

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



---

# An Adaptive Resolution Method Using Discrete Wavelet Transform for Humanoid Robot Vision System

---

Chih-Hsien Hsia, Wei-Hsuan Chang and  
Jen-Shiun Chiang

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/55114>

---

## 1. Introduction

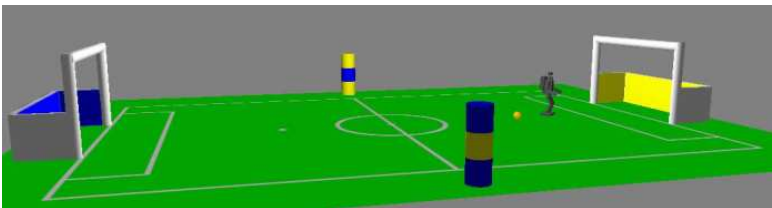
The RoboCup (Kitano et al., 1995) is an international joint project to stimulate research efforts in the field of artificial intelligence, robotics, and related fields. According to the rules for the 2009 RoboCup, in the league for kid-sized robots (Avalable, 2009), the competitions were to take place on a rectangular field with an area of  $600 \times 400 \text{ cm}^2$  containing two goals and two landmark poles, as shown in Fig. 1. A goal was placed in the middle of each goal line, with one of the goals colored yellow and the other colored blue. As shown in Fig. 2, each goal for the kid-sized robot field had a crossbar height of 90 cm, a goal wall height of 40 cm, a goal wall width of 150 cm, and a 50 cm depth for the goal wall. The two landmark poles were placed on each side of the two intersection points between the touch line and the middle field line. The landmark pole was a cylinder with a diameter of 20 cm. It consisted of three segments, each 20 cm in height, stacked on top of each other. The lowest and the highest segments have the same color as the goal on its left side, as shown in Fig. 3. The ball is the standard size orange tennis ball. All of the above objects are the most critical characteristics in the field, and they are also the key features which we have to pay attention to.

The functions of humanoid robot vision system include image capturing, image analyses, and digital image processing by using visual sensors. For digital image processing, it is to transform the image into the analyzable digital pattern by digital signal processing. We can further use image analysis techniques to describe and recognize the image content for the robot vision. The robot vision system can use the environment information captured in front of the robot to recognize the image by means of the technique of human vision system. An object recognition algorithm is thus proposed to the humanoid robot soccer competition.

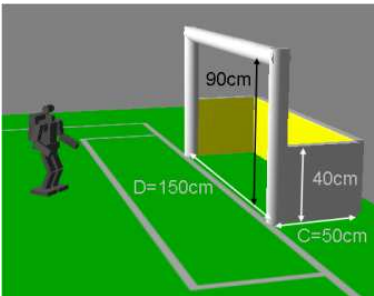
Generally speaking, object recognition uses object features to extract the object out of the picture frame, and thus shape (Chaumette, 1994) and (Jean & Wu, 2004), contour (Sun et al, 2003), (Kass et al., 1988), and (Canny, 1986), color (Herodotou et al., 1998) and (Ikeda, 2003), texture, and sizes of object features are commonly used. It is important to extract the information in real-time because the moving ball is one of the most critical object in the contest field. The complex feature such as contour is not suited to recognize in our application. The objects don't have the obvious texture which is not suited to use in the contest field. However the object color is distinctive in the contest field, we mainly choose the color information to determine the critical objects.

Although this approach is simple, the real-time efficiency is still low. Because there is a lot of information to be processed in every frame for real-time consideration, Sugandi et al. (Sugandi et al, 2009) proposed a low resolution method to reduce the information. It can speed up the processing time, but the low resolution results in a shorter recognizable distance and it may increase the false recognition rate. In order to improve the mentioned drawbacks, we propose a new approach, adaptive resolution method (ARM), to reduce the computation complexity and increase the accuracy rate.

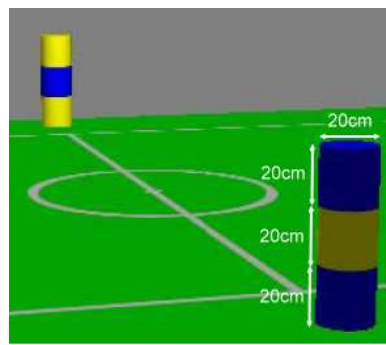
The rest of this study is organized as follows. Section 2 presents the related background such as the general color based object recognition method, low resolution method, and encountered problems. Section 3 describes the proposed approach, ARM. The experimental results are shown in Section 4. Finally, the conclusions are outlined in Section 5.



**Figure 1.** The field for the competitions.



**Figure 2.** The goal information.

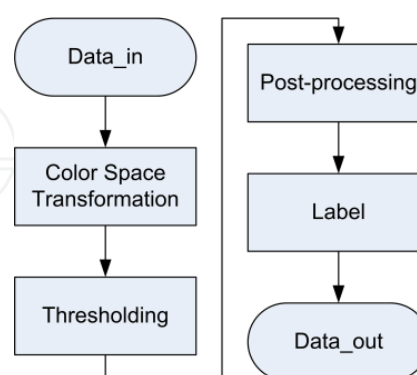


**Figure 3.** The landmark information.

## 2. Background

### 2.1. Color based object recognition method

An efficient vision system plays an important role for the humanoid robot soccer players. Many robot vision modules have provided some basic color information, and it can extract the object by selecting the color threshold. The flow chart of a traditional color recognition method is shown in Fig. 4. The RGB color model comes from the three additive primary colors, red, green, and blue. The main purpose of the RGB color model (Gonzalez & Woods, 2001) is for the sensing, representation, and display of images in electronic systems, such as televisions and computers, and it is the basic image information format. The X, Y, and Z axes represent the red, green, and blue color components respectively, and it can describe all colors by different proportion combinations. Because the RGB color model is not explicit, it can be easily influenced by the light illumination and make people select error threshold values.



**Figure 4.** The flow chart of the traditional color recognition method.

An HSV (HSV stands for hue, saturation, and value) color model relates the representations of pixels in the RGB color space, which attempts to describe perceptual color relationships more accurately than RGB. Because the HSV color model describes the color and brightness

component respectively, the HSV color model is not easily influenced by the light illumination. The HSV color model is therefore extensively used in the fields of color recognition. The HSV transform function is shown in eqs. (1)-(3) as follows:

$$H = \begin{cases} \left(6 + \frac{G - B}{\max - \min}\right) \times 60^\circ, & \text{if } R = \max \\ \left(2 + \frac{B - R}{\max - \min}\right) \times 60^\circ, & \text{if } G = \max \\ \left(4 + \frac{R - G}{\max - \min}\right) \times 60^\circ, & \text{if } B = \max \end{cases} \quad (1)$$

$$S = \begin{cases} 0, & \text{if } \max = 0 \\ \frac{\max - \min}{\max}, & \text{otherwise} \end{cases} \quad (2)$$

$$V = \max \quad (3)$$

In (1), (2), and (3), the range of  $H$ , hue, is  $0^\circ \sim 360^\circ$ ; the range of  $S$ , is  $0 \sim 1$ , and the range of  $V$ , value, is  $0 \sim 255$ . The RGB values are confined by (4):

$$\begin{aligned} \max &= \text{MAX}(R, G, B) \\ \min &= \text{MIN}(R, G, B) \end{aligned} \quad (4)$$

where “max” indicates the maximum value in the RGB color components and “min” indicates the minimum value in the RGB color components. Hence, we can directly make use of  $H$  and  $S$  to describe a color range of high environmental tolerance. It can help us to obtain the foreground objects mask,  $M(x, y)$ , by the threshold value selection as shown in (5).

$$M(x, y) = \begin{cases} 1, & \text{if } T_{H1} < H(x, y) < T_{H2} \cap S(x, y) > T_s \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where  $T_{H1}$ ,  $T_{H2}$ , and  $T_s$  are the thresholds of hue and threshold of saturation by manual setting. The foreground object mask usually accompanies with the noise, and we can remove the noise by the simple morphological methods, such as dilation, erosion, opening, and closing. It needs to separate the objects by labeling when many objects with the same colors are existed in the frame. The following procedure is the flow for labeling (Gonzalez & Woods, 2001):

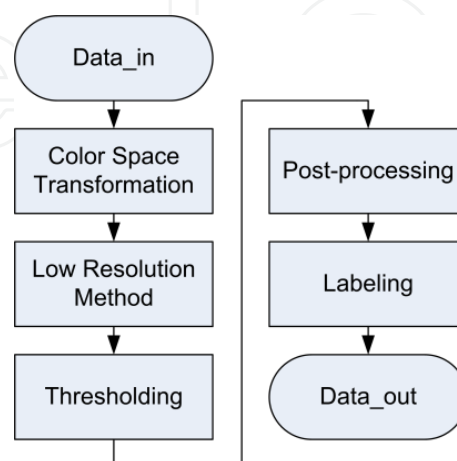
Step 1: Scan the threshold image  $M(x, y)$ ; Step 2: Give the value  $Label_{color}^i$  to the connected component  $Q\{n\}$  of pixel  $(x, y)$ ; Step 3: Give the same value to the connected component of  $Q\{n\}$ ;

Step 4: Until no connected component can be found; Step 5: Update,  $i = i+1$ . Then go to Step 1 and repeat Steps 2~4; Step 6: Completely scan the image.

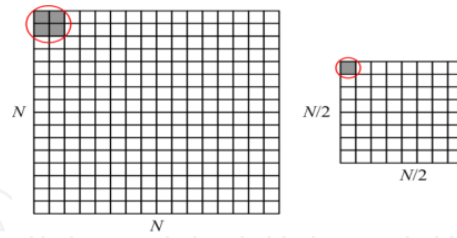
By using the above-mentioned procedure, the objects can be extracted. Although this method is simple, it is only suitable for low frame rate sequences. For a high resolution or noisy sequence, this approach may need very high computation complexity.

## 2.2. Low resolution method

To overcome the above-mentioned problems, several approaches of low resolution method were proposed (Sugandi et al., 2009), (Cheng & Chen et al., 2006). The flow chart of a general low resolution method is shown in Fig. 5. Several low resolution methods, such as the approach of applying 2-D discrete wavelet transform (DWT) and the using of  $2 \times 2$  average filter (AF), were discussed. (Cheng & Chen, 2006) applied the 2-D DWT for detecting and tracking moving objects and only the  $LL_3$ -band image is used for detecting motion of the moving object (It is suggested that the  $LL_3$ -band is a good candidate for noise elimination (the user can choose a suited decomposition level according to the requirement, and actually there is no need to do the reconstruction for these applications). Because noises are preserved in high-frequency, it can reduce computing cost for post-processing by using the  $LL_3$ -band image. This method can be used for coping with noise or fake motion effectively, however the conventional DWT scheme has the disadvantages of complicated calculation when an original image is decomposed into the  $LL$ -band image. Moreover if it uses an  $LL_3$ -band image to deal with the fake motion, it may cause incomplete moving object detecting regions. In (Sugandi et al., 2009) proposed a simple method by using the low resolution concept to deal with the fake motion such as moving leaves of trees. The low resolution image is generated by replacing each pixel value of an original image with the average value of its four neighbor pixels and itself as shown in Fig. 6. It also provides a flexible multi-resolution image like the DWT. Nevertheless, the low resolution images generated by using the  $2 \times 2$  AF method are more blurred than that by using the DWT method. It may reduce the preciseness of post-processing (such as object detection, tracking, and object identification), because the post-processing depends on the correct location of the moving object detecting and accuracy moving object.



