2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Video Scene Text Frames Categorization for Text Detection and Recognition

Longfei Qin[1], Palaiahnakote Shivakumara[2], Tong Lu[1], Umapada Pal[3] and Chew Lim Tan[4]

[1] National Key Lab for Novel Software Technology, Nanjing University, Nanjing, China

[2] Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia

[3] Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India

[4]School of Computing, National University of Singapore, Singapore

18205188190@163.com;hudempsk@yahoo.com, lutong@nju.edu.cn, umapada@isical.ac.in, tancl@comp.nus.edu.sg

***Abstract*—Developing a unified text detection and recognition method is hard for different video types due to varying characteristics in video. This paper proposes a new method for categorizing different types of video text frames, namely, videos containing advertisement, signboard, license plate, front page of book or magazine, street view, and video of general items, for better text detection and recognition rate. We propose symmetry features using gradient vector flow for Canny and Sobel edge images of each input frame to identify candidate edge components. Then for a candidate edge component image, we extract both global and local features using colors from different channels in a new way. Besides, the proposed method extracts statistical and structural features from the spatial distribution of candidate pixels in a multi-scale environment. Lastly, the extracted features are fed to a logistic classifier for categorization. The features extracted locally and globally are tested both separately and altogether in terms of confusion matrix. The performance of the proposed categorization method is evaluated through several text detection and recognition experiments before and after categorization. We noted that the proposed categorization method is very useful in improving text detection and recognition performance.**

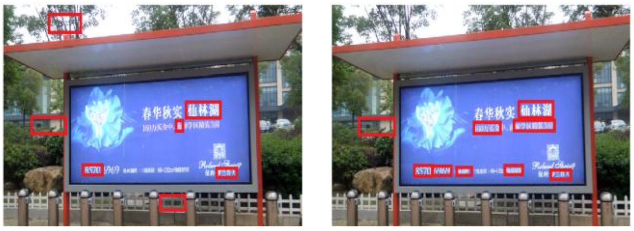
Keywords*—*Multi-scale global features, Multi-scale local features, Scene text Detection, Scene text recognition, Video scene text frame categorization.

I. Introduction

Text detection and recognition in video and natural scene images is a popular research topic currently in the field of video document image analysis because day by day the number of real time applications (e.g., creating smart digital cities by iTown, assisting tourists to identify interesting spots, safe driving, license plate tracking and recognition, various surveillance applications [1], etc) on text detection and recognition is increasing exponentially. As a result, a large database collection includes a variety of text frames from video as well as natural scene images. Video contains two types of texts, namely, graphics text and scene text [2, 3]. The methods which focus on graphics texts are generally developed for the purpose of indexing and retrieval. On the other hand, the methods which focus on scene texts are developed for the purpose of real time applications as mentioned above. When we compare graphics text based applications with scene text based ones, we find that achieving a good text detection and recognition accuracy for scene text databases is not as easy as graphics text detection and recognition because scene text suffers from background complexity, low resolution, contrast variations, font, font size variation, color bleeding and different orientations in contrast to graphics text [2, 3]. Therefore, achieving a good accuracy for scene text in video and natural scene images is challenging.

There are a plenty of methods proposed in the past years for detecting and recognizing scene texts in video and natural scene images [4-10]. However, when we apply the same method on different scene text datasets, the methods give inconsistent results. In other words, there is no universal or unified method which can give good accuracies for different databases because each dataset has its own complexity and characteristics. For instance, Zhang and Kasturi [6] proposed a method for text detection in video and natural scene images, which gives f-measure as 0.67 for the ICDAR 2003/2005 dataset, while for the Microsoft street view dataset, the same method gives f-measure as 0.44. Phan et al. [7] proposed a method for text detection in natural scene images, which gives f-measure as 0.66 for ICDAR 2003 data but for Microsoft street view dataset, it gives f-measure as 0.48. Kang et al. [8] proposed a method for robust text line detection in natural scene images, which gives f-measure as 0.66 for MSRA-TD500 database while for OSTD dataset, it gives f-measure as 0.76. Thus there is a big difference in f-measures for different datasets. In the same way, Mishra et al. [9] proposed a method for scene text recognition in natural scene images, which gives recognition rate 81.7% for ICDAR 2003 data and for Street View Data (SVT), it gives 73.2%. Phan et al. [10] proposed a method for recognizing perspective text in natural scene images, which gives 82.2% recognition rate for ICDAR 2003 data and 73.7% recognition rate for SVT data.

