Engineering International Conference "UNNES Conservation" 2013 Proceeding
p- ISBN: 979-25-2783
January 08, 2013, Semarang, Indonesia

Optimizing Ball Detection Algorithm using Center of Gravity method and Servo Pan and Tilt Controller for Humanoid Soccer Robotic based on Android

Reva Diaz Airlangga[1], Saptadi Nugroho[2], Banu Wirawan Yohanes[3]
Electronic and Computer Engineering Department
Satya Wacana Christian University
Jalan Diponegoro 52-60 Salatiga 50711, Jawa Tengah, Indonesia
[1]*(rev_1204@yahoo.co.id)*
[2]*(saptadi_nugroho@yahoo.com)*
[3]*(bona_yo@yahoo.co.id)*

Abstract—*The humanoid soccer robotic which contested in Kontes Robot Cerdas Indonesia (KRCI) has to recognize the orange tennis balls in order to response as quick and accurate as possible. This paper presents an improving performance of a ball detection algorithm using Center of Gravity method on YUV420sp colorspace compared to the one on RGBA colorspace. And it also proposes a non-linear servo pan and tilt controller algorithm to increase system response. Testing has been done by conducting series of simulations based on scenarios according to the contest rules. The experiments took an average from 100 sample of differentiate times. The result is an increasing ball detection runtime performance from 80ms to 22ms on average, or improving by 363%. The better a robotic doing ball detection and image processing, the faster it will move to follow the ball's moving by controlling its servo. From the experiment results, it concluded that the humanoid soccer robotic has increasing its system response.*

Keywords —*Humanoid Soccer Robotic, Center of Gravity, Servo Controller*

I. INTRODUCTION

Kontes Robot Cerdas Indonesia (KRCI) since 2011 included the Robosoccer Humanoid League. It is a contest where robotics plays football-like or soccer-like game. Two teams compete to score goals, and the team with the highest scoring goals is the winner of the game. However, since 2012 the contest allowed contestants to use an addition robotic as a goal keeper. But, algorithm implementation for the goal keeper role is still so simple. Although research on the development of humanoid soccer robotic has been long time, there are some inadequacies in the functionality and performance features, compared to human beings [1].

Image processing used in the goal keeper robotic consumes much time, so that the robotic often misses the ball and poorly responses the ball that coming unto it. Therefore, robotics needs to optimize the image processing algorithm used in the game. In addition, ball detection accuracy performance is another important task. When the contest took place, there were lots of environmental distractions in the soccer field, such as ball's color similarity with the poles and referee's uniforms, lights disparity, and uneven surface. Those problems could reduce the robotic performance, hence algorithm used by the robotic has to be simple and fast yet accommodate such scenarios in the soccer field.

This paper presents an algorithm optimization for the humanoid soccer robotic to improve the speed and accuracy of the ball detection for the goal keeper role. The paper's structure consists of the humanoid soccer robotic implementation, ball detection algorithm, servo controller, experiments and conclusion. By means of the algorithm research, it is expected that students much more attracted to the robotic, image processing, and mobile programming area.

II. HUMANOID SOCCER ROBOTIC

The ball detection system for the humanoid soccer robotic composed of hardware and software parts. The system diagram is appeared in Fig. 1.

II.1. Hardware

The hardware component is an actuator system to control the camera for the robotic vision. The goal keeper robotic utilized an Android smartphone as the sensors and the main processor, and also a microcontroller embedded system [2] to control servos pan and tilt.

Because of the Android smartphone has been included by many sensors such as camera and compass [3], besides it has communication modules such as bluetooth and wifi. The cost using the smart phone may be less expensive than using the modules separately.