**Figure 5.** The flow chart of a general low resolution method.



**Figure 6.** Diagram of the 2x2 AF method

In order to detect and track the moving object more accurately, we propose a new approach, adaptive resolution method (ARM), which is based on the 2-D integer symmetric mask-based discrete wavelet transform (SMDWT) (Hsia et al, 2009). It does not only retain the features of the flexibilities for multi-resolution, but also does not cause high computing cost when using it for finding different subband images. In addition, it preserves more image quality of the low resolution image than that of the average filter approach (Sugandi et al., 2009).

### 2.2.1. Symmetric Mask-Based Discrete Wavelet Transform (SMDWT)

In 2-D DWT, the computation needs large transpose memory and has a long critical path. On the other hand SMDWT has many advanced features such as short critical path, high speed operation, regular signal coding, and independent subband processing (Hsia et al, 2009). The derivation coefficient of the 2-D SMDWT is based on the 2-D 5/3 integer lifting-based DWT. For computation speed and simplicity considerations, four-masks,  $3 \times 3$ ,  $5 \times 3$ ,  $3 \times 5$ , and  $5 \times 5$ , are used to perform spatial filtering tasks. Moreover, the four-subband processing can be further optimized to speed up and reduce the temporal memory of the DWT coefficients. The four-matrix processors consist of four-mask filters, and each filter is derived from one 2-D DWT of 5/3 integer lifting-based coefficients.

In the ARM approach, we can select only the LL-band mask of SMDWT (The moving object is low-frequency energy). Unlike the conventional DWT method to process row and column dimensions respectively by low-pass filter and down-sampling, the LL-mask band of SMDWT can be used to directly calculate the LL-band image. The matrix function of the LL-mask is shown in (6) and the coefficients of the LL-mask are shown in Fig. 7 (Hsia et al, 2009). SMDWT (using the LL-band mask only) can reduce the image transfer computing cost and remove the noise. Besides, this approach can have accurate object tracking for various types of occlusions.

$$\begin{aligned}
 LL(i, j) = & (9/16)x(2i, 2j) + (1/64)\sum_{u=0}^1 \sum_{v=0}^1 x(2i-2+4u, 2j-2+4v) + \\
 & + (1/16)\sum_{u=0}^1 \sum_{v=0}^1 x(2i-1+2u, 2j-1+2v) + (-1/32)\sum_{u=0}^1 \sum_{v=0}^1 x(2i-1+2u, 2j-2+4v) + \\
 & + (-1/32)\sum_{u=0}^1 \sum_{v=0}^1 x(2i-2+4u, 2j-1+2v) + (3/16)\sum_{u=0}^1 [x(2i-1+2u, 2j) + x(2i, 2j-1+2u)] + \\
 & + (-3/32)\sum_{u=0}^1 [x(2i-2+4u, 2j) + x(2i, 2j-2+4u)].
 \end{aligned} \tag{6}$$



|       |       |       |       |       |
|-------|-------|-------|-------|-------|
| 1/64  | -1/32 | -3/32 | -1/32 | 1/64  |
| -1/32 | 1/16  | 3/16  | 1/16  | -1/32 |
| -3/32 | 3/16  | 9/16  | 3/16  | -3/32 |
| -1/32 | 1/16  | 3/16  | 1/16  | -1/32 |
| 1/64  | -1/32 | -3/32 | -1/32 | 1/64  |

**Figure 7.** The subband masks coefficients of the LL-mask.

### 3. The proposed method

#### 3.1. Adaptive Resolution Method (ARM)

ARM takes advantage of the information obtained from the image to know the area of the ball and chooses the most suitable resolution. The operation flow chart is shown in Fig. 8. After HSV color transformation, ARM chooses the most proper resolution by the situation at this moment in time. The high resolution approach brings a longer recognizable distance but with a slower running speed. On the other hand, the low resolution approach brings a lower recognizable distance but with a faster running speed. When we got the area information of the ball from the image last time, we could convert it as the “sel” signal through the adaptive selector to choose the appropriate resolution. The “sel” condition is shown in (7):

$$\text{sel} = \begin{cases} 0(\text{original size}), & \text{if } 0 \leq A_{\text{ball}} < A_{\text{thd1}} \\ 1(1\text{-level SMDWT}), & \text{if } A_{\text{thd1}} \leq A_{\text{ball}} < A_{\text{thd2}} \\ 2(2\text{-level SMDWT}), & \text{if } A_{\text{ball}} \geq A_{\text{thd2}} \end{cases} \quad (7)$$

In (7),  $A_{\text{thd1}}$  and  $A_{\text{thd2}}$  are the threshold values for the area of ball. The relationship between the resolution and the distance of the ball is described in Table 1. According to Table 1, we can conclude that  $A_{\text{thd1}}$  and  $A_{\text{thd2}}$  are set to 54 and 413, respectively. The threshold selection is performed for each different resolution of working environment. The threshold value is used to produce the recognizable distance. If the ball disappears in the frame, the frame will change into the original size to have a higher probability to find out the ball. Since the sizes of other critical objects (such as goal and landmark) in the field are larger than the ball, they can be recognized easily. Fig. 9 shows the results of different resolutions after the HSV transformation.

#### 3.2. Sample object recognition method

According to the above-mentioned color segmentation method, it can fast and easily extract the orange ball in the field, but it is not enough to recognize the goals and landmarks. The colors of the goals and landmarks are yellow and blue, and by color segmentation the extraction of goals and landmarks may not be correct as shown in Fig. 10. Therefore we have to use



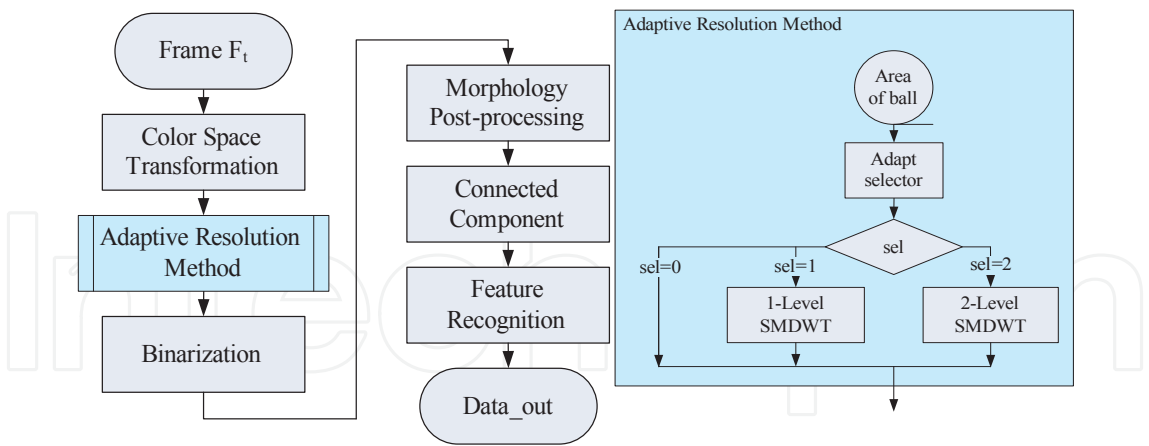


Figure 8. The flow chart of ARM.

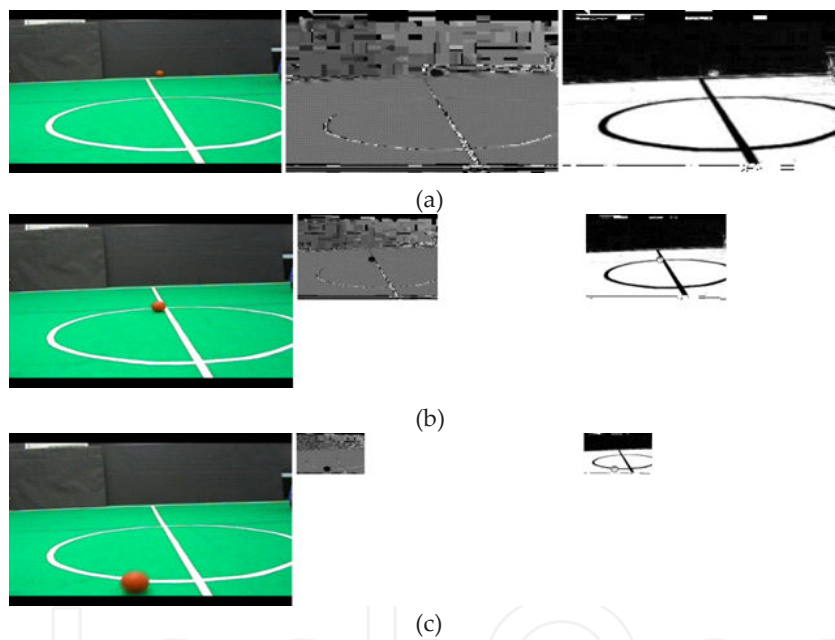


Figure 9. The results after the HSV transformation under the resolutions of video. (a) Recognizable max distance of ball in 320×240; (b) Recognizable max distance of ball in 160×120; (c) Recognizable max distance of ball in 80×60

more features and information to extract them. Since the contest field is not complicated, a simple recognition method can be used to reduce the computation complexity. The landmark is a cylinder with three colors. Let us look at one of the landmark with the upper and bottom layers in yellow, and the center layer in blue; this one is defined as the YBY-landmark. The diagram is shown in Fig. 11. The color combinations of the other one are in contrast of the previous one, and the landmark is defined as the BYB-landmark. The labels of the YBY-landmark can be calculated by (8). The BYB landmark is in the same manner as the YBY-landmark.

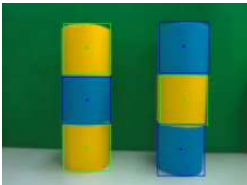
| Resolution | Level of DWT | Recognizable max distance of ball | *Area of ball |
|------------|--------------|-----------------------------------|---------------|
| 320×240    | 0(original)  | 404.6 cm                          | 18 pixels     |
| 160×120    | 1            | 322.8 cm                          | 54 pixels     |
| 80×60      | 2            | 121.3 cm                          | 413 pixels    |

\*The area means the pixel number in the original resolution.

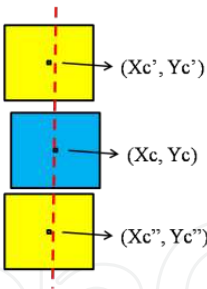
**Table 1.** The relationship between the resolution and the distance of the ball.

$$P_{YBY}(x,y)=L_Y^i(x,y)\cup L_Y^j(x,y)\cup L_B^k(x,y)$$
$$if\ \left\{ \left| L_Y^i(x_c)-L_Y^j(x_c) \right| < \beta_Y \right\} \cap \left\{ L_Y^i(y_{\max}) < L_B^k(y_c) < L_Y^j(y_{\min}) \right\}$$

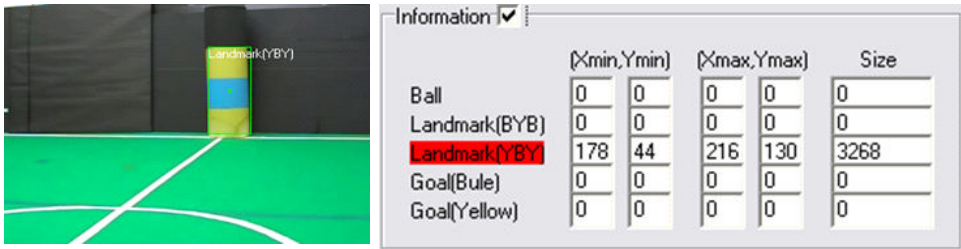
(8)



**Figure 10.** False segmentation of the landmark



**Figure 11.** The diagram of the landmark.



**Figure 12.** The result of landmark recognition.

According to the above-mentioned labeling procedure, we labeled all of the yellow and blue components in the frame and assigned the numbers to those components. Where  $L_Y^i$  is defined the pixels of the  $i$ -th yellow component (Y),  $y_{\min}$  and  $y_{\max}$  the minimum value and the maximum value for the object  $i$  at  $y$  direction respectively,  $x_c$  and  $y_c$  the center point of the object at the horizontal and vertical direction respectively. The vertical bias value  $\beta_Y$  is set as 15. The landmark is composed of two same color objects in the vertical line, and the center is in different color. If it can find an object with this feature, the system can treat this object as the landmark and outputs the frame coordinate data.