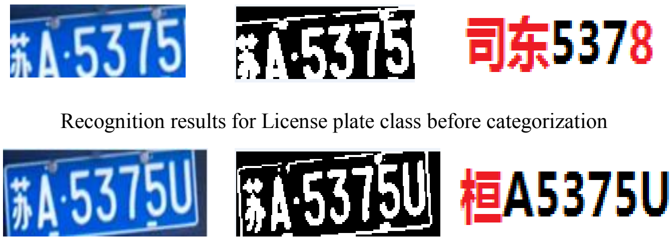
From the above discussions one can notice that there is a substantial margin in the accuracies for different datasets especially for text detection, and the gap sometimes almost reaches 20%. As the results depend on datasets, there is an urgent need for categorization of different video texts automatically before choosing appropriate text detection and recognition methods to achieve better results. The concept is in line with multi-script recognition, which identifies scripts before choosing an appropriate OCR engine to recognize texts of different scripts [11]. This is because developing a universal OCR for multiple scripts is difficult and it is not always advisable. In this work, we choose six classes, namely, Advertisement, Sign board, License plate, Book, Street view and Items (includes text on bottles, cards, computers and so on). The main reason to choose these six classes is that they pose different complexities and are used extensively for the above mentioned real time applications, particularly constructing smart and digital cities. For example, we can expect multiple color texts with fancy fonts for Advertisement video, distortions due to uneven illumination, vehicle movements for License plate videos, multiple fonts and colors for Book video, complex background due to greenery, sky and buildings for Street View video, and small fonts text on curved surfaces for video of General Items. This complex nature of different videos requires categorization before choosing proper text detection and recognition methods to enhance OCR performances.



(a) Text detection before categorization (b) Text detection after categorization

Fig. 1: Sample text detection results of an advertisement video frame before and after categorization by the text detection method [1]

One such illustration is shown in Fig. 1, where we can see text detection results before and after categorization for the sample images chosen from the Advertisement class. Fig. 1 shows that the text detection method (here we use the text detection method [1] which works well for complex background scene images) does not detect texts properly before categorization as shown in Fig. 1(a), while the same method works better to detect text lines after categorization as shown in Fig. 1(b). This is true because once we identify the class, we can modify the text detection method accordingly such that it gives better results. The same conclusion can be drawn from Fig. 2 also, where we can see the recognition results for the detected text lines before and after categorization. In this case, recognition results depend on the output of the text detection method before and after categorization. We use Niblack [12] thresholding technique for binarization and the publicly available OCR [13] for the illustration on our recognition experiments.



Recognition results for License plate class before categorization

Recognition results for License plate class after categorization

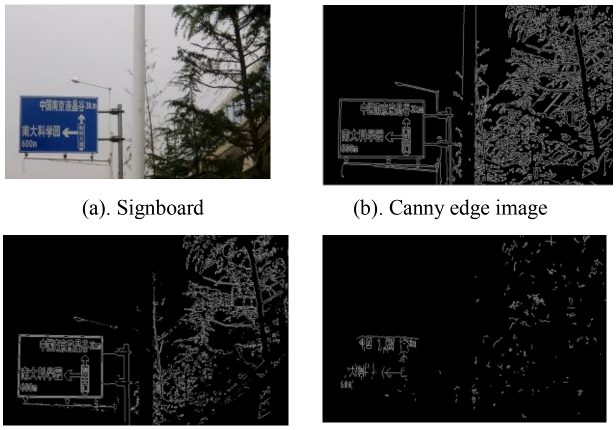
Fig. 2: Recognition results for the detected text lines corresponding to text detection before and after classification. Note: errors are indicated in red.

