Computers and Mathematics with Applications 64 (2012) 1291–1300

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

A novel color detection method based on *HSL* color space for robotic soccer competition

Shun-Hung Tsai *, Yu-Hsiang Tseng

Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, 10608, Taiwan, ROC
1, Sec. 3, Chung-hsiao E. Rd., Taipei, 10608, Taiwan, ROC

ARTICLE INFO

Keywords:
Vision system
Hue, saturation, and lightness (*HSL*)
Randomized circle detection (*RCD*)
Robotics soccer

ABSTRACT

In a robotic soccer competition, the locations of the ball and the robot are recognized through the vision system in a strong dynamic environment; therefore, the robot and the computer need to identify the object in real-time and accurately through the vision system. Due to the influence of the game ball under different brightnesses, the color clusters of glossy materials will produce a nonlinear behavior and will cause difficulty of recognition. Thus, we propose a novel color detection method based on hue, saturation, and lightness (*HSL*) color space in this paper. Firstly, by utilizing the hue histogram, the hue and saturation (*HS*) plane and the hue and lightness (*HL*) plane can be obtained respectively. In addition, based on the obtained information from two planes, the databases can be built for decreasing the running-time and increasing the recognition success rate of identifying object. Moreover, a novel algorithm is presented for the game ball in the robotic soccer competition. Finally, some examples under different brightnesses are illustrated to demonstrate the preciseness and the accuracy of the proposed method are better than the traditional method.

1. Introduction

Recently, the robotic soccer competition has become one of the most important robotic competitions. The robotic soccer competition provides a standardized competition platform for the integration of artificial intelligence. The platform can be regarded as a strong dynamic environment where robots can communicate, consult and cooperate with one another when competing in games. Based on this competition platform, many recent applications and studies can be demonstrated and implemented. Therefore, there are many robotic soccer competitions held every year, such as FIRA, MiroSot and AndroSot [1]. The vision system server is a base of robots' decision-making and motion-controlling [2,3]. Thus, if the vision system can work quickly and effectively, the robot will move smoothly and the algorithm will work well. For this reason, a successful vision system has to possess the characteristics of identifying objects in real-time and accurately [4–6].

In robotic soccer competition, the table-tennis balls and golf balls are commonly used as the game ball. Since the material of game ball is a glossy material, the glossy surface is easily affected by brightness. This problem may leads to image distortion and also affect the mis-recognition. In the FIRA, MiroSot and AndroSot games, each robot is set the color tag on the top. In general, the color tag is composed of two parts; one is team color and the other is the robot color. The color tag is usually used to identify the locations of every robot [7,8]. In a 3 versus 3 robotic soccer competition, there are at least six colors that need be identified. Therefore, if the color definition with a big range is set, it will result in the overlap phenomenon between the two similar colors.

* Corresponding author at: Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, 10608, Taiwan, ROC.
E-mail address: shtsai@ntut.edu.tw (S.-H. Tsai).