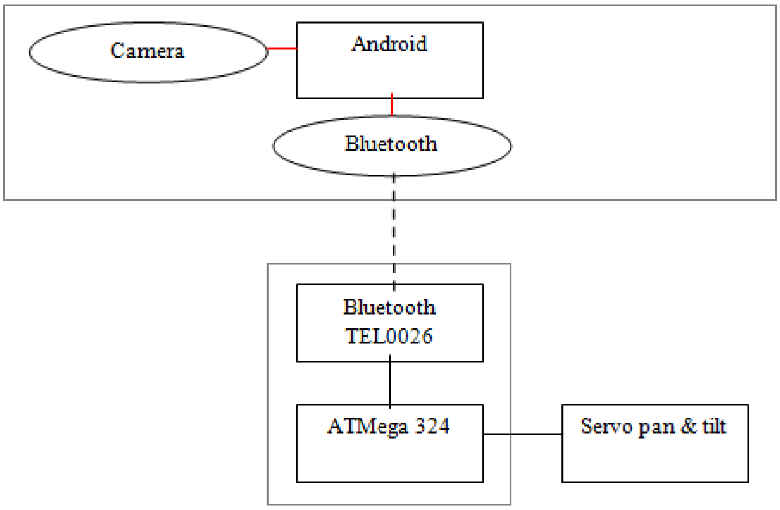

Fig 1. Humanoid soccer robotics block diagram

The Android smartphone's tasks are to take picture using its camera, processing images to detect a ball, communicate with the microcontroller through bluetooth, and take decisions based on image processing system.

The microcontroller embedded system is the support party which communicates with the smartphone using bluetooth module and controls the servos pan and tilt.

II.2. Software

The software parts are image processing system and servo controller that create the robotic's intelligence. Software programs implemented using Java [4] and Android [5] software development kit (SDK).

The Android software compiled of four main tasks, that is, first to create user interface include display layouts and components' event listener. Second, to control the whole system include to deactivating auto white balance feature because it disrupts the color segmentation process. Third, image processing includes creating binary images and calculating the image's center of gravity. Fourth, to manage the communication to microcontroller embedded system using bluetooth.

The microcontroller embedded systems software is to control the servos' movement using fast pulse width modulation (PWM) method.

III. BALL DETECTION

III.1. Colorspace

Colorspace is a mathematical abstraction model [6, 7] to describe color representation in numbers, and it usually consists of three or four color elements, such as RGB, CMYK, HSV, HSL [8], or YUV. Each base element in a colorspace is color composer.

III.1.1. RGBA colorspace

Media that transmitting color, such as television and LCD display, creating image using mix of additive color from prime color red, green, and blue. In addition to RGB, RGBA colorspace added another element, alpha, which contains transparent information from mixing RGB color. Alpha value 100% will display color opaquely.

III.1.2. YUV colorspace

YUV colorspace defines color using luminance element (Y) and two chrominance or color components (UV). It is an improvement from color information used in old black and white television which using only Y element. The Y element is a brightness level, and the U and V components are color information. The Android smartphone keeps the image taken by its camera in YUV420 colorspace. YUV 420sp colorspace defines in Fig. 2 where each 4 pixel consists of 4 bytes Y, a byte U and a byte V.

III.2. Taking Picture

Images being taken by an Android smartphone are in YUV420sp colorspace and kept in array of bytes. Previous works based on images obtained from an omnidirectional camera [9]. In the 2012 contest, the humanoid soccer robotic using OpenCV library [10] to take picture and kept in RGBA colorspace using Mat type. To synchronize the screen resolution, pictures are taken in 480 x 320 pixel resolution.

III.3. Color Segmentation

Color segmentation is used to separate orange ball color with other environmental color. The segmentation conducted by set upper and lower bounds for element values in images. As a results, binary image which consists of 0 for black and 1 for white.

III.4. Center of Gravity method

Center of mass or center of gravity on an image is a point where image object are spread evenly. After color segmentation had been done, the center of gravity on binary or grayscale image is calculated [11].

