing points given by Laplacian operations in different domains for identifying stroke pixel candidates without using edge images. Based on the fact that character components generally have constant stroke widths and uniform gray values, a new set of filters are proposed in the gray and frequency domains for identifying probable stroke candidate pairs. Furthermore, spatial relationships and the distances between probable candidate stroke pairs are explored to eliminate false probable stroke candidate pairs, which results in seed stroke candidate pairs. Finally, we grow the seed stroke candidate pairs along tangent directions with color differences to restore missing candidate pairs. Our future work involves exploring temporal information and deep learning for extracting features that are invariant to blur to expand the scope of the proposed technique.

REFERENCES

[1] R. Minetto, N. Thome, M. Cord, N. J. Leite, and J. Stolfi, “SnooperText: A text detection system for automatic indexing of urban scenes,” *Comput. Vis. Image Understand.*, vol. 122, pp. 92–104, May 2014.

[2] Q. Ye and D. Doermann, “Text detection and recognition in imagery: A survey,” *IEEE Trans. PAMI*, vol. 37, no. 7, pp. 1480–1500, Jul. 2015.

[3] B. Epshtein, E. Ofek, and Y. Wexler, “Detecting text in natural scenes with stroke width transform,” in *Proc. CVPR*, Jun. 2010, pp. 2963–2970.

[4] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, “Detecting texts of arbitrary orientations in natural images,” in *Proc. CVPR*, Jun. 2012, pp. 1083–1090.

[5] A. Mosleh, N. Bouguila, and A. B. Hamza, “Automatic inpainting scheme for video text detection and removal,” *IEEE Trans. IP*, vol. 22, no. 11, pp. 4460–4472, Nov. 2013.

[6] J. Zhang and R. Kasturi, “A novel text detection system based on character and link energies,” *IEEE Trans. IP*, vol. 23, no. 9, pp. 4187–4198, Sep. 2014.

[7] Y. Li, W. Jia, C. Shen, and A. V. D. Hengel, “Characterness: An indicator of text in the wild,” *IEEE Trans. IP*, vol. 23, no. 4, pp. 1666–1677, Apr. 2014.

[8] L. Neumann and J. Matas, “Scene text localization and recognition with oriented stroke detection,” in *Proc. ICCV*, 2013, pp. 97–104.

[9] P. X. Nguyen, K. Wang, and S. Belongie, “Video text detection and recognition: Dataset and benchmark,” in *Proc. WACV*, Mar. 2014, pp. 776–783.

[10] T. Q. Phan, P. Shivakumara, T. Lu, and C. L. Tan, “Recognition of video text through temporal integration,” in *Proc. ICDAR*, Aug. 2013, pp. 589–593.

[11] T. Q. Phan, P. Shivakumara, S. Tian, and C. L. Tan, “Recognizing text with perspective distortion in natural scenes,” in *Proc. ICCV*, 2013, pp. 569–576.

[12] D. Chen and J. M. Odobez, “Video text recognition using sequential monte carlo and error voting methods,” *Pattern Recognit. Lett.*, vol. 26, no. 9, pp. 1386—1403, Jul. 2005.

[13] P. Shivakumara, T. Q. Phan, S. Bhowmick, C. L. Tan, and U. Pal, “A novel ring radius transform for video character reconstruction,” *Pattern Recognit.*, vol. 46, no. 1, pp. 131–140, Jan. 2013.

[14] S. Tian, P. Shivakumara, T. Q. Phan, and C. L. Tan, “Scene character reconstruction through medial axis,” in *Proc. ICDAR*, Aug. 2013, pp. 1360–1364.

[15] S. Tian, P. Shivakumara, T. Q. Phan, T. Lu, and C. L. Tan, “Character shape restoration system through medial axis points,” *Neurocomputing*, vol. 161, pp. 183–198, Aug. 2015.

[16] P. Shivakumara, D. B. Hong, D. Zhao, C. L. Tan, and U. Pal, “A New Iterative-Midpoint-Method for Video Character Gap Filling,” in *Proc. ICPR*, Nov. 2012, pp. 673–676.

