FAST AND ACCURATE VISION-BASED PATTERN DETECTION AND IDENTIFICATION USING COLOR AND GREY IMAGE SEGMENTATION

Piotr Fiertek

Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics, Department of Automatic Control, ul. Gabriela Narutowicza 11/12, 80-952 Gdańsk, Poland, e-mail: pfiertek@wp.pl

Abstract: *This paper describes a computationally inexpensive vision-based robot detection and identification method, which can be used in FIRA competition. The proposed method is based on detection of color patches located on the topside of a robot. It exploits binary color segmentation based on the color recognition system in the HSI system. We discuss the ways of improving accuracy and robustness of the localization system by using additional adaptive grey image segmentation.*

1 Introduction

Robot soccer games have recently become very popular in educational institutions around the world. The robot soccer system provides perfect platform for research students and scientists in different technical fields such as robotics, intelligent control, communication, computer technology, image processing, artificial intelligence, etc. Due to its compact size, relatively inexpensive robots (the size of each robot is limited to 7,5cm x 7,5cm x 7,5cm) and wide range of research opportunities, MiroSot has been one of the most attractive categories in robot soccer area ever since the inauguration of the series of FIRA competitions in 1997.

In the MiroSot system, robots are controlled based only on the information from the overhead global vision system which detects robot and ball movements through color patches. The vision subsystem is therefore a dominant factor deciding upon the control accuracy of the entire system. There are many papers describing various algorithms of detection, identification and tracking that can be used for monitoring the playground. Most of them concentrate on finding blobs (coherent areas of pixels of similar color). Identification is made by testing configuration of blobs [2, 6, 7, 8, 9] or by shape recognition [10, 11, 12]. This paper is focused on presenting fast methods of checking configurations of blobs and on improving accuracy by using adaptive gray color image segmentation.

The paper is organized as follows: Section 2 presents the system of pixels color recognition, Section 3 describes identification of objects, Section 4 presents the proposed adaptive grey color segmentation method and Section 5 describes the measurement methodology and presents the experimental results. Section 6 concludes.

2 Color Recognition

Our vision system uses UNIQ UC-930 Color Digital CCD Camera. This camera features a single ½" progressive scan CCD. We used it to sample 512x512 24bit RGB and 1024x1024 24bit RGB color images at 30 fps. The camera was connected to a regular PC equipped with a PXD1000 Digital Frame Graber, Pentium IV 2,66GHz Processor and 1GB of RAM.

The UC-930 provides a monochrome image in R, G, B primary color mosaic filter (Bayer arrangement) [14]. It transmits 1024x1024x8bpp, where only one of the RGB components is provided at each pixel (Fig.1).

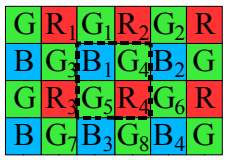


Fig. 1. Typical Bayer filter pattern showing the alternate sampling of red, green and blue pixels.

Unfortunately this frame grabber does not provide hardware acceleration for transformation of monochrome images in Bayer arrangement into 24bit RGB color images. To provide 24bit color at each pixel, special software is used to interpolate missing color components by using simple convolution matrices to generate RGB values at each pixel from the camera output. To reduce noise and improve reliability of color recognition we used low-pass filtering. To obtain satisfactory results for 512x512 24bit color images (Fig. 2a) we use the following equations (1).

$$\begin{cases} R = \frac{1}{16}\left(R_1 + 3R_2 + 3R_3 + 9R_4\right) \\ G = \frac{1}{16}\left(G_1 + G_3 + 6G_4 + 6G_5 + G_6 + G_8\right) \\ B = \frac{1}{16}\left(9B_1 + 3B_2 + 3B_3 + B_4\right) \end{cases} \quad (1)$$

A filter which is computationally less expensive, but also less accurate, is given by

$$\begin{cases} R = R_4 \\ G = \frac{1}{2}\left(G_4 + G_5\right) \\ B = B_1 \end{cases} \quad (2)$$

To generate RGB values at each pixel for a 1024x1024 24bit color image (Fig. 3a) we have to use 4 filters of this kind.