Recently, several methods have been developed for arbitrary oriented text detection in video and natural scene images. Shivakumara et al. [14] proposed a gradient vector flow and grouping method for arbitrarily oriented scene text detection in video images. Suyu et al. [15] proposed a two-level algorithm for text detection in natural scene images, which uses connected component analysis based features and an SVM for text detection. Tang et al. [16] proposed a spatial-temporal approach for video caption detection and recognition, which uses fuzzy clustering and neural networks. Phan et al. [17] proposed a method for the recognition of video texts through temporal integration, which uses stroke width information. Yao et al. [18] proposed a unified framework for multi-oriented text detection and recognition in natural scene images. To the best of our knowledge, there is no work on video scene text frame categorization and this is the first attempt to solve video scene text frame categorization to enhance the performance of text detection and recognition methods for different video types. Another advantage of this categorization is that rather than developing a new method, we can modify the existing methods to get better results for different categories of videos after categorization.

II. PROPOSED METHODOLOGY

When we observe the images corresponding to the six classes, namely, Advertisement, Sign Board, License Plate, Street View, Book and other Items, we note that color feature can play a prominent role for classifying Advertisement, Book, License Plate and Items because generally the color at text line level for these classes may not change much compared to its background. At the same time, the color of background may play a prominent role for classifying Signboard and License Plate because texts embedded in these classes of frames usually have homogenous background. Street View is very complex because in such data, texts exist with complex backgrounds consisting of trees, buildings, greenery, etc. As a result, color based features and background features are not sufficient to classify them correctly. Therefore, we need new features which can combine color, text background, textual properties and spatial distribution of text pixels as well as background pixels. With this notion, the proposed method first identifies candidate edge components from each input frame. Inspired by the work presented in [7] for identifying text candidates using the Canny and Sobel edge images of an input frame, we explore the same concept for identifying candidate edge components in this work. The basis is that Canny and Sobel edge images share the common properties for text regions and share different properties for non-text regions at the same time. This is true because Canny and Sobel edge detectors produce the same edge patterns for characters in text regions. In order to find common candidate pixels which satisfy this property in both Canny and Sobel images, we propose to use symmetry that exists inter and intra characters, which is estimated using Gradient Vector Flow (GVF). This results in candidate edge components.

For a candidate edge component image, we further extract features based on color histograms in gray, RGB and HSV color spaces because it is known that these color features play an important role in classifying the above mentioned classes. Besides, it is also true that due to contrast variation, one color sub-band may miss text lines and at the same time, the same text pixel may appear in other color sub-band. Therefore, the use of different color spaces help in restoring missing text pixels. Additionally, to extract the appearance property of text pattern and background texture, we propose to extract statistical features and then spatial features based on end, junction and intersection points of the candidate pixels image [19]. These features help in extracting textual property even where complex background exists in the image because these features will consider spatial distributions of text and background pixels. Moreover, to extract the same observation for different fonts and font sizes, we extract the same set of features in multi-levels, which looks like a pyramid structure. This results in multi-scale global features. Similarly, to extract the same observations for local regions, we extract the same set of features for the divisions given by a Quad tree at the first level. This results in multi-scale local features. We combine both the multi-scale global and local features and then feed them to a logistic classifier to obtain the final categorization results.



(a). Signboard (b). Canny edge image

(c). Sobel edge image (d).Candidate edge components

Fig. 3: Candidate edge components selection

A. Candidate Edge Components Selection

As discussed in the previous section, for the image shown in Fig. 3(a) which belongs to the Signboard class, the proposed method obtains its Canny and Sobel edge images as shown in Fig. 3(b) and (c), respectively, where one can see the same text patterns in both the edge images. This is the advantage of combining Canny and Sobel edge images for identifying candidate edge components. In order to extract the common pattern which looks similar in both the images, we propose a symmetry property based on GVF which finds symmetry that exists in intra and inter characters. The components in the Canny and Sobel images that satisfy the symmetry are considered as Candidate Edge Components (CEC) as shown in Fig. 3(d), where we can see CECs which represent text information as well as some background information. Essentially, GVF is the extension of gradient information. Unlike the normal gradient which gives little information in homogenous regions, GVF propagates gradient information from nearby regions into homogenous ones. Hence, it helps to increase the capture range of the edges and attract active contours into concave regions [20]. GVF field is computed by minimizing the following expression:

$$\varepsilon = \iint u\left({u_x}^2 + {u_y}^2 + {v_x}^2 + {v_y}^2\right) + |\nabla f|^2 |g - \nabla f^2| dxdy \quad (1),$$