[17] *Tesseract*, accessed on Oct. 15, 2014. [Online]. Available: http://code.google.com/p/tesseract-ocr/

[18] Z. Zhang and W. Wang, “A novel approach for binarization of overlay text” in *Proc. ICSMC*, Oct. 2013, pp. 4259–4264.

[19] K. Wang, B. Babenko, and S. Belongie, “End-to-end scene text recognition,” in *Proc. ICCV*, Nov. 2011, pp. 1457–1464.

[20] D. L. Smith, J. Field, and E. L. Miller, “Enforcing similarity constraints with integer programming for better scene text recognition,” in *Proc. CVPR*, Jun. 2011, pp. 73–80.

[21] A. Mishra, K. Alahari, and C. V. Jawahar, “Top-down and bottom-up cues for scene text recognition,” in *Proc. CVPR*, Jun. 2012, pp. 2687–2694.

[22] L. Kang, Y. Li, and D. Doermann, “Orientation robust text line detection in natural scene images,” in *Proc. CVPR*, Jun. 2014, pp. 4034–4041.

[23] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, “PhotoOCR: Reading text in uncontrolled conditions,” in *Proc. ICCV*, 2013, pp. 785–792.

[24] L. Neumann and J. Matas, “Real-time scene text localization and recognition,” in *Proc. CVPR*, Jun. 2012, pp. 3538–3545.

[25] X. C. Yin, Z. Yin, K. Huang, and H. W. Hao, “Robust text detection in natural scene images,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 970–983, May 2014.

[26] X. C. Yin, W. Y. Pei, J. Zhang, and H. W. Hao, “Multi-orientation scene text detection with adaptive clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1930–1937, Sep. 2015.

[27] H. Li, D. Doermann, and O. Kia, “Automatic text detection and tracking in digital video,” *IEEE Trans. Image Process.*, vol. 9, no. 1, pp. 147–156, Jan. 2000.

[28] X. Zhao, K. H. Lin, Y. Fu, Y. Hu, Y. Liu, and T. S. Huang, “Text from corners: A novel approach to detect text and caption in videos,” *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 790–799, Mar. 2011.

[29] V. Khare, P. Shivakumara, and P. Raveendran, “A new histogram oriented moments descriptor for multi-oriented moving text detection in video,”*Expert Syst. Appl.*, vol. 42, no. 21, pp. 7627–7640, Nov. 2015.

[30] G. Liang, P. Shivakumara, T. Lu, and C. L. Tan, “Multi-spectral fusion based approach for arbitrarily oriented scene text detection in video images,” *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4488–4501, Nov. 2015.

[31] L. Wu, P. Shivakumara, T. Lu, and C. L. Tan, “A new technique for multi-oriented scene text line detection and tracking in video,” *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1137–1152, Aug. 2015.

[32] G. Liang, P. Shivakumara, T. Lu, and C. L. Tan, “A new wavelet-laplacian method for arbitrarily-oriented character segmentation in video text lines,”in *Proc. ICDAR*, Aug. 2015, pp. 926–930.

[33] P. Shivakumara, T. Q. Phan, and C. L. Tan, “A laplacian approach to multi-oriented text detection in video,” I*EEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 412–419, Feb. 2011.

[34] F. Yuan, “Rotation and scale invariant local binary pattern based on high order directional derivatives for texture classification,” *Digit. Signal Process.*, vol. 26, pp. 142–152, Mar. 2014.

[35] D. Karatzas *et al.*, “ICDAR 2013 robust reading competition,” in *Proc. ICDAR*, Aug. 2013, pp. 1484–1493.

[36] D. Karatzas *et al.*, “ICDAR 2015 competition on robust reading,” in *Proc. ICDAR*, Aug. 2015, pp. 1156–1160.

[37] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, “ICDAR 2003 robust reading competitions,” in *Proc. ICDAR*, Aug. 2003, pp. 682–687.

[38] T. E. de Campos, B. R. Babu, and M. Varma, “Character recognition in natural images,” in *Proc. VISAPP*, vol. 2. 2009, pp. 273–280.

[39] K. Wang and S. Belongie, “Word Spotting in the Wild,” in *Proc. ECCV*, 2010, pp. 591–604.