The implemented color classifiers use huge look-up tables indexed by the pixel's color components, with the value of the selected table entry indicating the set of different color classes that colors may belong to. A color class is defined as a rectangular volume in the HSI space. This solution guarantees fast and robust color recognition.

Despite using low-pass filtering and robust color recognition, after thresholding we obtain binary images with large number of the background pixels which do not belong to any of the blobs (Fig. 2b). To obtain coherent color regions the binary median filter is used.

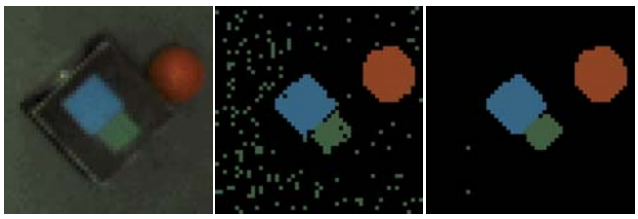


Fig. 2. Color recognition and image segmentation (a) original image (b) image obtained after color recognition (c) image obtained after using the binary median filter.

3 Identification of objects

The FIRA and RoboCup rules assume using one main blue or yellow patch on the topside of robot to recognize robots belonging to one of the teams. In FIRA competition one is allowed to use the second patch (in RoboCup more extra patches can be used) to identify one's robots. Such additional patch must not be colored in orange (which is reserved for the ball), blue, yellow and white. There are many papers describing variety of methods for robot detection and identification.

To reduce the size of the inspected image area, the vision system is provided with tracking algorithm which starts to search objects on the image from the most probable place of their localization. Most algorithms check all pixels inside the square window searching for colors that belong to the main and additional patches.

The need for interpolation and median filtering limits the time required to check a single pixel to approximately 0,27μs for a low resolution image and to 0,20μs for a high resolution image. In a low resolution image, the diagonal size of the robot is about 36 pixels. In a high resolution image it is approximately equal to 72 pixels. The corresponding minimal sizes of inspection windows are 36x36 pixels and 72x72 pixels respectively. In the first case the vision system checks 1296 pixels (which takes approximately 350μs), in the second it checks 5184 pixels (in approximately 1,04ms). To minimize the time of robot identification, in the presented method only some of the pixels are checked.

The first step is searching for pixels belonging to the main patch (blue or yellow blob). This is achieved by following a spiral search line originating at the start point. The next step consists of finding the left and the right boundary of the blob (Fig. 3b). When both boundaries are found we move up and down along the blob boundary until the left path meets with the right path. At this same time we count pixels belonging to the blob (pixels localized between the left and the right side of the boundary). The last step is devoted to calculation of coordinates of the blob centre.

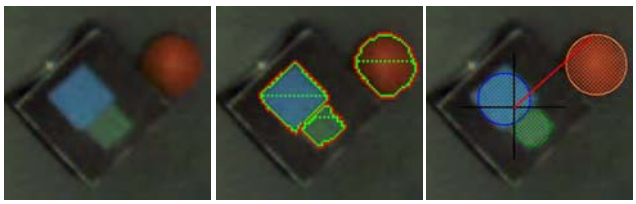


Fig. 3. Robot identification (a) original image (b) detection of the ball and identification of the robot in 1024x1024 resolution. (c) founded ball and robot markers.

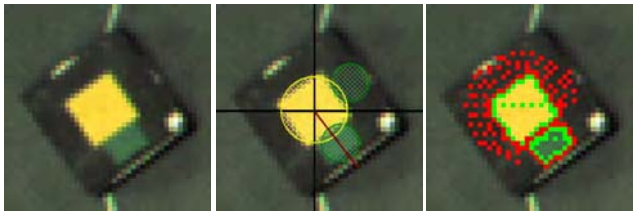


Fig. 4. Searching additional patches (a) original image (b) definition of robot's pattern (c) process of searching additional patches.

When the main patch is found, the search for additional patches is started. We assume that three additional patches can be used. The supporting blobs are localized in the same way as the main blob (Fig. 3 and 4). After finding all additional patches, their positions are compared with positions in known robot patterns. Figure 4 illustrates this process in the case where there is only one additional patch, i.e. the reference pattern consist of two patches.

