

# CHARACTER SEGMENTATION METHOD FOR DIGITAL BACK-LIGHT CONSOLE UNDER DIFFERENT LIGHTING CONDITIONS

## Article history

Received

17 June 2015

Received in revised form

13 December 2015

Accepted

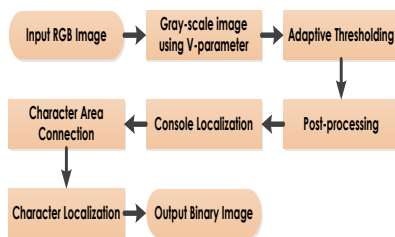
10 January 2016

Siriwuttichai Niruttisai, Settha Tangkawanit, Surachet Kanprachar\*

Department of Electrical and Computer Engineering, Faculty of Engineering, Naresuan University, Phitsanulok, Thailand

\*Corresponding author  
surachetka@nu.ac.th

## Graphical abstract



## Abstract

Nowadays, the amount of electronics data has been significantly increased especially for the case of capturing images. One particular application regarding this is the image capturing of the monitoring console. The real information in such console image capturing is the characters on the console. To transfer or store the captured images, a large amount of data is required. Instead of storing the whole image, some image processing techniques could be applied in order to reduce the amount of data required. In this paper, an image processing is done to the captured image by considering the value (V) parameter of the HSV (Hue, Saturation, and Value) color system. An adaptive threshold algorithm on V parameter is adopted for segmenting the console area and then the character areas from the whole image. Under different lighting conditions from 0 to 450 lux, the console area can be correctly selected. And, the characters appearing on the console area can be retrieved with 98 percent of accuracy.

**Keywords:** Color system; character segmentation; image processing; monitoring console; lighting conditions

© 2016 Penerbit UTM Press. All rights reserved

## 1.0 INTRODUCTION

Currently, the optical character recognition or OCR technology has been widely used in order to translate the written or typed message to be an editable message. For example, there are many available free online OCR websites that convert images or pdf documents to editable files. The OCR accuracy for character segmentation will be degraded if the background image of the message is contained in the character image in the character detection process. To improve the accuracy of OCR, many algorithms and techniques have been adopted so that the background image and character image can be clearly separated. One important factor affecting the accuracy of character segmentation in many applications is the light intensity of the considered image. This factor can lessen the effectiveness of character segmentation.

In the past 10 years, many techniques have been proposed in order to segment objects or characters from an image. For object segmentation, in [1], a comparison of using RGB (Red, Green, and Blue) and HSV (Hue, Saturation, and Value) color systems in detecting cars in parking area was studied. It was found that under different lighting condition, HSV color system is more suitable than RGB color system. Two parameters of HSV color system; namely, S (Saturation) and V (Value), were shown to be important parameters in this object segmentation. Character segmentation has been done in [2, 3] for separating the characters from the background using the images obtained from a scanner and the internet. Wavelet transform [2] and Sub-layer analysis technique [3] were adopted for extracting the characters from the whole image. It was shown that these two techniques could improve the performance of the character segmentation. Hill climbing algorithm was used in

segmenting the characters from a car license plate [4]. It was shown that applying such algorithm with the HSV color system, the accuracy of the character segmentation was increased. Segmenting characters from video file was done in [5] using character edge extraction with Robert high-pass filtering. It is seen that there are many techniques applying to object or character segmentation in variety of applications in order to improve the accuracy of the segmentation.

In this paper, capturing and segmenting characters from a digital back-light [6 – 10] console is considered. The image of the console is captured and the area of interest is extracted. Then, the characters in such area are segmented. The proposed technique is to use V-parameter in HSV color system with the adaptive threshold algorithm in the image segmentation process. To show the effectiveness of the proposed technique, an image in RGB color system is also considered. Additionally, the result of applying a widely used algorithm called Otsu's method is obtained so that the adaptive thresholding algorithm can be compared. The images used are captured with different lighting conditions.

Organization of the paper is done as follow. Theory and technology used in the paper are reviewed in Section 2. Processes in capturing and segmenting the digital back-light console are given in Section 3. And, in Section 4, the results of segmenting the area of interest and the characters in such area are shown. Finally, the work done in the paper is summarized in Section 5.

## 2.0 RELATED TECHNOLOGY OVERVIEW

Many related technology and algorithms are adopted in this work. Information about these is described below.

