ISSN: 2277-3754
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 1, Issue 4, April 2012

Extraction of Text from Moving Vehicles

Rajesh Chhikara, Sunil Kumar, Rita Chikara, Gopinath S.

***Abstract*—With the decline in cost of security cameras and processing devices, the feasibility of video based security systems has increased. With a large amount of data to be processed, we need an efficient system which analyze the video and extract useful information which can later be indexed. In this paper we present a method which helps in extracting text from moving vehicles from a video captured by a surveillance camera. The algorithm is tested on a captured video of a typical street scene. That extracted information can later be indexed and used for further activities.**

***Index Terms*—Frame Differencing, Background Subtraction, Segmentation, Connected Components.**

I. Introduction

The problem of detecting text in video has been approached using three kinds of methods: edge-based, texture-based, and color-based methods. Edge-based methods [1, 5] use the observation that text has strong edges between the character and background pixels. Texture-based methods [4, 8] rely on the texture feature uniformity across the text object. Color-based methods [3, 2, 6] assume that the text pixels are similar in color or intensity. The three cited methods particularly have the advantage of making no assumption about the color and intensity of the text pixels, whereas others assume a white or bright text foreground color. Color features are first extracted and spatial rules are applied to find the spatial boundaries. The major difference between the three methods is the amount of information used in extracting the color features. Jain and Yu [3] used the information in the whole image in obtaining multi-valued image decomposition. Connected components are used to determine the spatial boundary. This method would work well if either the text occupies a large portion of the image or if the image has a few distinct colors, such as a magazine. In contrast, Gargi et al [2] examined very small features – small segments of the same color are fused together and grown to become text regions. The work of Mariano and Kasturi [6] strikes a balance between the two by examining a single row of pixels at a time. For every third row of the image, color clusters are computed, and a local search is used to find text boundaries in the row's vicinity.

With the availability of low priced hardware we can place surveillance cameras to a lot of locations and can have better security systems. With that a lot of things can be monitored including vehicles. An automatic system has been thought of which constantly monitor traffic and passing vehicles, and also extract any useful information or vehicle feature that includes vehicle type, model, color and other distinct marks(like some text present on the vehicle). Text recognition in document images has been an active research area for some time. However, text detection and recognition in video frames is a different domain and requires a very different approach. This is because text in printed documents is restricted to unicolor characters on a uniform background (plain paper). It only requires a simple thresholding to separate the text from the background. In video, however, characters in scaled down scene images suffer from a variety of noise components including uncontrolled illuminating conditions. Further, the background is a moving one and the characters can be of varied color, sizes and fonts.

With the help of our method we try to extract text marks present on moving vehicles captured from a still surveillance camera. The camera covers a wide area, which gives a clear view of the vehicles. This system can be used by the local authorities in terms of keeping a record of the text marks present on vehicle. This gives us a complete database giving us information of the frequency and type of vehicles passing a particular area. Any suspicious and unauthorized vehicle can be detected and the local police can be informed. Hence it provides us with a highly effective security system.

In this paper, we have solved this problem of text extraction in two steps. First the video is converted into frames. Then with the help of those generated frames we find out the moving vehicle. This frame containing the moving vehicle is segmented and the text is removed as an image. The first task is to extract frames from the video so that we can implement text detection algorithms on the frame containing the moving vehicle. To implement this thing we took a recorded video recorded from a fixed camera having typical street scene in which vehicles are clearly visible. From the recorded video it is important to first remove the frames and hence find the moving vehicle. We have followed the following algorithm to accomplish this task.

```
WhileindexFrame (1, 1) <= numFrames
indexFrame(1, 2) = min(numFrames, indexFrame(1, 1) +
frameLimit);
frameAll = read(videoObject, indexFrame);
for i = indexFrame(1, 1):1:indexFrame(1, 2)   %save frames
to pic
imgFrame = frameAll(:,:,:,i - indexFrame(1, 1) + 1);
saveFormat = strcat('%s\\%s_%0', int2str(numOrder),
'd.%s');
picName = sprintf(saveFormat, outputPath, fileName, i,
picFormat);
imwrite(imgFrame, picName);
```

ISSN: 2277-3754
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 1, Issue 4, April 2012

End

We have implemented this algorithm in MATLAB and were successful in extraction of frames from the recorded video.

II. TEXT EXTRACTION

After extraction of frames we took some frames in which the moving object is clearly visible. This selection of frames is done manually. Then, text extraction is performed on individual video frames.

A. RGB to GRAY conversion

To eventually detect the text marks on the moving vehicle in the pictures and hence using the pixel coordinates of the character obtained, we need to convert the RGB image of suitable video frame into grayscale to implement this algorithm.

B. Image Enhancement

Before any further processing is performed, the frame's edges are enhanced using a 3x3 mask. Further, a salt and pepper noise removal is performed to remove any noise, which was not removed in the previous step. For this purpose we use a median filter.

C. Edge Detection

After removing noise from the grayscale image by median filtering, Edge detection is the next process which is performed on that image. This function is solely responsible for detecting text as edge detection aims at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. In MATLAB, there are different methods to perform edge detection but in our algorithm we are using sobel method for edge detection. The Sobel method finds edges using the Sobel approximation to the derivative and returns edges at those points where the gradient of Image is maximum.

