Final Report

EEL 5666 Intelligent Machines Design Laboratory

TAs: Mike Pridgen & Thomas Vermeer
Instructors: Dr. A. Antonio Arroyo & Dr. Eric M. Schwartz

Hao (Hardy) He

Dec 08th, 2009

Table of Contents

Abstract

This final report describes in detail the design of my robot named SoccerFan. SoccerFan is a wheel-based robot that can locate a colored ball and push it into a colored goal. System configuration, actuation, sensors, behaviors and experimental results are covered in line.

Executive Summary

SoccerFan uses an ATmega128A microcontroller to drive all peripherals including three IR proximity detectors, a CMU camera, two servos (pan/tilt for the camera), a dc motor driver with two motors, an LCD screen, two LED lights, a bumper switch, a CdS light sensor and a buzzer. The main function is to use the IRs and the CMU camera (which is color sensitive) to locate green balls and a red goal. Then SoccerFan will push the ball into the goal.

A salient feature of SoccerFan is that the CMU camera can be steered in many directions by use of the pan and tilt servos. This enables the robot to accurately pinpoint a small colored object, as well as to see far-away objects. This feature is crucial to the whole task since SoccerFan must be able to locate small balls and then to locate the goal without leaving the ball.

Another feature is the use of median filters in all sensor readings, including the IRs, the camera and the CdS. The median filter significantly reduces the impact of noise and system instability and thus improves the performance a lot.

The most involved part of SoccerFan is the CMU camera which communicates with the microcontroller by a serial port. The Baud rate, white balance, poll mode and many other parameters/settings have to be carefully adjusted to make the camera work smoothly.

Introduction

I like playing soccer, as well as soccer video games. This motivates me to build a robot that can play soccer. SoccerFan will be an autonomous robot that can perform basic soccer skills: locate a ball and hit/bring the ball into a goal. SoccerFan uses differential drive of two wheels to move, IRs to determine distances and a CMU camera to detect the ball and goal (painted with different colors). With the knowledge of objects' positions and distances, SoccerFan can carefully push the ball into the goal. The original plan of adapting SoccerFan to any ball/goal positions was not very successful, mainly due to the imprecision of motors and weak color-sensitiveness of

the camera. However, in relatively simple setups (such as the ball/goal in front of SoccerFan not too far away), the task can be successfully completed.

Integrated System

The system configuration is outlined in Figure 1.

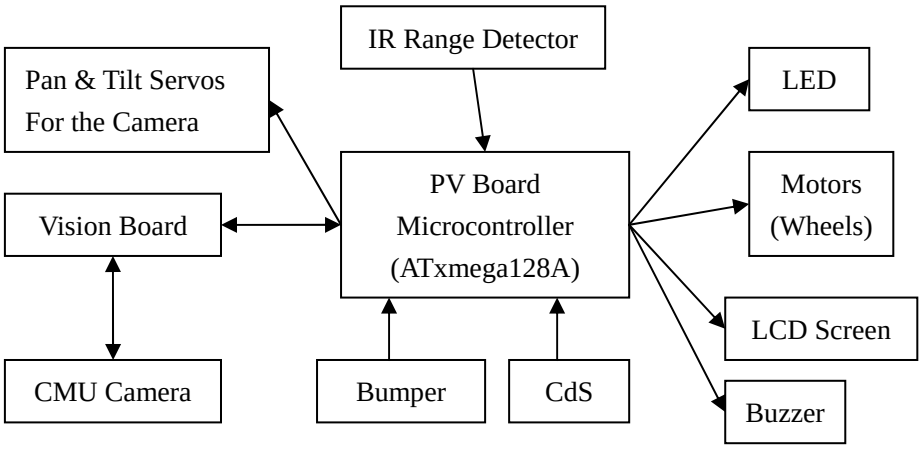

Figure 1

An ATmega128A microcontroller seated on the Pridgen-Vermeer board interfaces with all peripherals, including various sensors and servos. The CMU camera is used to locate the ball and the goal by color. IR range detectors assist the camera in locating the objects. Moreover, IR range detectors, which are effective for calculating short ranges, help the robot move close to the ball/goal with a proper distance. By analyzing the information from sensors, the microcontroller sends signals to servos which drive the motors to perform certain movements. An LCD screen shows the IR and camera readings, mostly for debug convenience. Two LED lights (one green and the other red) show the current status of the robot. A bumper switch and a CdS light sensor are used to receive task-switch commands from outside.