0898-1221/$ – see front matter
doi:10.1016/j.camwa.2012.03.073

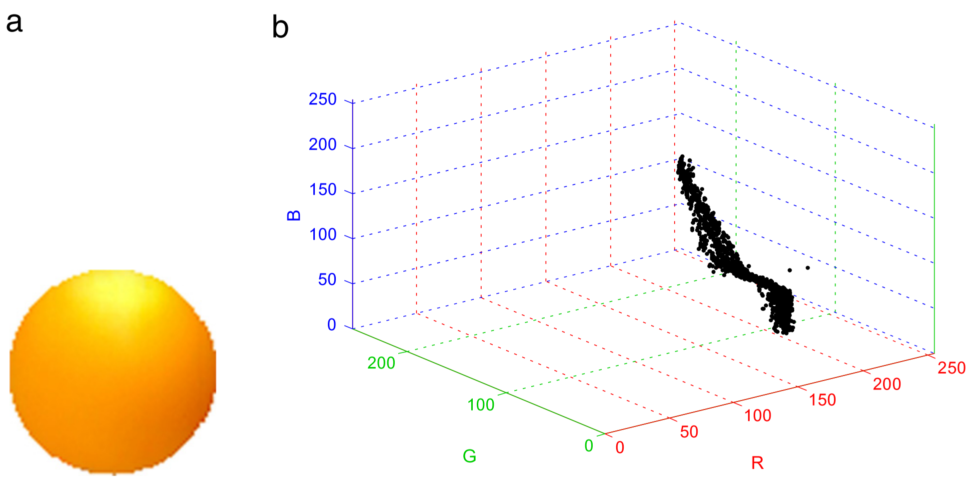

Fig. 1. The distribution of the color clusters in *RGB* color space. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The distribution of color cluster presents a linear model in *RGB* color space under different brightness [9–11], where *R* is red color, *G* is green color and *B* is blue color. Unfortunately, the color clusters of glossy materials under different brightnesses will vary with a non-linear trend [12,13]. In other words, in the image pixels of the same color with differential brightnesses, the pixels values of *R, G* and *B* won't be changed with a linear trend in the *RGB* color space under the different brightnesses. Therefore, if we install a light source on the top of the game ball, then a part of game ball will be brighter than the other parts, as shown in Fig. 1(a). Fig. 1(b) shows the distribution of *RGB* color space. Obviously, the color clusters of glossy materials show a non-linear trend in *RGB* color space under different brightness.

In general, the images are captured in terms of the *RGB* color space, but the *RGB* color space is seldom used for machine vision. That is because the brightness information from *RGB* color space cannot be obtained. Fortunately, through some coordinate transforms, *RGB* color space can be transformed into other color spaces such as, *YCbCr*, *YUV*, *YIQ*, *HSI*, *HSL* and so on [14–16], which have been built to improve the shortcomings of *RGB* color space. Thus, we can adopt these color spaces for further analysis under different brightnesses.

According to the above discussions, we propose a pre-processing of images to modify the problem of the object under different brightness in this paper. The main contributions of this paper are highlighted as follows: (i) a color detection method is presented to improve the distortion colors from glossy materials under different brightness; (ii) an algorithm is propounded to increase the accuracy and the preciseness of the similar colors; (iii) a database is constructed to reduce the execution-time of the follow procedures; (iv) a design procedure is proposed to enhance the success rate of the follow-up identifying.

The paper is organized as follows. In Section 2, the proposed color detection method is introduced. In Section 3, a novel circle detection algorithm is propounded. The experiment result is shown in Section 4. Finally, the conclusions are given in Section 5.

2. The color detection

Generally, if one wants to detect colors, then one has to transform *RGB* color space into the other color spaces with different brightness information. In this paper we adopt the *HSL* color space, where *H* is hue which means the color name, such as yellow, red, and the range lies in [0°, 360°], *S* is saturation which means the color of purity, the range is [0, 100], *L* represents the lightness and the range is [0, 100], as shown in Fig. 2.

2.1. Traditional color detection method

In this subsection, we will review the traditional color detection method based on *HSL* color space. In *HSL* color space, H, S and L represent three different dimensions respectively, and each dimension comprises two threshold values (maximum and minimum values), as shown in Fig. 3(a). Since six threshold values are set, a cube can be built in the *HSL* color space, shown in Fig. 3(b). Therefore, if the undefined color pixel falls in the cube, then one can declare the pixel belongs to this color [17].

A glossy-surfaced object under the different brightness will cause a non-linear variation in the color-cluster distribution. From Fig. 4, we can know that even though the *RGB* color space is transformed into *HSL* space, the distribution of the color clusters still vary with a non-linear trend. It is well known that the non-linear variation may cause a huge error by using

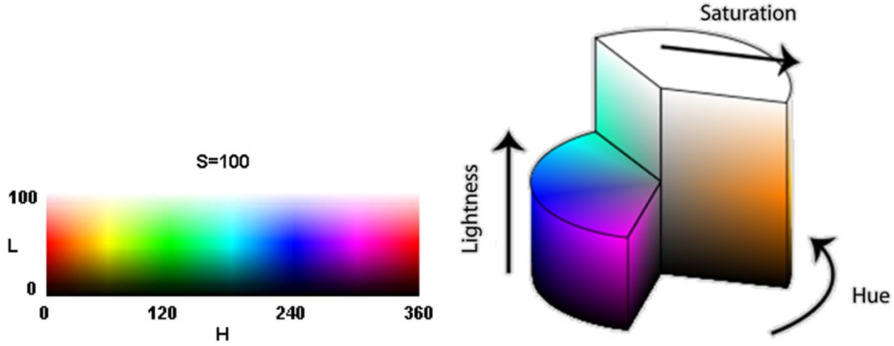

Fig. 2. *HSL* color model. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

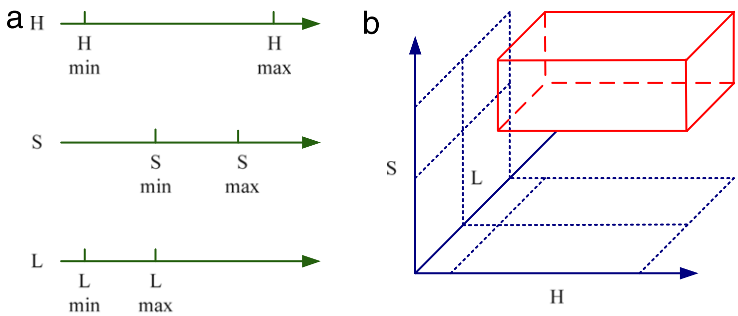

Fig. 3. Traditional color detection method.