The proposed method guarantees that the minimal number of pixels is checked to detect and identify a robot with high level of accuracy.

4 Adaptive grey color segmentation

The two main factors that limit accuracy of the vision system are: sampling quantization and non-linear lens distortion [13]. The quantization error is a function of the size of the viewed area and of the capture resolution. High resolution provides high level of accuracy, but this requires checking a large number of pixels. This is the reason of considering image processing with two resolutions (512x512 and 1024x1024, respectively).

The cells of a CCD video camera are arranged in a rectangular grid and averaging occurs when two or more colors are incident on a single CCD cell. For example, on the boundary between red and yellow, the pixels localized contain all of the colors along the line connecting red and yellow [4]. Depending on color definition or lighting conditions, thresholding and segmentation stages localize objects in slightly different positions (Fig. 5). It is possible to reduce this effect by using more restrictive color definition but this causes wrong color recognition when lighting conditions change.

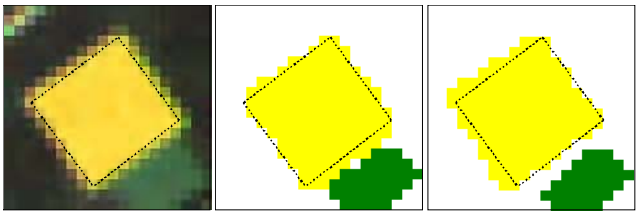


Fig. 5. Color thresholding (a) original image (b) color threshold when definition of 'yellow' is 'more' green (b) color threshold when definition of 'yellow' is 'more' red.

The nonuniform playground illumination and rapid lighting changes caused by fluorescent lamps make the object recognition task truly nontrivial. The solution proposed in this paper is based on adaptive grey color thresholding. In this case the threshold levels are calculated for each image frame.

In the first step the standard detection and identification method is used (color thresholding and segmentation) under low resolution (512x512). As a result, the preliminary (rough) estimates of the robot's position and orientation are obtained. Then the position and the grey level of the main blob (G_m), the grey levels of the supporting blobs (G_a), and the grey level of the background color (G_b) are evaluated. To calculate the grey level of a pixel we use the following equation:

$$G = \tfrac{1}{3}(R+G+B) \tag{3}$$

Next we calculate the grey threshold levels for the main and supporting blobs (cf. Fig. 6).

$$\begin{cases} P_{Hm} = G_m + MIN(35,\ \ 255 - G_m) \\ P_{Lm} = G_m - MIN(35,\ \ G_m - \tfrac{3}{4}G_a - \tfrac{1}{4}G_m) \\ P_{Ha} = G_a + MIN(35,\ \ \tfrac{1}{2}G_a + \tfrac{1}{2}G_m - G_a) \\ P_{La} = G_a - MIN(25,\ \ G_a - \tfrac{3}{4}G_a - \tfrac{1}{4}G_b) \end{cases} \tag{4}$$

Finally, image segmentation is performed by using the binary median filter.

To reduce the jitter error effect, for each of the checked pixels we calculate its influence on the final position of the blob. This influence is described by functions $W_m(J)$ and $W_a(J)$ where J denotes the grey level of the pixel obtained from (3) (Fig. 6). (for pixels that are not checked and lie inside of blob: W_a=1 and W_m=1).

If a supporting blob lies near the main blob, some pixels belonging to the main blob boundary may be assigned to the supporting blob (Fig. 7b). There are two solutions. First, one can appropriately design the robot patterns. Second, one can use an extra condition when testing the supporting blob's pixels. When the supporting blob is investigated, the checked pixels are not assigned to it if at least one of the neighboring pixels belongs to the main blob (Fig. 7c).

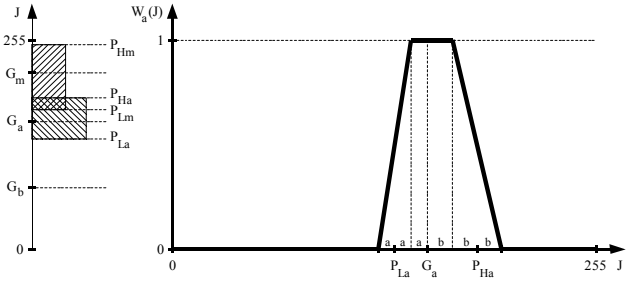


Fig. 6. Adaptive grey color threshold (a) grey color threshold (b) fuzzy function of pixel significance.