The result of landmark recognition is shown in Fig.12. Eq. (9) is used to define the label of the ball:

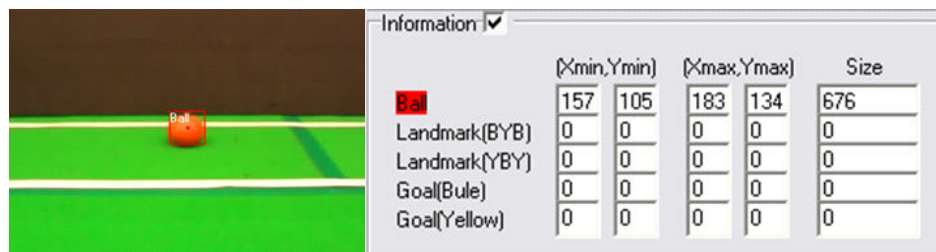
$$B(x, y) = L_O^s(x, y), \text{ if } \alpha_1 \leq \frac{L_O^s(x_{\max}) - L_O^s(x_{\min})}{L_O^s(y_{\max}) - L_O^s(y_{\min})} \leq \alpha_2 \cap A_s \text{ is the maximum} \quad (9)$$

where is the pixel of the  $s$ -th orange component in a frame. Since the ball is very small in the picture frame, in order to avoid noise, the ball is treated as the maximum orange object and with a shape ratio of height to width approximately equal to 1. Here  $\alpha_1$  and  $\alpha_2$  are set to 0.8 and 1.2, respectively. The result of ball recognition is shown in Fig. 13. The goal recognition is defined in (10).

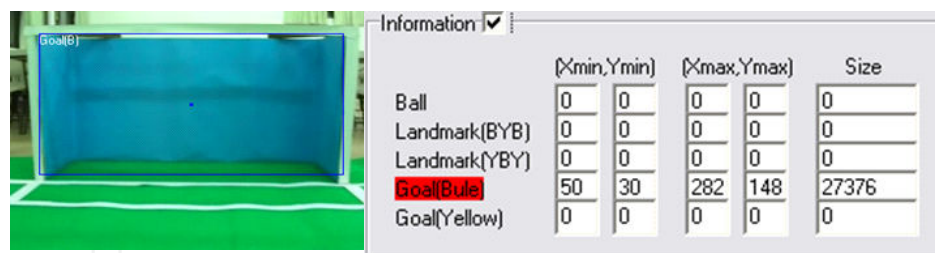
$$G_B(x, y) = L_B^m(x, y), \quad (10)$$

$$\text{if } L_B^m(x, y) \notin P_{BYB}(x, y) \cap L_B^m(x, y) \notin P_{YBY}(x, y) \cap A_B^m > \gamma_B$$

where is the pixel of the  $m$ -th blue component in a picture frame. Since the blue goal is composed of the blue object, it is not a part of the YBY-landmark and BYB-landmark. The size of the goal in the field is the largest object, and therefore we set the parameter  $\gamma$  as 50. The result of goal recognition is shown in Fig. 14. The yellow goal is in the same manner as the blue goal.



**Figure 13.** The result of ball recognition.



**Figure 14.** The result of goal recognition.

### 3.3. Coordinate transformation

Because our proposed approach, ARM is using the different resolutions in the object recognition, we transform the coordinate into the original resolution by level-based of DWT when the object information is outputted. The transform equation is defined in (11).

$$O(x, y) = LL_n(x \times 2^n, y \times 2^n) \quad (11)$$

where  $O$  is the original image,  $LL_n$  is the LL-band iamge after transformation, and  $n$  is the transformation level.

## 4. Experimental results

In this work, the environment information is extracted by the Logitech QuickCam Ultra Vision (Using the monocular vision technique). The image resolution is 320×240, and the frame rate is 30 FPS (frame per second). For the simulation computer, the CPU is Intel Core 2 Duo CPU 2.1GHz, and the development tool is Borland C++ Builder 6.0. The graphical interface is shown in Fig. 15.

This work is dedicated to the RoboCup soccer humanoid league rules of the 2009 competition. In order to prove the robustness of the proposed approach, many scenes of various situations are simulated to verify the high recognition accuracy rate and fast processing time. For the analyses of recognition accuracy rate, it is classified as a correct recognition if the critical object is labeled completely and named correctly such as the objects of Goal[B] and Ball shown in Fig. 16(a). On the other hand there are two categories for false recognition, “false positive” and “false negative”. “False positive” means that the system recognizes the irrelevant object as the critical object, such as the Goal[Y] shown in Fig. 16(b). “False negative” means the system cannot label or name the critical object, such as those balls shown in Figs. 16(c) and 16(d).

### 4.1. Low resolution analysis

Several low resolution methods, such as down-sampling (DS), AF, and SMDWT, were implemented and simulated in this experiment and the noise removing capabilities with these

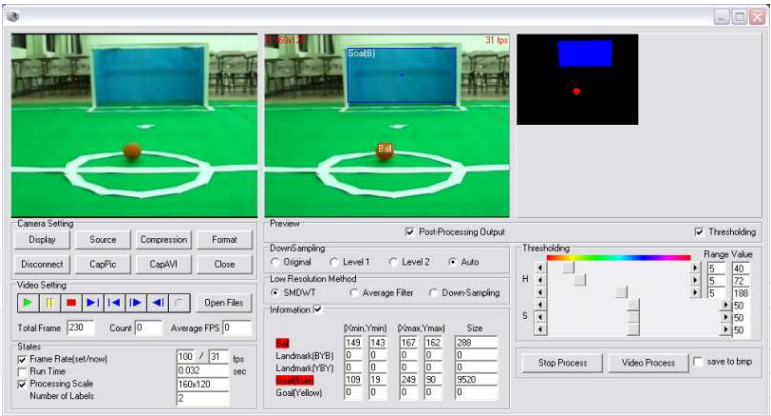


Figure 15. The graph interface for simulation.

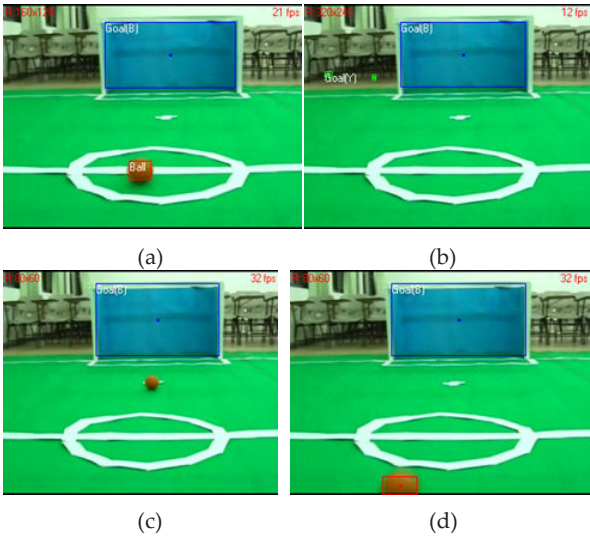
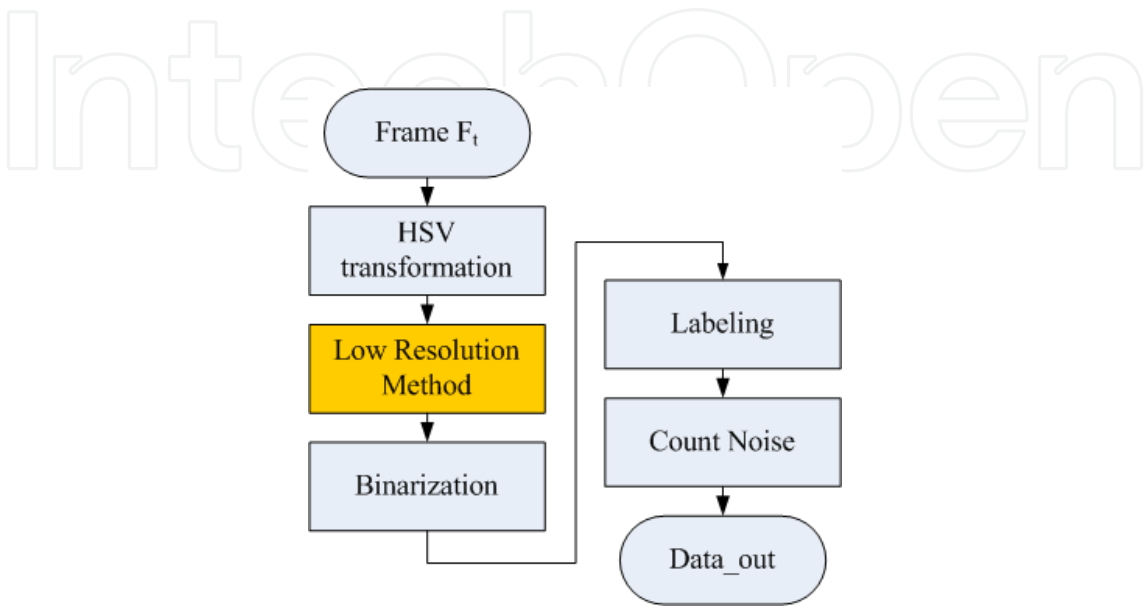


Figure 16. The determination of recognition accuracy. (a) correct recognition. (b) false positive. (c) false negative. (d) false negative.

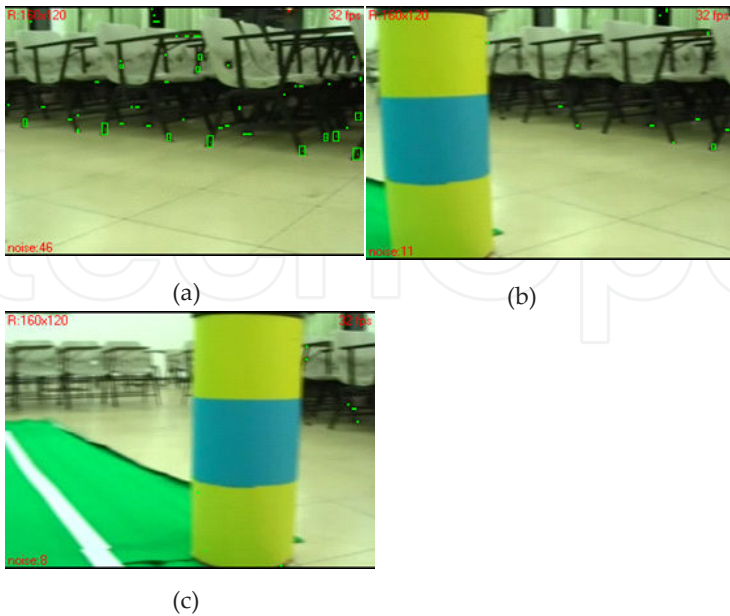
methods were analyzed. The flow chart of noise removing for the low resolution approaches is shown in Fig. 17. The input frame resolution is 320×240, and the resolution turns to be 160×120 after the low resolution processing. The noise numbers under different low resolution methods were counted. The contents of the simulated scene are obtained by turning the camera to left to see the YBY-landmark and keeping turning until the YBY-landmark disappeared from the camera scope. In this situation the background of the scene produces noise very easily. The hue threshold values of the orange, yellow, and blue colors are set as 35~45, 70~80, and 183~193, respectively, and the saturation threshold of the orange, yellow, and blue colors values are all set as 70. The experimental results under different low resolution methods, DS, AF, and SMDWT, are shown in Figs. 18-20, respectively.