[40] L. J. Latecki, R. Lakamper, and U. Eckhardt, “Shape descriptors for non-rigid shapes with a single closed contour,” in *Proc. CVPR*, vol. 1. Jun. 2000, pp. 424–429.

[41] N. Otsu, “A threshold selection method from gray-level histograms,” *Automatica*, vol. 11, nos. 285–296, pp. 23–27, 1975.

[42] W. Niblack, *An Introduction to Digital Image Processing*. Copenhagen, Denmark: Strandberg, 1985.

[43] J. Sauvola and M. Pietikainen, “Adaptive document image binarization,” *Pattern Recognition*, vol. 33, no. 2, pp. 225–236, 2000.

[44] J. Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide-baseline stereo from maximally stable extremal regions,” *IVC*, vol. 22, no. 10, pp. 761–767, Sep. 2004.

Yirui Wu is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Nanjing University. His current interests are in the areas of image processing, computer vision, and pattern recognition algorithms.

Palaiahnakote Shivakumara was a Research Fellow with the Video Text Extraction and Recognition Project, Department of Computer Science, School of Computing, National University of Singapore, from 2008 to 2013. He is currently a Senior Lecturer with the Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia.

He received the B.Sc., M.Sc., M.Sc. Technology, by research, and Ph.D. degrees in computer science from the University of Mysore, Mysore, Karnataka, India, in 1995, 1999, 2001, and 2005, respectively. He has authored over 190 research papers in national, international conferences, and journals. He co-authored the book entitled *Video Text Detection* (Springer, 2014). His research interests are in the area of image processing, pattern recognition, and video text analysis. He has been a Reviewer for several conferences and journals. He has been serving as an Associate Editor of the *ACM Transactions Asian Language Information Processing*.

Tong Lu (M’15) received the M.Sc. and B.Sc. degrees from Nanjing University in 2002 and 1993, respectively, and the Ph.D. degree in computer science from Nanjing University in 2005. He served as an Associate Professor and an Assistant Professor with the Department of Computer Science and Technology, Nanjing University, from 2007 to 2005. He is currently a Full Professor with Nanjing University. He has served as a Visiting Scholar with the National University of Singapore. He also served as a Visiting Scholar with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology. He is a member of the National Key Laboratory, Novel Software Technology, China. His current interests are in the areas of multimedia, computer vision, and pattern recognition algorithms/systems. He has authored over 60 papers and two books in his area of interest, and issued over 20 international or Chinese invention patents.

Chew Lim Tan received the B.Sc. degree (Hons.) in physics from the University of Singapore in 1971, the M.Sc. degree in radiation studies from the University of Surrey, U.K., in 1973, and the Ph.D. degree in computer science from the University of Virginia, USA, in 1986. He is currenlty a Professor with the Department of Computer Science, School of Computing, National University of Singapore. His research interests include document image analysis and text and natural language processing. He has authored over 460 research publications in these areas. He is a fellow of the International Association of Pattern Recognition.

Michael Blumenstein (SM’11) was the Head of the School of Information and Communication Technology, and was the earlier Dean (Research) of the Science, Environment, Engineering and Technology Group, Griffith University, QLD, Australia. He is currently a Professor and the Head of the School of Software, University of Technology Sydney, Australia. He is an internationally and nationally renowned Expert in the areas of pattern recognition and artificial intelligence (specifically Machine learning and Neural Networks). 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., the application of artificial neural networks to a flood emergency decision support system), neurobiology (e.g., the 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). He has authored over 160 papers in refereed conferences, journals, and books in these areas. He is a fellow of the Australian Computer Society. In 2009, he was named as one of Australia’s top ten emerging leaders in innovation in the Australian’s Top 100 Emerging Leaders Series supported by Microsoft.

Govindaraj Hemantha Kumar received the B.Sc., M.Sc., and Ph.D. degrees from the University of Mysore. He is currently a Professor with the Department of Studies in Computer Science, University of Mysore. He has authored over 200 papers in journals, edited books, and refereed conferences. His current research interests include numerical techniques, digital image processing, pattern recognition, and multimodal biometrics.