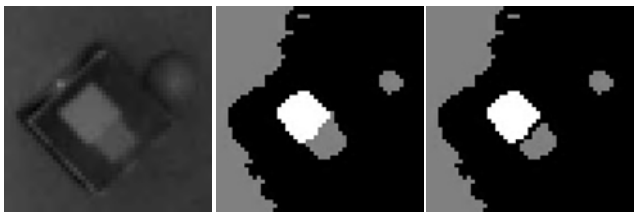


Fig. 7. Grey color threshold and segmentation (a) original image (b) grey color threshold (b) grey color threshold with an extra neighborhood condition.

5 Measurements

To obtain reliable information about accuracy and robustness of the presented method, we performed a several series of measurements for both 512x512 and 1024x1024 resolutions followed by the color and grey thresholding and segmentation. Measurements were gathered for four jackets, shown in Figure 8, and for a

whole range of angles (with the 5° step for the first jacket and 15° steps for the remaining jackets).

To minimize influence of non-linear lens distortion error on the measurements result, the investigated robot was placed directly under the camera.

Table 1. Average bias and variance position and orientation errors, time of robot identification.

Pattern with one add. blob	Color 512x512	Color 1024x1024	Grey 512x512	Grey 1024x1024
$E(\sigma_\theta^2)$ [°]	0,727	0,287	0,118	0,0348
$E(b_\theta^2)$ [°]	4,268	3,110	3,301	3,070
$E(e_\theta)$ [°]	2,058	1,700	1,544	1,521
$MAX(e_\theta)$ [°]	4,385	3,381	3,821	3,430
$E(\sigma_x)$ [mm]	0,174	0,107	0,075	0,045
$E(\sigma_y)$ [mm]	0,166	0,102	0,080	0,046
$E(T_{Ident})$ [µs]	48,5	94,4	84,7	144,6
Pattern with two add. blobs	Color 512x512	Color 1024x1024	Grey 512x512	Grey 1024x1024
$E(\sigma_\theta^2)$ [°]	0,186	0,0500	0,0361	0,0108
$E(b_\theta^2)$ [°]	1,963	1,457	0,519	0,332
$E(e_\theta)$ [°]	1,290	1,057	0,674	0,494
$MAX(e_\theta)$ [°]	2,715	2,298	1,255	1,113
$E(\sigma_x)$ [mm]	0,176	0,094	0,068	0,040
$E(\sigma_y)$ [mm]	0,132	0,093	0,057	0,035
$E(T_{Ident})$ [µs]	70,8	132,2	126,3	215,4
Pattern with three add. blobs	Color 512x512	Color 1024x1024	Grey 512x512	Grey 1024x1024
$E(\sigma_\theta^2)$ [°]	0,370	0,111	0,228	0,0264
$E(b_\theta^2)$ [°]	1,425	0,970	0,750	0,316
$E(e_\theta)$ [°]	1,214	0,872	0,870	0,507
$MAX(e_\theta)$ [°]	2,683	1,831	1,785	1,116
$E(\sigma_x)$ [mm]	0,191	0,078	0,070	0,064
$E(\sigma_y)$ [mm]	0,144	0,080	0,058	0,039
$E(T_{Ident})$ [µs]	97,8	160,2	159,8	244,4
Standard FIRA pattern	Color 512x512	Color 1024x1024	Grey 512x512	Grey 1024x1024
$E(\sigma_\theta^2)$ [°]	0,0861	0,0407	0,0241	0,0081
$E(b_\theta^2)$ [°]	2,259	1,551	0,776	0,417
$E(e_\theta)$ [°]	1,352	1,109	0,670	0,552
$MAX(e_\theta)$ [°]	2,729	2,270	2,611	1,292
$E(\sigma_x)$ [mm]	0,129	0,112	0,057	0,041
$E(\sigma_y)$ [mm]	0,111	0,082	0,060	0,039
$E(T_{Ident})$ [µs]	62,4	115,8	109,3	188,3

Fig. 8. Robot's jacket.