where$g(x,y) = (u(x,y), v(x,y))$ is the GVF field and $f(x,y)$ is the edge map of the input image.To extract the common patterns of edge components in candidate edge images, we define symmetry as follows. The symmetry edge components [9] are defined as the set of locations where two neighboring GVF arrows are opposite to each other, because this indicates that the region is at the center of two edges. Concretely, (x, y) is a symmetry point in the vertical direction, if and only if:

$$g(x,y) \cdot g(x+1,y) < \theta \quad (2)$$

Intuitively, the above condition requires that the inner product between the GVF vector at $(x,y)$and the GVF vector at $(x+1,y)$ must be smaller than a negative value. In addition to the vertical direction, we derive similar constraints for symmetry points in three other directions: horizontal, left-diagonal and right-diagonal. As long as one of the four directions is satisfied, we view the point as a symmetry point.

The CEC can be viewed as the intersection of Sobel symmetry edge components and Canny symmetry edge components. Let $C = \{c_i\}$ and $S = \{s_j\}$ be the sets of Canny and Sobel symmetry edge components. Also let the intersection of C and S is $C \cap S$. For each element $sc \in C \cap S$, if the size of $sc \geq \alpha$,then sc is retained as symmetry edge components CEC. In order to extract color features, we need to map the candidate edge components to the input image, and the mapped result is defined as map_CEC:

$$map\_CEC = Adjacency(I \cap CEC) \quad (3)$$

where Adjacency(X) represents the 8-adjacency region for each pixel in X; I is the original image and CEC is the candidate edge components.

B. Feature Extraction for Candidate Edge Components in Different Color Spaces

As discussed in the proposed methodology section, color provides a vital clue for the categorization of video text frame classes. Therefore, we plot a histogram for the gray values corresponding to the edge components in the CEC image by quantizing the histogram bins into 10. We determine the number of bins, i.e. 10, for quantization based on our experimental studies. The proposed method calculates the percentage of pixel values in each bin. This results in 10 features for the gray color CEC image. To extract more color features, we split the input color image into RGB spaces separately. The same feature extraction scheme is deployed for these three RGB spaces. This gives 30 features. In the same way, the method extracts another 30 features for HSV spaces. In total, 70 (10+30+30) features are extracted from different color spaces. The feature extraction can be represented mathematically as follows.

$$[Gray, RGB\ , HSV] = Hist(map\_CEC) \quad (4)$$

where map_CEC denotes the candidate edge components of the respective color spaces. It is true that color features alone may not be sufficient to solve this complex categorization problem. Inspired by the features extracted in [19] for the identification of different scripts in video, we explore the same statistical and spatial relationship based features for the categorization of video text frames in this work. These features require the dominant points for each candidate edge component in the CEC image.

The proposed method finds dominant points, such as end, junction and intersection points as defined in equation (5). For any pixel P, the adjacent pixels of P are defined as ADJ_P, the connected component which contains P is defined as CON_P:

$$\mathrm{CON_P} \cap \mathrm{ADJ_P} = \begin{cases} 1, \quad P \in \quad end\ point \\ 2, \ P \in \quad pixels\ witH\ two\ neigbs \\ 3, \quad P \in \ junction\ point \\ 4, P \in intersection\ point \end{cases} \quad (5)$$

To study the relationship between dominant pixels, the proposed method computes a proximity matrix by calculating the geodesic distance between respective dominant points. For instance, the proximity matrix for an end point can be computed as defined in equation (6), where an end point is represented by $EP_{i,j}$ and P_{End} is a set of end points.

$$EP_{i,j} = NGD(P_{End_i}, P_{End_j}) \quad (6)$$

Here, $\mathrm{NGD}(\mathrm{P_{End_i}}, \mathrm{P_{End_j}})$ calculates the Nearest Geodesic Distance between $\mathrm{P_{End_i}}$ and $\mathrm{P_{End_j}}$. We use a dynamic programming algorithm to obtain the distance between dominant points. Similarly, we estimate the proximity matrices for junction points, intersection points and all the pixels. Overall, the procedure gives four proximity matrices respectively for end, junction, intersection and pixels. The proposed method computes the means and variances for the respective proximity matrices. For example, the mean and variance for the proximity matrix of the end points can be calculated by equation (7) and equation (8), respectively:

