2012 10th IAPR International Workshop on Document Analysis Systems

New Spatial-Gradient-Features for Video Script Identification

[a]Danni Zhao,[a]Palaiahnakote Shivakumara, [b]Shijian Lu and [a]Chew Lim Tan
[a]School of Computing, National University of Singapore, Singapore
[a]{zhao89, shiva, and tancl}@comp.nus.edu.sg
[b]Institute for Infocomm Research, Singapore, [b]slu@i2r.a-star.edu.sg

***Abstract*—In this paper, we present new features based on Spatial-Gradient-Features (SGF) at block level for identifying six video scripts namely, Arabic, Chinese, English, Japanese, Korean and Tamil. This works helps in enhancing the capability of the current OCR on video text recognition by choosing an appropriate OCR engine when video contains multi-script frames. The input for script identification is the text blocks obtained by our text frame classification method. For each text block, we obtain horizontal and vertical gradient information to enhance the contrast of the text pixels. We divide the horizontal gradient block into two equal parts as upper and lower at the centroid in the horizontal direction. Histogram on the horizontal gradient values of the upper and the lower part is performed to select dominant text pixels. In the same way, the method selects dominant pixels from the right and the left parts obtained by dividing the vertical gradient block vertically. The method combines the horizontal and the vertical dominant pixels to obtain text components. Skeleton concept is used to reduce pixel width to a single pixel to extract spatial features. We extract four features based on proximity between end points, junction points, intersection points and pixels. The method is evaluated on 770 frames of six scripts in terms of classification rate and is compared with an existing method. We have achieved 82.1% average classification rate.**

Keywords- Video text blocks, Gradient blocks, Dominat video text pixels, Spatial-gradient-features, Video scrpt identification

I. INTRODUCTION

The main problem of the current OCR methods is that when the video contains frames of different scripts then the OCR methods fail to perform because the methods are designed for single script recognition and there is no universal OCR for recognizing multiple script frames in video. Therefore, to enhance the capability of the current OCR readability, it is necessary to identify scripts of different frames in video. However, low resolution, complex background, orientation, different fonts and font sizes of video text make this problem difficult and challenging [1-3].

There are methods for text detection, text region detection and text block detection in video [4-6], which can be classified as connected component-based, texture-based, and gradient and edge-based methods. Although these methods achieve good accuracy for text detection and text block detection irrespective of fonts, font sizes, types of text and orientation, they do not have the ability to differentiate multi-script frames in video because the goal of these methods was to detect text block or text region regardless of scripts in the video [4-6] and not to identify the script.

Similarly, we can see text recognition methods which take care of segmentation and binarization of the text areas/blocks with the help of enhancement criteria to increase the contrast of text lines before feeding them to OCR [7]. However, these methods are limited to single script frames in video. In other words, text recognition methods fail to perform on multi-script frames in video because the extracted features and OCR are usually designed for specific languages. Furthermore, in multi-lingual countries like Singapore and India, it is common to have multiple languages in the same video frame. Therefore, there is immense scope for identifying the scripts before selecting the appropriate OCR to improve the OCR performance as it is hard to develop a universal OCR. To the best of our knowledge, the work on video script identification is not much reported in the literature. However, we found one paper which takes text lines detected by the text detection methods as input and uses statistical and texture features with k-nearest neighbor classifier to identify Latin and Ideographic text in images and videos [1]. This method works well for English and Chinese but not for other scripts. In addition, its performance depends on the classifier. The new features, namely smoothness and cursiveness based on text lines without classifier are proposed for video script identification recently by us [2]. This method considers only English, Chinese and Tamil scripts and it is noted from the experimental results that the features are not good enough to handle more than three scripts present in video frames. To overcome this limitation, in this work, we propose new Spatial-Gradient-Features for identifying six scripts, namely Arabic, Chinese, English, Japanese, Korean and Tamil.

