VOL. 11, NO. 21, NOVEMBER 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

SOCCER FIELD DETECTION BASED ONHISTOGRAM OF S-RGB

Panca Mudjirahardjo, Nurussa'adah and Ponco Siwindarto
Department of Electrical Engineering, Faculty of Engineering, Universitas Brawijaya, Indonesia
Jl. MT. Haryono 167, Malang, Indonesia
E-Mail: panca@ub.ac.id

ABSTRACT

Field detection and extraction are the first task in the object recognition in soccer robot. In this paper we propose a novel method of background detection. It based on color appearance. In this method, we assume that the background has a dominant color in a frame. First, we calculate the sum of RGB intensity (we called as s-RGB) in each pixel. Then we group s-RGB in 16 bins to form a histogram of s-RGB. Second, we obtain a modus of s-RGB. This modus is used to determine the interest bin. Third, from this interest bin, we create histogram of R, G and B intensity, then we calculate modus of R, G and B intensity. These intensities are supposed to be the background color. Applying a threshold, we can extract foreground. For image size of 640×480 pixels, the computation time is 97.29 ms, suitable for real time application.

Keywords: histogram of S-RGB, object recognition, field detection.

1. INTRODUCTION

The goal of pattern recognition is to recognize the foreground as the interest object. For this task, the first process is separation of foreground from the background. After we extract the foreground, then we process it for the next computation, such as feature extraction, probability measurement, segmentation or template measurement. For soccer robot, background is the soccer field.

There are many automatic foreground extraction methods, for one frame processing [1-5] or sequence frames processing [6-9]. Tong *etal*. [1] performed field extraction based on the color distance between a pixel and the mean value of the field. They used only hue and saturation components in HSV color space. Huang *etal*. [2] employed a histogram learning technique to detect the background pixels. Color models were learned for foreground and background pixels, using a training set of soccer videos. The color model is an RGB color histogram with N bins per channel in the RGB color space. Ali etal. [3] employed their algorithm for soccer field and assumed the background color is green color. They determined the background pixel if the color intensity of $G > R > B$. Mudjirahardjo *etal* [4][5] extracted the foreground from background by using Euclidean distance function of the determined pixels as foreground and the surrounding pixels. If the distance was less than a threshold, then the pixel was a foreground. They calculated the distance in RGB color space [4] and HSV color space [5].

Chakraborty *etal*. [6] employed a background subtraction method along with a frame differencing method using three consecutive frames were employed for segmentation on the moving objects in the foreground. Durus [7] accomplished the segmentation of background by using the dominant color distribution. They calculated the mean of the peak intensity values of the histogram of several frames. Then they measured each pixel intensity respect to the mean value by using Mahalanobis distance. If the distance less than the variance, then the pixel is supposed to be background. Mudjirahardjo *etal*. [8] extracted the interest object in dynamic background by using velocity histogram based on Harris corner detector and Lucas-Kanade tracker. And in [9] they extracted the interests object as [8] and shift histogram based on motion history image (MHI).

This research aims to build an algorithm to detect background based on the dominant color in RGB color space. This algorithm has robust and fast processing time to be used in robot application, such as a soccer robot.

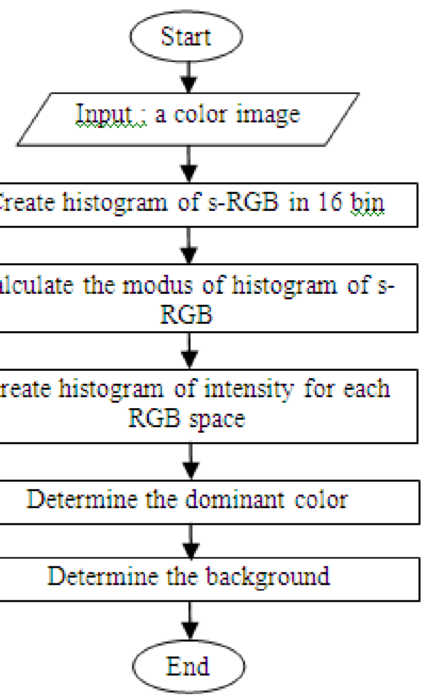

Figure-1. Overview of the proposed method.

The overview of this experiment is shown in Figure-1.The structure of this paper is as follows. Creating the histogram of s-RGB, histogram of intensity and detection of background are overviewed in section 2. Experimental results are shown in section 3. Finally the paper is concluded in section 4.

2. OVERVIEW OF THE PROPOSED METHOD

In this paper, we evaluate the RGB color space to detect background.

2.1 Creating the histogram of s-RGB

To provide the dominant color in a frame image, the process is explained in the following sub chapter. First, we calculate the sum of RGB intensity (s-RGB) at each pixel. This calculation as in equation (1),

$$s-RGB(x,y) = I_R(x,y) + I_G(x,y) + I_B(x,y) \quad (1)$$

where s-RGB(x,y) is sum of RGB intensity at pixel (x,y), I_R(x,y), I_G(x,y), and I_B(x,y) are red, green and blue intensity at pixel (x,y), respectively. When we use 8 bit to code a color intensity, then we can get the s-RGB value of 0-765.Second, we divide the s-RGB value into 16 bins, then create the histogram of s-RGB as shown in Figure-2.

