Team KMUTT: Team Description Paper

Thavida Maneewarn, Xye, Pasan Kulvanit, Sathit Wanitchaikit,

Panuvat Sinsaranon, Kawroong Saktaweekulkit, Nattapong Kaewlek

Djitt Laowattana

King Mongkut's University of Technology Thonburi,
Institute of Field Robotics (FIBO)
126 Pracha u-tid Rd., Bangmod, Thungkru, Bangkok 10140, Thailand
praew@fibo.kmutt.ac.th
http://fibo.kmutt.ac.th

Abstract. This paper explains the scientific achievement of our Robocup 2010 Humanoid league team. The new locomotion software is implemented in our humanoid robot this year which allow us to design more complex control algorithm into the robot. With the current robot design, the robot can achieve a fast dynamic walk and more precise ball manipulation. We implement a vision-based navigational scheme that can give accurate visual tracking. We also improve the localization module which use multiple visual landmarks to identify the location of the robot in the game field. Our two-legged robot can autonomously traverse the game field while recognizing the target (colored ball) and environment (opponents, the obstacles, game field) and execute the actions according to appropriate decision.

1 Introduction

Team KMUTT has been participating in the RoboCup Humanoid League since 2005. The Team KMUTT was in the quarterfinals in the 3-3 games of RoboCup 2009. Figure 1 shows the robots from the Team KMUTT (taken at RoboCup 2009) that will enter the Robocup 2010 Humanoid league competition. Details specification can be found at http://fibo.kmutt.ac.th

In the game of soccer, an individual player must run fast, think in real-time and be agile in order to gain advantage over the other players. In this year, we focus our research and development on the modular design of locomotion and motor cortex layer as well as the localization algorithm.

In this paper, we will describe our humanoid robot systems that will be used in the competition. The main focus of this paper is what we have done to improve our team from last year. Section 2 shows the competition-ready system overview. Section 3 demonstrates our idea of vision based navigation. Section 4 illustrates some of our AI strategic game plan. Section 5 concludes the paper.

2 System overview

This section explains the hardware used in our biped robots. Each robot is composed of mechanical hardware, sensors, and computing hardware. Figure 2 shows the over-all systems. Mechanical hardware is composed of robot structure and motor. The structure of both robots is made from aluminum alloy sheet metal with some parts are made from Kevlar carbon fiber in order to keep the weight low while benefiting from the high strength property. Both robots use 20 RS-485 networked servo-motors.

All robots use the same set of sensors. These are 2-axis accelerometer [+/-2g], 3 rate gyros [+/-100 deg/sec] and one CCD USB camera. The camera installed on the robot is a single lens USB webcam which has the horizontal field of view at 60 degrees. The capture resolution is 320x240 pixels2. The workspace of pan-tilt unit is -120 to 120 degrees in panning and 0 to -80 degrees in tilting. The accelerometer tells the robot if there is any longitudinal and/or transversal tilt. The three rate gyros measure angular velocity in three axes of rotation. Kalman filter is used to estimate the body angle from gyro and accelerometer information. The angular position and velocity information will be used to adapt the attitude of the body during walking. The camera is used to track the ball and other objects of interest, which is crucial for navigation decision-making software.

The main computer for all robots, which was PC-104 with 500MHz processor last year, is replaced by the PICOITX (PICO820) SBC with ultra low power Intel AtomTM processor Z500 (1.6GHz CPU) . The PICO820 board computer receives information from the CCD camera via the USB port. The computer computes the walking path and sends locomotion command to the ARM 7 (60MHz RISC microprocessor) motor controller via RS485 port.