Identification of different scripts in a document with plain background and high resolution is a familiar problem in document analysis. In this regard, we can see lot of methods in the literature. An overview of script identification methodologies based on structure and visual appearance is presented in [8]. It is noted from this review that the proposed methods work well for camera-based images but not for video frames since the latter has low contrast and complex background. The rotation invariant features for automatic script identification are proposed by Tan [9] based on Gabor filter. This work considers six scripts for identification. Busch et al. [10] have explored the combination of wavelet and Gabor features to identify the scripts. However, these approaches expect a large number of training samples to achieve good classification rate. Lu and Tan [11] have proposed a method for script identification in noisy and degraded document images based on document vectorization. Although the method is tolerant to various types of document degradations, it does not perform well for Tamil because of the complexity of the script. Texture features based on Gabor filter and Discrete Cosine Transform are used for script identification at the word level in many papers [12-14] where the methods expect high contrast document for segmentation of words. Similarly, a study of character shape for identifying scripts is proposed in [15, 16]. These methods perform well as long as segmentation works well and the character shape is preserved. Online script identification is addressed in [17] where the spatial and temporal information is used to recognize the words and text lines. Recently, composite script identification and orientation detection for Indian text images is proposed by Ghosh and Chaudhuri [18]. This method considers eleven scripts for identification purpose. The features presented in this work are derived from the connected component analysis. These features are

978-0-7695-4661-2/12 $26.00 © 2012 IEEE
DOI 10.1109/DAS.2012.57

good only if the connected components preserve their shapes. Based on literature review on both camera and video documents, it is observed that most of the papers used Roman and Devanagiri as the common scripts and a few papers consider other scripts for identification purpose [14]. In addition, the main focus of these methods is that the identification of scripts in documents with plain background and high contrast but not the scripts in video. To the best of our knowledge, none of the papers addressed the problem of script identification in video, in particular the identification of Arabic, Chinese, English, Japanese, Korean and Tamil.

Therefore, this paper presents new features at the block level based on spatial-gradient information for six script identification in video. The main advantage of this method is that it is insensitive to segmentation of words as most of the existing methods in document analysis require words for script identification.

II. PROPOSED METHOD

Since several sophisticated methods for text block detection and text region detection in video frame are available in literature, we use one of our developed methods for text frame classification at the block level for identifying text block in video [19]. This method divides the whole video frame of size 256×256 into 16 blocks of size 64×64 and it analyses each block based on wavelet-moments and mutual nearest neighbor concept to identify the text blocks among 16 blocks. The reason for choosing block size 64×64 is to have few words in a block, to make implementation simpler and to speed up the computation process. This is because generally text does not occupy the full video frame, instead, it scatters in the frame as small clusters. The advantage of this method is that it identifies the text block irrespective of scripts, fonts, font size and orientation. Therefore, this method provides text block of six scripts and is considered as input for script identification in this work. More details may be found in [19]. Since our aim is to identify the text frame of different scripts, the method analyses the blocks given by the text block detection method [19] and it expects at least one block to satisfy the criteria to identify the frame of particular script. Therefore, the scope of this work is limited to text frame containing single script but not the text frame containing multi-script.

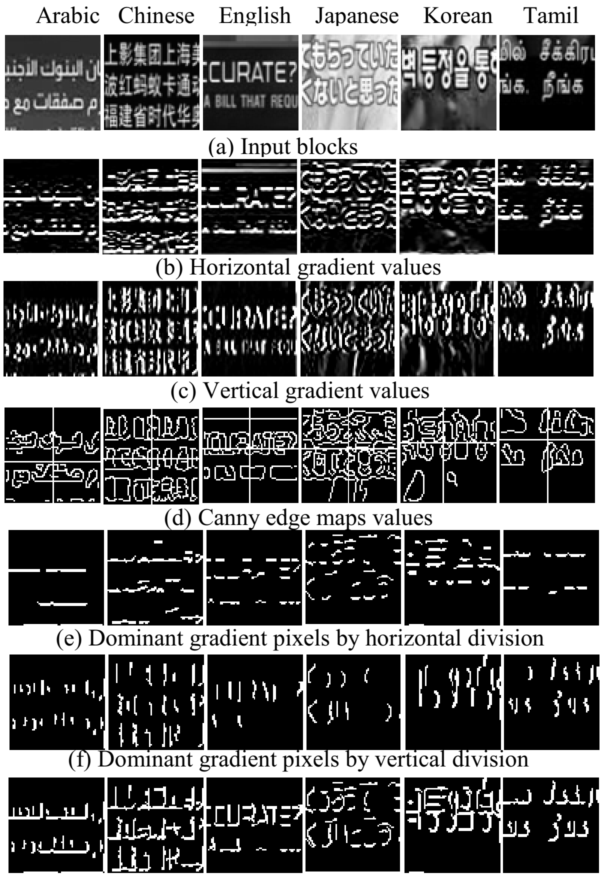
The proposed method consists of four sub-sections. Section A introduces a method based on gradient histogram for obtaining dominant text components. Finding candidate text components based on skeleton and filtering is presented in section B. Section C proposes new features based on distance between end points, junction points, intersection points and pixels for classification of scripts. In section D, we explain how to create a template for identifying scripts.

