Eagle Knights
Small Size League Team

Luis A. Martinez-Gomez, Francisco Moneo, David Sotelo, Victor Soto, Juan P. François, Sahima Murguía, Alfredo Weitzenfeld

Robotics Laboratory, ITAM

Rio Hondo 1, San Angel Tizapan

Mexico City, MEXICO, 01000

email: alfredo@itam.mx

***Abstract*—We present the description of our Small Size League Team – Eagle Knights. With this team we have won 3rd place in the American Open 2003 and 2nd place in the US Open 2004.**

First we will explain the three main components of our architecture: Vision System, AI System and Robots. Each element is an independent entity and therefore the explanation focus in the overall functionality of the component and the way it interacts with the rest of the elements in the team architecture.

I. INTRODUCTION

RoboCup[1] is an international joint project to promote AI, robotics and related field. In the Small Size League, two teams of five robots up to 18 cm in diameter play soccer on a 4 by 5.4 m carpeted soccer field.

We have a classic architecture of a team in the *Small Size League* (SSL) with four main elements: the vision system, the AI system, five robots and the referee box.

The vision system digitally process two video signals from the cameras mounted on top of the field. It computes the position of the ball and robots on the field, including orientation of the robots in its own team transmitting the information back to the AI system.
The AI system receives the information and makes strategic decisions. The actions of the team are based in a set of roles (goalkeeper, defense, forward) that exhibit behaviors according to the current state of the game. To avoid collision with robots of the opposite team an obstacle avoidance module is used. The decisions are converted to commands that are sent back to the robots via a wireless link. The robots execute these commands and produce mechanical actions as ordered by the AI system. This cycle is repeated 60 times per second. Finally the referee can communicate additional decisions (infraction, goal scored, start of the game, etc.) sending a set of predefined commands to the AI system through a serial link.
A picture of the Eagle Knights robots is shown in Figure 1.

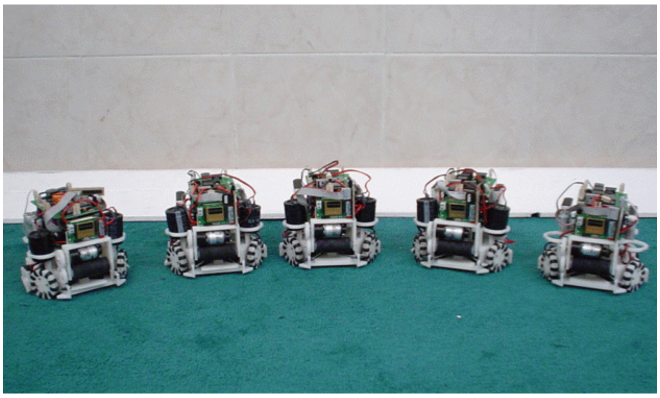

Fig 1. Eagle Knights robots

II. VISION SYSTEM

The vision system is the only source of feedback in the whole architecture, if the data given by the vision system is wrong the overall performance of the team will be severely affected. That's why the vision system should be robust enough to compensate for any possible mistakes.

The main object characteristics used by the vision system are the colors defined in the rules [2] of the SSL. The ball is a standard orange golf ball. The robots of one team must have on top of them a 50 mm blue colored circle while the other team must have a yellow patch.

The main tasks of the vision system are:
1. Capture video from the cameras mounted on top of the field in real time.
2. Recognize the set of colors assigned in the rules to the objects of interest in the field (robots and ball).
3. Identify and compute the orientation and position of the robots in the team
4. Compute the position of the robots of the opposite team.
5. Transmit the information back to the AI system.

6. Adapt to different light conditions (color calibration procedure).

The system is modular to allow future updates and improvements. It has several modules, each module it's a functional block with a specific task. Figure 2 shows the vision system architecture.