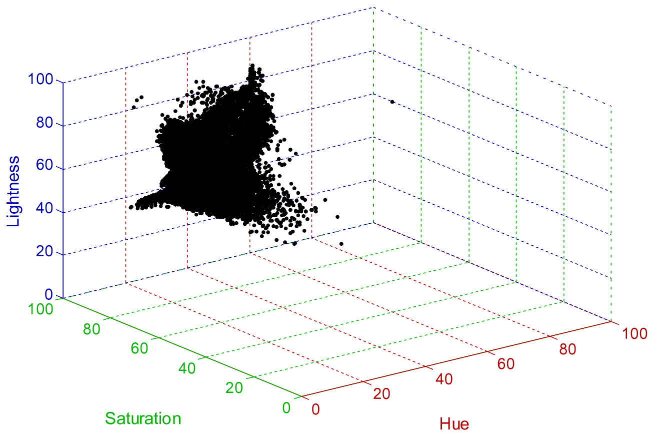

Fig. 4. The distribution of the color clusters in *HSL* color space. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the cube method to detect color. In addition, these errors will also cause the overlap phenomenon of color definitions in different brightness. Furthermore, it will cause the recognition difficulty for two similar colors under different brightness.

2.2. The construction of data base

For solving the shortcoming of traditional color detection method, some images under different brightness have to be captured firstly for observing intensity trend. In this subsection, we utilize the table-tennis ball images under different brightness to describe the process of the database construction, shown in Fig. 5.

At first, we select several ball images under different brightnesses, and then we can obtain the information for the pixel number per hue, shown in Fig. 6. Fig. 6 shows that most of the color distributions lie in the [25°, 65°].

Fig. 5. Table-tennis ball image under different brightness.

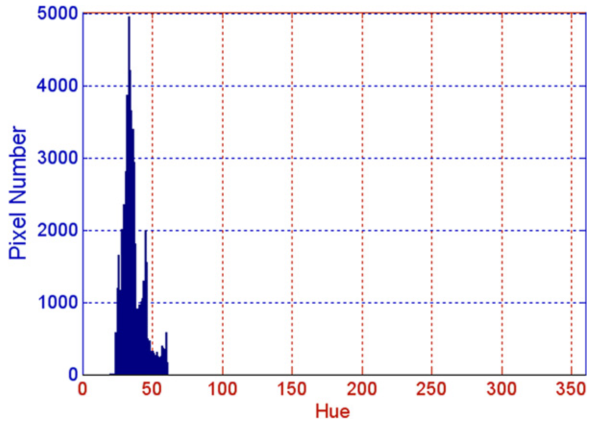

Fig. 6. The hue histogram of the table-tennis ball image under different brightnesses.

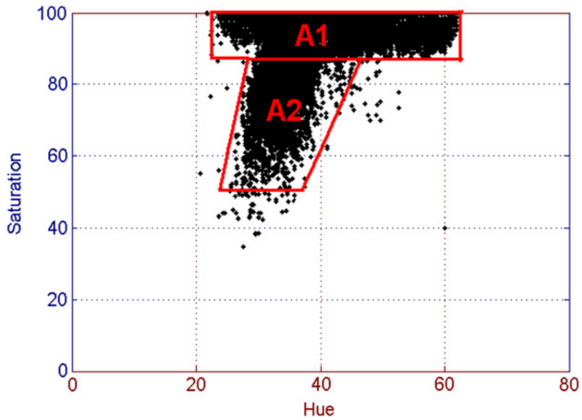

Fig. 7. The color distribution pixel of game ball in *HS* plane.

The recognition success rate of the ball color in an image usually is influenced by different saturations and intensities (lightness) of the image. Hence, the distributions of intensity, saturation and hue need also be defined for obtaining more precise color detection. Fig. 7 shows the color distribution of the saturation region for the table-tennis ball in *HS* plane. From Fig. 7, we can find that the most distributions are composed of two clusters. If we divide the two clusters by utilizing some linear inequalities, shown in (1) and (2), then we can obtain two regions, i.e. **A1** and **A2**.

$$\mathbf{A1} = \begin{cases} H > 22, \quad H < 63 \\ S > 85, \quad S \leq 100 \end{cases} \tag{1}$$

$$\mathbf{A2} = \begin{cases} S > 50, \quad S \leq 85 \\ S - 4.5H < -35 \\ S - 4H > -90. \end{cases} \tag{2}$$