The experiment data are listed in Table 2. According to Table 2, the DS approach has the worst noise removing capability; the 2×2 AF approach also has a bad noise removing capability for

big noise block even though this method can make the image smoother. On the other hand, the SMDWT approach (using LL-mask only) has a better noise removing capability than the other methods, and it can retain the information of low-frequency component and remove the noise of high-frequency component in the image.

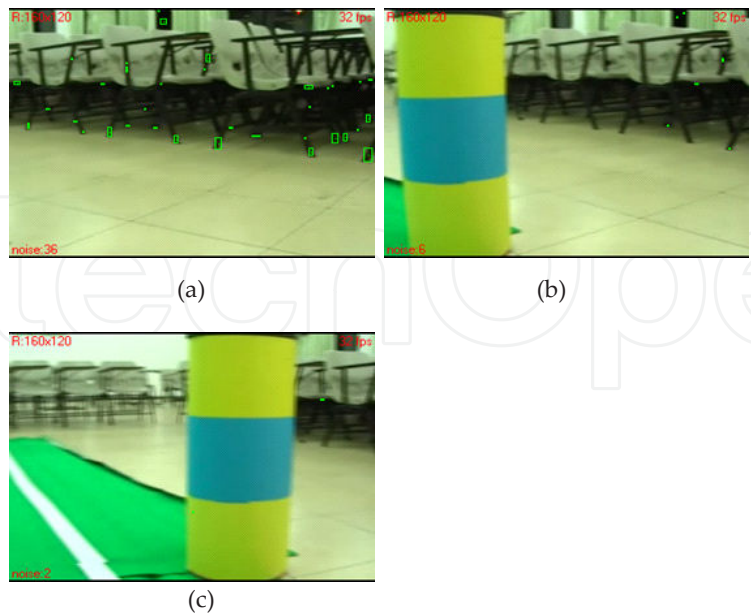


**Figure 17.** The flow chart of noise removing capability.

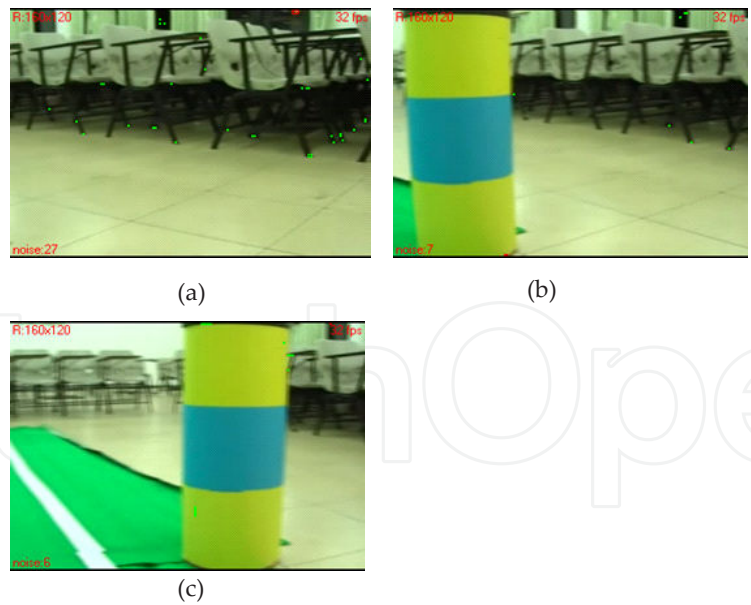


**Figure 18.** Fig. 18. The noise removing capability by using DS. (a) frame 37. (b) frame 67. (c) frame 97.





**Figure 19.** The noise removing capability by using 2x2 AF method. (a) frame 37. (b) frame 67. (c) frame 97.



**Figure 20.** The Gaussian noise removing capability by using SMDWT. (a) frame 37. (b) frame 67. (c) frame 97.

In order to improve the noise removing capability of the whole system, we added the opening operator (OP) of mathematical morphology after labeling in the flow chart of Fig. 17. The results after adding the opening operator are shown in Figs. 21-23.



| Method | Total frame | Total noise number | *Average noise number | Average frame rate |
|--------|-------------|--------------------|-----------------------|--------------------|
| DS     | 153         | 4,133              | 27.01                 | 42.24 FPS          |
| AF     |             | 3,191              | 20.86                 | 41.03 FPS          |
| SMDWT  |             | 2,670              | 17.45                 | 38.13 FPS          |

\*Average noise number = (Total noise number) / (Total frame)

Table 2. The noise counts under different low resolution methods.

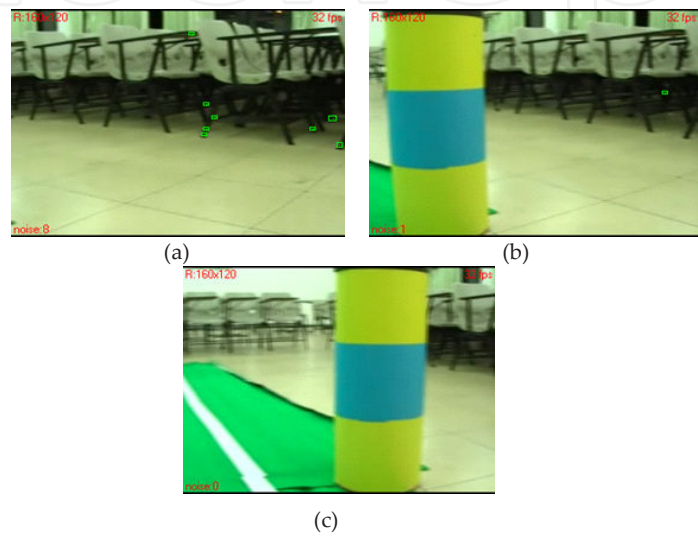


Figure 21. The noise removing capability by using DS and opening operator. (a) frame 37. (b) frame 67. (c) frame 97.

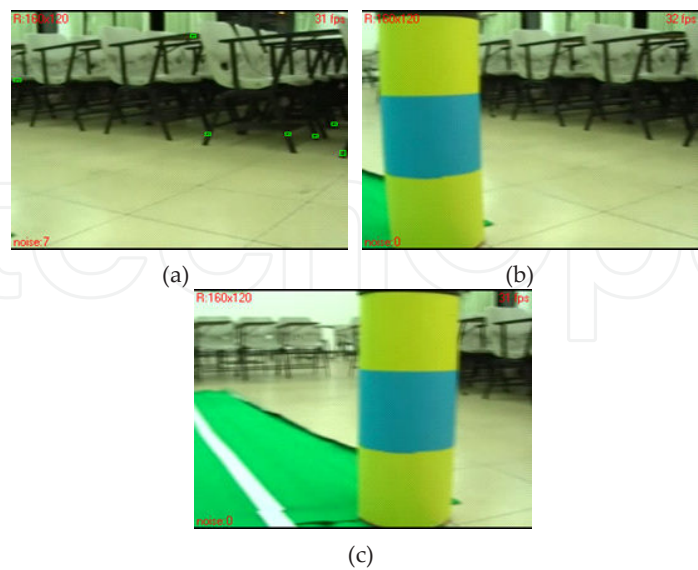
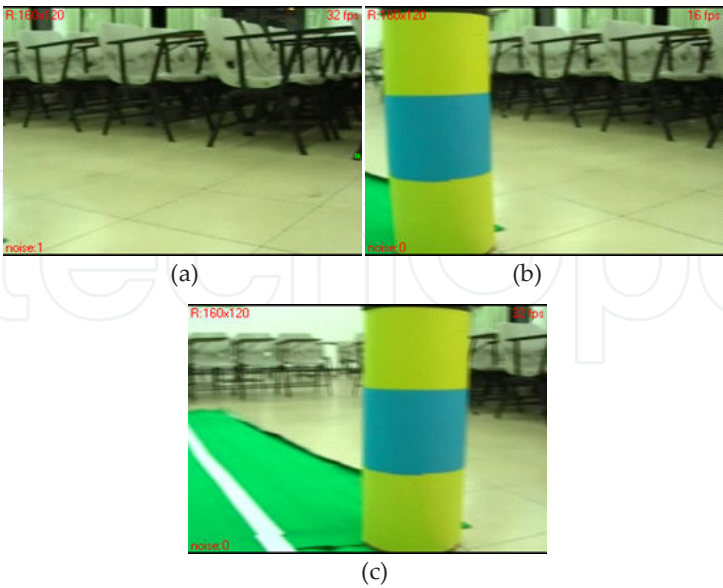


Figure 22. The noise removing capability by using 2x2 AF method and opening operator. (a) frame 37. (b) frame 67. (c) frame 97.



**Figure 23.** The noise removing capability by using SMDWT and opening operator. (a) frame 37. (b) frame 67. (c) frame 97

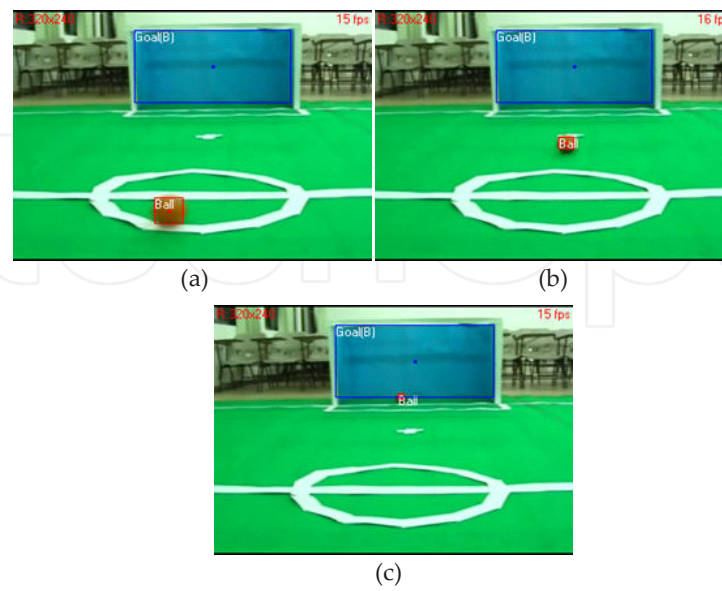
The experiment data after adding the opening operator are shown in Table 3. Compared with the results of Table 2, the noise numbers are reduced significantly after adding the opening operator, and it can reduce the unnecessary computation. The SMDWT approach has the best performance and the frame rate can be as high as 30 FPS. Therefore this work adopts the SMDWT approach as the low resolution method.

| Method     | Total frame | Total noise number | Average noise number | Average frame rate |
|------------|-------------|--------------------|----------------------|--------------------|
| DS + OP    | 153         | 408                | 2.67                 | 38.01 FPS          |
| AF + OP    |             | 334                | 2.18                 | 37.73 FPS          |
| SMDWT + OP |             | 60                 | 0.39                 | 30.95 FPS          |