1. CAPTURE MODULE. Assign a physical capture device to the cameras, the type of connection (IEEE1394, S-Video, Composite, etc), the resolution of the image, and the frame rate.
2. PREPROCESSING MODULE. Modify the quality of the image, such as brightness, contrast, gamma, etc.
3. OBJECT CALIBRATION MODULE. This module is a tool to establish the thresholds of each component according to the space color for every object of interest (robots and ball).
4. SEGMENTATION MODULE. Separate each pixel of the images into object classes. The module consist of two segmenters, each one using the thresholds values assigned to the camera for every object of interest.
5. BLOB BUILDER MODULE. Connects the segmented pixels into blobs. Before reaching this module the image is composed of separate pixels; when a blob is constructed useful information is computed such as the area, centroid, bounding box, etc. A joint list of blobs for the two cameras is generated for each color.
6. ACTIVATION/DESACTIVATION MODULE. Enables or disables the use of a particular robot. Sometimes a team can play with less robots so this information is useful to avoid unnecessary searching processes.
7. RECOGNITION MODULE. Selects the regions that adjust better to the objects searched. It has a selection criteria for every kind of object. For the ball we select the orange blob that is nearest to an area of 85 pixels (with an image resolution of 640x480). For the robots of the opposite team the selection criteria consists in selecting the blobs of the corresponding color of the central patch with an area nearest to 115 pixels (the area of the patch is bigger than the ball). The number of blobs selected are determined in the Activation/ Desactivation module. For the team the procedure is similar to the one used for the robots of the opposite team, but additional to the central patch, a search for extra patches is necessary. The extra patches are employed for identification and orientation computation.
8. GEOMETRIC CALIBRATION MODULE. This module computes the internal and external parameters of the cameras using the Tsai method [3]. This parameters are used to correct the distortion produced by the lenses of the camera.
9. LOCALIZATION MODULE. Computes the position of all objects in the field. It uses the camera parameters obtained in the Geometric Calibration module to undistort the image. Also computes the orientation of robots in the team.
10. GRAPHIC DISPLAY MODULE. It's responsible for displaying video images in the screen and for generating basic drawing functions such as lines, circles, etc. in the video image.
11. TRANSMISSION MODULE. A UDP network link is setup for the communication between the vision system an the AI system. The module builds a structure appropriate for data transmission. In practice the vision system can perform communication with two or more hosts allowing for distributed AI system processing if necessary.

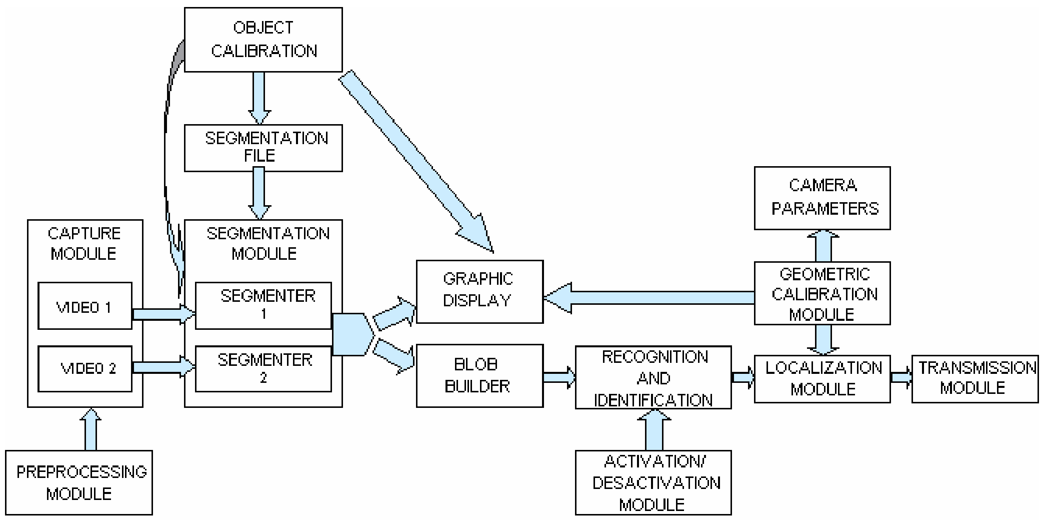

Fig 2. Vision System Architecture

III. AI SYSTEM

The AI System or High Level Control Application is formed by seven modules named: Artificial Intelligence, Collision Detection, Transceiver Communication, Omni-directional Drive Control, User Interface, Vision's System Communication and Game Control. The system has a main thread that loops all the time and calls function in each of the different modules. This system is designed in a way that the user can test each module separately. It doesn't need all modules connected and working to make simple tests. The system also has a dynamics simulator so that we don't need the robots to make collision detection, AI or drive control tests.