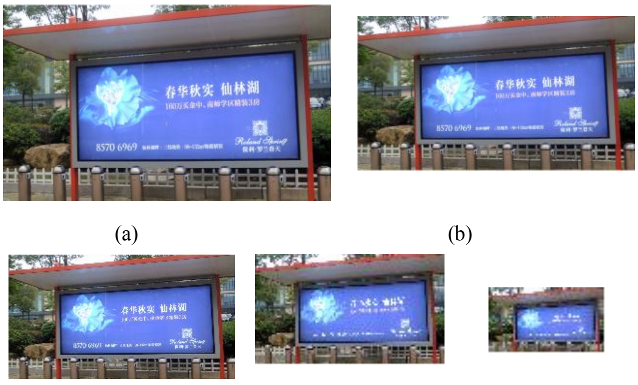
$$Mean_{EP} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{i} EP_{i,j} \ (7)$$

$$Var_{EP} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{i}(EP_{i,j} - Mean_{EP})^2 \quad (8)$$

where $Mean_{EP}$ denotes the mean of the proximity matrices of end points, Var_{EP} denotes the variance of the proximity matrices, and n is the number of end points. This process gives a total of 8 features. In the same way, we estimate the covariance matrices for dominant points and pixels in the respective points. This helps in extracting the correlation between the dominant points. This procedure gives 8 more features. In total, 16 features are extracted from the proximity matrices of the dominant points and pixels in the CEC image of the input frame of each class. With these 16 features, the total number of features becomes 86.

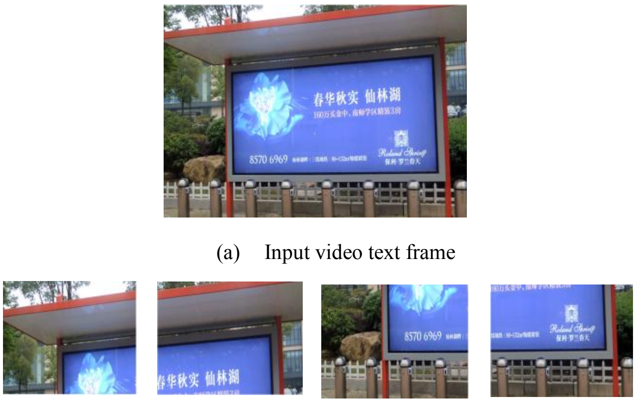
C. Multi-Scale Global and Local Features for Categorization

In order to cope with the problems of font and font size variations, we propose to extract the global and local features in a multi-scale environment using pyramid structure, scaling and quad tree division, respectively. Therefore, the method reduces the size of each input frame to a quarter of the input frame size by downsizing. Again, the downsized image is further reduced to a quarter of the downsized image. The process continues till the method gets five downsized images from the original input frame as shown in Fig. 4, where we can see (a)-(e) represent four downsized images, respectively. According to our experiments, five levels are sufficient to achieve good results. Then the proposed method extracts all the 86 features from the five levels, which gives 430 (86×5) features. Since the method uses the whole image for downsizing, we name them multi-scale global features. To extract the features which consider local information, we divide the given input frame size into four equal sub-parts by Quad tree division as shown in Fig. 5, where (a) is the input frame and (b) gives the four quadrants of the image in (a). Our experimental results show that one level containing four quadrants is enough for categorization. The proposed method extracts 86 features from the four quadrants of the given Quad tree. This gives 344 (86×4) features. In total, we obtain 774 (430+344) dimensional feature vectors for categorization. Finally, these features are fed to a logistic classifier as proposed in [21] for categorization.


(a) (b)

(c) (d) (e)

Fig. 4: Down scaling like pyramid structure for multi-scale global features: Candidate edge components selection: (a) actual size (b) reduced to half size to (a), (c) size reduced to half of (b), (d) size reduced to half of (c), and (e) size reduced to half of (d).


(a) Input video text frame

(b) Four equal sized sub-parts of (a)

Fig. 5: Dividing given frame into sub-parts like Quad tree for multi-scale local features

III. EXPERIMENTLAL RESULTS

We create our own dataset because as per our knowledge, this is the first attempt to solve video scene text frame categorization to enhance the performance of text detection and recognition methods. This dataset includes 100 frames per class. In total, 600 frames for the six classes, namely, Advertisement, Signboard, License plate, Street view, Book and Items. For categorization, we feed the feature vectors to a logistic classifier as proposed in [21], where the classifier has been used for image categorization but not video text frame categorization. In this work, we follow the same procedure given in [21] for categorization of our dataset. To evaluate the proposed categorization method, we perform 10 fold cross validation scheme. We present the average of confusion matrix results given by respective 10 fold experimentation. We use categorization rate for measuring the performance of the proposed categorization method, recall, precision and f-measure for evaluating text detection methods, and recognition rate for evaluating recognition results in this work. For text detection and recognition, we follow the instructions given in [22] for calculating the measures.