Fig. 1. *KM-Series* at RoboCup 2009

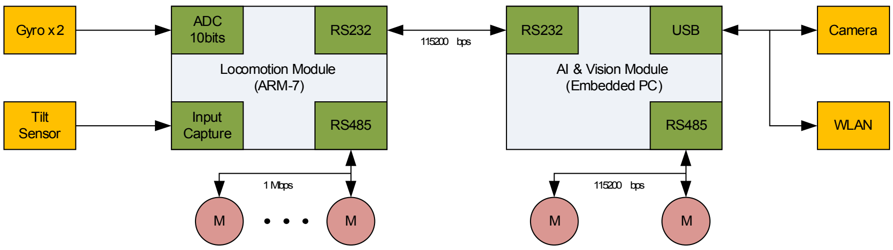

Fig. 2. System overview of the humanoid *KM-Series*

Low-level interface

In the 2009 humanoid robot, we only send the command and control parameters to the motors without any feedback readings. In the 2010 system, the position, velocity, load , voltage and temperature of all motors are constantly reported back to the PC via the ARM-7 that acts like a smart router controller. The accelerometer and gyro sensor readings also automatically report to the PC via the smart router. This allow us to gather all information of the robot in real-time into the PC. With this system, the locomotion module was moved to the PC. The locomotion code is written in Python, which allow us to adjust the locomotion and display all sensor feedback during the development phase.

We also add the Kalman filter into the sensor fusion module, therefore, the body angle around X and Z axis of the robot can be more accurately estimated. This ability allow us to experiment with new algorithm of balancing on the robot, where not only angular velocity is concerned but also the body angles.

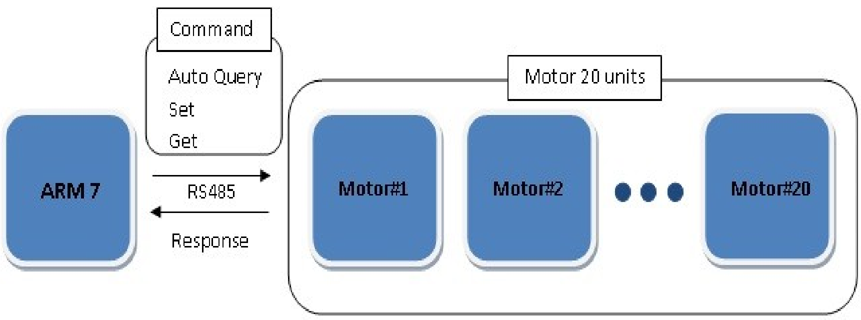

Fig. 3 . Block diagram of the communications between ARM7 and motors

Motion Editor

With the new low-level interface that the information of all motors and sensors are sent to the PC all the time, the configuration of the robot can be easily captured, displayed, adjusted and playback. The motion editor software is also developed using Python (with libraries such as pyFLTK and OpenGL) The GUI interface shows the configuration of the robot that are updated from sensor readings in the low-level interface. These readings are converted to the robot coordinate frame and displayed graphically.

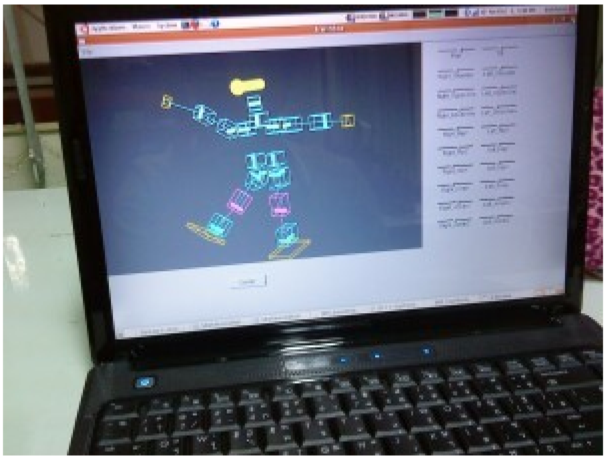

Fig. 4 Motion editor program

In the near future, these features will be implemented in the motion editor applications

1. sensor feedback visualization
2. posture and dynamical simulation
3. motion debugging capability

3 Vision based navigation

To obtain objects information associated to the game play, the robots must have abilities to identify and distinguish each object in the field including the abilities of locating the objects and their movement. These can be solved by implementing vision-based navigation system which consist of two major parts, the object recognition and objects observation. After the objects information is known, the robot can determine a proper action to play the game as shown by the simple algorithm for tracing the ball in the end of this section.

3.1 Object Recognition

Each object type in the field is clearly assigned a specific color, so the recognition can be done by focusing on the color feature of each type. To do this, a color segmentation technique, called color scoring, which based on fuzzy concept are implemented.