IV. SERVO CONTROLLER

The humanoid soccer robotic used microcontroller ATMega 324 to control servo pan and tilt. Because of the communication between an Android smartphone and microcontroller is not only used to control servo, but also to control the robotic. Hence the robotic needs to differentiate both data. Each data for controlling servo and robotic has different header and tail. The byte configuration is presented in Table 1.

Table 1. Byte configuration for communication between Android smartphone and microcontroller

Function	Byte 1	Byte 2	Byte 3	Description
Pan	255	Value	Tail	Controlling servo pan position using value between 0°-180°
Tilt	254	Value	Tail	Controlling servo tilt position using value between 0°-90°

Servo pan configured to rotate 180° with 1° for each step, so there are 180 steps. Meanwhile, servo tilt configured to rotate 90°, so there are 90 steps. Servo rotation is used to search and follow the ball movement.

V. RESULTS AND ANALYSIS

Simulations are drawn up in accordance with the latest contest rules. In 2012 contest, servo is moved linearly one step per frame or 1° per frame. When each frame has average runtime 80 ms, so the maximum servo speed is only 12.5° per second. On that account, it slowed the servo movement. The new servo pan and tilt controller is not moving linearly, but adjust the distance of the ball with center of image. The farther the distance then servo movement will be quicker. Highest speed achieved is 6 steps per frame. The average runtime at 22 ms, then the average servo speed is 272.3° per second. The servo motor used has a 2.7 kg.cm torsion and speed at 60° per 130 ms or 461.5° per second.

The experiments to check algorithm performance on the humanoid soccer robotic are categorized into four groups, that is, image processing speed, noise invulnerability, light intensity alteration tenacity, and ball detection accuracy. The experiments are conducted 100 times to compare system runtime to detect a ball for each frame. A ball is located in an environment that has many objects with similar color with the orange ball. The objects are located on different distance from robotic and also varied lights intensity. When testing conditions are changed, the value boundaries are kept constant and the robotic is turned to show a different circumstances. System can detect the ball correctly when the crossing red-line located in the center of image. The simulation results are presented in Table 2.

It can be seen that YUV420sp colorspace is perform better than RGBA colorspace. When lights intensity is reduced, the ball detection performance of both colorspace is degraded, but the reduced accuracy of YUV420sp colorspace is less than RGBA colorspace.

\

Table 2. Contest simulation results

Condition (lights brightness, number of lamp, ball distance from robotic)	YUV420sp colorspace	RGBA colorspace
Bright lights, 2 lamps, 20 cm	100%	100%
Bright lights, 2 lamps, 40 cm	100%	100%
Bright lights, 2 lamps, 100 cm	100%	85%
Bright lights, 2 lamps, 200 cm	90%	60%
Bright lights, 2 lamps, 300 cm	80%	40%
Dreary lights, 1 lamp, 20 cm	100%	100%
Dreary lights, 1 lamp, 40 cm	100%	95%
Dreary lights, 1 lamp, 100 cm	90%	70%
Dreary lights, 1 lamp, 200 cm	75%	40%
Dreary lights, 1 lamp, 300 cm	65%	20%