1. MAIN THREAD. The Control System has a function that loops all the time that we call the main thread. In the iteration we call each of the functions of the different modules to finally get the commands sent to each one of the robots. The main thread first checks for the vision system's information to know where the robots are. Then we have to check the game state we are in (game state is controlled by the referee). With the robot's coordinates and the game state we call the AI function that returns the desired position to move for each robot and the actions to take. Each robot has to avoid collisions so the future positions are sent to the collision detection function that returns the moving vector that avoids the obstacles. With the moving vector we have to calculate the speed for each one of the three wheels. This task is done by the drive control function. Finally we build the packets for all the robots with the speed and the actions and we send them with the help of the transceiver.
2. AI MODULE. This module receives the robot's positions, the ball position, the robot's angles, the game state, the robot's roles, the strategy and the direction we're shooting. With all this information the system calculates the future position and the actions for each robot. We can have different strategies that give us the chance to change the game play for each team or situation. We can program and integrate a new strategy in a very easy way thanks to the simple design of this module.
3. COLLISION DETECTION MODULE. This module receives the movement vector, robot's positions, the ball position, and the robot's angles and returns the new movement vector that assures that there will be no collisions on the movement.
4. TRANSCEIVER COMMUNICATION MODULE. This module receives the speed for each one of the robot's motors and the actions to take. This module builds the packets that we will send using our transceiver. It also makes sure the communication is active at all times.
5. OMNI-DIRECTIONAL DRIVE CONTROL MODULE. This module receives the movement vector and returns the speed for each one of the three robot's motors. Since we have three omni-directional wheels we need this module to know the speed on each motor so that we move on the desired direction.
6. USER INTERFACE MODULE. This module is constantly displaying all the information we need for each robot. We have the coordinates, the angle, the motor's speeds, the desired positions, the id, the actions, the game state, the referee's commands and the game positions for each robot. The coordinates, angles, desired positions and actions are displayed graphically in an GUI programmed using OpenGL.
7. VISION SYSTEM COMMUNICATION MODULE. In this module we receive the packets sent by the vision system containing what we call the scenario that has the robot's coordinates, the ball coordinate and the robot's angles.
8. GAME CONTROL MODULE. This module receives all the referee commands through the serial interface and returns the game state of the game. This module is based an the SSL rules.

Figure 3 shows the interaction between the modules.

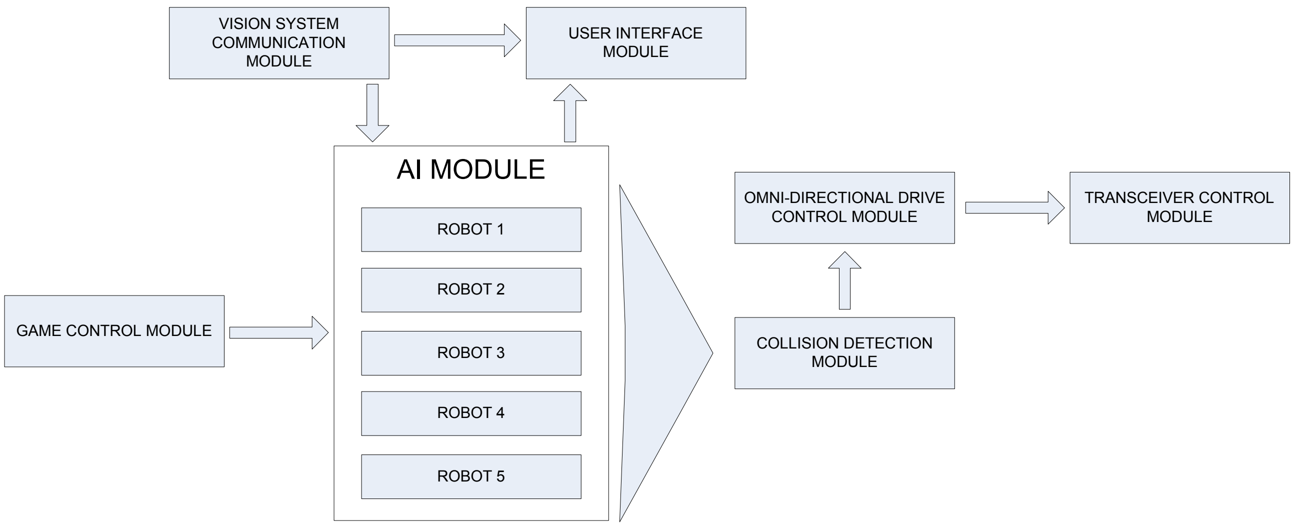

Fig 3. Artificial Intelligence System

IV. ROBOTS

We design and built five omni-directional robots. Each robot has four motors Maxon [4] A-Max 22 (three for the wheels and one for the dribbler), a solenoid, a DSP, a transceiver, two printed circuit boards and two batteries.