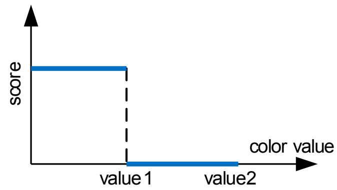

Fig5: Color Segmentation using HSV with exact values, 0 and 1

Typically, color segmentation is performed by using hard threshold on every pixel value of an image in a traditional color space, such as RGB or HSV. The pixels which have their values according to the threshold values are given a score 1 and declared as member of the color class. Otherwise, the score is 0 and declared as the member of the complement set of the color class. This is shown in the Fig 5.

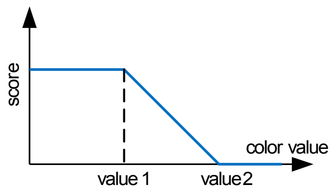

Fig6a: Color segmentation using image scoring with low brightness of high values

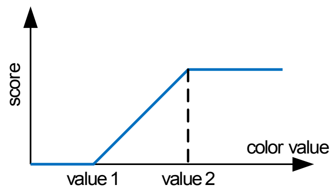

Fig6b: Color segmentation using image scoring with high brightness of high values

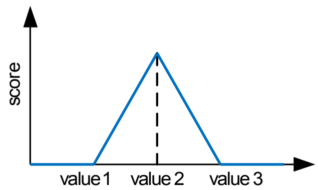

Fig6c: Color segmentation using image scoring with high value of medium brightness

Instead of doing above approach, the image scoring approach uses the relation among each color channel value and the fuzzy concept to assign a score for each pixel as shown in the Fig 6. The scores of each color channel are defined between 0 and 1. The score to the color class of each pixel are the product of score of its channels.

After testing both segmentation schemes, the image scoring is more efficient than HSV transform because the image scoring method adds up brightness values of each pixel to the values between 0 and 1. Therefore, it is possible to get more efficiency than fixed values

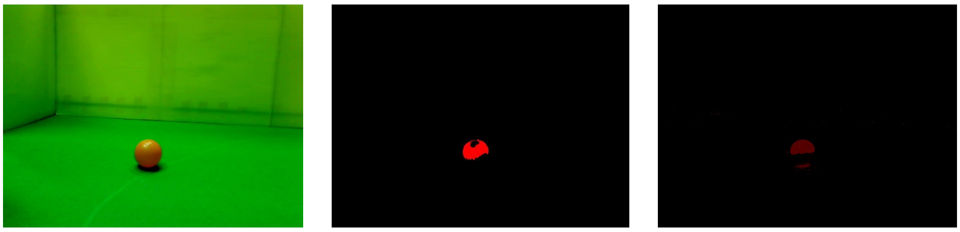

Fig7 Evaluated image (Left), HSV transform (middle), image scoring(right)

3.2 Object Observation

After recognizing every object in the scene, the objects position referenced to the robot are determined. This process assumes the camera is well calibrated, so the intrinsic and extrinsic parameters of the camera installed on the head pan-tilt is obtained and the robot knows the head pan and tilt angle of each scene.

To compute the angle to the object from robot reference, the horizontal pixel position of the object centroid in the scene is converted to the angle deviation from the center of the scene. Then, this value is combined with the head pan angle to obtain the object angular position referenced to the robot reference. For the distance to an object, the head tilt angle associated with the known height of the robot is used to compute the distance. This is shown in the Fig 9.

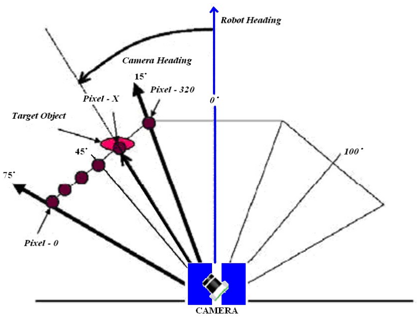

Fig. 8 Determination of robot heading toward the targeted object (Top View)

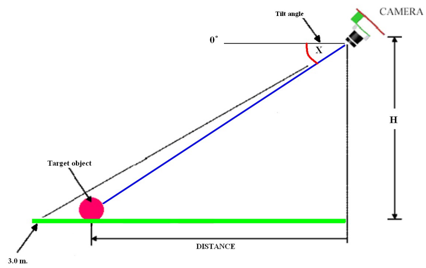

Fig. 9 Determination of the distance from the robot to the target object (Side View)

3.3 Simple Goal Scoring Algorithm

The robot implements a visual servo algorithm for walking toward the ball. The ball and the goal positions are fed to the control unit. Then, proper actions for guiding the robot to the ball are determined. The advantage of this algorithm is that the robot always reaches the ball by this guiding as long as it sees the ball. The operation of each robot can be simplified as shown by the following state diagram.