The mean square error of the robot's position and orientation can be broken down to the sum of two components: the bias error b^2 and variance error σ^2. The standard deviation of the error can be obtained from:

$$e = \sqrt{\sigma^2 + b^2} \tag{5}$$

The bias error component depends mainly on natural lighting conditions and on the jitter error at fixed robot orientation (Fig. 5). The variance component is primarily due to rapid lighting changes introduced by fluorescent lamps.

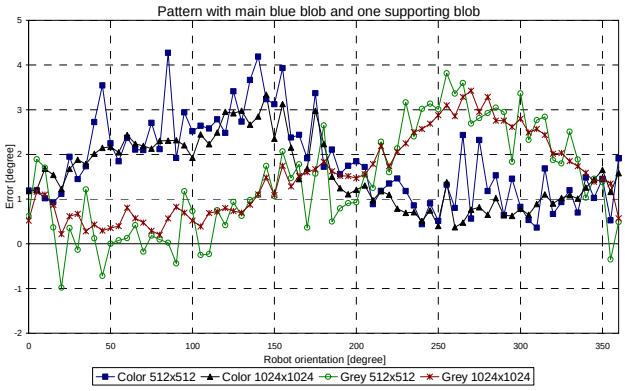
The orientation errors are showed in Figure 9 and 10. Figure 9 shows orientation bias errors and Figure 10 shows the variance components of the mean square estimation orientation errors. Table 1 shows the average bias and variance orientation errors for all tested orientations.

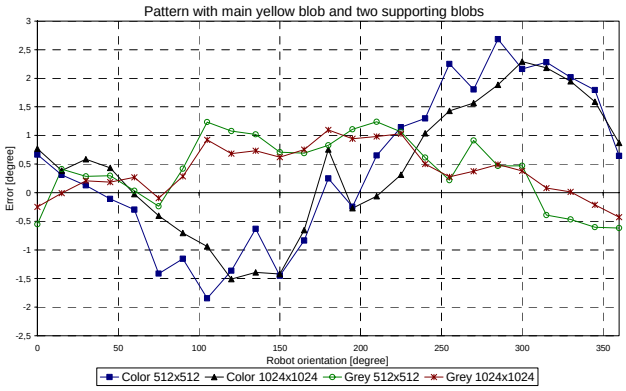
Not surprisingly standard deviations of variance position and orientation errors decrease with increasing image resolution. A similar effect can be observed if the proposed grey color segmentation is used. The same remains true for the bias error components.

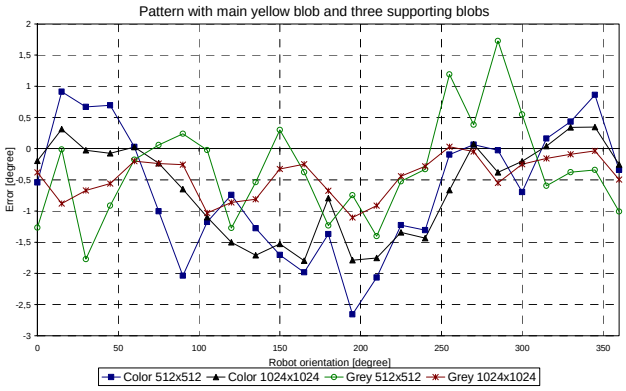
In a 512x512 image one pixel represents 3,2mm x 3,2mm square on the playground field. The average error yielded by the robot identification algorithm based on color thresholding is 1,5-2,5mm for position measurements and 2,7°-4,4° for orientation measurements. The identification algorithm which incorporates grey color segmentation yields position errors on the 1,5-2,5mm level, and orientation errors on the 1,3°-3,5° level (the actual values depend on the "quality" of the jacket).

Even though processing the image under high resolution or using the grey color thresholding increase the time needed for robot identification, it allows one to significantly increase accuracy of the vision system and hence to improve performance of the control system.