Similarly, the color recognition is also influenced by intensity, so table-tennis balls image in picture will show different colors under different brightnesses. Under the circumstances, the intensity distribution also needs to be defined. Fig. 8 shows the ball color distribution in the *HL* plane. The distribution clusters can be divided into **B1**, **B2**, and then the regions can be

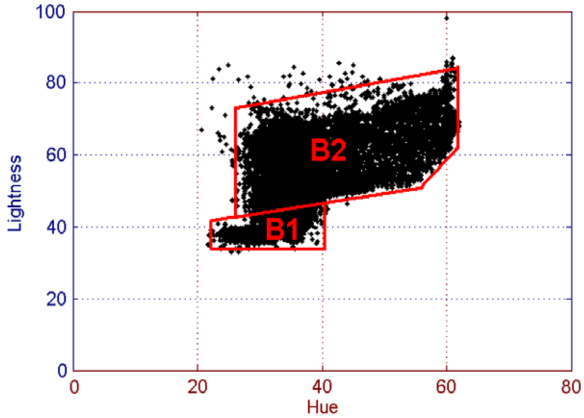

Fig. 8. The color distribution pixel of game ball in *HL* plane.

defined by utilizing these linear inequalities, as shown in Eqs. (3), (4).

$$\mathbf{B1} = \begin{cases} H > 22, \quad H < 40 \\ L > 32 \\ L - 0.22H \leq 35.56 \end{cases} \tag{3}$$

$$\mathbf{B2} = \begin{cases} H > 25, \quad H < 63 \\ L - 0.22H > 35.56 \\ L - 2H > -60 \\ L - 0.43H < 59.2. \end{cases} \tag{4}$$

Finally, the data satisfy the Eqs. (1)–(4) which can be regarded as the color database. Therefore, if the pixel data satisfy these linear inequalities, then the pixel can be easily determined.

3. A novel image processing algorithm

In this section, we will introduce a novel image processing algorithm, which will be adopted to identify the game ball image from the picture. Before discussing the proposed image algorithm, we will introduce the *RCD* algorithm [18–20] briefly. Firstly, the three non-collinear edge points $p_1(x_1, y_1), p_2(x_2, y_2),\ p_3(x_3, y_3)$ are chosen randomly from the image, and the three points which can be represented in candidate circle form, follow (5a)–(5c).

$$a_{123} = \frac{\begin{vmatrix} x_2^2 + y_2^2 - (x_1^2 + y_1^2) & 2(y_2 - y_1) \\ x_3^2 + y_3^2 - (x_1^2 + y_1^2) & 2(y_3 - y_1) \end{vmatrix}}{4((x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1))} \tag{5a}$$

$$b_{123} = \frac{\begin{vmatrix} 2(x_2 - x_1) & x_2^2 + y_2^2 - (x_1^2 + y_1^2) \\ 2(x_3 - x_1) & x_3^2 + y_3^2 - (x_1^2 + y_1^2) \end{vmatrix}}{4((x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1))} \tag{5b}$$

$$r_{123} = \sqrt{(x_i - a_{123})^2 + (y_i - b_{123})^2} \tag{5c}$$

where a_{123} and b_{123} represent the center point of the circle, and r_{123} is the radius. Secondly, calculate the distance between the circumference and the other edge points repeatedly. If the distance is smaller than the designate threshold value T_d, then the vote number of the possible circle will be added one. If the vote results of the circle exceed a certain ratio threshold T_r, then this possible circle will be regarded as a real circle, otherwise, the number of failures will be added one. Thus, we can declare that the threshold T_r and T_d can be considered as the sensitivity of *RCD*. By contrast, another threshold T_f can be set, as the numbers of failures reach the threshold, and then it means that there is no circle in this image. Thus, by utilizing the *RCD* algorithm, we can easily identify whether there exists a circle in the image or not.

Fig. 9 shows the complete flowchart of the proposed color detection method. Firstly, we obtain the input image from the *RGB* color space, and then transform the *RGB* color space into the *HSL* color space. Secondly, color detection is started and it is compared with the color database. If the results obtained from the color detection lie in the color database, then the procedures of the opening and edge detection will be carried out. Otherwise, the original image will be outputted. After processing the edge detection procedure, the procedure of the circle detection will be executed. If the result is fail then the threshold value will be reset and re-execute the *RCD* algorithm. Here, an error threshold is set. If the reset threshold is reached then the program will be stopped. After processing a series of procedures, the location and size of the game ball can be identified from the original image, as shown in Fig. 10(a)–(d).