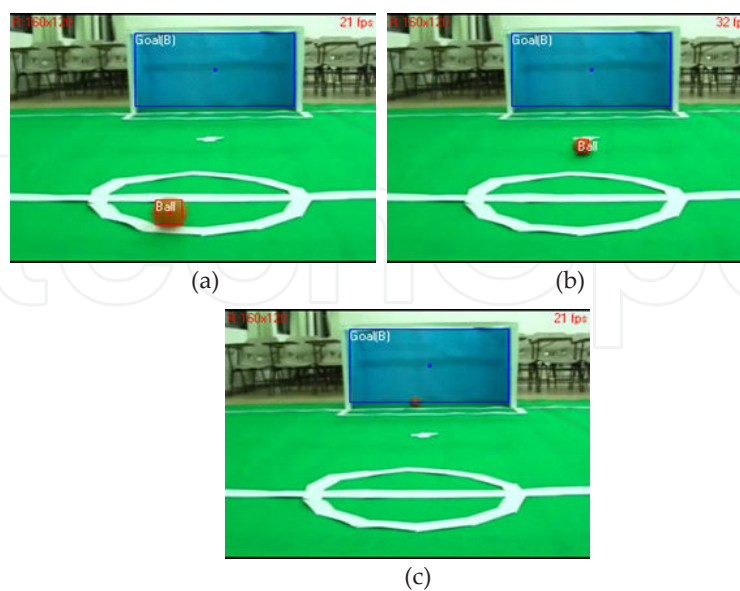
**Table 3.** The noise counts under different low resolution methods with opening operator

4.2. Adaptive Resolution Method (ARM) analyses

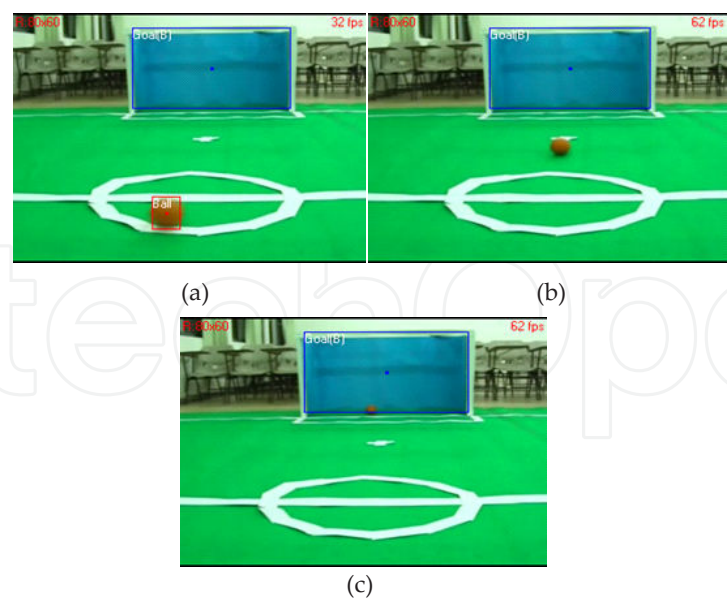
In this experiment, we try to verify that ARM does not only retain high recognition accuracy rate, but also can raise the system processing efficiency. The hue threshold values of the orange, yellow, and blue colors are set as 35~45, 70~80, and 183~193, respectively. The saturation threshold values of the orange, yellow, and blue colors are all set as 70. To verify the ARM approach, the camera is set in the center of the contest field. The scene tries to simulate that the robot kicks ball into the goal and the vision system will track the ball. The results under resolutions of 320×240, 160×120, 80×60, and ARM are shown in Figs. 24-27, respectively.



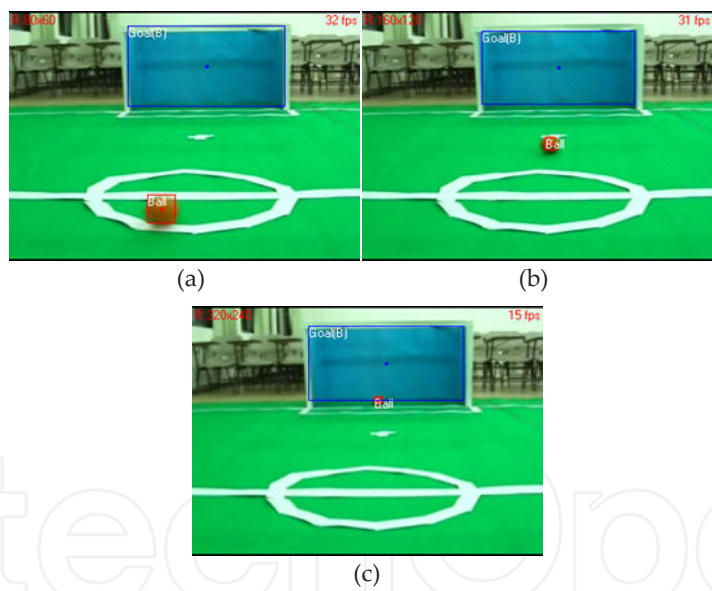
**Figure 24.** The result of object recognition under resolution 320×240. (a) frame 20. (b) frame 35. (c) frame 110.



**Figure 25.** The result of object recognition under resolution 160×120. (a) frame 20. (b) frame 35. (c) frame 110.



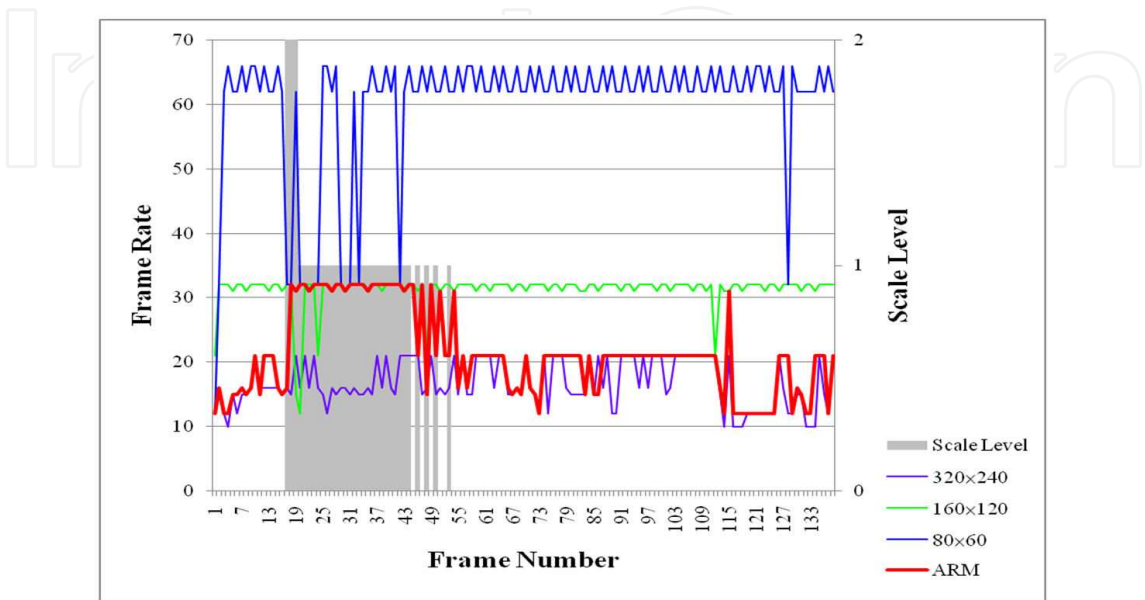
**Figure 26.** The result of object recognition under resolution 80×60. (a) frame 20. (b) frame 35. (c) frame 110.



**Figure 27.** The result of object recognition under the ARM approach. (a) frame 20. (b) frame 35. (c) frame 110.

The experiment data of the accuracy rate and average FPS under different resolutions and ARM are shown in Table 4 and Fig. 28. According to Table 4, although the 320×240 resolution has a high accuracy rate, the processing speed is slow. The 80×60 resolution has the highest processing speed, but it has the lowest accuracy rate. By this approach, it gets high accuracy rate only when the object is close to the camera. On the other hand, the proposed ARM approach does not only have a high accuracy rate, but also keeps high processing speed. According to Fig. 28, the result shows that ARM selects the most proper resolution when the

ball is in different distances. ARM uses the 80×60 resolution when the level is equal to 2 and uses the 160×80 resolution when the scale level is equal to 1. As the scale level is equal to 0, ARM selects the original input frame size (320×240).



**Figure 28.** The relationship between frame number and frame rate under different resolutions and ARM.

| Resolution | Total frame | Object frame | False positive | False negative | Accuracy rate | Average frame rate |
|------------|-------------|--------------|----------------|----------------|---------------|--------------------|
| 320×240    | 138         | 138          | 0              | 6              | 95.65%        | 16.93 FPS          |
| 160×120    |             |              | 0              | 52             | 62.32%        | 31.46 FPS          |
| 80×60      |             |              | 0              | 109            | 21.01%        | 59.84 FPS          |
| ARM        |             |              | 0              | 7              | 94.93%        | 21.17 FPS          |

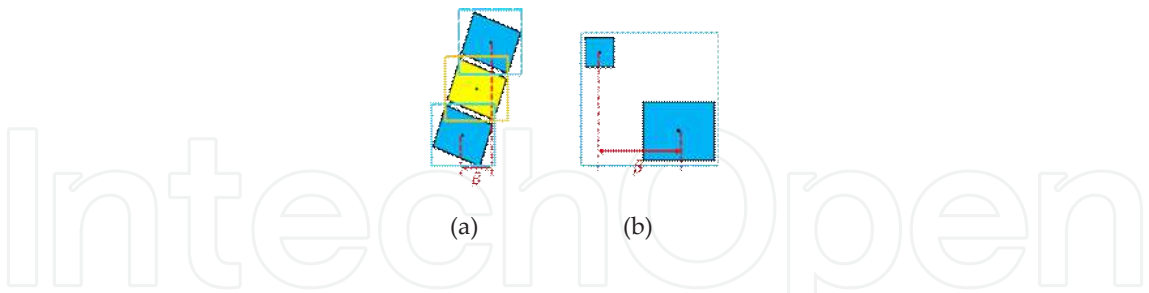
**Table 4.** The experimental results of the accuracy rate and average FPS under different resolutions and ARM.

4.3. The critical objects recognition analysis

In this experiment, several scenes were simulated to improve the robustness of feature recognition approaches proposed in this work.

4.3.1. Landmark recognition analysis

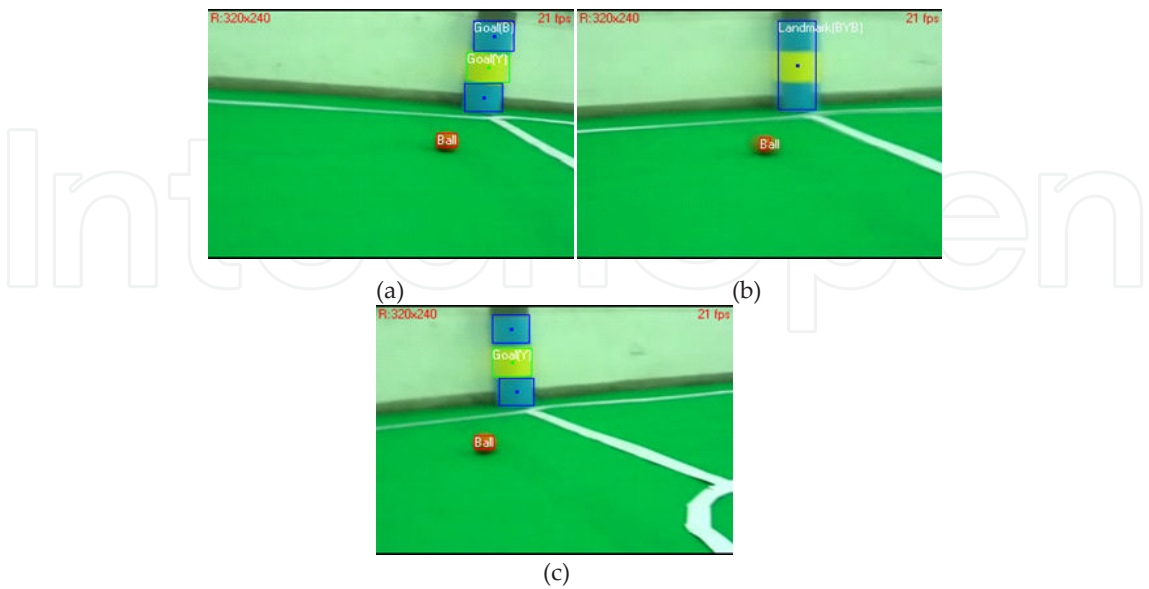
According to (8), the landmark is composed of two same color objects in the vertical line, and the bias value  $\beta$  is the key point to make sure whether this block is a landmark or not. A small bias value  $\beta$  will cause the missing recognition, however a large  $\beta$  may recognize an irrelevant block as a landmark, and these two situations are shown in Fig. 29.



