International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622
NATIONAL CONFERENCE on Developments, Advances & Trends in Engineering Sciences (NCDATES- 09th & 10th January 2015)

RESEARCH ARTICLE OPEN ACCESS

An Extraction of overlay text from Digital Videos

[1]Vani Kasireddy, [2]Swathi.G, [3]Santhosh Kumar
[1]Department Of Electronics And Communication Engineering *CMR Engineering College, Hyderabad.*
[2]*Department Of Electronics And Communication Engineering* CMR Engineering College, Hyderabad.
[3]*Department Of Electronics And Communication Engineering AVANTHI Engineering College, HYDERABAD.*

Abstract:
With the increasing popularity of the World Wide Web (WWW) and the introduction of streamed digital audio and video on the Internet, the amount of multimedia information available to consumers continues to grow. As content becomes readily available, automatic indexing during archiving and searching large volumes of multimedia data will become difficult. An important and integral part of video that contains high-level semantic information is overlay text and scene text. Overlay text brings important semantic clues in video content analysis such as video information retrieval and summarization, since the content of the scene or the editor's intention can be well represented by using inserted text. Most of the previous approaches to extract overlay text from videos are based on low-level features. However, existing methods experience difficulties in handling texts with various contrasts or inserted in a complex background. The framework proposed in this is used to detect and extract the overlay text from the video scene. Based on the observation that there exist a transient color between the inserted text and its adjacent background, a transition map is generated first and Linked maps are generated to make connected components of candidate region. Then candidate regions are extracted by a reshaping method and the overlay text regions are determined based on the occurrence of overlay text in each candidate. The detected overlay text regions are localized accurately using the projection of overlay text pixels in the transition map and the text extraction is finally conducted. This method is robust to different character size, position, contrast, and color. It is also language independent. Overlay text region update between frames is also employed to reduce the processing time.

Keywords— Optical character recognition (OCR) overlay text, transition map, video information retrieval, and video summarization.

I. Introduction

In the development of video editing technology, there are growing uses of overlay text inserted into video contents to provide viewers with better visual understanding. Most broadcasting videos tend to increase the use of overlay text to convey more direct summary of semantics and deliver better viewing experience. The present-day development of various multimedia compression standards combined with a significant increase in desktop computer performance, and a decrease in the cost of storage media, has led to the widespread exchange of multimedia information. To enable users to quickly locate their interested content in an enormous quantity of video data, many research efforts have been put into video indexing and summarization. Textual, visual and audio information is most frequently used for this purpose. Among them, text in video, especially the overlay text, is the most reliable clue for three reasons:

1) It is closely related to the current content of video.

2) It has distinctive visual characteristic and

3) The state-of-art optical character recognition (OCR) techniques are far more robust than the existing speech analysis techniques and visual object analysis.

Therefore, almost all video indexing research work begins with text recognition. Text in video (especially the overlay text) is one powerful source of high-level semantics. If these text occurrences could be detected, segmented, and recognized automatically, they would be a valuable source of high-level semantics for indexing and retrieval.

A keyword based video indexing system illustrates in Fig 1. Text in videos is extracted as keywords and stores in the database in an off-line fashion. User browses the video by inputting keyword, and then video clips which contain keywords are retrieved for the users. Keyword based indexing system is especially useful in news program and sports broadcast because the overlay text is highly correlated with the video content. For this purpose, to detect and extract text in video effectively and accurately is the key technique to keyword based video indexing system.

Unfortunately, it is not an easy problem to reliably detect and localize text embedded in videos. In a single frame, the size of characters can change from very small to very big. The font of text can be different and can have multiple colors. Text can also occur in a much cluttered background. Furthermore, text can be either still or moving in an arbitrary direction. The background can also be moving or changing, independent of the text. In response to such difficulties, we try to find out an efficient and robust text detection and extraction framework to solve the problems.