### 2.1 HSV Color System

Color elements in HSV color system are categorized as Hue (H), Saturation (S), and Value (V). In each particular pixel in an image, these three elements are assigned. HSV color elements can be viewed as a circular cylindrical coordinate as shown in Figure 1.

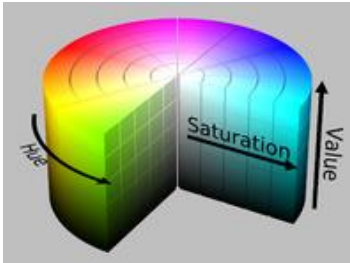


Figure 1 HSV color system [11]

From Figure1, it is seen that Hue or H-parameter is the color value determined by the radial slice of the cylinder. Saturation or S-parameter is the brightness

obtained from the distance from the center of the cylinder. And, Value or V-parameter is the light intensity determined from the distance from the bottom of the cylinder.

HSV color system parameters can be calculated from RGB system parameters [12]; that is,

$$V = \text{Max}(R, G, B) \quad (1)$$

$$S = \begin{cases} \frac{V - \text{Min}(R, G, B)}{V} & \text{if } V \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$H = \begin{cases} \frac{60(G - B)}{V - \text{Min}(R, G, B)} & \text{if } V = R \\ \frac{120 + 60(B - R)}{V - \text{Min}(R, G, B)} & \text{if } V = G \\ \frac{240 + 60(R - G)}{V - \text{Min}(R, G, B)} & \text{if } V = B \end{cases} \quad (3)$$

From (1) to (3), it is seen that if the values of R, G, and B are known, the values of H, S, and V can also be determined. For example, if a pixel in an image has the values of R, G, and B as 50, 128, and 100, respectively. Applying (1), (2), and (3) to the given values of R, G, and B, we get that

$$V = \text{Max}(50, 128, 100) = 128 \quad (4)$$

$$S = \frac{128 - \text{Min}(50, 128, 100)}{128} = 0.61 \quad (5)$$

$$H = \frac{120 + 60(100 - 50)}{128 - \text{Min}(50, 128, 100)} = 40 \quad (6)$$

From (4) to (6), it is seen that from the known R, G, and B values, the H, S, and V values are 128, 0.61, and 40, respectively.

### 2.2 Image Segmentation

Image segmentation is a process that separates the features in an image depending on a specific criterion. This process is done in order to reduce the amount of data required in analyzing the image. One important technique that has been used in image segmentation is the adaptive threshold technique for transferring a color image to a binary image. This technique allows the threshold to be changed according to the surrounding pixels of a particular area of interest of an image. The process is done as follows.

- A 2-dimension filter with  $N \times N$  pixels is established.
- A constant, C, is set for being used in the thresholding process. The threshold value in a particular area can be determined by (7).

$$\text{ThresholdValue} = \text{Mean} - C \quad (7)$$

where Mean is the average gray-scale value of the pixels in the  $N \times N$  filter.

In (7), the threshold value is determined by subtracting Mean by C. The constant, C, can be modified accordingly depending on many factors; for example, lighting condition. It is seen from (7) that the threshold value is varied or changed adaptively as the

filter moves from one area to the next depending on the surrounding pixels. There are two popular methods in determining Mean, the average gray-scale value of the pixels in an  $N \times N$  filter; that is, the mean value method and the Gaussian method. In the mean value method, the gray-scale values of all pixels are averaged without any weighting function applied. And, in Gaussian method, the gray-scale values of all pixels are weighted differently using Gaussian function and then averaged. Applying one of these methods, the threshold value can then be obtained. The threshold value is used in comparison to the middle pixel of the filter in order to transfer a gray-scale value of such pixel into a binary one.

**2.3 Image Detection and Localization**

Image detection and localization are the processes where an object of interest in an image is identified and located. In this work, a function called Blob detection from EmguCV library version 2.4.2 [13] is adopted. This function is based on the work from Fu Chang in [14] that the external contour and internal contour of the object of interest can be determined by connect components labeling with contour tracing technique. Such technique is done with group of pixels in order to divide the area of pixels to be the internal or external areas.