**Figure 29.** The diagram of false recognition of landmark. (a) the case of small  $\beta$ . (b) the case of large  $\beta$ .

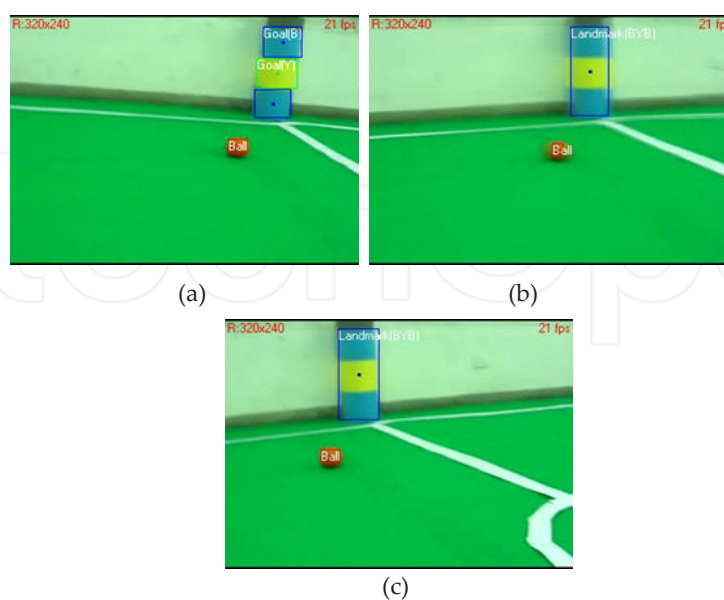
In this experiment, different values of  $\beta$  were set to test the effects. The scene is used for simulating that the camera captures a slantwise landmark when the robot is walking. The hue threshold values of the orange, yellow, and blue colors are set as 35~45, 65~75, and 175~185, respectively. The saturation threshold values of the orange, yellow, and blue colors are all set as 60. The results under values of  $\beta$  equal to 5, 10, 15, and 20 are shown in Figs. 30-33, respectively.

The experiment data of landmark recognition is shown in Table 5. According to this table, we can have a higher recognition accuracy rate when  $\beta$  is greater than 15. Generally speaking, the vibration of robot walking is not more intense than the simulation, and therefore  $\beta$  is set as 15 in this work. It will increase the chance of false recognition as a larger  $\beta$  is used.

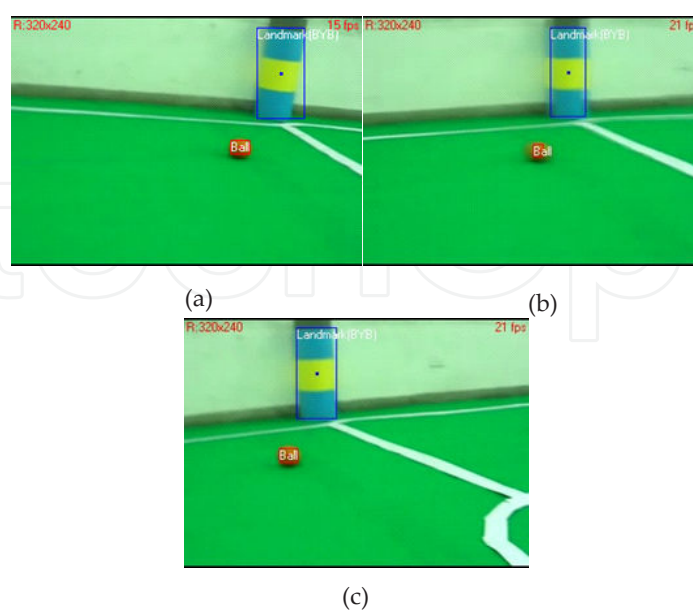


**Figure 30.** The result of object recognition with  $\beta$  equal to 5. (a) frame 246. (b) frame 253. (c) frame 260.

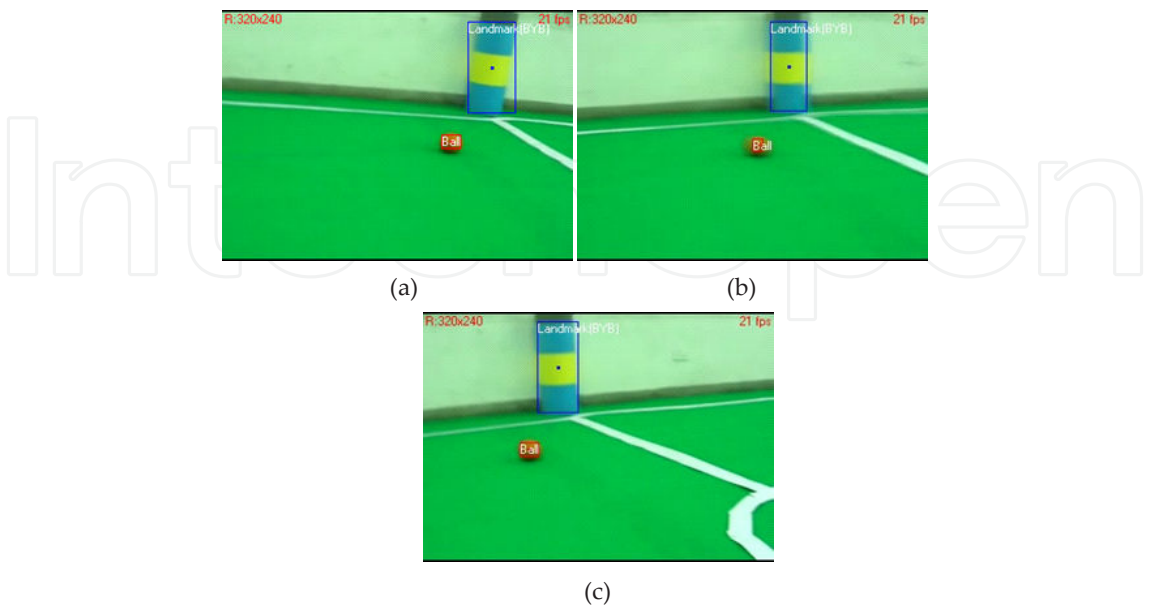




**Figure 31.** The result of object recognition with  $\beta$  equal to 10. (a) frame 246. (b) frame 253. (c) frame 260.



**Figure 32.** The result of object recognition with  $\beta$  equal to 15. (a) frame 246. (b) frame 253. (c) frame 260.



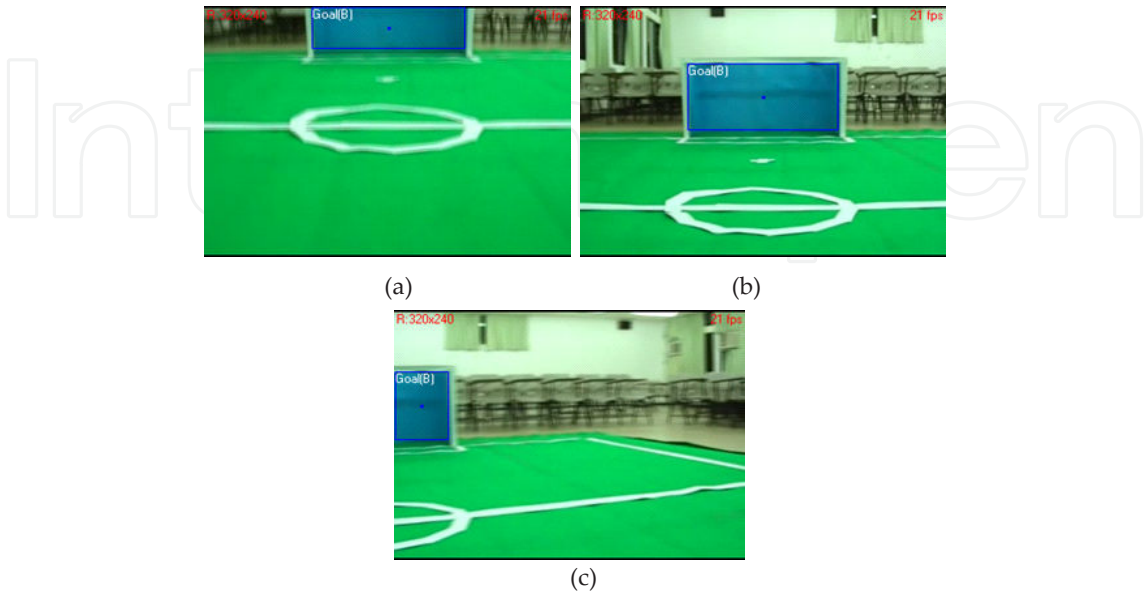
**Figure 33.** The result of object recognition with  $\beta$  equal to 20. (a) frame 246. (b) frame 253. (c) frame 260.

| $\beta$ | Total frame | Object frame | Correct recognition | Accuracy rate | Average frame rate |
|---------|-------------|--------------|---------------------|---------------|--------------------|
| 5       | 664         | 664          | 304                 | 45.78%        | 20.02 FPS          |
| 10      |             |              | 545                 | 82.08%        | 20.07 FPS          |
| 15      |             |              | 637                 | 95.93%        | 20.15 FPS          |
| 20      |             |              | 631                 | 95.03%        | 20.18 FPS          |

**Table 5.** The experimental results under different values of  $\beta$ .

4.3.2. Goal recognition analysis

The goal is the largest critical object in the field, and hence the camera always captures the incomplete goal in the frame when the robot is walking in the field. It causes a false recognition easily by using the feature of the shape ratio to recognize the goal. We improve this drawback by using the proposed method in Section 3.2 and the experimental results are shown here. The camera is set in the center of the contest field. The scene tries to simulate that the robot raises its head to see the goal and turns right to see the YBY-landmark and then turns left to see the BYB-landmark. The hue threshold values of the orange, yellow, and blue colors are set as 35~45, 70~80, and 183~193, respectively. The saturation threshold values of the orange, yellow, and blue colors are all set as 60. The results are shown in Fig. 34 and the experiment data are listed in Table 6. According to the result, the system can make the correct recognition of goal even though the goal is occluded.