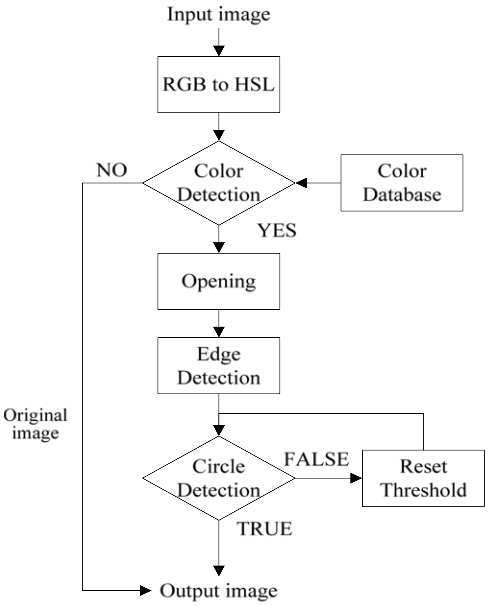

Fig. 9. The procedure of the ball recognition.

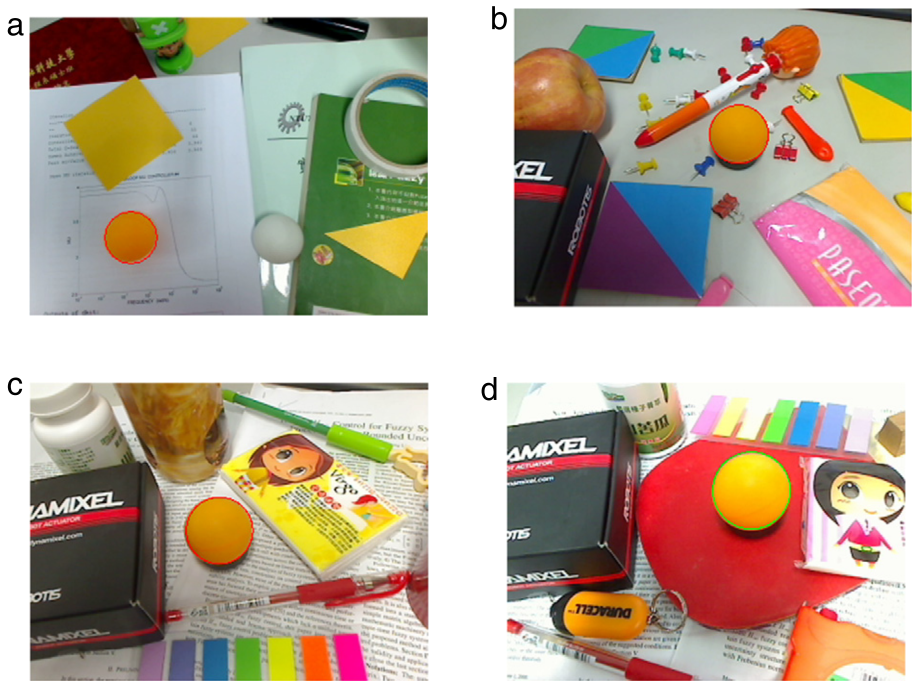

Fig. 10. The results by processing *RCD* algorithm.

4. Experiment results

In this section, some experiments will be illustrated to demonstrate the effectiveness and the feasibility of our proposed approach. In this section, all experiments are implemented by using the Pentium D personal computer with 3.4 GHz and the C programming language.