Besides, the MPEG-7 standard, a novel definition for content-based multimedia description and retrieval, provides a videotext descriptor to represent the media content. To this end, it is for us to develop a novel framework to detect and extract the overlay text from the video scene.

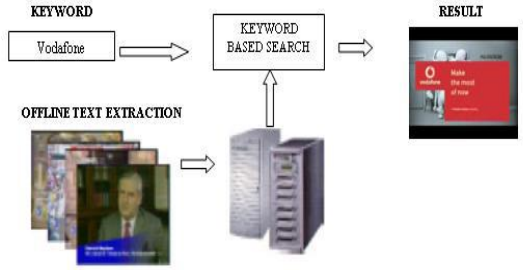


Fig 1. Keyword based indexing using automatic extracted text.

II. Related Work

Text can appear in video anywhere in the frame and in different contexts. It appears as either scene text or as overlay text. Text that appears as part of the scene and recorded with the scene is referred to as scene text and its presence can be in the scene as part of street and shop name broads, or on a person's clothing. It is difficult to extract scene text reliably due to the unconstrained nature of its appearance. On the other hand, overlay text is intended to carry and stress important information in video. It is typically generated by video title machines or graphical font generators in studios. Therefore, most of the researches are concentrated on the overlay text detection, localization and extraction. Video text recognition is generally divided into four steps:

1. Detection.
2. Localization.
3. Extraction.
4. Recognition.

- The detection step roughly classifies text regions and non-text regions.
- The localization step determines the accurate boundaries of the text strings.
- The extraction step filters out background pixels in the text strings, so that only the text pixels are left for the recognition.
- Since the above three steps generate a binary text image, the recognition step can be done by commercial document OCR software.

A. Text Detection Method

Videotext detection is the first step toward automatic videotext recognition. The goal of videotext detection is to classify the text region and non-text region. The videotext detection is based on the special characteristics of the text. In general, the special characteristics are Contrast, Color, Font, Size, Font Shape, Orientation, and Stationary Location.

Based on the analysis, videotext detection methods can be classified into three classes. The first class treats text as a type of texture, the second class assumes that a text string contains a uniform color. The third method depends on the edge information of the text.

1) Texture Based Method:

For texture based method, it assumes that the video text has similar frequency and orientation. Therefore, videotext can treats as a special type of texture. These methods usually divide a whole image into blocks. They first use various approaches, e.g., Gabor filter, spatial variance, or wavelet transform, to calculate the texture features of blocks. Then they employ proper classifiers, e.g., neural network or a support vector machine, to classify text and non-text blocks. Li *et al*. [4] proposed to utilize the wavelet transform to extract the features in three sub-bands (LH: Horizontal High Frequency, HL: Vertical High Frequency, HH: Horizontal & Vertical High frequency). After extracting and selecting the features, it trains a neural network to eliminate the false text region. With the phase of neural network training, this method not only deals with simple background but also complex background. Indeed, it can detect videotext in a blur scene (the contrast of the videotext is not high enough against background). The drawback of this method is that the training cost very high. It is hardy to reach real-time requirement.

2) Color Based Method:

The method based on the assumption that videotext contains a uniform color. Hence, it only detects the videotext in a simple background, e.g., the background is back or white. Based on this method, it is difficult to detect videotext correctly in a complex background. Thus, at first they perform color reduction and segmentation in some selected

color channel or color space, and then they perform connected-component analysis to detect text regions. Jain *et al*. [1] proposed a detection method to deal with color image. The aim is to calculate the similarity of different color values, i.e., in RGB color space, the range of R, G, B are from 0 to 255, total 256 different color values. Therefore, color reduction is needed to raise the speed of similarity calculation. Jain adopted bits reduction and color quantization used to perform the color reduction. Originally, it needs 8 bits to represent the R, G, B component of a pixel. It truncates lower 6 bits so that the higher 2 bits remain. Only 6 bits left for similarity measurement. It greatly reduces the color values of the color space which is form 224 to 26. Then, they apply a histogram method to quantize the color, in other words, it merge the region of similar color to find out the text region.