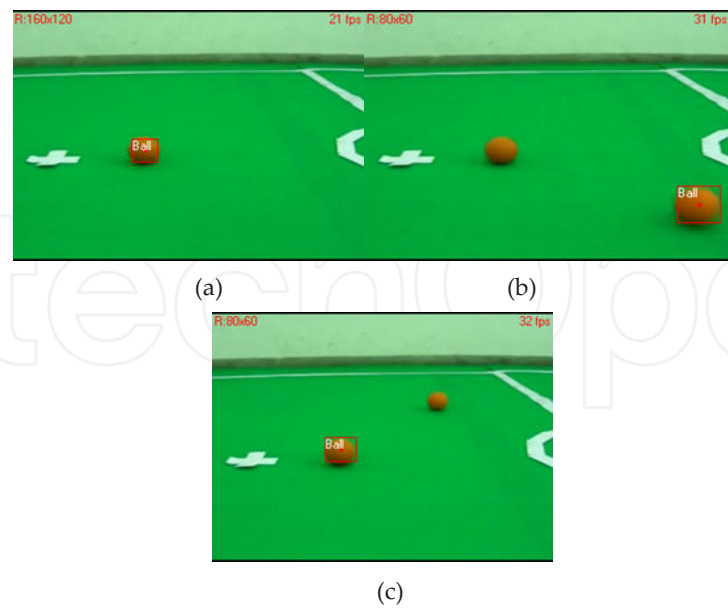
**Figure 34.** The results of goal recognition. (a) frame 34. (b) frame 76. (c) frame 118.

| condition        | Total frame | Object frame | False positive | False negative | Accuracy rate | Average frame rate |
|------------------|-------------|--------------|----------------|----------------|---------------|--------------------|
| Goal Recognition | 328         | 297          | 0              | 7              | 97.64%        | 21.98 FPS          |

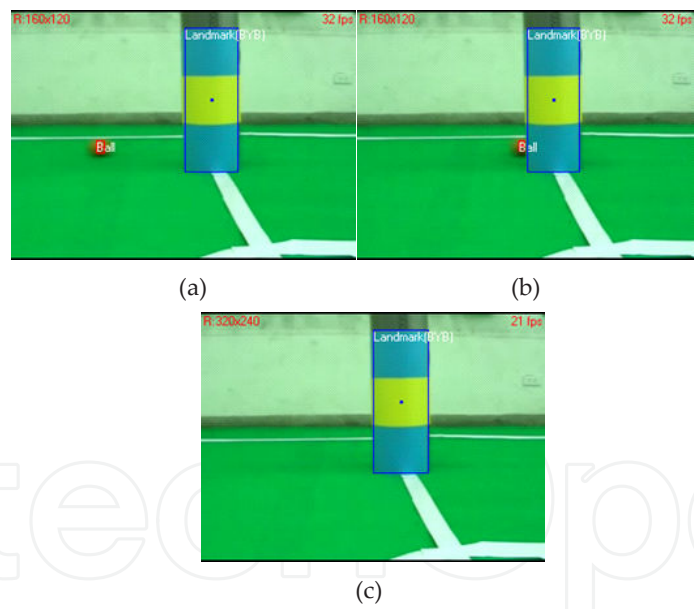
**Table 6.** The experiment data of goal recognition.

4.3.3. Ball recognition analysis

For the ball recognition, the system determines the orange block which has the maximum pixels as a ball for preventing the influence of noise. In this experiment, two balls are used in the scene. One ball is static in the field, and the other one moves into the frame and then moves away from the camera. The hue threshold values of the orange, yellow, and blue colors are set as 35~45, 70~80, and 183~193, respectively. The saturation threshold values of the orange, yellow, and blue colors are all set as 60. The result is shown in Fig. 35 and the experiment data are shown in Table 7. The static ball is labeled absolutely if only one ball is in the field, and the result is shown in Fig. 35(a). Because another ball has a bigger area when it is moving into the frame, the system will label the moving ball and determine the static ball as noise, and the result is shown in Fig. 35(b). When the moving ball is distant from the camera, the static ball is labeled again, and the result is shown in Fig. 35(c).



**Figure 35.** The result of ball recognition. (a) frame 124. (b) frame 131. (c) frame 156.



**Figure 36.** The result of ball recognition. (a) frame 145. (b) frame 151. (c) frame 153.

| condition        | Total frame | Object frame | False positive | False negative | Accuracy rate | Average frame rate |
|------------------|-------------|--------------|----------------|----------------|---------------|--------------------|
| Ball Recognition | 274         | 274          | 0              | 0              | 99.99%        | 30.93 FPS          |
| Ball Occlusion   | 289         | 289          | 3              | 0              | 98.96%        | 20.69 FPS          |

**Table 7.** The experiment data of ball recognition.

Besides, it can also handle the situation when the ball is occluded partially by using the feature recognition proposed. We use the scene that the ball is occluded by the landmark during the ball moving in the frame from left to right. The hue threshold values of the orange, yellow, and blue colors are set as 35~45, 70~80, and 175~185, respectively. The saturation threshold values of the orange, yellow, and blue colors are all set as 50. The results are shown in Fig. 36 and the experiment data are shown in Table 7.

4.4. Environmental tolerance analysis

The color deviation by luminance variation has the most influence to the result of the color-based recognition method proposed in this work. Before the robot soccer competition we usually have one day to prepare for the contest, and therefore we can regulate the threshold values easily by the graph interface according to the luminance of the field. The results under different luminance are shown in Fig. 37. The reference threshold values are shown in Table 8.

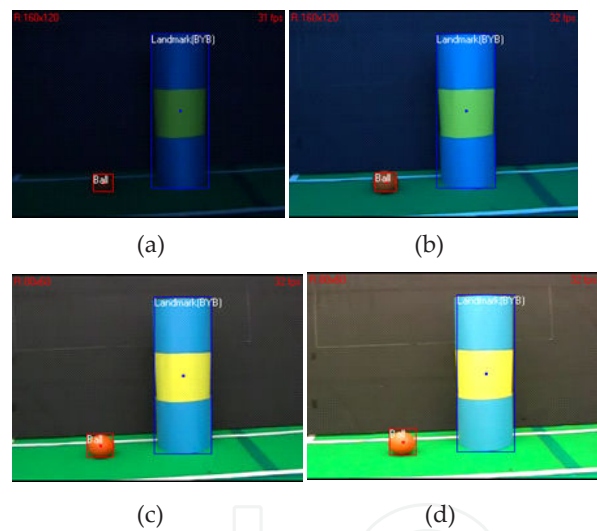
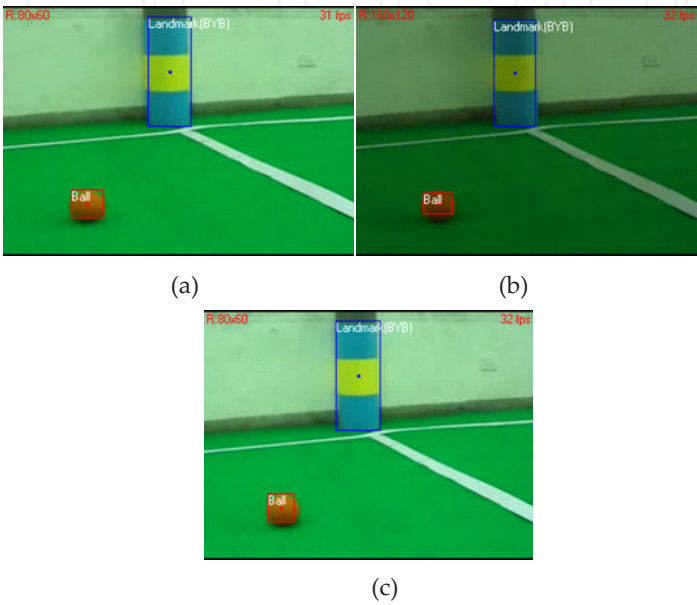


Figure 37. The results of object recognition under different luminance. (a) 16 lux. (b) 178 lux. (c) 400 lux. (d) 893 lux.

| Luminance | Hue_O | Sat_O | Hue_Y   | Sat_Y | Hue_B   | Sat_B |
|-----------|-------|-------|---------|-------|---------|-------|
| 16 lux    | 3~13  | 10    | 118~128 | 50    | 220~230 | 96    |
| 178 lux   | 13~23 | 60    | 119~129 | 60    | 205~215 | 96    |
| 400 lux   | 17~27 | 50    | 61~71   | 50    | 190~200 | 50    |
| 596 lux   | 17~27 | 50    | 57~67   | 50    | 180~190 | 50    |
| 893 lux   | 23~33 | 50    | 57~67   | 45    | 180~190 | 50    |

Table 8. The threshold values used under different luminance.

The system cannot only recognize the critical objects under different luminance, but it can also accommodate the light changing suddenly. This experiment simulates that the robot recognizes the BYB-landmark and ball in the field under the light changing suddenly. The hue threshold values of the orange, yellow, and blue colors are set as 33~43, 67~77, and 175~185, respectively. The saturation threshold values of the orange, yellow, and blue colors are all set as 50. The results are shown in Fig. 38 and the experiment data are listed in Table 9. According to the result, the proposed method has a good performance about environmental tolerance.



**Figure 38.** The result of object recognition under the light changing suddenly. (a) frame 388. (b) frame 440. (c) frame 535.

| condition       | Total frame | Object frame | False positive | False negative | Accuracy rate | Average frame rate |
|-----------------|-------------|--------------|----------------|----------------|---------------|--------------------|
| Light Influence | 1,219       | 1,219        | 0              | 0              | 99.99%        | 31.14 FPS          |

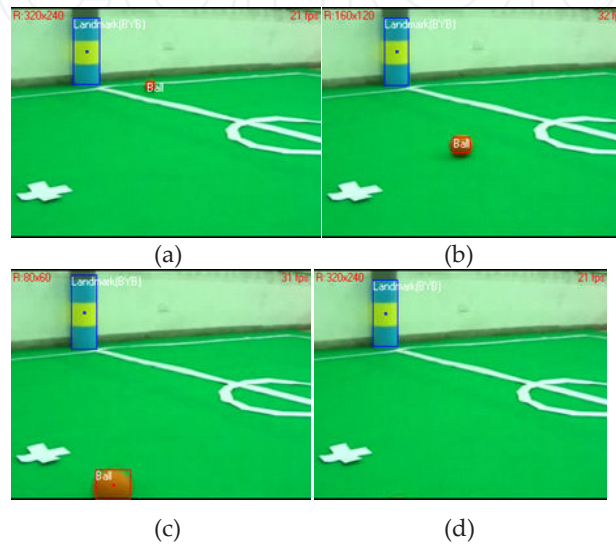
**Table 9.** The experiment data of light influence.

4.5. Synthetic analyses

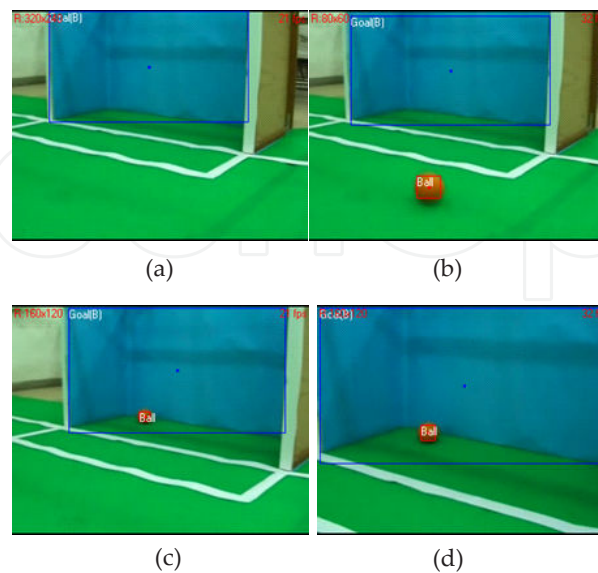
In this experiment, several scenes were simulated to compare the recognition accuracy rate and processing time between the 320×240 resolution and ARM. Scene 1: the ball is approach it to the camera slowly. Scene 2: the robot is approaching the ball after shooting the ball to the goal. Scene 3: the robot finds the ball and then tries to get approaching and kick it. Scene 4: the camera captures a blurred image when the head motor of the robot is rotating very fast. Scene 5: the robot localizes itself by seeing the landmarks. The hue threshold values of the orange, yellow, and blue colors are set as 35~45, 70~80, and 185~190, respectively. The saturation threshold values of the orange, yellow, and blue colors are all set as 50. The experiment data of these scenes are shown in Table 10 and the experimental results are shown in Figs. 39-43,



respectively. According to the simulation results, our proposed method accommodates many kinds of scenes. It has the accuracy rate of more than 93% on average and the average frame rate can reach 32 FPS. It does not only maintain the high recognition accuracy rate for the high resolution frames, but also increases the average frame rate for about 11 FPS compared to the conventional high resolution approach. Furthermore, all of the experimental result videos mentioned in this section are appended in.

