

Research Article

ViOC-optical Alphanumeric Character Extraction from Video Frames

¹Resmi R. Nair, ¹A. Shobana, ¹T. Abhinaya and ²S. Sibi Chakkaravarthy

¹Department of ECE, Vel Tech,

²Department of CSE, Vel Tech RR and SR Technical University, Chennai-600062,
Tamilnadu, India

Abstract: The main motto of the study is to provide the new distinct method to extract the optical characters in the form of alpha numeric characters from the video frames. In this study we proposed a new methodology to recognize the optical characters from the video frames; the methodology is taken with two step process, in first step the video frames are separated into frame by frames. Then the text detection phase is revoked with text localization and text verification and the second step is to recognize the characters. In this phase the text is verified and recognized. The final outcome is the recognized characters from the video frames. The experimental results are demonstrated clearly and the proposed method has an optimality over the video frames without any jitter or noise sequence in processing the extraction phase. The method performs better result than the existing algorithm and the results yields 94% accuracy in the MATLAB R2013b simulation environment.

Keywords: Localization, OCR, text extraction, text recognition, text verification, video frame

INTRODUCTION

OCR mainly deals with recognizing offline optical characters (Mokhtarian *et al.*, 1996; Tollmar *et al.*, 2004). Input sequence may be video or images which were scanned document or printed images. Major processing elements of OCR be denoted as (Chakkaravarthy *et al.*, 2013):

- Scanning
- Pre-processing
- Segmentation
- Feature extraction
- Recognition

Basics of OCR: The algorithm mentioned above shows the various steps involved in obtaining a text using the Optical Character Recognition (O.C.R). The block of text is obtained using the text detection process. These blocks of text is sent to the O.C.R, where these block of text is segmented into a single character and the templates are generated.

After the segmentations of the connected regions, the templates are compared with the segments. This is done to match the connected regions with the alphabets, numbers etc., (Fig. 1), to obtain the text. The text is got as the output from the O.C.R. The text got by comparison (Chen *et al.*, 2001; Koteeswaran *et al.*, 2012) is written to the text file like notepad, wordpad etc. Hence the non-editable text, which was got from the text detection phase (Srihari *et al.*, 2000;

Koteeswaran and Kannan, 2013) is converted into the editable text by passing it to O.C.R.

METHODOLOGY

The Algorithm mentioned above explains the various steps involved in the proposed system. Firstly, the Video is given as the input to the proposed system. The Input video is divided into individual frames and each individual frames are passed through the rest of the two phases and the individual frame represents the RGB image. The RGB image to gray scale conversion is done. The edge detection (i.e., horizontal and vertical edge detection) is done to the gray scale image using the Sobel and canny masks. Using edges as the prominent feature of our system gives the opportunity to detect characters with different fonts and colours since every character present strong edges, despite its font or colour, in order to be readable. Canny edge detector is applied to gray scale images. Canny uses Sobel (<http://www.intelligent-systems.info/classes/ee509/gui.htm>; <http://www.ele.uri.edu/~hansenj/projects/ele585/> OCR; Hidayatullah *et al.*, 2012; Neumann and Matas, 2010; Tou and Gonzalez, 1995; Swain and Ballard, 1991) masks in order to find the edge magnitude of the image, in gray scale and then uses non-Maxima suppression and hysteresis threshold. With these two post-processing operations Canny edge detector (Swain and Ballard, 1991) manage to remove non-maxima pixels, preserving the connectivity of the contours.