A. *Text Components based on Gradient Histogram Method*

For each pixel in the block as shown in Figure 1(a), the method obtains gradient values by convolving Sobel horizontal mask and vertical mask shown in equation (1) over the block to increase the contrast as shown in Figure 1(b) and (c), respectively. It is noted from literature that the gradient information is useful for video text pixel classification as it gives high gradient values for text pixels and low gradient values for non-text pixels [6]. In order to select dominant text pixels which usually will have high gradient values, the method divides the horizontal gradient block at the centroid horizontally, which results in two equal parts namely the upper part and the lower part. The centroid is computed based on edge pixels in Canny edge map of input block as shown in Figure 1(d). For the upper and the lower part of gradient block, we plot a histogram to find gradient values which give the highest peak in the histogram and these values are considered as dominant values of text pixels. The dominant text pixels are represented as white pixels and other gradient values are represented as black pixels as shown in Figure 1(e). In the same way, the method divides the vertical gradient block vertically at the centroid to obtain the left part and the right part of the gradient block. The same histogram criterion is used to select dominant pixels from both the right and the left parts of the gradient block as shown in Figure 1(f). Then we combine the dominant pixels obtained from the above horizontal and vertical divisions to obtain the total text information as shown in Figure 1(g). The flow diagram for the selection of text components from horizontal and vertical gradient division is shown in Figure 2.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I, \qquad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I \qquad (1)$$

Where G_x and G_y represents gradient in the horizontal and vertical direction respectively and I is the image and $*$ represents 2 dimensional convolution operation.



(a) Input blocks

(b) Horizontal gradient values

(c) Vertical gradient values

(d) Canny edge maps values

(e) Dominant gradient pixels by horizontal division

(f) Dominant gradient pixels by vertical division

(g) Text components (horizontal + vertical dominant pixels)

Figure 1. Intermediate results for text components

B. *Candidate Text Components Selection*

To reduce the pixel width of the components in the results given by the previous section, we use skeleton criteria to get single pixel width components and to preserve the shape of the components as shown in Figure 3(a) as it helps in identifying end points, junction and intersection points of the components accurately. The method computes area (number of pixels in the component) of the components and uses the area as feature to eliminate unwanted components. For this purpose, the method uses k-means clustering algorithm with k=2 instead of thresholding. Then the method chooses the cluster which gives low mean compared to mean of other cluster to eliminate unwanted components such as small components as defined in equation (2). This results in candidate text components as shown in Figure 3(b). With these candidate text components, we extract features which will be discussed in the next section.

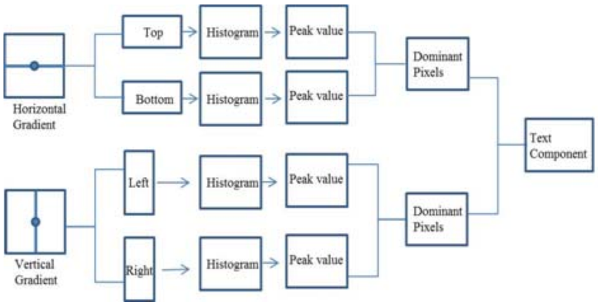
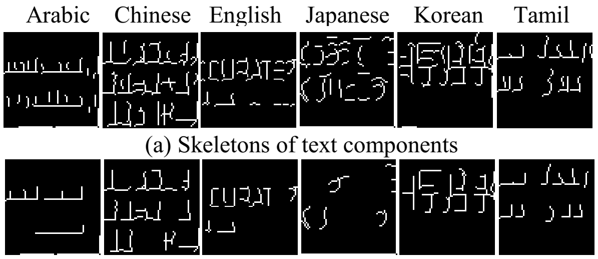