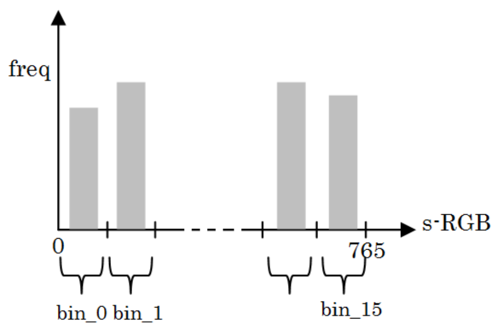

Figure-2. A histogram of s-RGB.

Third, from this histogram we determine modus bin of s-RGB by using equation (2),

$$\text{mod}_{\text{s-RGB}} = \arg\max_{bin}\left(histogram\_s-RGB\right) \quad (2)$$

This modus value restricts the pixels having bin of s-RGB for further processing.

2.2 Creating the histogram of intensity

After we get the modus value, for every pixel which has s-RGB value within the modus bin will be processed to create histogram of intensity for each RGB intensity. These histograms show frequency of intensity appearance of color. To determine the dominant color within modus bin, we calculate the modus of color from each histogram of intensity as in equation (3),

$$\text{mod}_{\text{hist\_R}} = \arg\max_{I_R}\left(histogram\_R\right) \quad (3.a)$$

$$\text{mod}_{\text{hist\_G}} = \arg\max_{I_G}\left(histogram\_G\right) \quad (3.b)$$

$$\text{mod}_{\text{hist\_B}} = \arg\max_{I_B}\left(histogram\_B\right) \quad (3.c)$$

Then the dominant of color, mod_{RGB}, can be calculated as in (4),

$$\text{mod}_{\text{RGB}} = \max\left(\text{mod}_{\text{hist\_R}}, \text{mod}_{\text{hist\_G}}, \text{mod}_{\text{hist\_B}}\right) \quad (4)$$

From (4) we will see what color is dominant in the frame image.

If we define the identity of pixel in image, I_P, as in (5),

$$I_P = \begin{cases} 1 & if \text{ s-RGB within } \text{mod}_{\text{s-RGB}} \\ 0 & otherwise \end{cases} \quad (5)$$

Then we decide a pixel in image as background or foreground as in (6),

$$piksel = \begin{cases} I_P \times background & \text{mod}_{RGB} - th < I_{\text{mod}_{RGB}}(x,y) < \text{mod}_{RGB} + th \\ I_P \times foreground & otherwise \end{cases} \quad (6)$$

where th is a threshold value; I_{modRGB}(x,y) is an intensity of mod_{RGB} at pixel (x,y). For example, if mod_{RGB} is 'green', then I_{modRGB} is I_G. If I_P equals 0, then this pixel should be the foreground.

2.3 Detection the background

After we process every pixel by equation (1) until (6), then we create a binary image. The 'foreground' pixel is labeled as 'white', while the 'background' pixel is labeled as 'black'.

This result image still has noise, i.e. there are many dots of foreground in background area, and vice versa. To remove this noise we apply an opening operation on it.

3. EXPERIMENTAL RESULT

To conduct the experiment, we set-up the experimental environment as follows: Operating system is Windows 7 professional; the processor is Intel® core™ i5-2410 M CPU @ 2.30 GHz; 2048MB RAM; and the used software is Microsoft Visual Studio 2010.

We evaluate the image size of 275×183 pixels and 640×480 pixels. The image size of 640×480 pixels is an image frame of real time. We set the threshold value in (6) of 15. For opening operation, we apply the structuring element 5×5 pixels.We perform a comparison of three processes. The first process, we don't use histogram of s-RGB. We use histogram of intensity directly, then calculate eq. (3)(4). The determination of background as in (7),

$$piksel = \begin{cases} background & \text{mod}_{RGB} - th < I(x,y) < \text{mod}_{RGB} + th \\ foreground & otherwise \end{cases} \quad (7)$$

The second process, we process the image to follow equation (1)-(6), using 8 bins of s-RGB. And the third process, similar with the second process, the difference is this process using 16 bins of s-RGB.

The process results of determination of the dominant color andthe results of background detection are shown in Figure 3and 4, respectively.

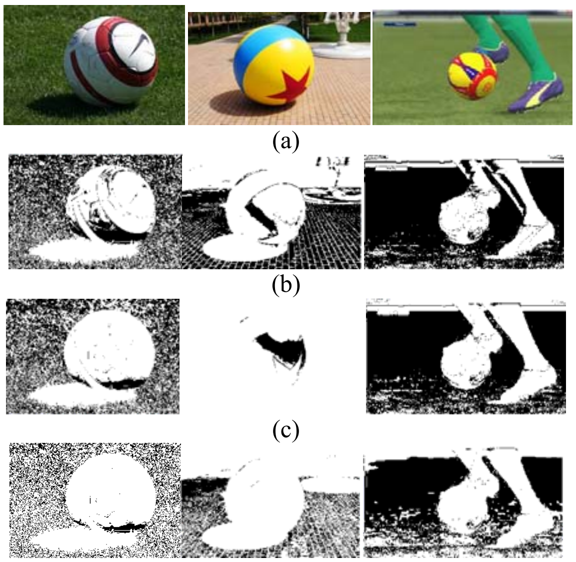

Figure-3. Process of background detection based on dominant color appearance (a) original images (b) using the first process (c) using the second process (d) using the third process.

