RESEARCH ARTICLE OPEN ACCESS

Text Extraction in Images

Shraddha [1], Soumya B [2], Swathi S Hegde [3], Thejaswini [4], Prof Kaushik K.S [5]
Department of Computer Science and Engineering
Canara Engineering College, Benjanapadavu
India

ABSTRACT

Content-based image retrieval, structuring and indexing of images is of great importance and interest in today's world. Text extraction from images is an extremely difficult and challenging job due to the variations in the text such as text scripts, style, font, size, colour, alignment and orientation; and due to extrinsic factors such as low image contrast (textual) and complex background. However, this is realizable with the integration of the proposed algorithms for each phase of text extraction from images using Open CV libraries and python programming language. Initially, the pre-processing phase involves gray scaling of the image. Thereafter, Text detection process takes place, which is followed by text binarization, localization and Character segmentation. Further, recognition of the processed and segmented characters is done. Experimental results for a set of static images confirm that the proposed method is effective and robust.

Keywords :- Image Pre-processing, Gray Scaling, Text Detection, Binarization, Text Localization, Character Segmentation and Text Recognition.

I. INTRODUCTION

Nowadays, information libraries that originally contained pure text are becoming increasingly enriched by multimedia components such as images, videos and audio clips. They all need an automatic means to efficiently index and retrieve multimedia components. If the text occurrences in images could be detected, segmented, and recognized automatically, they would be a valuable source of high-level semantics. For instance, in the Informedia Project at Carnegie Mellon University, text occurrences in images and videos are one important source of information to provide full-content search and discovery of their terabyte digital library of newscasts and documentaries [1]. Therefore, content-based image annotation, structuring and indexing of images is of great importance and interest in today's world.

Text appearing in images can be classified into: Artificial text (also referred to as caption text or superimposed text) and scene text (also referred to as graphics text). Artificial text is artificially overlaid on the image at a later stage (e.g. news headlines appearing on television, etc.), whereas, scene text exists naturally in the image (e.g. the name on the jersey of a player during a cricket match, etc.) [2]. Scene text is more difficult to extract due to skewed or varying alignment of the text, illumination, complex background and distortion. This paper focuses on artificial text and its extraction from still images. Existing OCR engines can only deal with binary text

Images (characters against clean background), and it cannot handle characters embedded in shaded, textured or complex background [3]. This is not always the case, as there exist many disturbances (noise) in the input text images. These disturbances have a high influence on the accuracy of the text extraction system.

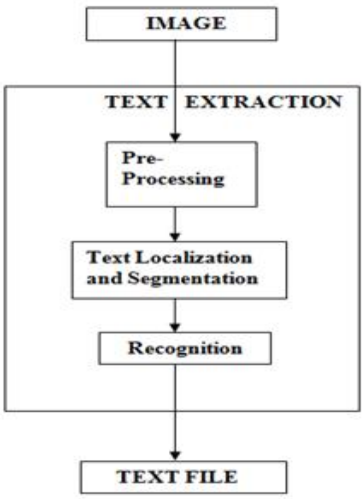

Fig. 1 the proposed model

The process of extracting text from images consists of various stages as seen in Fig. 1. Each stage consists of steps that are explained with the help of select algorithms in Section II and are finally demonstrated by presenting experimental results for a set of static images in Section III.

II. METHODOLOGY

The input image to our proposed system has a complex background with text in it. The first stage is image pre-processing, which serves to remove the noise from the input image and generates a clear binary image. Text segmentation is the next stage, where we differentiate each character from the entire word by circumscribing them into boxes and saving them each separately. The final stage is text recognition, where the segmented characters are compared to the stored character matrices and as a result, the closest match for each character is displayed.

A. Image Pre-processing

The main purpose for pre-processing the image is to produce an image containing the text to be recognized without any other disturbing elements or noise. This step contributes significantly to boost the performance of text recognition. To remove the disturbances and noise (i.e. everything else except the text) from the image, gray scaling is done firstly, followed by text detection, which is lastly followed by text recognition.

1) Gray scaling: The main purpose for gray scaling is to produce a binary image (containing of black or white pixels only), thus, making it easier to distinguish text from the background. First, all the pixels in the image are converted to shades of gray. To do this, the RGB (R: Red, G: Green, B: Blue) color components of each pixel in the image are extracted using bitwise shift operators. Each of these values of the R, G and B components vary from 0-255[4]. Then, these values are added in a proportion of Red: 30%, Green: 59% and Blue: 11% [5] to get the gray scaled equivalent of that particular pixel. This method is applied to each pixel in the image to convert the entire image into gray scale.

2) Text Detection: The main purpose of this step is to convert the text into unreadable format so that maximum amount of noise will be removed. The implementation method makes use of morphological operation to carry out the step. Here the gray scaled image becomes the input for the detection step and the output would be the detected image with much less noise.