Figure 2. Flow diagram for text components selection using gradient histogram.



(a) Skeletons of text components

(b) Candidate text components after filtering

Figure 3. Candidate text components

$$\mu_{NC} = \min(\mu_{C1}, \mu_{C2}) \quad (2)$$

where $\{\mu_{C1}, \mu_{C2}\}$ are the means of the two clusters and μ_{NC} is the mean of the noise cluster.

C. *Features based on Spatial information*

It is observed from the candidate text components in the block of six scripts that the spatial distribution of end points, intersection points, junction points and pixels exhibit distinctive appearances among the different scripts. For instance, one can notice from the candidate text components of English and Chinese that the end points in Chinese have close proximity, while in English, proximity between end points is not as close as Chinese. This is because components in Chinese contain many sub components, hence more end points with close proximity, while component in English have single components with few end points. Similarly, intersection points have close proximity for Korean script, thus there are more intersection points in the case of Korean script due to its cursive nature. In the case of Arabic script, the intersection points are not as close in proximity as in Korean script due to its less cursiveness and less number of intersection points. To extract such observation we compute four variance features with respect to end points, intersection points, junction points and pixels. The method finds distances from each end point to the remaining end points, thereby generating a proximity matrix for all end points. The same is done for intersection points, junction points and pixels.Then a single vector comprising the four variance features is formed. The end points, junction and intersection points are defined as follows.

For any pixel P with surrounding pixels $P_{surround}$ in a component with pixels $P_{componen}$:

$$P_{surround} \wedge P_{component} = k \quad (3)$$

P is an end point when k = 1 in equation (3). P is a junction point when k = 3 in equation (3) and P is an intersection point when k = 4 in equation (3).

The proximity matrices for end points, junction points, intersection points and all pixels are defined as follows. The proximity matrix of end points is represented by $ED_{(i,j)}$ where E is the set of all end points and E' is E transpose.

$$ED_{(i,j)} = \sum_{r=1}^{n} \sqrt[2]{{E_{i,r}}^2 + {E'_{r,j}}^2} \quad (4)$$

The proximity matrices of junction points, intersection points and all pixels are represented by $JD_{(i,j)}$, $ID_{(i,j)}$ and $PD_{(i,j)}$, respectively are computed as in equation (4). In the same way, the variances for proximity matrices are computed as follows. The variance of proximity of end point distances is represented by $Var(ED)$ where ED is the set of all end point distances and μ_{ED} is the mean of the end point distances.

$$Var(ED) = \frac{1}{n}\sum_{i=1}^{n}(ED_i - \mu_{ED})^2 \quad (5)$$

The variance of junction point distances, intersection point distances and all pixel distances represented by $Var(JD)$, $Var(ID)$ and $Var(PD)$, respectively are computed as in the equation (5).

D. *Template Formation for Script Identification*

We select 50 frames from each class of scripts for template creation. The idea of creating templates for script identification is inspired by the work presented in [11] in which script identification for camera images is addressed. The four variance features are computed for the blocks corresponding to 50 frames of each script class. Then the average of the variance features of the blocks of 50 frames for each script is computed as defined in equation (6) below.

$$Avg(Var) = \frac{1}{50}\sum_{i=1}^{50} Var_i \quad (6)$$

This gives six templates (vectors) for the six scripts containing four average variance features in each template. For the given block, the method extracts four variance features and compares them with the six templates to find the minimum Euclidean distance to classify the frame into a particular class. This procedure gives a confusion matrix for the six scripts. The sample templates for the scripts Arabic, Chinese, English, Japanese, Korean and Tamil can be seen respectively in Figure 4(a)-(f) where all the six templates have distinct features. This shows that generated templates are good enough to classify the six scripts with good classification rate.