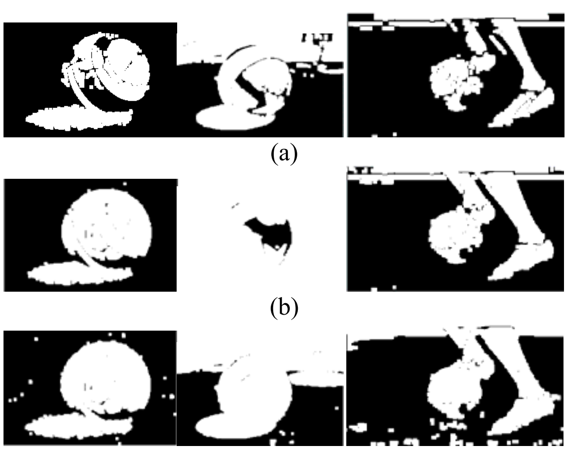

Figure-4. The result after applying the opening operation using structuring element of 5×5 pixels (a) using the first process (b) using the second process (c) using the third process.

The average of computation time for image size of 640×480 pixels are shown in Table-1.

Table-1. Computation time for image size of 640×480 pixels.

	Computation time (ms)
Process 1	97.39
Process 2	98.17
Process 3	97.29

DISCUSSIONS

Figure 3(a) and 4(a) are the process result which relies on the global dominant color in the image. These results are more noisy than the other processes. There are many parts of foreground recognized as background. Due to it has to compute for all pixels as in (7), it takes the computation time.

Differ from the first process, the second and the third processes only create the histogram of intensity for pixels which have s-RGB value within the modus s-RGB bin. Determination of dominant color and background are also performed for pixels which have s-RGB value within the modus s-RGB bin. For pixels which have s-RGB value outside the modus s-RGB bin are considered as foreground. If we set many bins, the number of pixels within the modus s-RGB bin will decrease. Then the computation time to perform the background detection will decrease. The background detection results are better than the first process, as depicted in Figure 4(b) and 4(c).

4. CONCLUSIONS

In this paper, we evaluate and demonstrate a histogram of s-RGB for background detection. With this histogram, determination of dominant color and background are more accurate than without. The numbers of pixels to be processed are reduce. Then the computation time to detect background is low, suitable for real time application.

As future work, we will develop an algorithm to extract the shape of object as a foreground.

ACKNOWLEDGEMENT

This work was supported by DIPA-FT Universitas Brawijaya, Grant No. 45/UN10.6/PG/2016.

REFERENCES

[1] X.F. Tong, H.Q. Lu, Q.S. Liu. 2004. An Effective and Fast Soccer Ball Detection and Tracking Method. Proceeding of the 17th International Conference on Pattern Recognition (ICPR), Cambridge, UK. pp. 1-4.

[2] Y. Huang, J. Llach, S. Bhagavathy. 2007. Players and Ball Detection in Soccer Videos Based on Color Segmentation and Shape Analysis. Proceeding of Multimedia Content Analysis and Mining (MCAM), Weihai, China. pp. 416-425.

[3] M.N.N. Ali, M.A. Wadud, S.L. Lee. 2012. An Efficient Algorithm for Detection of Soccer Ball and Players. Proceeding of Conference on Signal and Image Processing (SIP). pp. 1-8.

[4] P. Mudjirahardjo, M.F.E. Purnomo, R.N. Hasanah, H. Suyono. 2015. Multi Scale Performance of Feature Extraction for Human Head Recognition. International Journal of Applied Engineering Research. 10(17): 37731-37736.

[5] P. Mudjirahardjo, M.F.E. Purnomo, R.N. Hasanah, H. Suyono. 2016. Histogram of Transition for Human Head Recognition. Jurnal Teknologi. 78(5-9): 53-58.

[6] B. Chakraborty, S. Meher. 2013. A Trajectory-Based Ball Detection and Tracking System with Applications to Shooting Angle and Velocity Estimation in Basketball Videos. Proceeding of Annual IEEE India Conference (INDICON), Mumbai. pp. 1-6.

[7] M. Durus. 2014. Ball Tracking and Action Recognition of Soccer Player in TV Broadcast Videos. Doctoral Disertation, Technische Universitat Munchen.

[8] P. Mudjirahardjo, J.K. Tan, H. Kim, S. Ishikawa. 2013. Abnormal Motion Detection in an Occlusive Environment. Proceeding of SICE Annual Conference, Nagoya-Japan. pp. 1398-1402.

[9] P. Mudjirahardjo, J.K. Tan, H. Kim, S. Ishikawa. 2015. Temporal Analysis for Fast Motion Detection in a Crowd. Journal Artificial Life and Robotics. 20: 56-63.