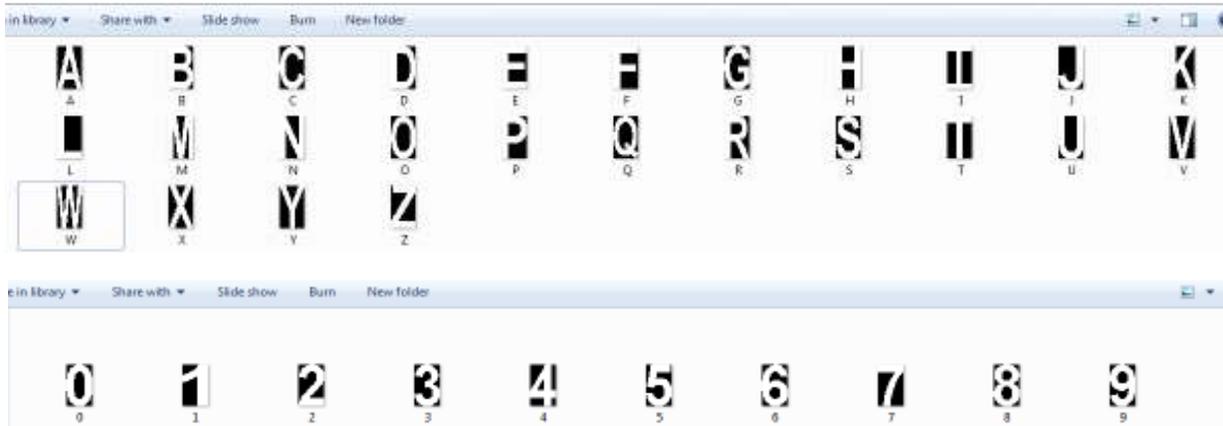


Fig. 1: Template files of alphabets and numbers

Later, dilation on the resulted image is done to find the text like region. Dilation by a cross-shaped element is performed to connect the character contours of every text line. The common edge between the vertical and the horizontal edges is extracted and it is dilated again to get the accurate text like regions. The groove filling mechanism is applied to fill the gap between the non-connected pixels. The co-ordinates of the dilated regions are sent to find whether the text like regions extracted is text or not. Once, the regions extracted are verified as the text and then this detected text is passed to the Optical Character Recognition (O.C.R). In the O.C.R, the block of detected text is segmented into characters and then the non-editable text is converted into editable text. The text is saved in the text file (i.e., notepad or WordPad).

Pseudo code:

```

Begin
Function OTE
    Input: Video file (.avi)
Output: Image/frames
    Step: Pre-processing
Convert RGB→grey;
Sobel (horizontal, videoframes)
Canny (vertical, videoframes)
Plot (octagon);
Plot (rectangle);
Dilate (videoframes);
    for i = 1:m
    for j = 1:n
    FindText (min (n), max (m), min (m), max (n));
    End
    End
Joincharparts (FindText);
End
Pseudocode for text verification
Begin
Function TextOrNot

```

```

Input: Four coordinates of the dilated regions (X,
X1, Y, Y1) and the edge map image (H)
Output: Text or Not
for i = X:X1
for j = Y:Y1
hcount = hcount+1
end
end
tcount = (X1-X) * (Y1-Y)
Ratio = hcount/tcount
If (Ratio >= 0.065)
Result = TRUE (Its Text)
end
end

```

EXPERIMENTAL RESULTS

Based on optical character recognition, our proposed method is used to recognize the text characters from the video streams. A video sequence of 12 frames is used as the data set model, OTE is applied in order to retrieve the text from the video frames. The text recognized in the video frames was “Rhino is rare”. Figure 2a and b denotes the output sequence of the OTE. Firstly the character recognized by OTE is considered to be the segmented frames of the video sequence. Various filters are used (which was denoted in the Fig. 3) in order to plot the diagonal mapping on the text characters. Full implementation is done through MATLAB (<http://www.mathworks.com/products/matlab/>; <http://www.mathworks.in/>).

Various frames are consider, pre-processed with basic filters and processed using OTE methodology; its results are denoted at the final frame of Fig. 2a and b.

Figure 3 denotes the output sequence of text extracted from the video frames. In frame 1 the word “Rhino” is extracted. In frame 2 the word “is” is extracted and in frame 3 the word “rare” is extracted, in frame four, all the extracted words from previous frames are processed and at final frame the words are recognized and displayed.