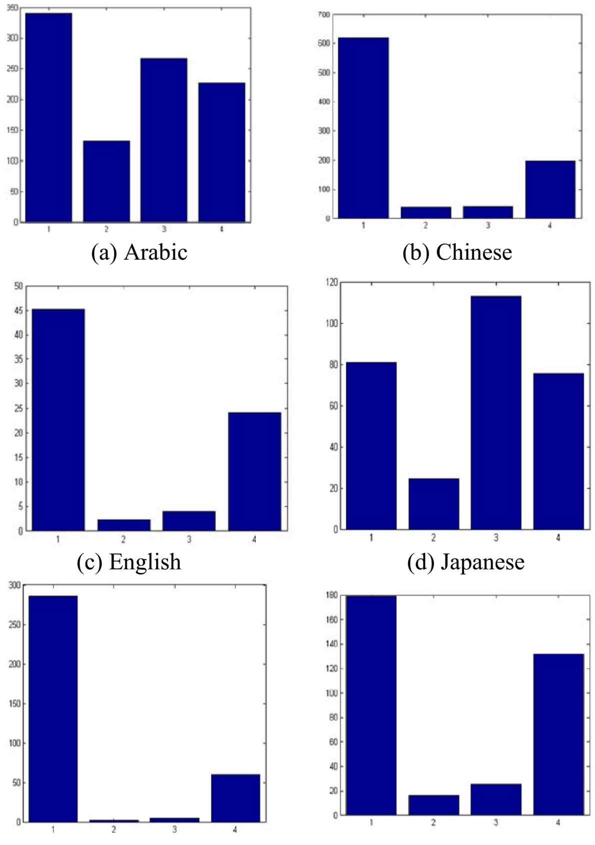
The Euclidean distance for classification is defined as follows. For a text block with extracted feature vector FV, the set of Euclidean distance ED = {ED_1, ED_2 … ED_6} between FV and the set of template vector T = {T_1, T_2 … T_6} is given by,

$$ED_i = \sqrt[2]{\sum_{r=1}^{d}(FV_r - T_{i_r})^2} \qquad (7)$$

Where d is the dimension, in this case, d = 4 since there are 4 features.

The classified script is given by $S \in \{1, 2 \dots 6\}$ where

$$ED_S = \min(ED)$$



(a) Arabic (b) Chinese

(c) English (d) Japanese

(e) Korean (f) Tamil

Figure 4. Six templates for script identification

III. Experimental Results

Since there is no standard dataset or benchmark dataset available publicly, we create our own dataset selected from different sources such as sports news, weather news, entertainment video etc to show that the proposed method works well for different varieties of video frames. Our dataset include 100 Arabic frames, 100 Chinese frames, 260 English frames, 100 Japanese frames, 100 Korean frames and 100 Tamil frames. In total, 770 frames are used for experimentation purpose. To evaluate the performance of the method, we consider classification rate as a measure and we present a confusion matrix containing classification rate/misclassification rate for each script. In subsequent sections, we present analysis of the individual features, all four features together and a comparative study with an existing method.

A. *Experiments on Individual Features*

The aim of this experiment is to find contribution of each feature in terms of average classification rate (average of diagonal elements of confusion matrices of each feature). Therefore, Figure 5 shows the average classification rate for the four features. It is noticed from Figure 5 that feature-1 and feature-4 contribute more than feature-2 and feature-4 for classification because proximity between end points (feature-1) and all pixels (feature-4) give distinct features for the six scripts. In the same way, proximity between junction points (feature-2) and intersection points (feature-3) contributes less compared to feature-1 and feature-4. Therefore, we combine these four features to get better results for classification of the six scripts.