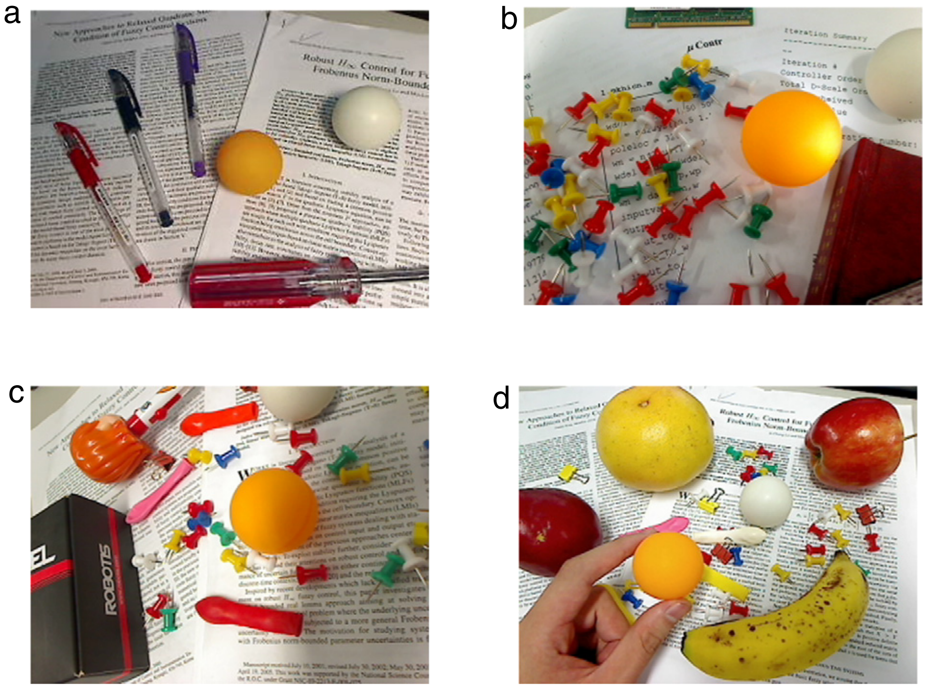

Fig. 11. Four different interference environments.

4.1. Comparison of the color detection

In this subsection, we will compare the traditional color detection method and the proposed method for four 320×240 pictures, shown in Fig. 11(a)–(d). There is a game ball image in Fig. 11(a) under the normal background. In Fig. 11(b), there are some partial images of game ball under high brightness and there are some similar color objects in this picture. In Fig. 11(c), the picture comprises the game ball image in the shadow and some similar color objects. In Fig. 11(d), there are many similar color objects overlap with the game ball image under a high brightness circumstance. After processing the color detecting procedure, these edge points will be used to detect the circle; therefore, the integrity of circle edge will affect execution-time and accuracy of circle detection directly. For this reason, we have to execute the color detection, opening operation and Sobel edge detection method for these images.

Firstly, we adopt the traditional color detection method for the Fig. 11, and the 6 threshold values are set as $25 < H < 60$, $50 < S \leq 100$ and $35 < L < 75$. Fig. 12 shows the experimental results by processing the traditional method. From Fig. 12(b)–(d), we can find that the traditional method cannot discriminate the similar colors clearly, such as the yellow thumbtack and fruits. Besides, in the Fig. 12(b) and (c), we can find that the detected circle is incomplete, which means the traditional method cannot detect the distortion color under the high and low brightness environments. For solving the similar colors problem, we can narrow the range of the threshold values, but the incomplete circle problem will become more aggravated. In contrast, if we loosen the range of threshold values for avoiding the incomplete circle problem, the similar colors problem will be highlighted.

Fig. 13 shows the results of four different images by using the proposed method. Fig. 13(b) and (c) show that the proposed method can detect the distortion color accurately under the higher and lower brightness. In addition, for an image with many noises, as shown in Fig. 13(d), the proposed method still can accurately distinguish the desired color. From the results of Figs. 12(a) and 13(a), we can find that the two color detections methods do not have a conspicuous difference under a low interference environment or normal brightness. However, the proposed method can provide a better performance under many interference environments than the traditional color detection method, and the results are shown in Figs. 12(d) and 13(d).

4.2. The comparisons of recognition success rate and the execution-time

In this experiment, we will compare the recognition success rate and the execution-time for the same pictures with the proposed method and the traditional method. The recognition success rate is defined as: "the game ball image can be recognized from the image after processing the *RCD* algorithm", and we execute the program 100 times for the four pictures in this paper. In addition to recognition success rate, we also explored the execution-time problem in this paper. In this