In order to show the effectiveness and usefulness of the categorization, we implement the state of the art text detection methods, such as Minetto et al. [1] which uses urban scene images for text detection, Yi et al. [4] which detects text in bottle and items etc, and Xu et al. [22] detects text regions of different scripts with arbitrary orientations to conduct the experiment before and after categorization. Before categorization experiments we consider the frames of all the six classes for calculating recall, precision and f-measure, while after categorization experiments we also consider the frames of the six classes separately. In the same way, we implement three baseline binarization methods, namely, Niblack [12], Otsu [23] and Sauvola [24], which are classical methods for binarizing scanned and camera based documents to conduct experiments before and after categorization through recognition rate. We also use a recent method [25], which detects texts for blur images based on inverse rendering. For calculating recognition rate, we use Tesseract OCR which is available publicly [13] and considers the output of binarization methods for recognition.

A. Experiments on Categorization

To know the contributions of multi-scale global and local features, we estimate the confusion matrix for each feature extraction scheme separately as shown in Table I and Table II, respectively. When we compare the average categorization rates (the average of the diagonal elements in Table I and Table II), the categorization rate of multi-scale global features is lower than that of multi-scale local features. However, the difference is only around 3%. This shows that both the feature extraction schemes contribute significantly. This can be seen from the results reported in Table III, where it provides the confusion matrix for the combined features (multi-scale global + local features). It is interesting to note that the categorization rate for the combined features is 88.1%, which is a significant improvement compared to 77.3% of multi-scale global and 80.8% of multi-scale local features.

Table I: Confusion matrix for Multi-scale global features

Classes	Adv	Sign	License	Street	Book	Items
Adv	79.0	2.0	1.0	9.0	4.0	5.0
Sign	3.0	89.0	3.0	4.0	1.0	0.0
License	3.0	10.0	67.0	12.0	5.0	3.0
Street	3.0	5.0	9.0	79.0	0.0	4.0
Book	4.0	4.0	3.0	2.0	77.0	10.0
Items	2.0	1.0	6.0	1.0	14.0	76.0

Table II: Confusion matrix for Multi-scale local features

Classes	Adv	Sign	License	Street	Book	Items
Adv	81.0	4.0	7.0	6.0	2.0	0.0
Sign	3.0	88.0	3.0	3.0	2.0	1.0
License	3.0	11.0	79.0	5.0	0.0	2.0
Street	6.0	3.0	8.0	82.0	0.0	1.0
Book	3.0	6.0	3.0	1.0	76.0	11.0
Items	3.0	1.0	1.0	1.0	15.0	79.0

Table III: Confusion matrix for combining Multi-scale global and local features

Classes	Adv	Sign	Items	Street	License	Book
Adv	90.0	3.0	0.0	1.0	3.0	3.0
Sign	3.0	81.0	6.0	5.0	3.0	2.0
Items	1.0	6.0	88.0	3.0	1.0	1.0
Street	5.0	2.0	1.0	89.0	1.0	2.0
License	2.0	0.0	0.0	2.0	94.0	2.0
Book	6.0	1.0	1.0	3.0	2.0	87.0

B. Validating Classficaiton Through Text Detection

As mentioned above, we report the results of the three text detection methods before and after categorization in Table IV. Table IV shows that text detection methods when used after categorization and shows better results compared to that before categorization. Therefore, we can conclude that categorization is useful in improving the performance of text detection methods. Since the video frames of different classes have different complexities, text detection methods, when used before categorization, give poor accuracies. The same text detection methods give better accuracies after categorization because the methods can be tuned and modified according to the complexity of the data. Since our goal is to show inconsistent accuracies for different videos, we report the detection results by changing datasets after categorization without tuning and modifying the existing methods. This is the advantage of the categorization. It is also observed from Table IV that Minetto et al.'s method is better at precision before categorization and better at recall, precision and f-measure after categorization as compared to the other methods. Therefore, we use the same text detection method to test on individual classes as reported in Table V, where we note that the method gives better results after categorization. It is found that the text detection method gives low f-measure for street view data compared to other data as reported in Table V because street view data is much more complex than the other data. Furthermore, the output of this method is used for binarization experiments to calculate recognition rate in the next section.