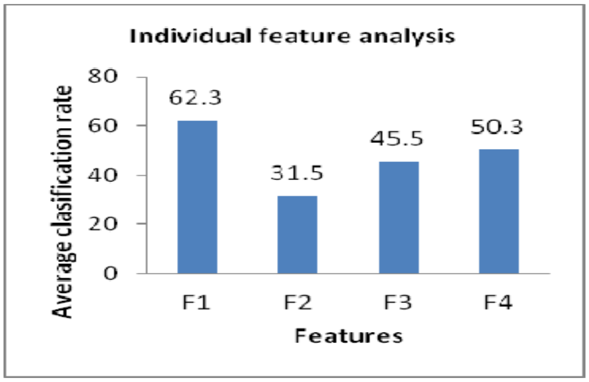
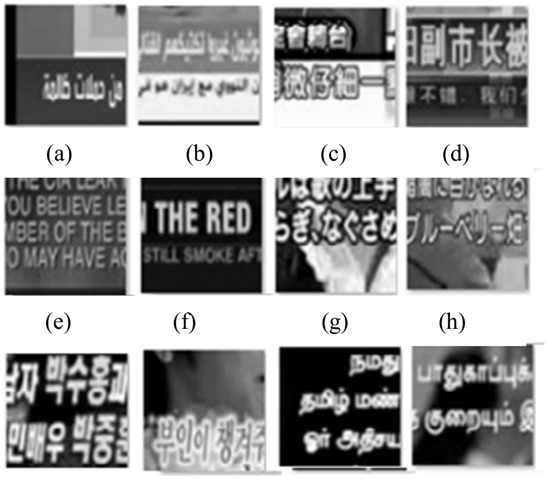


Figure 5. Average classification rate for each feature



(a) (b) (c) (d)

(e) (f) (g) (h)

(i). (j) (k) (l)

Figure 6. Sample blocks for six scripts from our database

B. *Experiments on four Variance Features*

Sample blocks of the six scripts are shown in Figure 6 where Figure 6(a) and (b), (c) and (d), (e) and (f), (g) and (h), (i) and (j), (k) and (l) show Arabic, Chinese, English, Japanese, Korea and Tamil script, respectively. One can notice from Figure 6 that we have considered varieties of blocks including complex background, low resolution text, different fonts, fonts size etc to show that the proposed method work well for different situations. The results for the four features for classification of the six scripts is reported in Table 1 where classification rate is good for Chinese, English, Japanese, Korean but is not as good for Arabic and Tamil. This is because the extracted features are confused with English scripts as the prominent features such as horizontal, vertical and cursive edge information of English script at text components level may share the vertical, horizontal features of Arabic and cursive features of Tamil. Despite low classification rate for Arabic and Tamil, we achieve 82.1% average classification rate (average of diagonal elements in Table 1).

Table 1. Confusion matrix of the proposed method (in %)

Scripts	Arabic	Chinese	English	Japanese	Korean	Tamil
Arabic	**66**	3	1	4	0	4
Chinese	7	**94**	1	3	1	4
English	11	1	**96**	7	5	17
Japanese	1	0	0	**83**	2	4
Korean	7	0	1	2	**90**	7
Tamil	4	2	1	1	2	**64**

C. Comparative Study

We have found one paper on video script identification based on text lines [1], which uses descriptive features and classifiers. Since this work consider only two scripts such as English and Chinese, our method also consider the two scripts for comparative study in this work. The existing method considers text lines as input for script identification while the proposed method considers block as input, therefore we run the existing method on blocks as our method does to identify the scripts at the block level. The classification rate for both the proposed method and existing method on English and Chinese is reported in Table 2 where the proposed method gives better classification rate compared to the existing method. The reason for poorer results for the existing method is that the extracted features are not good enough to handle broken segments and touching between adjacent components as these features expects some regularity of text pattern in each zone. The proposed method is capable of overcoming these problems because the dominant pixel selection and their spatial study preserve the uniqueness of scripts in spite of broken segments and touching caused by low resolution and complex background. In addition, the existing method is sensitive to the classifier and samples. On top of this, our method work well for six scripts. Thus the proposed method is superior to existing method in terms of classification rate and number of scripts.