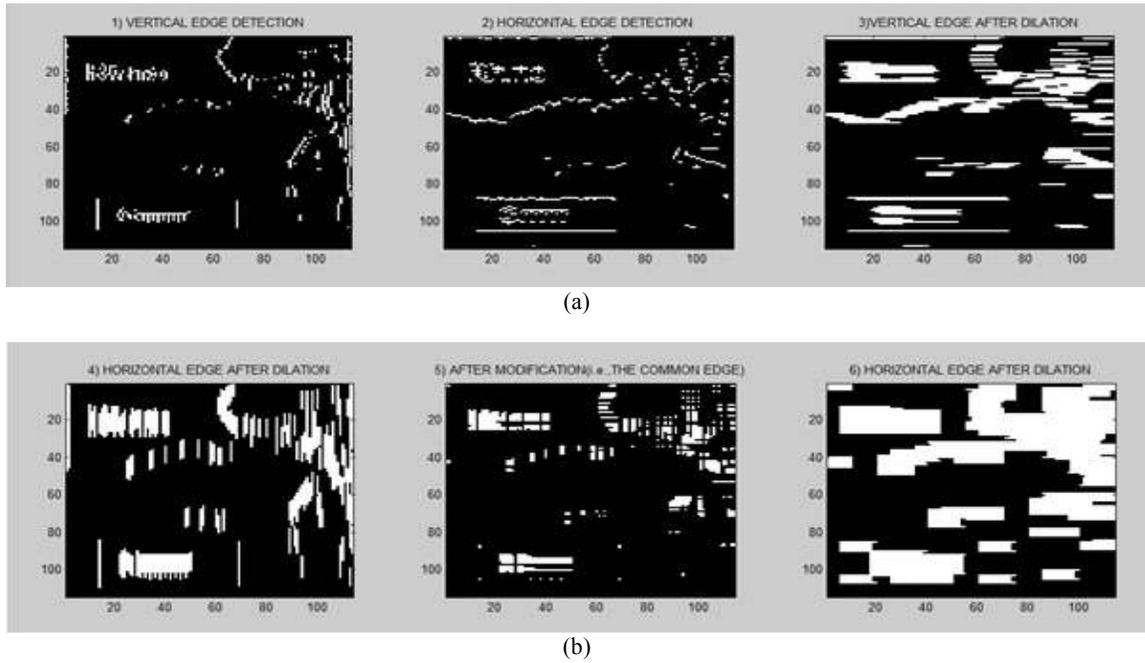


Fig. 2: Edge detection before and after dilation



Fig. 3: Output for the video sequence-O TE method

Table 1: Result analysis

Number of video frames	12
Detection percentage	97.08%
Extracted, recognized percentage	91.60%

CONCLUSION

As the conclusion of this study we experimentally performed some basic test in order to recognize the characters. Optical character recognition is playing a vital role in the field of image processing research and used in various applications. OCR processes mainly with segmentation and classification. The proposed method is evaluated experimentally with the video sequence and its results are shown clearly in Fig. 2a, b and 3.

Results shows that our proposed method has better performance in recognizing the characters in the video frames. Our proposed methodology has reduced noise with an accuracy of 97.08% (Table 1). In future we extend this study with more samples and increased accuracy to recognize the text and its patterns in the video sequences.

REFERENCES

- Chakkaravarthy, S.S., R.R. Nair and P.Visu, 2013. A novel methodology to identify and recognize the composite human gesture for Kinect based human-machine interface. *Life Sci. J.*, 10(2): 1261-1266.
- Chen, D., H. Boulard and J.P. Thiran, 2001. Text identification in complex background using SVM. *Proceeding of the International Conference on Computer Vision and Pattern Recognition*, pp: 621-626.
- Hidayatullah, P., N. Syakrani, I. Suhartini and M. Wildan, 2012. Optical character recognition improvement for license plate recognition in Indonesia. *Proceeding of the 6th UKSim/AMSS European Symposium on Computer Modeling and Simulation (EMS, 2012)*, pp: 249-254.
- Koteeswaran, S. and E. Kannan, 2013. Analysis of Bilateral Intelligence (ABI) for textual pattern learning. *Inform. Technol. J.*, 12(4): 867-870.
- Koteeswaran, S., J. Janet and E. Kannan, 2012. Significant term list based metadata conceptual mining model for effective text clustering. *J. Comput. Sci.*, 8(10): 1660-1666.
- Mokhtarian, F., S. Abbasi and J. Kittler, 1996. Robust and efficient shape indexing through curvature scale space. *Proceeding of the British Machine Vision Conference*, pp: 9-12.
- Neumann, L. and J. Matas, 2010. A method for text localization and recognition in real-world images. *Proceedings of the 10th Asian Conference on Computer Vision Volume Part III (ACCV'10)*, pp: 770-783.
- Srihari, R.K., Z. Zhang and A. Rao, 2000. Intelligent indexing and semantic retrieval of multimodal documents. *J. Inform. Retrieval*, 2(2/3): 245-275.
- Swain, M. and H. Ballard, 1991. Color indexing. *Int. J. Comput.*, 7(1991): 11-32.
- Tollmar, K., T. Yeh and T. Darrell, 2004. IDeixis: Image-based Deixis for finding location-based information. *Proceedings of the Conference on Human Factors in Computing Systems (CHI'04)*, pp: 781-782.
- Tou, J.T. and R.C. Gonzalez, 1995. *Pattern Recognition Principles*. Addison-Wesley Publishing Co. Inc., Reading, Massachusetts.