D. Edge Filtering

Once the edges are detected, a preliminary edge filtering is performed to remove areas which possibly do not contain text or, even if they do, they cannot be reliably detected. Edge filtering can be performed at different levels. One is at a frame level and the other is at a sub-frame level. On the frame level, if more than a reasonable portion of the frame is composed of edges, which might be due to high density of objects in the frame, we disregard that frame and take the next input frame. Using frame level needs only one counter to be maintained to keep the count of the number of edges in the image. This, however, can lead to the loss of text in some clean areas in an image and may result in false negatives. In order to overcome this problem, edge filtering is performed at a sub-frame level. To find text in an "overcrowded" frame, we maintain six counters to keep the count of the subdivided portions of the image. Three counters are used for three vertical portions of the image (one third of the area of the frame). Similarly three counters are used for three horizontal stripes. Text lines found in high-density edge areas (stripes) are rejected at a subsequent step. This filtering can be performed using smaller areas, in order to retain areas that are clean and contain text in a region smaller than one-third of an image.

E. Connected Component

Connected component (CC) analysis is performed on edges generated in the previous step. Each character is assumed to give rise to a connected component or a part thereof. All the edge pixels that are present at a certain distance (we use an eight-pixel neighbourhood), are merged into a single CC structure. This CC structure contains the location of the pixels that are connected together. This structure also contains the value of the leftmost, rightmost, top, and bottom pixel in the structure along with the location of the centre in the x and y directions. It also contains the count of the pixels giving rise to the CC. This is the area of the CC.

III. CONCLUSION AND FUTURE DIRECTION

When we implemented this method on MATLAB (Example shown later), we got around 40 connected components. From these 40 components, 3 components were of our interest. These 3 components contain the text written on the vehicle. Screen shots of the output of the program (ran on MATLAB) are shown in figure 1to 3. With the help of the above described method, we were successful in extracting text written on the moving vehicle from the recorded video. A method can be formulated in which the moving vehicle can be extracted automatically. Also if the connected components containing the text are selected automatically then the whole system can be implemented as a complete automatic system in which written text can be indexed and can be processed for further use. Also if optical character recognition is implemented properly, then text can be searched easily.

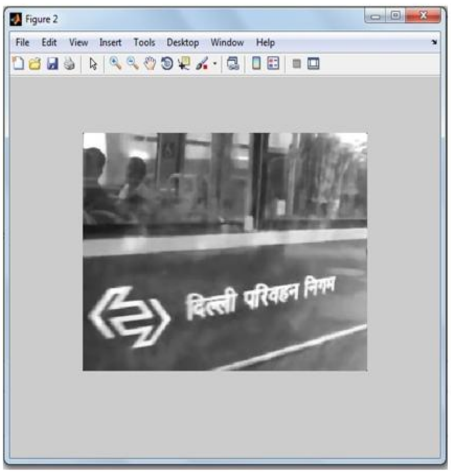

Figure 1: Selected frame of moving vehicle converted to gray scale

ISSN: 2277-3754
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 1, Issue 4, April 2012

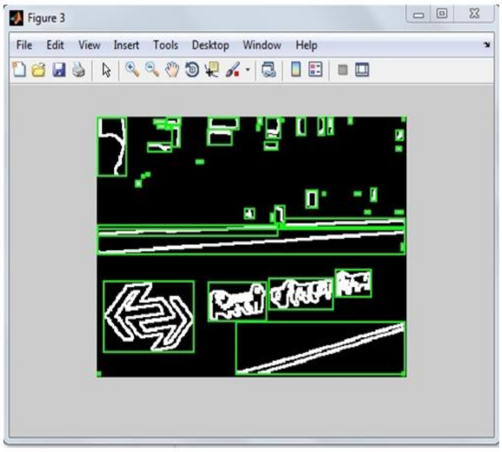

Figure 2: Connected components

Figure 3: Selected components showing text written on the vehicle

REFERENCES

[1] L. Agnihotri and N. Dimitrova.Text detection for video analysis. In Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries, pages 109–113, June 1999.

[2] U. Gargi, S. Antani, and R. Kasturi. Indexing text events in digital video databases. In Proc. International Conference on Pattern Recognition, pages 916–918, 1998.

[3] A. K. Jain and B. Yu.Automatic text location in images and video frames. Pattern Recognition, pages 2055–2076, December 1998.

[4] H. Li, D. Doermann, and O. Kia. Automatic text detection and tracking in digital video. IEEE Transactions on Image Processing, 9(1):147–156, January 2000.

[5] R. Lienhart and A.Wernicke. Localizing and segmenting text in images and video. IEEE Transactions on Circuits and Systems for Video Technology, 12(4):256–268, 2002.

[6] V. Y. Mariano and R. Kasturi. Locating uniform-colored text in video frames. In Proc. International Conference on Pattern Recognition, volume 4, pages 539–542, 2000.

[7] T. N. Tan and K. D. Baker. Efficient image gradient based vehicle localization. IEEE Transactions on Image Processing, 9(8):1343–1356, 2000.

[8] Yu Zhong, Hongjiang Zhang, and Anil Jain. Automatic caption localization in compressed video. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(4):385–392,