Table 2. Performance of the proposed and existing methods at the block level (in %)

Proposed method			Gllavata and Freisleben [1]		
Scripts	Confusion matrix		Confusion matrix		
	English	Chinese	Scripts	English	Chinese
English	97	3	English	60	40
Chinese	9	91	Chinese	44	56

IV. CONCLUSION AND FUTURE WORK

We have proposed new spatial-gradient-features for identifying six scripts. The dominant text pixel selection is done based on the histograms of horizontal gradient and vertical gradient. The four variance features with respect to distances between end points, intersection points, junction points and pixel in the block are proposed for script identification. The experimental results show that the four features are sufficient to identify the six scripts. The performance of the proposed method is compared with an existing method at the block level to show that the proposed method is superior to the existing method. We are planning to extend this method for more script identification at the word level in future.

ACKNOWLEDGMENT

This research is supported in part by A*STAR grant 092 101 0051 (WBS no. R252-000-402-305).

REFERENCES

[1] J. Gllavata and B. Freisleben, “Script Recognition in Images with Complex Backgrounds”, In Proc. IEEE International Symposium on Signal Processing and Information Technology, 2005, pp 589-594.

[2] T. Q. Phan, P. Shivakumara, Z. Ding, S. Lu and C. L. Tan, “Video Script Identification based on Text Lines”, In Proc. ICDAR, 2011, pp 1240-1244.

[3] D. Doermann, J. Liang and H. Li, “Progress in Camera-Based Document Image Analysis”, In Proc. ICDAR, 2003, pp 606-616.

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

[5] K. Jung, K.I. Kim and A.K. Jain, “Text information extraction in images and video: a survey”, Pattern Recognition, 2004, pp. 977-997.

[6] P. Shivakumara, T, Q. Phan and C. L. Tan, “A Laplacian Approach to Multi-Oriented Text Detection in Video”, IEEE Transactions on PAMI, 2011, pp 412-419.

[7] D. Chen and J. M. Odobez, “Video text recognition using sequential Monte Carlo and error voting methods”, Pattern Recognition Letters, 2005, pp 1386-1403.

[8] D. Ghosh, T. Dube and A. P. Shivaprasad, “Script Recognition-Rview”, IEEE Ttansactios on PAMI, 2010, pp 2142-2161.

[9] T .N. Tan, “Rotation Invariant Texture Features and Their Use in Automatic Script Identification”, IEEE Transactions on PAMI, 1998, pp 751-756.

[10] A. Busch, W. W. Boles and S. Sridharan, “Texture for Script Identification”, IEEE Transactions on PAMI, 2005, pp 1720-1732.

[11] L. Shijian and C. L. Tan, “Script and Language Identification in Noisy and Degraded Document Images”, IEEE Transaction on PAMI, 2008, pp 14-24.

[12] S. Jaeger, H. Ma and D. Doermann, “Identifying Script on Word-Level with Informational Confedence”, In Proc. ICDAR, 2005, pp 416-420.

[13] P. B. Pati and A. G. Ramakrishnan, “Word level multi-script identification”, Pattern Recognition Letters, 2008, pp 1218-1229.

[14] S. Chanda, S. Pal, K. Franke and U. Pal, “Two-stage Apporach for Word-wise Script Identification”, In Proc. ICDAR, 2009, pp 926-930.

[15] S. Chanda, O. R. Terrades and U. Pal, “SVM Based Scheme for Thai and English Script Identification”, In Proc. ICDAR, 2007, pp 551-555

[16] L. Li and C. L. Tan, “Script Identification of Camera-based Images”, In Proc. ICPR, 2008.

[17] A. M. Namboodiri and A. K. Jain, “On-line Script Recognition”, In Proc. ICPR, 2002, pp 736-739.

[18] S. Ghosh and B. B. Chaudhuri, “Composite Script Identification and Orientation Detection for Indian Text Images”, In Proc. ICDAR, 2011, pp 294-298.

[19] P. Shivakumara, A. Dutta, Trung Quy Phan, C. L. Tan and U. Pal, “A Novel Mutual Nearest Neighbor based Symmetry for Text Frame Classification in Video”, Pattern Recognition, 2011, pp 1671-1683.