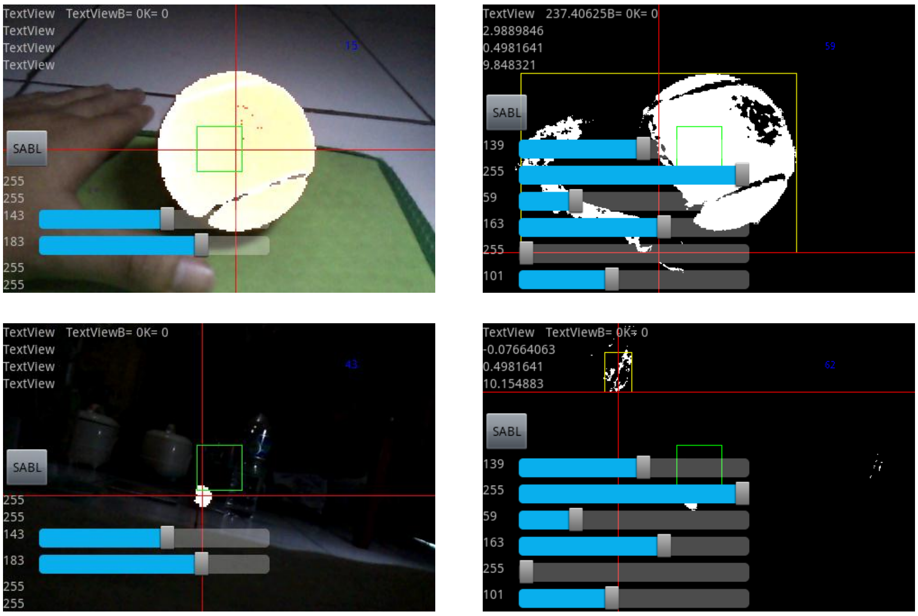

Fig 2. Experiment results, using YUV420sp colorspace (left) and using RGBA colorspace (right)

The color segmentation is done using YUV420sp colorspace, but only U element employed to segment image. Besides shorten time to process images, the segmentation result is not degraded significantly. Optimized algorithm using YUV420sp colorspace takes 15-25 ms to process an image, its speedier than the old algorithm RGBA colorspace which takes 60-100ms. Besides the optimized algorithm also presents a better color segmentation than the old one, the color segmentation can perform well even when hand approach the ball captured in image. The new algorithm is also more endurable to noise than the old one, when lights intensity are varied color segmentation still performs well and center of gravity coordinate is settled. At last, the new algorithm still can process ball object positioned at 2 meters far but the old one cannot. The experiment results can be seen in Fig. 2.

VI. CONCLUSION

This paper has discussed a comparison in color segmentation and calculating center of gravity between two different colorspace, that is, RGBA and YUV420sp. Both colorspace are implemented on an Android smartphone. From the experiment results, it can be concluded that YUV420sp colorspace is perform better than RGBA colorspace for humanoid soccer robotic which used to compete in KRCI. There have been many improving features on image processing speed and accuracy, besides a quicker non-linear servo controller.

Further research can be planned on the shape detection algorithm to improve image processing performance and not to display image processing when the contest is going on, not only to lighter the processing but also to give a better security.

ACKNOWLEDGEMENTS

The reported work was supported by Robotic Research Center (R2C), Electronic and Computer Engineering Department, and Satya Wacana Christian University, Salatiga, Indonesia.

REFERENCES

[1] H. Mobalegh, *Development of an autonomous humanoid robot team*, Ph.D. Thesis, Freie University of Berlin, 2011.

[2] D. Susilo, *48 jam kupas tuntas mikrokontroler MCS51 & AVR* (Andi, 2010).

[3] Robocup, *RoboCup soccer humanoid league rules and setup*, 2011.

[4] S. Hermawan, *Mudah membuat aplikasi Android* (Andi, 2011).

[5] U. Ahmad, *Pengolahan citra digital* (GrahaIlmu, 2005).

[6] T. Acharya, A.K. Ray, *Image processing principles and applications, first edition* (New Jersey: John Wiley & Sons, 2005).

[7] N.S. Widodo, A. Rahman, Vision based self localization for humanoid robot soccer. *Telkomnika*, 10(4), December 2012, 637-644.

[8] F.M. Carlos, U.L. Pedro, *Vision-based self-localization for soccer robots, Proceeding of the 2000 IEEE/RS International Conference on Intelligent Robots and Systems*, 1193-1198.

[9] G. Bradsky, *O'Reilly Learning OpenCV computer vision with OpenCV library* (O'Reilly Media, 2008).

[10] Schildt, Herbert, *Java : the complete reference* (McGraw-Hill, 2011)

[11] Nugroho S, Utomo D, Rotation Invariant Indexing For Image Using Zernike Moments and R–Tree, *Telkomnika Journal*, 9(2), 2011, 335-340.