3) ***Edge Based Method:*** According to the stroke density and contrast characteristic of text string, it allows us to manipulate the edge detection method to detect text region in image or video frames. By the method, it generates an edge map first and utilizes the merge methods to connect text character to form meaningful text string. These methods include connected-component analysis and morphological operations. This method can detect overlay videotext with simple or complex background. To the scene videotext, it only detects high contrast videotext with simple background. Besides, this method may detect small object as text character because it also has the stroke density and contrast characteristics. The edge based text detection method can classify into two categories, one is in compressed domain and the other is in pixel domain. In compressed domain, it utilizes the DCT coefficients to measure the horizontal and vertical intensity variation of each DCT block in an I-frame. In pixel domain, it employs Canny, Sobel or Gaussian filter to perform edge detection. In the flowing section, we give a brief introduction of this method in two categories. Lyu [2] use a modified edge map with strength for text region detection and localize the detected text regions using coarse-to-fine projection. They also extract text strings based on local thresholding and inward filling. The authors consider the strokes of text in horizontal, vertical, up-right, and up-left directions and generate the edge map along each direction. Then they combine statistical features and use k-means clustering to classify the image pixels into background and text candidates.

B. Text Localization Method

Most text recognition researches focus on static text recognition, but text regions are not always static. Nowadays, there are a lot of motion texts provide us useful information in general TV programs, especially in news program. In general, text motion can divide into three classes. There are static, simple linear motion (for example, scrolling movie credits) and complex nonlinear motion (for example, zooming in/out, rotation, or free movement of scene text), respectively. Generally, there are three approaches for text localization. The first approach assumes that the text is static, so it can only deal with static text. The second approach is based on the text pixel value; it can deal with three type of text motion. The third approach applies the vertical/horizontal projection profile to localize the text, but it is not robust to deal with complex nonlinear text motion. The details are described in the following.

1) First Approach: Chirstian Wolf [3] assumed that text is static, once the text region is detected; it localized the text region in the same place in successive frames to reduce the text detection computation. The text region detection process is based on the maximum successive frames which a single text may stand to avoid the false localization.

2) Second Approach: Huiping Li [4] proposed a robust method to localize both static text and motion text (simple linear motion and complex nonlinear). The detection process still determined by the maximum successive frames of which a single stand. Li proposed two methods to localize text in successive frames to reduce text detection computation. One is SSD (Sum of Squared Difference)–Based Module Based Image Matching, the other is Contour-Based Text Stabilization. SSD–Based Module Based Image Matching can deal with static text or simple linear motion text and Contour-Based Text Stabilization deal with complex nonlinear motion text.

3) Third Approach: Rainer Lienhart [5] proposed a projection profile method to localize text block in images. A project profile of an image region is a compact representation of the spatial pixel content distribution and has been successful employed in document text segmentation. While histograms only capture the frequency distribution of some image feature such as the pixel intensity (all spatial information is lost), intensity projection profiles preserve the rough spatial distribution at the cost of an even higher aggregation of the pixel content. Vertical and horizontal projection profiles are depicted as bar charts along the x and y axes of the feature image. Based on projection profile method, it can localize static text or simple motion text block. As to complex nonlinear motion text, it only localizes parts of the whole text block. To avoid

this circumstance, Lienhart proposed to detect the entire image every four frames.

C. Text Extraction Method

After text detection and localization process, the text region can be localized well. But it is still not suitable for recognition by OCR software because of the embedded complex background. To this end, a binarization process is needed to solve this problem. It segments the text character from the background. The text extraction methods can divide into two groups. One group includes color-based methods and the other includes stroke-based methods. The former holds the assumption that the text pixels are of different color of from the background pixels, so that they can be segmented by thresholding.