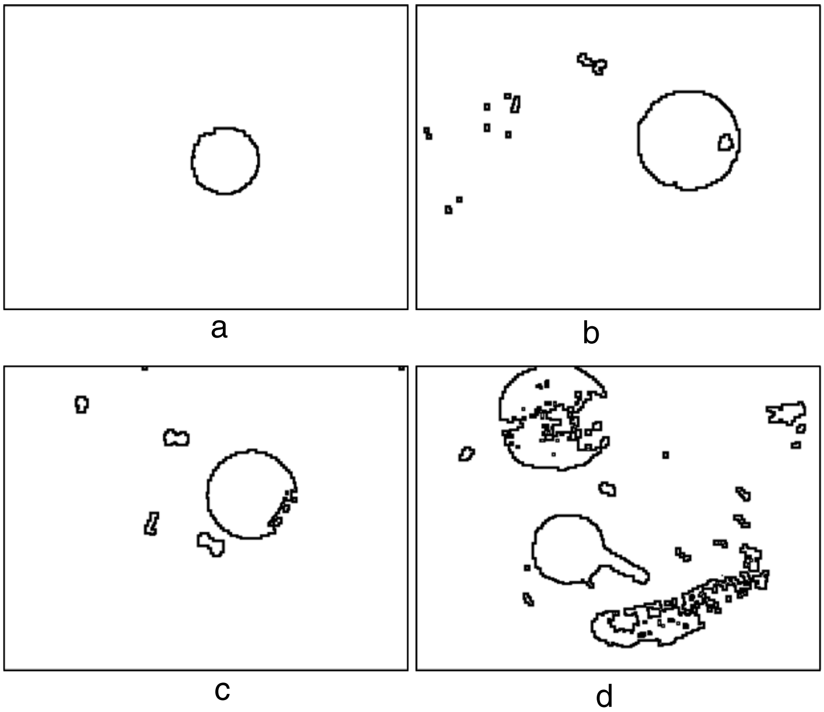

Fig. 12. The process results by adopting the traditional method for Fig. 11.

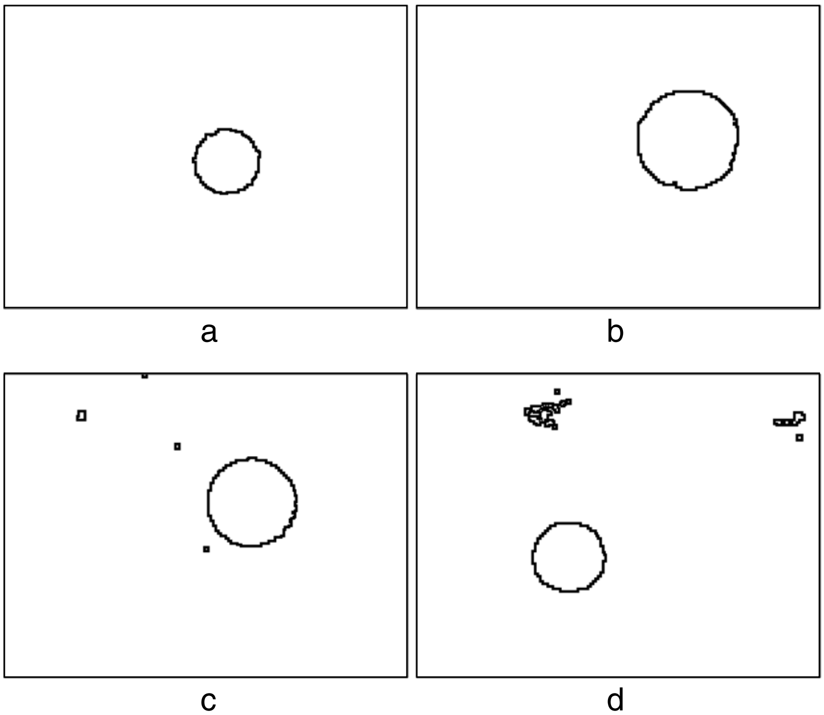

Fig. 13. The process results by adopting our proposed method for Fig. 11.

experiment, the execution-time is defined as: ''carrying out the location and size identifications of the game ball from an original image'', and its measured unit is milliseconds.

While executing the *RCD* algorithm for the four images, the thresholds T_f, T_d, and T_r are set as 8000, 2, and 70% respectively. The thresholds T_f denote the number of toleration failures. When it is set larger, it might not miss identifying

Table 1
The recognition success rate comparison.

	Fig. 11(a) (%)	Fig. 11(b) (%)	Fig. 11(c) (%)	Fig. 11(d) (%)
Traditional method	100	99	95	76
Proposed method	100	100	99	99

Table 2
Execution-time comparison.

	Fig. 11(a)	Fig. 11(b)	Fig. 11(c)	Fig. 11(d)
Traditional method	15.61	20.76	29.43	141.58
Proposed method	15.57	16.34	17.91	19.25
Improvement rate	0.0026	0.2129	0.3914	0.8640

Improvement rate = (traditional method − proposed method)/traditional method.

the possible circle. On the contrary, it will need more execution-time when there is no circle in an image. The thresholds T_d and T_r can be regarded as the sensitivity of the detection. When these threshold values are tighter, it can identify the location and size of the circle with more accuracy. However, this movement will increase the execution-time of image detection.

Table 1 shows the comparison results between the traditional method and the proposed method. From Table 1, we can find that the proposed method has a better recognition success rate than the tradition method, and this situation is especially obvious for the image under brighter light. From the three columns (Fig. 11(a)–(c)) of Table 1, we can find the recognition success rates are almost the same in the low interference environment, and their gap of recognition success rate is less then 5%. But in a high interference environment (Fig. 11(d)), the recognition success rate gap increases to 23%. The main reason is that there is more noise overlap with the circle edge, which will cause difficulty in recognition. From the above discussions, we can find that when the noise increases, the recognition success rate will decrease.