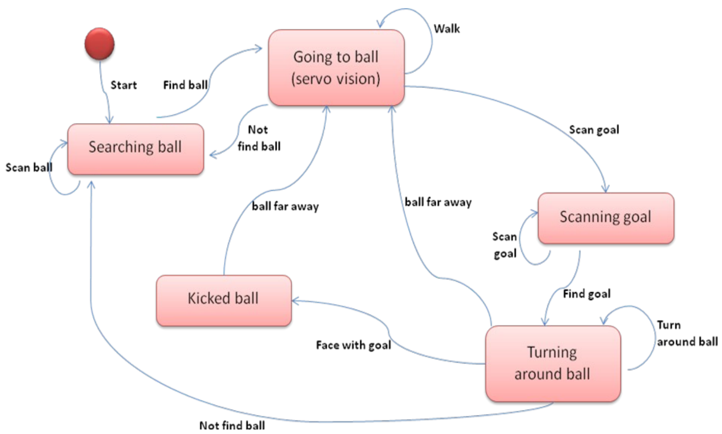

Fig 10 State diagram of walking ability toward the ball

4 Game plan

Game plan is crucial for soccer team playing. This is especially more important this year since the game will be 3 VS 3. Each of the three robots in a team will be assigned duties as follows:

Striker	Search for ball, shoot the ball to opponent goal
Mid-field	Search for ball, feed the ball up field, defense
Goalie	Search for ball, feed the ball up field, defense

All robots can communicate with each other via peer to peer wireless LAN. The Host PC is only used for monitoring robots status. The team network is shown in Fig. 11. The communication is use for tasks as follows:

Peer avoidance	Avoiding the collision between robots from tracking the same target
Roles swapping	Increasing the chance of attacking or defending for open players
Target declaration	Helping peer robot look for target when target is locally out of sight

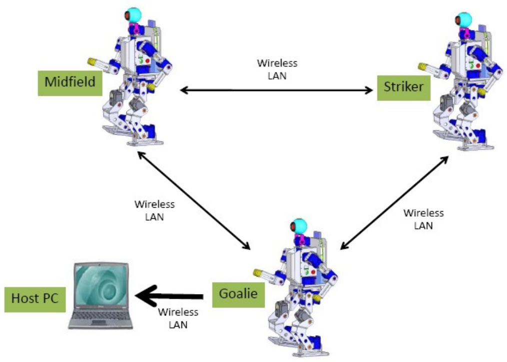

Fig. 11. Wireless communication network for Team play

5 Conclusions

Since 2008, we have designed and built a complete humanoid robot system KM-series that can play the game of soccer autonomously. For the 2010 RoboCup, we have refined our system by modularized the locomotion control that can be seamlessly integrated into the existing high level decisioning module. The new system will be more robust due to the modularized design in both hardware and software. We are still focus on improving four major parts of the system- smarter dynamic walking, quicker vision tracking, robust localization, and cooperative team play.

References

1 Laowattana D., et al.: Team KMUTT, Team description paper. Robocup 2009, Humanoid League (2009)

2 Kulvanit, P., et al.: Team KMUTT, Team description paper. Robocup 2008, Humanoid League (2008)

3 Kulvanit, P.: Design and Realization of bipedal walking robot with optimal fast gait. Ph.d. dissertation, King Mongkut's University of technology, Thonburi, 2007

4 Kulvanit, P., Srisuwan, B., Laowattana, D.: Optimum Performance of The Fast Walking Humanoid Soccer Robot: Experimental Study, International Conference on Climbing and Walking Robots, CLAWAR, 2007

5 Sawasdee, W., Kulvanit, P., and Maneewarn, T., "Locomotion Control Scheme for Fast Walking Humanoid Soccer Robot", International Conference on Climbing and Walking Robots, CLAWAR, 2007