**3.0 METHODOLOGY**

In this section, the processes for preparing images and testing the proposed system are explained. It starts by first generating images of digital back-light console. The image resolution is  $1080 \times 980$  pixels. 10 different light intensities varying from 0 to 450 lux are used. The digit presented in the console is varied from 1 to 10. Varying the lighting condition and the digit presented, 100 images of digital black-light console can then be generated.

These 100 images are passed through the proposed system in order to determine the effectiveness of the system. The console area of each image will be segmented. Then, the character areas in the console area are segmented and the character is each character area is recognized. The diagram in the system is shown in Figure 2.

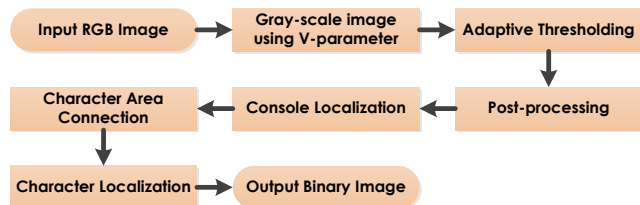


Figure 2 The proposed system diagram

From Figure 2, it is seen that there are 6 processes in the diagram. In the first process, the generated digital black-light console image in RGB is transformed into an HSV image. And, the only V-parameter of the HSV image is used to produce a gray-scale image. An example of this process is shown in Figure 3.

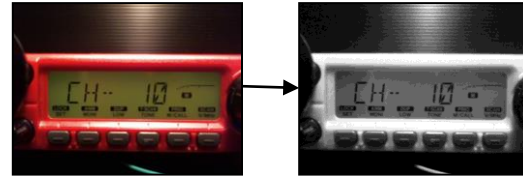


Figure 3 Transforming from an RGB image to a V-parameter gray-scale image

The second process is about applying an adaptive threshold to the image. This is done by using  $N$  by  $N$  filter in order to determine the average value of a particular area of the image. Then, the average of such area is subtracted by a constant,  $C$ , in order to get an adaptive threshold. This was shown in (7) in Section II. Applying the achieved threshold to the middle pixel of the considered area, a binary value can then be assigned. The values of  $N$  and  $C$  are set to be 19 and 3, respectively. The resulting image after having this process done is shown in Figure 4.

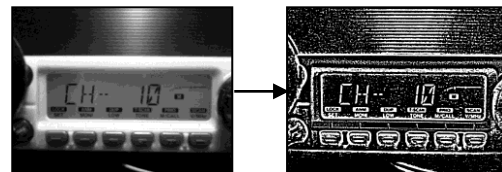


Figure 4 Adaptively thresholding the V-parameter gray-scale image to a binary image

To reduce the noise in the image obtained from the previous process, in this work, a post-processing is applied. This is done by using  $M$  by  $M$  median filter with  $M = 7$  pixels in order to smooth the image. The image after applying this median filter is shown in Figure 5. It is seen that the resulting image has less number of small white dots.

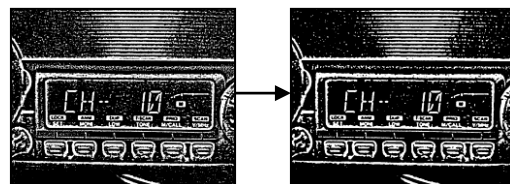


Figure 5 Noise reduction in the image using median filtering

Next, the console area has to be detected. In this work, Blob detection is used. It is seen that after applying Blob detection to the image, there will be many possible areas detected. To be able to localize the area of interest, some conditions have to be

made. First, the area of each detected area is considered. If the area of the detected area is between 10 and 50 percent of the whole image, such area is passed to the next step. The ratio between the width and height of the area is then considered. If the ratio of the considered area is approximately 3.5, the area will be chosen. And, the rest of the image will be discarded. The selected console area is shown in Figure 6. It is seen that only the true console area is correctly chosen.

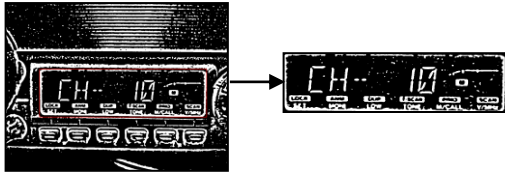


Figure 6 Selected console area using Blob detection

Considering the characters (or digits) shown in Figure 6, it is seen that in each character, the pixels are not totally connected. Hence, in the next process, the pixels in each character are connected using  $5 \times 3$  filter. The filtering is done to all pixels in the selected console area twice. The resulting image is shown in Figure 7.