Figure 4 shows the final design of the Eagle Knights robots.

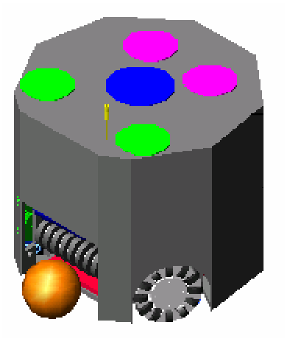

Fig 4. Robot Design

The robot receives commands from the AI system and executes them. To accomplish that each robot has the next functional elements:

1. ROBOT ID. Each robot incorporates an identification circuit manually setup with a dipswitch making it easy to modify its id if necessary any time during the match.
2. DSP. The robot micro controller is the Texas Instruments TMS320LF2407A fixed-point single chip DSP (Digital Signal Processor). This device offers low-power and high-performance processing capabilities, optimized for digital motor and motion control. The DSP consists of six major blocks of logic: (1) External program and data memory, (2) Analog Interface, (3) I/O Interface, (4) Expansion interface, (5) JTAG Interface, and (6) Parallel Port JTAG Controller Interface, while we currently use only three of these:
    - External program and data memory. The RAM module is used in debugging the software with the Parallel Port JTAG Controller Interface.
    - I/O Interface. The interface contains different kinds of pins:
        - Capture units: used for capturing the rising pulses generated by the motor encoders which can be used to measure speed and direction of the moving motor.
        - PWM outputs: these pins have a compare unit associated. A period value is established to determine the size of the PWM, and the compare value is used to change the duty cycle.
        - Standard I/O: used to read and write values for transceiver communication, motor, kicker and dribbler control.
3. MOTOR CONTROL. The motor encoders generate a number of quadrature pulses for each completed turn. Each pulse is captured using the DSP and the *feedback* speed is computed. *Feedback* speed along with *received* speed from the transceiver are used as inputs to the PID algorithm in calculating an adjusted PWM signal sent back to the motor. Figure 5 illustrate the procedure.

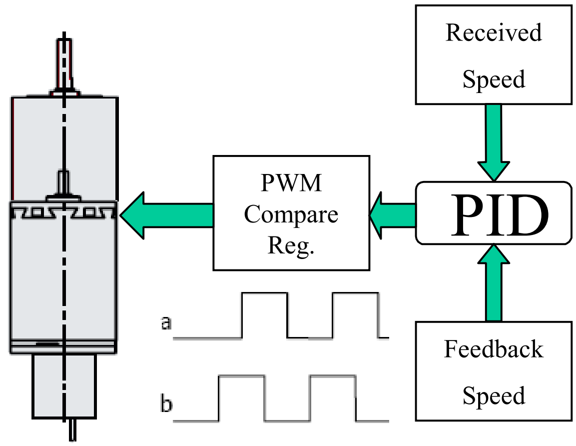

Fig. 5 Motor control

4. WIRELESS COMMUNICATION. Wireless communication is controlled by two Radiometrix RPC-914/869-64 transceivers with radio frequency at either 914MHz or 869MHz. The transceiver module is a self-contained plug-in radio incorporating a 64kbit/s packet controller with a parallel port interface.
   Data is transferred between the RPC and the host (either DSP or PC) four bits at a time using a fully asynchronous protocol. The nibbles are always sent in pairs to form a byte, having the Least Significant Nibble (bits 0 to 3) transferred first, followed by the Most Significant Nibble (bits 4 to 7). Two pairs of handshake lines REQUEST & ACCEPT, control the flow of data in each direction.
   Each data packet has 21 bytes, 1 preamble byte (sync byte and an error check sum) and 20 bytes for the 5 robots (4 bytes for each robot). The first byte (control) specify the status of kicker and dribbler (activated / deactivated), and the other three specify the desired speed for each of the three motors in the robot.
5. KICKER CONTROL SYSTEM. To win a Small Size soccer robot match, the team needs to score. Scoring requires a ball impulse system. In the Eagle Knights team scoring is possible with a push type solenoid that kicks the ball.