The stroke-based methods, on the other hand, employ some filters to output only those pixels likely on the strokes to the final results, such as the asymmetric filter, the four-direction character extraction filter, and the topographical feature mask. These filters are intended to enhance the stripe (i.e. stroke-like) shapes and to suppress others. Due to the small resolution and noise occurring in the frames, to enhance the contrast of text against the background is necessary. There are several images enhancement methods, such as multiple frame integration and bi-linear interpolation. The color based method holds the assumption that the text pixels are of different color from the background pixels, so they can be segmented by thresholding. Difficulty in this approach is the color polarity of text (i.e., light or dark), must be determined. Antani [6] proposed a color-based method that generates 2 segmented results in both polarities for text string and then selects the one with a higher score on text-like characteristics as the final result. In this paper, we propose a new overlay text detection and extraction method using the transition region between the overlay text and background. First, we generate the transition map based on our observation that there exist transient colors between overlay text and its adjacent background. Then the overlay text regions are roughly detected by computing the density of transition pixels and the consistency of texture around the transition pixels. The detected overlay text regions are localized accurately using the projection of transition map with an improved color-based thresholding method to extract text strings correctly.

III. Proposed Work

The main objective of the proposed method is to detect and extract the overlay text from the video frames. Fig 2 shows the Overall architecture for the proposed framework.

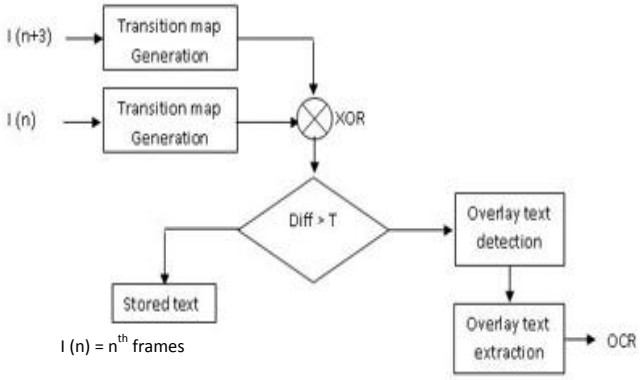


Fig 2. Overall Architecture

A. Transition Map Generation

On the contrary, the overlay text tends to be dark if the background of overlay text is bright. Therefore, there exists a transient colors between overlay text and its adjacent background due to color bleeding, the intensities at the boundary of overlay text are observed to have the logarithmical change. Basically the intensities of three consecutive pixels are decreasing logarithmically at the boundary of bright overlay text due to color bleeding by the lossy video compression. It is also observed that the intensities of three consecutive pixels increases exponentially at the boundary of dark overlay text. The change of intensity at the boundary of overlay text may be small in the low contrast image, to effectively determine whether a pixel is within a transition region, the modified saturation is first introduced as a weight value based on the fact that overlay text is in the form of overlay graphics. The value of modified saturation $S(x,y)$ is defined as follows:

$$S(x,y) = 1 - \frac{3}{(R+G-B)}[\min(R,G,B)]$$

$$\tilde{S}(x,y) = \frac{S(x,y)}{\max\,(S(x,y))}$$

$$Where, \max\big(S(x,y)\big) = \begin{cases} 2\times\big(0.5-\tilde{I}(x,y)\big), & if\ \tilde{I}(x,y) > 0.5 \\ 2\times\tilde{I}(x,y), & otherwise. \end{cases}$$

$S(x,y)$ And $\max\,(S(x,y))$ denote saturation value and maximum saturation value at the corresponding intensity level respectively. $\tilde{I}(x,y)$, denotes the intensity at (x,y). The transition can thus be defined by combination of the change of intensity and the modified saturation as follows:

$$T(x,y) = \begin{cases} 1, & if\ D_H > D_L + TH \\ 0, & otherwise \end{cases}$$

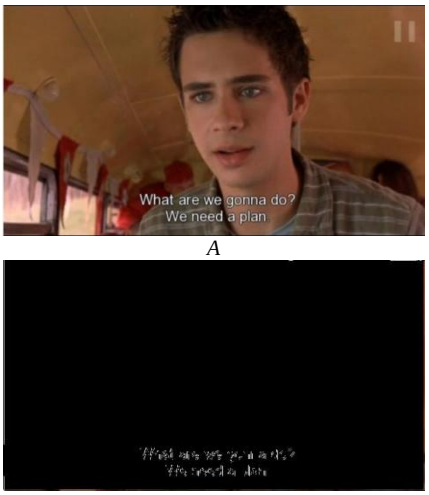
Where,

$$D_L(x,y) = \left(1 + dS_L(x,y)\right) \times |\, I(x-1,y) - I(x,y)$$
$$D_H(x,y) = \left(1 + dS_H(x,y)\right) \times |\, I(x,y) - I(x+1,y)$$

Where,

$$dS_L(x,y) = |\, \tilde{S}(x-1,y) - \tilde{S}(x,y)$$
$$dS_H(x,y) = |\, \tilde{S}(x,y) - \tilde{S}(x+1,y).$$

The thresholding value *TH* is empirically set to 0.2 or 0.3 (based on our observation) in consideration of the logarithmical change.



A

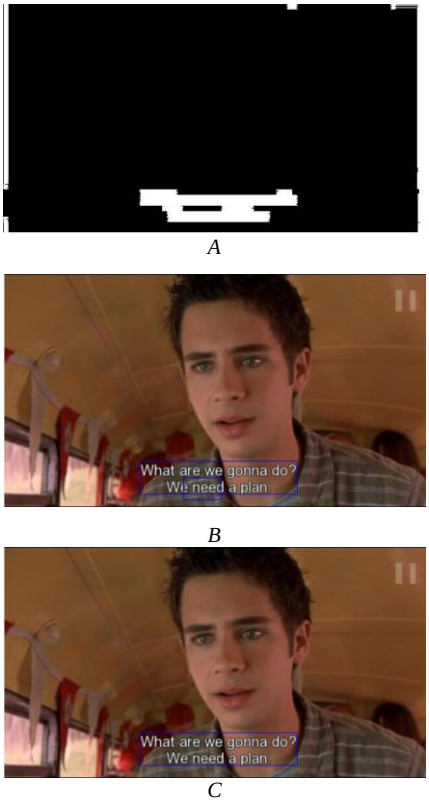
B

Fig 3. A) Original Video Frame. B) Transition Image

B. Overlay Text Detection

The transition map can be utilized as a useful indicator for the overlay text region. To generate the connected components, a linked map is generated using the condition, if a gap of consecutive pixels between two nonzero points in the same row is shorter than 5% of the image width, they are filled with 1s. If the connected components are smaller than the threshold value, they are removed. The threshold value is empirically selected by observing the minimum size of overlay text region. Then each connected component is reshaped to have smooth boundaries. Since it is reasonable to assume that the overlay text regions are generally in rectangular shapes, a rectangular bounding box is generated by linking four edge points.

1) Overlay text determination: The next step is to determine the real overlay text region among the boundary smoothed candidate regions by some useful clues, such as the aspect ratio of overlay text region. Since most of overlay texts are placed horizontally in the video, the vertically longer candidates can be easily eliminated. The density of transition pixels is a good criterion as well. Nevertheless, a more refined algorithm is needed to minimize the false detection due to the complex background. In this subsection, we introduce a texture-based approach for overlay text region determination.



A

B

C

Fig 4. A) Linked map. B) Image with false detection region. C) Overlay text determined image without false detection.

The intensity variation around the transition pixel is big due to complex structure of the overlay text, we employ the local binary pattern (LBP) introduced in to describe the texture around the transition pixel. LBP is a very efficient and simple tool to represent the consistency of texture using only the intensity pattern. LBP forms the binary pattern using current pixel and its all circular neighbor pixels and can be converted into a decimal number.