For grey color thresholding and segmentation it is very important to maintain good quality of the robot jackets surface. The surface should be uniformely dark and matt. The grey levels of the blobs and of the background should be well separated.







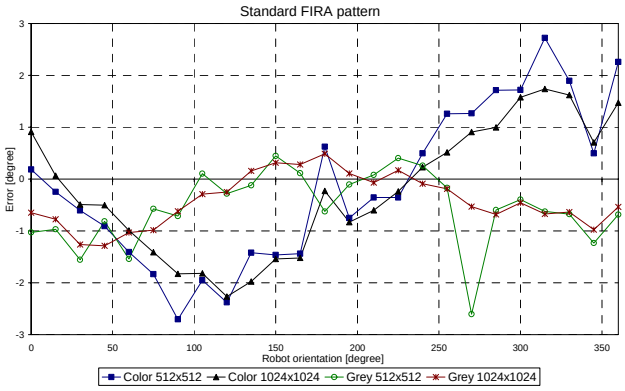
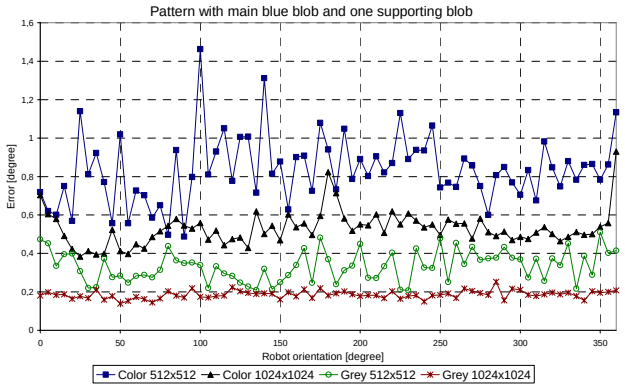
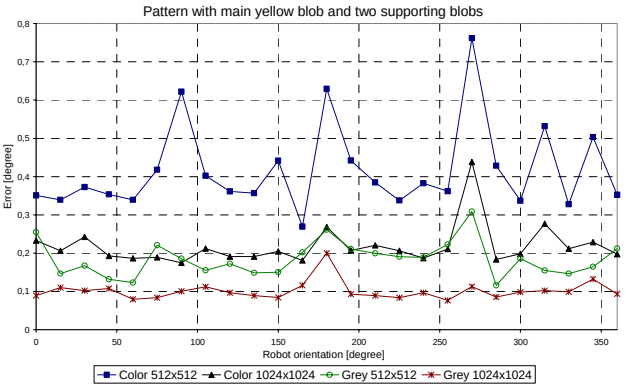
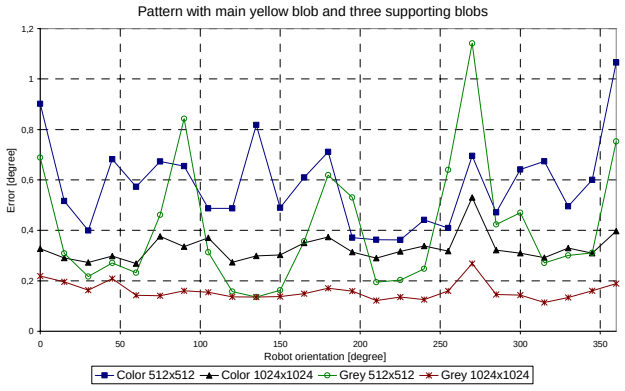


Fig. 9. Orientation bias error.







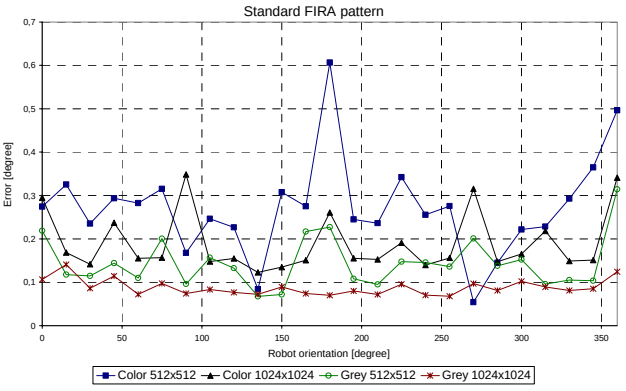


Fig. 10. Variant component of mean square estimation orientation error.