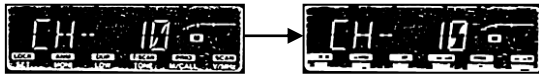


Figure 7 Character pixel connection after filtering

It is seen from Figure 7 that in each particular area, the pixels in such area are connected and viewed as a whole. For example, the character 'C' on the left side, it is seen that the corner and the middle parts are disconnected. But, after applying filtering twice, those parts are connected and can be viewed as a character 'C' clearly. These connections are also done to other areas.



Figure 8 Selected character areas using Blob detection

After connecting the character, Blob detection is applied again in order to separate the character areas from the whole console area. This process is done by considering the areas detected by Blob detection technique with the following conditions; that is, the area must contain at least 30 percent of the whole console area, and the ratio between the height and the width must be between 0.3 and 0.7. After having this process done, only characters in the console area can be obtained as shown in Figure 8.

## 4.0 RESULTS AND DISCUSSION

The result of having the proposed system done is obtained and compared to the case of using gray-scale RGB image. Also, it is compared to the case of using a widely used technique called Otsu's method.

The effectiveness of console area detection is first studied. All 100 images generated are used with the proposed system; that is, all images are first transformed into HSV images and only V-parameter is used for making gray-scale images. The result of getting console area in all 100 images is shown in Figure 9. Also, the result of using RGB images is shown.

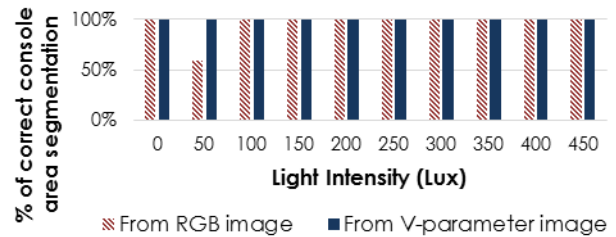


Figure 9 Percentage of correct console area segmentation comparing between the proposed technique and the case of gray-scale RGB images

From Figure 9, it is seen that for the case of using the proposed technique, with different lighting conditions, the console area can be selected correctly. While, for the case of using RGB images, in most lightning conditions, the console area can be selected properly. However, at the light intensity of 50 lux, the effectiveness of the console area selection is dropped to 60 percent. Since at 50-lux of light intensity, the pixels of gray-scale RGB images are not properly connected and using Blob detection cannot separate the area of interest. It is clearly seen that the proposed technique is less affected by lighting condition and superior to the case of using RGB images.

The effectiveness of the proposed technique in terms of the thresholding process is studied and compared to that from the case of using Otsu's method. The result of selecting console area is shown in Figure 10.

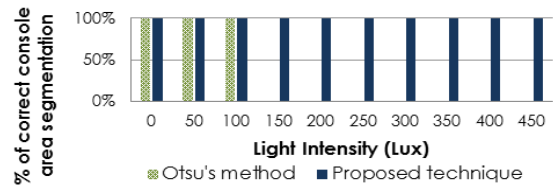
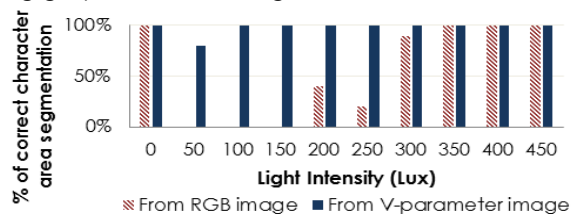


Figure 10 Percentage of correct console area segmentation comparing between the proposed technique and Otsu's method

It is seen that for the proposed technique, the console area can be chosen correctly in all lighting conditions. However, for Otsu's method, the console area can be chosen correctly only for the low light

intensity; that is, 0 to 100 lux. At higher level of light intensity, using Otsu's method does not give a proper console area. It is clearly seen that with Otsu's method, light intensity can affect the effectiveness in terms of console area selection while this does not happen with the proposed technique.

Next, after having a correct console area, the character areas in the console area have to be selected. This is done by using Blob detection with some conditions as mention in Section III. If the correct character areas in the considered console area are segmented correctly, the following process can then be done accordingly. That is, the character in the character area will be matched to a set of pre-defined characters using normalized correlation coefficient so that such character can be interpreted. In Figure 11, the correctness of character area segmentation in different light intensity is shown for the case of using the proposed technique and the case of using gray-scale RGB image.