2) Overlay Text Region Refinement: The overlay text region obtained in the preceding subsection needs to be refined for better accurate text extraction, which will be addressed in Section C. In this subsection, we use a modified projection of transition pixels in the transition map [2] to perform the overlay text region refinement. First, the horizontal projection is performed to accumulate all the transition pixel counts in each row of the detected overlay text region to form a histogram of the number of transition pixels. Then the null points, which denote the pixel row without transition pixels, are removed and separated regions are re-labeled. The projection is conducted vertically and null points are removed once again.

C. Overlay Text Extraction

Before applying video OCR application, the refined overlay text regions need to be converted to a binary image, where all pixels belonging to overlay text are highlighted and others suppressed. Since the text color may be either brighter or darker than the background color, an efficient scheme is required to extract the overlay text dealing with complex backgrounds and various text appearances.

1) Color Polarity Computation: Sometimes the overlay text is darker than the surrounding background or the text is brighter than its neighbors. The binarized images obtained by simple thresholding represent the overlay text as either 1 (or "White") or 0 (or "*Black*"), respectively. The main goal in this subsection is to check the color polarity and inverse the pixel intensities if needed so that the output text region of the module can always contain bright text compared to its surrounding pixels. The binarized text region, the boundary pixels, which belong to left, right, top, and bottom lines of the text region, are searched and the number of white pixels is counted. If the number of white boundary pixels is less than 50% of the number of boundary pixels, the text region is regarded as "bright text on dark background" scenario, which requires no polarity change. In other words, the overlay text is always bright in such scenarios. If the number of white pixels is greater than that of black pixels, we conduct a task to turn on or off the "bright_text_flag".

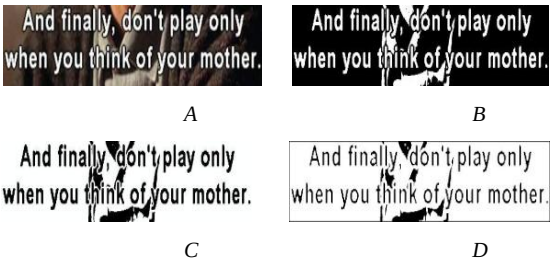


Fig 5 A) Determined text region. B) Binarized text region. C) Inverted Region. D) Extracted text.

2) Overlay text extraction: First, each overlay text region is expanded wider by two pixels to utilize the continuity of background. This expanded outer region is denoted as *ER*. Then, the pixels inside the text region are compared to the pixels in *ER* so that pixels connected to the expanded region can be excluded. We denote the text region as *TR* and the expanded text region as *ETR*. Next, sliding-window based adaptive thresholding is performed in the horizontal and the vertical directions with different window sizes, respectively. Finally, we can obtain characters correctly from each overlay text region by the inward filling. If a pixel is "*White*" during the scanning of binarized pixels in *ER*, all the connected "*White*" pixels including the pixel itself are filled with "*Black*". After the inward filling, all non-"*Black*" pixels are set to be "*White*". Now we can see that the background of text is well removed.

Conclusion

A novel method for overlay text detection and extraction from complex videos is proposed in this paper and the detection method is based on the observation that there exists a transient colors between inserted text and its adjacent background. The transition map is first generated based on logarithmical change of intensity and modified saturation. Linked maps are generated to make connected components for each candidate region. We compute the density of transition pixels and the consistency of texture around the transition pixels to distinguish the overlay text regions from other candidate regions. The local binary pattern is used for the intensity variation around the transition pixel in the proposed method. The boundaries of the detected overlay text regions are localized accurately using the projection of overlay text pixels in the transition map. Overlay text region update between frames is also exploited to reduce the processing time. Based on the results of overlay text detection, the overlay texts are extracted based on Lyu's extraction method. We add a simple constraint based on detected transition pixels in the transition map to improve the extraction performance. To validate the performance of our detection and extraction method, various videos frames have been tested. The proposed method is very useful for the real-time application. Our future work is to detect and extract the moving overlay text to extend the algorithm for more advanced and intelligent applications.