Since the inspected robot jackets were designed for color thresholding and segmentation, the grey level of the supporting blobs lied near the grey level of the robot jacket's background. Additionally, the background surface was not perfectly mate. This means that the obtained results can be further improved by designing the identification jackets more carefully.

6 Conclusions

In this paper we have described a method for fast and accurate identification of FIRA robots. First we presented the method based on color image thresholding and segmentation. Then we discussed the ways of improving accuracy and robustness of the localization system by using adaptive grey image segmentation. The presented methods were implemented and tested experimentally. The proposed method was found to be accurate and robust to changing lighting conditions.

References

[1] Bruce J., Balch T., Veloso M., *Fast and Inexpensive Color Image Segmentation for Interactive Robots*, Proceedings of the 2000 IEEE/RSJ, Vol. 3, October 2000, pp. 2061-2066.

[2] Bruce J., Veloso M., *Fast and Accurate Vision-Based Pattern Detection and Identification*, In Proceedings of ICRA'03, the 2003 IEEE International Conference on Robotics and Automation, Taiwan, May 2003.

[3] Gupta G. S., Messom Ch. H., Demidenko S., *Real-Time Identification and Predictive Control of Fast Mobile Robots Using Global Vision Sensing*, IEEE Transactions on Instrumentation and Measurement, Vol. 54, NO. 1, February 2005.

[4] Thomas P. J., Stonier R. J., Wolfs P. J., *Robustness of Colour Detection for Robot Soccer*, Seventh International Conference on Control, Automation, Robotics and Vision (ICARCV'02), Singapore, December 2002.

[5] McNaughton M., Zhang H., *Color Vision for RoboCup With Fast Lookup Tables*, Proceedings of the 2003 IEEE International Conference on Robotics, Inteligent Systems and Signal Processing, Changsha, China, October 2003.

[6] Jiang H., Peng Q., Lee HA., C Teoh EL., Sng HL., *Colour Vision and Robot/Ball Identification for a Large Field Soccer Robot System*, 2nd International Conference on Autonomous and Agents, Palmerston North, New Zealand, December 13-15 2004.

[7] Weiss N., Hildebrand L., *An Exemplary Robot Soccer Vision System*, CLAWAR/EURON Workshop on Robots in Entertainment, Leisure and Hobby, Vienna, Austria, December 2-4, 2004.

[8] Borensztejn P., Jacobo J., Mejail M., Stoliar A., (ed.) *Design of the Vision System for the UBA-Sot team*, Description of vision system UBA-Sot team, Robot-soccer FIRA competition 2002.

[9] Ball D., Wyeth G., Nuske S., *A Global Vision System for a Robot Soccer Team*, Proceedings of the 2004 Australasian Conference on Robotics and Automation (ACRA), Canberra, Australia, 2004.

[10] Dutkiewicz P., Kiełczewski M., *Sprzężenie Wizyjne w Sterowaniu Grupą Robotów Mobilnych*, Artykuł Umieszczony na Stronie Koła Naukowego Studentów Automatyki i Robotyki Politechniki Poznańskiej, http://cybair.cie.put.poznan.pl/system_wizyjny.htm, March 2003.

[11] Thornton J., Leonard J., Wisbey R., Lee Y. Ch., Spanks K., *Shape Recognition and Enhanced Control Systems for Robot Soccer*, FIRA Robot World Cup 2002 in Korea.

[12] Wong Ch., Huang Ch., *Role Construction and Recognition in Robot Soccer Games*, Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control, Taipei, Taiwan, March 21-23, 2004.

[13] Klančar G., Kristan M., Karba R., *Wide-angle camera distortions and non-uniform illumination in mobile robot tracking*, Robotics and Autonomous Systems 46 (2004) 125-133, Elsevier, received in revised form 15 May 2003.

[14] Bayer B. E., *Color imaging array*, U.S. Patent US3971065, 1976.