In the Table 2, one can clearly know that the traditional method and the proposed method have almost the same execution-time under a low interference environment. However, in a high interference environment, such as Fig. 11(d), the execution-time improvement rate is up to 86.4%. Table 2 shows that our proposed method indeed can improve the recognition success rate and shorten the execution-time.

5. Conclusion

In this paper, a novel color detection method based on *HSL* color space is proposed. The proposed method can identify colors more accurately from glossy materials under different brightnesses. Besides, an algorithm is proposed for decreasing the recognition failure rate of the traditional method. Moreover, some experiments under different brightness environments are illustrated to demonstrate that the proposed method can decrease the execution-time of recognizing an object from images and improve the recognition success rate.

Acknowledgments

This work was supported by the National Science Council of Taiwan, R.O.C., under grant NSC-99-2221-E-027-101- and NSC-100-2221-E-027-017.

References

[1] Official homepage of the FIRA robot soccer tournaments. Available online at: http://www.fira.net/.
[2] G. Novak, R. Springer, An introduction to a vision system used for a MiroSOT robot soccer system, in: Proceedings of the International Conference on Computational Cybernetics, 2004, pp. 101–108.
[3] J. Zhou, H.-S. Zhang, Y. Chen, C.-Z. Zhang, Design of vision system and recognition algorithm in MiroSOT, in: Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems, 2004, pp. 153–157.
[4] N. Razmjooy, B.S. Mousavi, F. Soleymani, A real-time mathematical computer method for potato inspection using machine vision, Computers and Mathematics with Applications 63 (1) (2012) 268–279.
[5] K.-Y. Huang, Detection and classification of areca nuts with machine vision, Computers and Mathematics with Applications (2011) http://dx.doi.org/10.1016/j.camwa.2011.11.041.
[6] S. Bettahar, A.B. Stambouli, P. Lambert, Numerical scheme for efficient colour image denoising, Computers and Mathematics with Applications 61 (9) (2011) 2903–2913.
[7] C.-N. Shyi, J.-Y. Lee, K.-C. Hung, C.-H. Chen, A fast method for robot location determination, Computers and Mathematics with Applications 18 (9) (1989) 771–777.
[8] Y.-H. Lee, B. Kim, H.-J. Kim, Efficient object identification and localization for image retrieval using query-by-region, Computers and Mathematics with Applications 63 (2) (2012) 511–517.
[9] H. Ragheb, E.R. Hancock, Lambertian reflectance correction for rough and shiny surfaces, in: Proceedings of the International Conference on Image Processing, 2002, pp. 553–556.

[10] R. Basri, D.W. Jacobs, Lambertian reflectance and linear subspaces, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (2) (2003) 218–233.
[11] R. Kumar, A. Barmpoutis, A. Banerjee, B. Vemuri, Non-Lambertian reflectance modeling and shape recovery for faces using tensor splines, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (3) (2011) 533–567.
[12] G. Klinker, S. Shafer, T. Kanade, A physical approach to color image understanding, International Journal of Computer Vision 4 (1) (1991) 7–38.
[13] J.H. Park, G.S. Lee, S.Y. Park, Color image segmentation using adaptive mean shift and statistical model-based methods, Computers and Mathematics with Applications 57 (6) (2009) 970–980.
[14] S. Peng, Y. Wen, Research based on the HSV humanoid robot soccer image processing, in: Proceeding of the International Conference on Communication Systems, 2010, pp. 52–55.
[15] B. Ahirwal, M. Khadtare, R. Mehta, FPGA based system for color space transformation RGB to YIQ and YCbCr, in: Proceeding of the International Conference on Intelligent and Advanced Systems, 2007, pp. 1345–1349.
[16] H. Ren, Q. Zhong, A new image segmentation method based on HSI color space for biped soccer robot, in: Proceeding of the International Symposium on IT in Medicine and Education, 2008, pp. 1058–1061.
[17] R. Jain, R. Kasturi, B.G. Schunck, Machine Vision, McGraw-Hill, 1995.
[18] T.-C. Chen, K.-L. Chung, An efficient randomized algorithm for detecting circles, Computer Vision and Image Understanding 83 (2) (2001) 172–191.
[19] B. Lamiroy, L. Fritz, O. Gaucher, Robust circle detection, in: Proceeding of the International Conference on Document Analysis and Recognition, 2007, pp. 526–530.
[20] K.-L. Chung, Y.-H. Huang, Speed up the computation of randomized algorithms for detecting lines, circles, and ellipses using novel tuning and LUT-based voting platform, Applied Mathematics and Computation 190 (1) (2007) 132–149.