   Solenoid kicker system needs a high power supply. For size restrictions robots have only two 9V/1600mA batteries, so we got 28.8 Watts; with this amount of power we obtain less than the solenoid requires for a minimum performance.

The main idea in power elevation is to store energy, then discharge it when solenoid is activated. To solve this power problem we implemented a four layer system described below:

a) Oscillation generation.
   Voltage transformers needs an AC signal but robots batteries are DC so, in order to accomplish a voltage transformation, we need an oscillating voltage source.
   Oscillating voltage can be obtained with a Schmitt trigger RC circuit. We use a CMOS CD40106, a 1KΩ resistor and a 0.1mF capacitor. With this configuration we can generate 5V, 50% duty cycle signal @ 44KHz.
b) Voltage transformation.
   The oscillating voltage obtained from Schmitt trigger is a low power signal, to increase signal power we use an H transistor bridge: L298N.
   The two direction control bits in L298N are feed from the Schmitt trigger oscillating signal, one is the inverse of the other. Enable bit is always in high level (5 VDC) so L298N operates all time.
   The input power signal in L298N is obtained from the two serial connected 9V/1600mA DC batteries. It represents an input 18V signal, so the output in L298N is a 36V peak to peak, 50%duty cycle @ 44KHz signal.
   L298N output signal feeds a 12/120V voltage transformer. The voltage in the output transformer signal is a 360V peak to peak, 50% duty cycle @ 44KHz signal.
c) Charge accumulation.
   The charge in a capacitor is the number of electrons on the two plates. This involves the difference in the quantity of electrons and the unit of quantity is the coulomb.

$$Q = CE$$

   Where:
   Q = Coulombs.
   C = Capacitance in Farads
   E = Volts

   We use a 690mF capacitor, before charge it, the input signal needs to be rectified, we use a full wave rectifier diode bridge across a DB106.
   The full charge capacitor time is 5 seconds approximately. It means solenoid can be activated every 5 seconds.
d) Discharge and solenoid activation.
   If robot has a ball and it is in score position, then robot needs to kick the ball. An infrared sensor system in the bottom of the robot senses if the robot has the ball, an output sensor system bit indicates it.
   DSP sends an output bit in high level when the robot is in score position. To discharge capacitor into the solenoid we need to assure that the robot has the ball too. The Discharge layer uses a TTL AND gate between DSP kick bit and the infrared ball detector output bit.
   Because capacitor charge level is too high, we need to discharge it using an electronic relay. We selected a Sun Hold RAS-0610 relay.
   AND gate output bit determines relay activation. RAS-0610 needs a 130mA current activation but the AND gate does not bring it. So we required a darlington transistor to increase the AND output signal in order to activate relay, discharge capacitor into solenoid and kick the ball.
   The Darlington transistor is a semiconductor device which combines two bipolar transistors in tandem in a single device. This gives it high gain, but takes up less space than using two discrete transistors in the same configuration.

Finally the robots were manufactured using a CNC ABC plastic machine in ITAM´s facilities. The circuits were tested successfully and PCBs were ordered. Robot integration took only a week after all parts were ready.

V. CONCLUSIONS

We present a software and hardware overview of the SSL Eagle Knights team. The functional blocks of the software systems (Vision and AI) has been exposed in detail. We described the robot’s hardware with insight information of the kicker control system.
Our team has been the first latinamerican team in getting to the top 3 of an official RoboCup competition (USOpen). We had released to the public the Vision System and documentation of our electronics and DSP software to promote the participation of others teams in this initiative. More information can be found in http://robotica.itam.mx/

VI. REFERENCES

[1] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa. Robocup: The robot world cup initiative. In Proceedings of the IJCAI-95 Workshop on Entertainment and AI/ALife, 1995.

[2] Verret, Ball, Kiat Ng. Laws of the F180 League - Release 3.00a. http://www.itee.uq.edu.au/~wyeth/F180%20Rules/index.htm.

[3] Tsai, R.Y. A versatile camera calibration technique for high accuracy 3D machine vision using off-the-shell TV cameras and lenses. IEEE Journal of robotics and Automation, 1987

[4] Maxon Motor AG. Product Catalog http://www.maxonmotor.com