Table IV: Performance of different text detection methods before and after categorization

Text Detection methods	Before Categorization			After Categorization		
	P	R	F	P	R	F
Minetto et al [1]	0.69	0.20	0.31	0.79	0.32	0.45
Yi et al. [4]	0.60	0.18	0.28	0.68	0.28	0.39
Xu et al. [22]	0.58	0.23	0.33	0.70	0.31	0.43

Table V: Performance of text detection method [1] for all the six classes before and after categorization

Classes	Before Categorization			After Categorization		
	P	R	F	P	R	F
Advertisement	0.66	0.18	0.29	0.76	0.30	0.43
Sign	0.79	0.19	0.31	0.87	0.27	0.41
Items	0.50	0.13	0.20	0.74	0.16	0.27
Street	0.67	0.17	0.27	0.65	0.23	0.34
License	0.80	0.42	0.55	0.88	0.58	0.70
Books	0.72	0.10	0.18	0.85	0.36	0.50

C. Validating Categorization Through Recognition

As we conclude from text detection experiments in the previous section, the same conclusion can be drawn from the recognition experiments of different binarization methods as reported in Table VI. We calculate the recognition rate (RR) for the original (Ori) text line image detected by the text detection method without applying binarization method using Tesseract OCR. The results show that achieving a good recognition rate for video text lines of different applications as well as data is not as easy as achieving the recognition rate for scanned document images due to background, contrast, font and font size variations. We also conduct experiments on binarization methods output to show the effectiveness of binarization methods on video scene text line images. Table VI shows that the recognition rates of the four binarization methods are lower before categorization compared to that of after categorization

IV. Conclusion and Future Work

In this paper, we propose a new method for video scene text frame categorization to improve the performance of text detection and recognition methods for video scene text frames data. We explore the common property of Canny and Sobel operation for identifying candidate edge components. For candidate edge components image, we extract different features such as color, statistical and spatial features at different scales: globally and locally. Further, we combine multi-scale global and local features for the final categorization using a logistic classifier. Experimental results on text detection and recognition obtained before and after categorization show that categorization is essential for improving the accuracy of text detection and recognition methods on video scene text data. In future, we will investigate further to improve the accuracy for document analysis using more classes with new categorization methods.

Acknowledgment

The work is supported by the Natural Science Foundation of China under Grant No. 61672273, No. 61272218 and No. 61321491, and the Science Foundation for Distinguished Young Scholars of Jiangsu under Grant No. BK20160021. The work is also partly supported by the University of Malaya HIR under Grant No. M.C/625/1/HIR/210.

References

[1] R. Minetto, N. Thome, M. Cord, J. Fabrizio, andB.Marcotegui,"Snoopertext: A multiresolution system for text detectionin complex visual scenes," In Proc. ICIP, 2010, pp.1–4.

[2] Q. Ye and D. Doermann, "Text Detection and Recognition in Imagery: A Survey", IEEE. Trans. PAMI, 2015, pp 1480-1500.

[3] J. Zang and R. Kasturi, "Extraction of Text Objects in Video Documents: Recent Progress", In Proc. DAS, 2008, pp 5-17.

[4] C. Yi and Y. Tian, "Text string detection from natural scenesby structure-based partition and grouping", IEEE Trans. IP, 2011, pp 2594 –2605.

[5] A. Jain, X. Peng, X. Zhuang, P. Natarajan and H. Cao, "Text detection and recognition in natural scenes and consumer videos", In Proc. ICASSP, 2014, pp 1245-1249.

[6] J. Zhang and R. Kasturi, "A Novel Text Detection System based on Character and Link Energies", IEEE Trans. PAMI, 2013, pp 4187-4198.

[7] T. Q. Phan,P. Shivakumara, C. L. Tan,"Detecting Text int the Real World". In Proc. ACMMM, 2012, pp 765-768.