REFERENCES

[1] A. K. Jain and B. Yu, "Automatic text location in images and video frames," *Pattern Recognition.*, vol. 31, no. 12, 1998, pp. 2055–2076.

[2] Lyu, M.R., Jiqiang Song, Min Cai, "A comprehensive method for multilingual video text detection, localization, and extraction", *IEEE Trans. Circuits Syst. Video Technol.*, Volume 15, Issue 2, Feb. 2005, pp. 243 – 255.

[3] C. Wolf, J.-M. Jolion, F. Chassaing, "Text localization, enhancement and binarization in multimedia documents" *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, Volume 2, 11-15 Aug. 2002, pp. 1037 – 1040.

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

[5] R. Lienhart and A. Wernicke, "Localizing and segmenting text in images, videos and web pages," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 4, Apr. 2002, pp. 256–268.

[6] S. Antani, D. Crandall, and R. Kasturi, "Robust extraction of text in video," in *Proc. 15th Int. Conf. Pattern Recognition.*, vol. 1, 2000, pp. 831–834.
[7] K. C. K. Kim et al., "Scene text extraction in natural scene images using hierarchical feature combining and verification," in *Proc. Int. Conf. Pattern Recognition*, Aug. 2004, vol. 2, pp. 679–682.
[8] R. C. Gonzalez and R. E.Woods, *Digital Image Processing*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 2002.
[9] S. U. Lee, S. Y. Chung, and R. H. Park, "A comparative performance study of several global thresholding techniques for segmentation," *Computer. Vis., Graph, Image Process.*, vol. 52, pp. 171–190, 1990.
[10] J. M. Pike and C. G. Harris, "A combined corner and edge detector," in *Proc. Alvey Vision Conf.*, 1988, pp. 147–151.
[11] Wonjun Kim and Changick Kim, "A New approach for overlay text detection and extraction from complex video scene", *IEEE Transactions on image processing*, vol. 18, no. 2, February 2009.
[12] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 1, pp. 62–66, Mar. 1979.
[13] Y. Liu, H. Lu, X. Xue, and Y. P. Tan, "Effective video text detection using line features," in *Proc. Int. Conf. Control, Automation, Robotics and Vision*, Dec. 2004, vol. 2, pp. 1528–1532.
[14] S. Kwak, K. Chung, Y. Choi, "Video Caption Image Enhancement for an Efficient Character Recognition", in *Proc. 15th Int. Conf. Pattern Recognition*, vol. 2, 2000, pp. 2606–2609.
[15] L. Agnihotri and N. Dimitrova, "Text detection for video analysis," in *Proc. IEEE Int.Workshop on Content-Based Access of Image and Video Libraries*, Jun. 1999, pp. 109–113.

AUTHOR PROFILES

VANI KASIREDDY received his M.Tech (VLSI SYSTEM DESIGN) degree from JNTUH University, TELANGANA, India, in 2012. She is serving as Lecture in the department of Electronics and Communication, CMR Engineering College, Hyderabad, India, since 2014. Her research interests include Digital signal processing, FPGA, Digital systems and Speech and video Processing and vlsi design.

SWATHI.G received his M.Tech (Electronics and Communication ENGINEERING) degree from JNTUH University, TELANGANA, India, in 2012. She is serving as Lecture in the department of Electronics and Communication, CMR Engineering College, Hyderabad, India, since 2014. His research interests include Digital signal processing ,communication systems and embedded systems.

G.SANTHOSH received his M.Tech (Communication Engineering) degree from VIT University, TamilNadu, India, in 2010. He is serving as Lecture in the department of Electronics and Communication, AVANTHI Engineering College, Hyderabad, India, since 2014. He has served as Lecturer in the department of Electronics and Communication, Guru Nanak Institute of Engineering & Technology, Hyderabad, India. His research interests include Digital signal processing, Digital systems, Mobile Wireless and Image Processing and Artificial Intelligence.