**Figure 11** Percentage of correctness of characters' interpretation comparing between the proposed technique and the case of gray-scale RGB images

Considering Figure 11, for the case of using gray-scale RGB images, it is seen that only at the light intensities of 0, 350, 400, and 450 lux; the character areas in the console can be segmented correctly. While at 50-lux to 250-lux of light intensity, the correctness is less than 50 percent. However, when considering the result from the proposed technique, it is obviously seen that for all light intensities, character areas in the console area can be segmented 100 percent except at the 50-lux intensity, 80 percent of the character areas in the console image are selected correctly. In another word, from 100 images considered, 98 images can be correctly segmented and character-recognized. This figure shows a promising performance of the proposed technique in character segmentation for the digital black-light console under different lighting conditions.

## 5.0 CONCLUSION

A proposed technique for segmenting the console and character areas from a digital black-light console image is given. The V-parameter in the image is used for transforming the image into a gray-scale image. Then, the adaptive thresholding and Blob detection algorithms are applied. With different lighting conditions, the performance of the proposed

technique is studied and compared to the case of using gray-scale RGB images and the case of using Otsu's method. It is shown that with the proposed technique, the console area can be selected correctly. And, the characters on the console area can be obtained correctly with 98-percent accuracy. This shows that the proposed technique works really well with different lighting conditions and can be useful to many digital black-light console applications.

## Acknowledgement

This work was supported by the research and development fund of the National Broadcasting and Telecommunications Commission, Thailand (Contract No. T2-1-0002/56).

## References

- [1] Kanprachar S. and Tangkawanit S. 2007. Performance of RGB and HSV Color Systems in Object Detection Applications under Different Illumination Intensities, in *IMECS*. 1943-1948.
- [2] Weijuan W., Xianglin H., Lifang Y., Zhao Y., and Pengju Z. 2009. An Efficient Method for Text Location and Segmentation, in *Software Engineering, 2009. WCSE '09. WRI World Congress*. 3-7.
- [3] Xianglin H., Lifang Y., and Zhao Y. 2009. A Method of Text Segmentation from Scanned Image with Complex Background, in *Management and Service Science, 2009. MASS '09. International Conference on, 2009*. 1-4.
- [4] Lihong Z., Junbin G., and Xiangjian H. 2010. Efficient character segmentation on car license plates, in *Control Automation Robotics & Vision (ICARCV), The 11th International Conference*. 1139-1144.
- [5] Shih-Chang H. and Cheng-Nan H. 2012. A High-Performance Videotext Detection Algorithm, in *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2012 Eighth International Conference*. 242-245.
- [6] Chao L., Wenming Y., and Qingmin L. 2011. An automatic interpretation method for LCD images of digital measuring instruments, in *Image and Signal Processing (CISP), 2011 4th International Congress*. 1826-1829.
- [7] Ghosh S. and Shit S. 2014. A low cost data acquisition system from digital display instruments employing image processing technique, in *Advances in Computing, Communications and Informatics (ICACCI), 2014 International Conference*. 1065-1068.
- [8] Ghugardare R. P., Narote S. P., Mukherji P., and Kulkarni P. M. 2009. Optical Character Recognition System For Seven Segment Display Images Of Measuring Instruments," in *TENCON 2009 - 2009 IEEE Region 10 Conference*. 1-6.
- [9] Huiying S. and Coughlan J. 2006. Reading LCD/LED Displays with a Camera Cell Phone," in *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on*, 119-119.
- [10] Zhuofu L., Zhongming L., Panpan G., and Ming G. 2012. The Research Of Character Recognition Algorithm For The Automatic Verification Of Digital Instrument, in *Measurement, Information and Control (ICMIC), 2013 International Conference*. 177-181.
- [11] SharkD, 2011. The HSV Color Model Mapped To A Cylinder. 22 March 2010, 01:02 (UTC).
- [12] Marks. K. 2010. *The OpenCV Reference Manual* Available: <http://docs.opencv.org/opencv2refman.pdf>
- [13] Emgu CV. 2011 Available: <http://www.emgu.com/>

- [14] Chang F., Chen C.-J., and Lu C.-J., 2004. A Linear-Time Component-Labeling Algorithm Using Contour Tracing Technique. *Computer Vision and Image Understanding*. 93: 206-220,