[8] L. Kang, Y. Li and D. Doermann, "Oreintation Robust Text Line Detection in Natural Scene Images", In Proc. CVPR, 2014, pp 4034-4041.

[9] A. Mishra, K. Alahari and C. V. Jawahar, "Top-Down and Bottom-Up Cues for Scene Text Recognition", In Proc. CVPR, 2012, pp 2687-2694.

[10] T. Q. Phan, P. Shivakumara, S. Tian and C. L. Tan, "Recognizing Text with Perspetive Distortion in Natural Scene Images", In Proc. ICCV, 2013, pp 569-576.

[11] N. Sharma, R. Mandal, R. Sharma, P. P. Roy, U. Pal and M. Blumenstein, "Multi-Lingual Text Recognition from Video Frames", In Proc. ICDAR, pp 951-955, 2015.

[12] W. Niblack, "An introduction to digital image processing", Strandberg Publishing Company, 1985.

[13] Tesseract. http://code.google.com/p/tesseract-ocr/.

[14] P. Shivakumara, T. Q. Phan, S. Lu and C. L. Tan, "Gradient Vector Flow and Grouping based Method for Arbitrarily-Oriented Scene Text Detection in Video Images", IEEE Trans. CSVT, pp 1729-1739, 2013.

[15] L. . R. W. Suyu and Z. X. Shi, "A two level algorithm for text detection in natural scene images", In Proc. IWDAS, 2014, pp 329-333.

[16] X. Tang, X. Gao, J. Liu and H. Zhang, "A Spatial-Temporal Approach for Video Caption Detection and Recognition", IEEE Trans. NN, 2002, pp 961-971.

[17] T. Q. Phan, P. Shivakumara, T. Lu and C. L. Tan, "Recognition of Video Through Temporal Integration", In Proc. ICDAR, 2013, pp 589-593.

[18] C. Yao, X. Bai and W. Liu, "A unified framework for multi-oriented text detection and recognition", IEEE Trans. IP, 2014, pp 4737-4749.

[19] P. Shivakumara,Z. Yuan,D. Zhao,T. Lu,C. L. Tan,"New Gradient-Spatial-Structural Features for Video Script Identification".Computer Vision and Image Understanding, 2015, pp . 35-53.

[20] C. Xu and J. L. Prince, " Snakes, Shapes, and Gradient Vector Flow", IEEE Trans. IP, 1998, pp. 359–369.

[21] D. Dai and L. V. Gool, "Ensemble Projection for Semi-supervised Image Classification", In Proc. ICCV, 2013, pp 4321-4328.

[22] J. Xu, P. Shivakumara, T. Lu, T. Q. Phan, C. L. Tan, "Graphics and Scene Text Classification in Video", In Proc. ICPR, 2014, pp 4714-4719.

[23] N. Otsu, "A theshold seelction method from gray level histogram", IEEE Trans. SMC, 1978, pp 62-66.

[24] J. Sauvola, T. Seeppanen, S. Haapakoski and M. Pietikainen, "Adaptive document binarization", In Proc. ICDAR, pp 147-152, 1997.

[25] Y. Zhou, J. Feid, E. L. Miller and R. Wang, "Scene Text Segmentation via Inverse Rendering", In Proc. ICDAR, 2013, pp 457-461.

Table VI: Recognition Rate (RR) based on different binarization methods before and after categorization

Binarization Methods	Before Categorization		After Categorization											
			Advertisement		Sign		License plate		Street view		Book		Item	
	Ori	Binary	Ori	Binary	Ori	Binary	Ori	Binary	Ori	Binary	Ori	Binary	Ori	Binary
	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR
Niblack [12]	0.16	0.22	0.18	0.22	0.20	0.19	0.26	0.35	0.13	0.14	0.20	0.21	0.17	0.18
Otsu [23]	0.23	0.25	0.22	0.22	0.20	0.23	0.21	0.22	0.13	0.15	0.16	0.16	0.20	0.21
Sauvola [24]	0.10	0.11	0.22	0.23	0.20	0.21	0.25	0.30	0.13	0.12	0.16	0.17	0.14	0.15
Y. Zhou[25]	0.28	0.32	0.31	0.39	0.30	0.38	0.4	0.42	0.15	0.16	0.20	0.24	0.22	0.24