3) Binarization: Binarization is one among the many conventional methods used for text extraction from images. These methods are based on the assumption that text pixels have a different color than the background pixels. Thus, a threshold color value is used to separate the text from the background. To be specific, this is done by comparing each pixel value to a threshold value (that lies between black and white) and setting that pixel value to black or white as its consequence. This yields the gray-scaled binary image that is the input for the next step of pre-processing. This step also provides a means for the storage of pixel values of the entire image in an array for further processing.

B. Text Localization and Segmentation

After we finish pre-processing the input image, all that remains is the text against the plain background. To separate each character from the entire word in the image, localization of individual characters is done, thus segmenting the text and generating distinct windows for each and every character in the image.

Text localization involves circumscribing the characters of the text one after the other. To do so, a three-line horizontal group scan is performed from left to right on the pre-processed image until a black pixel is encountered [7]. Later, all the connected black pixels in that cluster are discovered and the extreme X and Y coordinate values are extracted. Here, the change of colour is done in order to make sure that the character localized earlier is not re-encountered and that the next character is directly taken into consideration in search for the next black pixel. This process is iterated for all the characters until the scan reaches the end of the image without encountering a single black pixel. Once after segmentation, all the cropped images get saved in the original path which is given in the code.

C. Character Recognition

Now that we have discrete segmented character arrays localized from the pre-processed image, we arrive at the last stage of text extraction from images, i.e. recognition of characters. The segmented characters get compared with the characters present in database and then output the matching characters into the text file.

III. EXPERIMENTAL RESULTS

The proposed methodology in this paper is tested by building software in Python using OpenCV libraries and classes. The system accepts an input image with complex Background and applies the set methods presented in this paper to recognize the text from the image. Fig. 2 through Fig. 6 demonstrates the successful step-by-step extraction of the text from the input image (Fig. 2).

Fig 2: Input image

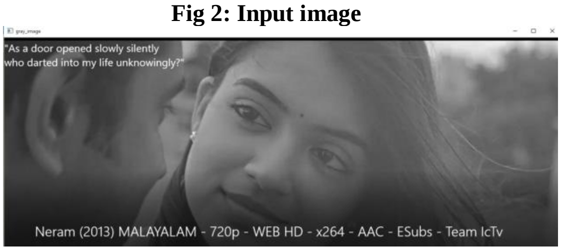

Fig 3: Gray scaled image

Fig 4: Binarized image

Fig 5: Localized image

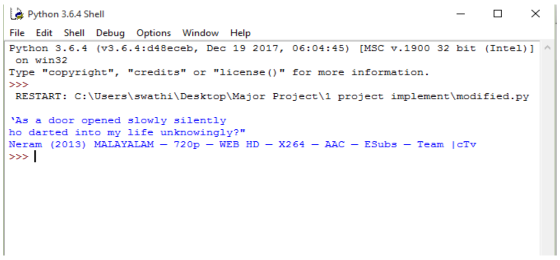

Fig 6: Recognized characters

The proposed methodology is robust and works well with a set of static input images. It is efficient enough to recognize maximum number of characters from the chosen input image.

IV. CONCLUSIONS

This paper introduces a text extraction system and discusses a methodology with promising directions for future research. This methodology has been successfully tested for a particular font in ASCII text (English language). This can be further extended to other fonts and scripts. Few applications of this system include: (1) Automatic number plate recognition (ANPR), (2) Content based image retrieval: Searching for the required keywords in the images and retrieving the images accordingly, (3) Document data compression: From document image to ASCII text, etc.

There might be a few cases in which certain characters may not get recognized correctly, i.e. a character may be recognized as some other character. This might be due to the discrepancy in the pixel intensity values of the processed input matrix and the various stored character matrices. It can be inferred from the experiments performed on the software that most of the text gets recognized successfully and that the proposed methodology is robust and efficient.

REFERENCES

[1] R. Lienhart and A. Wernicke, *Localizing and Segmenting Text in Images and Videos*, Transactions on Circuits and Systems for Video Technology, Vol. 12, No. 4, April 2002.

[2] C. Wolf and J. M. Jolion, *Extraction and Recognition of Artificial Text in Multimedia Documents*, Lyon research center for Images and Intelligent Information Systems.S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.

[3] S. Jianyong, L. Xiling and Z. Jun, *IEEE: An Edge-based Approach for Video Text Extraction*, International Conference on Computer Technology and Development, 2009

[4] J. Gllavata, Extracting Textual Information from Images and Videos for Automatic Content-Based Annotation and Retrieval, PhD thesis, Fachbereich Mathematik und Informatik der Philipps-Universität Marburg, 2007.

[5] C. Bunks, *Grokking the Gimp*, New Riders Publishing Thousand Oaks, CA, USA, 2000.

[6] K. Subramanian, P. Natarajan, M. Decerbo and D. Castañòn, *IEEE: Character-Stroke Detection for Text-Localization and* Extraction, Ninth International Conference on Document Analysis and Recognition (ICDAR), 2007.