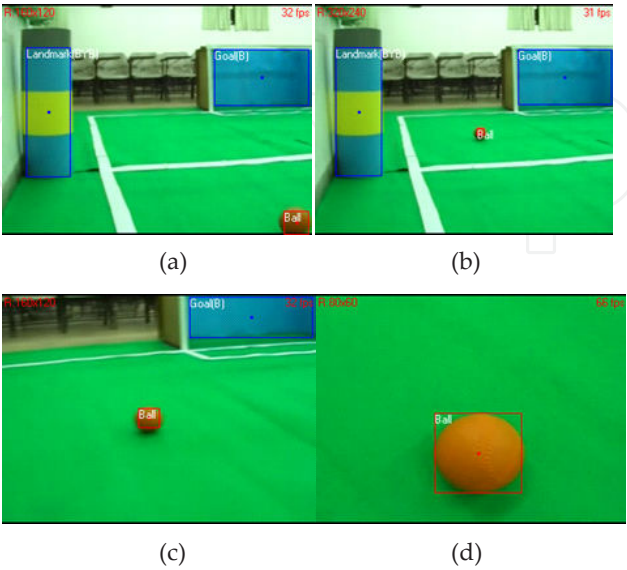


**Figure 39.** The result of Scene 1. (a) frame 11. (b) frame 46. (c) frame 63. (d) frame 65.



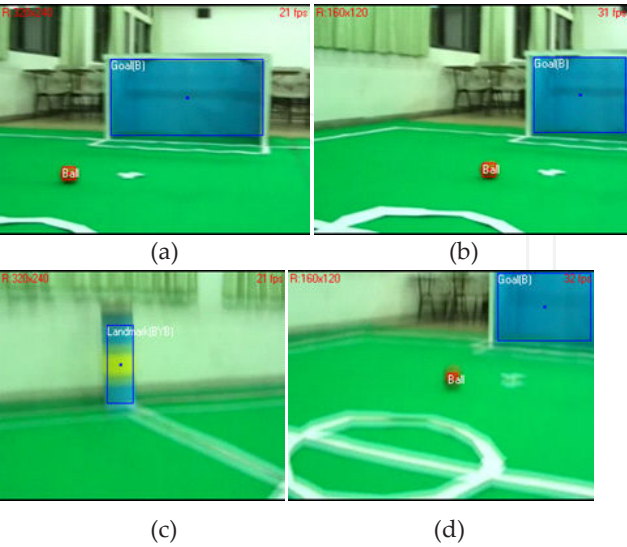
**Figure 40.** The result of Scene 2. (a) frame 89. (b) frame 98. (c) frame 240. (d) frame 347.

Int Open

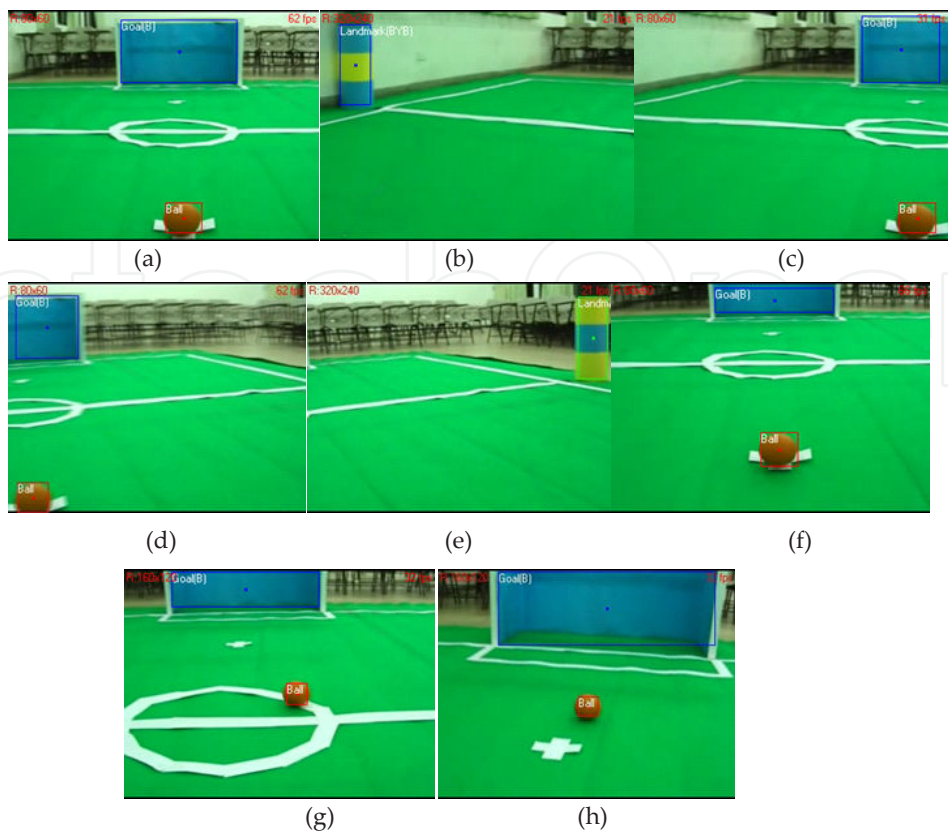


**Figure 41.** The result of Scene 3. (a) frame 81. (b) frame 159. (c) frame 456. (d) frame 793.

Int Open



**Figure 42.** The result of Scene 4. (a) frame 81. (b) frame 162. (c) frame 273. (d) frame 620.



**Figure 43.** The result of Scene 5. (a) frame 106. (b) frame 211. (c) frame 347. (d) frame 372. (e) frame 434. (f) frame 581. (g) frame 748. (h) frame 954.

| Scene | Resolution | Total frame | Object frame | False positive | False negative | Accuracy rate | Average frame rate |
|-------|------------|-------------|--------------|----------------|----------------|---------------|--------------------|
| 1     | 320×240    | 165         | 165          | 1              | 5              | 96.36%        | 20.49 FPS          |
|       | ARM        |             |              | 0              | 4              | 97.58%        | 23.31 FPS          |
| 2     | 320×240    | 409         | 409          | 27             | 1              | 93.15%        | 21.36 FPS          |
|       | ARM        |             |              | 11             | 2              | 96.82%        | 29.88 FPS          |
| 3     | 320×240    | 919         | 919          | 16             | 15             | 96.63%        | 19.75 FPS          |
|       | ARM        |             |              | 1              | 28             | 96.84%        | 28.48 FPS          |
| 4     | 320×240    | 679         | 627          | 3              | 83             | 86.28%        | 19.29 FPS          |
|       | ARM        |             |              | 2              | 88             | 85.65%        | 27.31 FPS          |
| 5     | 320×240    | 1,114       | 1,114        | 12             | 60             | 93.54%        | 22.38 FPS          |
|       | ARM        |             |              | 4              | 74             | 93.00%        | 40.58 FPS          |
| Total | 320×240    | 3,286       | 3,234        | 59             | 164            | 93.10%        | 20.78 FPS          |
|       | ARM        |             |              | 18             | 196            | 93.38%        | 32.25 FPS          |

**Table 10.** The experimental results of several kinds of scene simulation.

## 5. Conclusions

An outstanding humanoid robot soccer player must have a powerful object recognition system to fulfill the functions of robot localization, robot tactics, and barrier avoiding. In this study, we propose an HSV color based object segmentation method to accomplish object recognition. The object recognition system uses the proposed adaptive resolution method (ARM) and sample object recognition method, and it can recognize objects. The experimental results indicate that the proposed method is not only simple and capable of real-time processing but that it also achieves high accuracy and efficiency with the functions of object recognition and tracking. The method achieves a high accuracy rate of more than 93% on average, and the average frame rate can reach 32 FPS in indoor situations.

## Author details

Chih-Hsien Hsia<sup>1</sup>, Wei-Hsuan Chang<sup>2</sup> and Jen-Shiun Chiang<sup>2</sup>

1 Department of Electrical Engineering, National Taiwan University of Science and Technology Taipei, Taiwan

2 Department of Electrical Engineering, Tamkang University Taipei, Taiwan

## References

- [1] Available: <http://www.robocup2009.org/153-0-rules>.
- [2] Canny, J. (1986). A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, (June 1986) pp. 679-698.
- [3] Chaumette, F. (1994). Visual servoing using image features defined on geometrical primitives, *IEEE Conference on Decision and Control*, (December 1994) pp. 3782-3787.
- [4] Cheng, F.-H. & Chen, Y.-L. (2006). Real time multiple objects tracking and identification based on discrete wavelet transform, *Pattern Recognition*, Vol. 39, No. 6, (June 2006) pp. 1126-1139.
- [5] Chiang, J.-S., Hsia, C.-H., Hsu, H.-W., & Li C.-I. (2011). Stereo vision-based self-localization system for RoboCup," *IEEE International Conference on Fuzzy Systems*, (June 2011) pp. 2763-2770.
- [6] Gonzalez, R. C. & Woods, R. E. (2001). *Digital image processing*, Addison-Wesley Longman Publish Co., Inc., Boston.

- [7] Herodotou, N., Plataniotis, K. N., & Venetsanopoulos, A. N. (1998). A color segmentation scheme for object-based video coding, *IEEE Symposium on Advances in Digital Filtering and Signal Processing* (June 1998) PP. 25-29.
- [8] Hsia, C.-H.; Guo, J.-M. & Chiang, J.-S. (2009). Improved low complexity algorithm for 2-D integer lifting-based discrete wavelet transform using symmetric mask-based scheme, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 19, No 8, (August 2009) pp. 1201-1208.
- [9] Ikeda, O. (2003). Segmentation of faces in video footage using HSV color for face detection and image retrieval, *International Conference on Image Processing*, Vol. 2, (September 2003) pp. III-913-III-916.
- [10] Jean, J.-H. & Wu, R.-Y. (2004). Adaptive visual tracking of moving objects modeled with unknown parameterized shape contour, *IEEE International Conference on Networking, Sensing and Control*, (March 2004) pp. 76-81.
- [11] Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: active contour models, *International Journal of Computer Vision*, Vol. 1, (January 1988) pp. 321-331.
- [12] Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., & Osawa, E. (1995). Robocup: The robot world cup initiative, *IJCAI-95 Workshop on Entertainment and AI/ALife*, (1995) pp. 19-24.
- [13] Sun, S. J., Haynor, D. R., & Kim, Y. M. (2003). Semiautomatic video object segmentation using VSnakes, *IEEE Transactions on Circuits System Video Technology*. Vol. 13, Vol. 1, (January 2003) pp. 75-82.
- [14] Sugandi, B., Kim, H., Tan, J. K., & Ishikawa, S. (2009). Real time tracking and identification of moving persons by using a camera in outdoor environment, *International Journal of Innovative Computing, Information and Control*, Vol. 5, Vol. 5, (May 2009) pp. 1179-1188.

IntechOpen