Mobile Platform

The platform of SoccerFan is cut out of a wood board using the T-tech machine. Figure 2 shows the final platform (a quarter coin is placed for the size comparison).

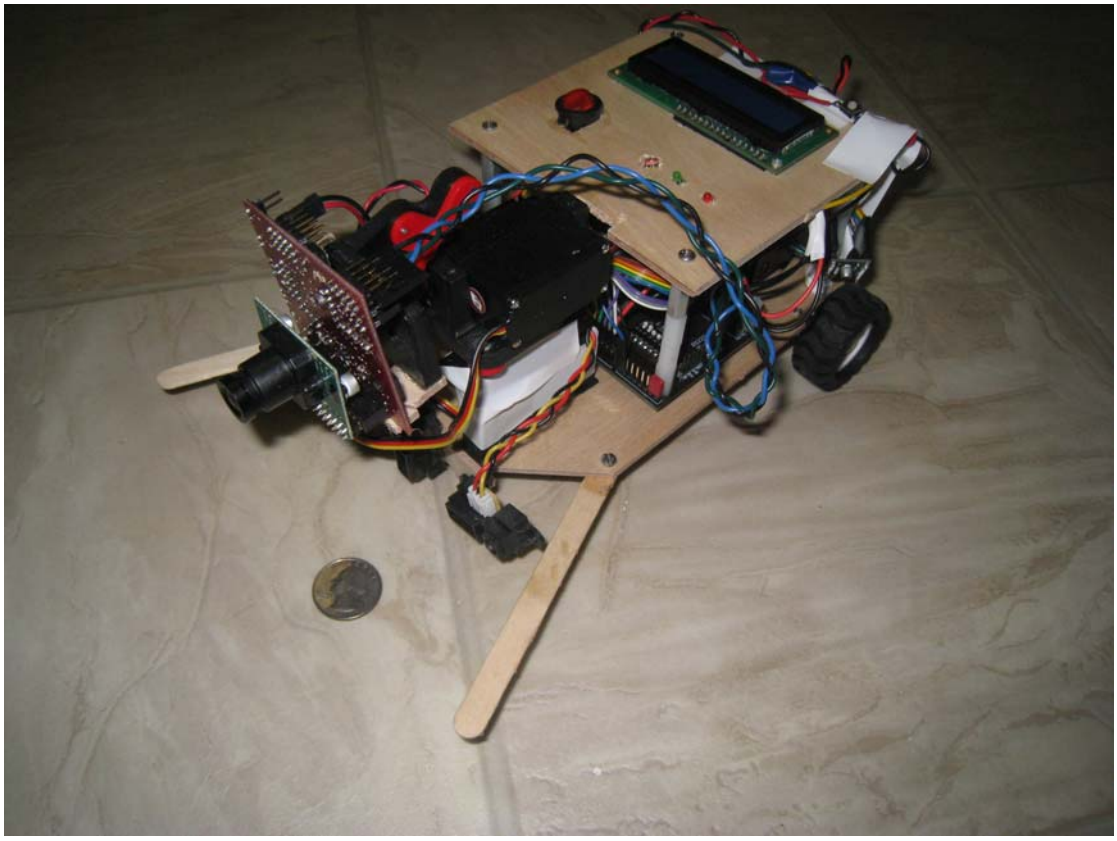

Figure 2

All components are compactly mounted on the two-layer wood platform. Screws and nuts are mainly used while white electrical tapes assist in fixing flimsy accessories. The main microcontroller board, motors/wheels, the CMU camera with the pan & tilt servos, three IR sensors are mounted on the bottom layer. The main power switch, an LCD screen, two LED lights, a CdS light sensor, a bumper switch, a buzzer (not easily seen) are mounted on the top layer. The two sticks stretched out in the front are used to constrain the movement of the ball when the ball is being pushed by SoccerFan.

Actuation

I use the Toshiba TB6612FNG DC motor driver with two 160rpm 100:1 gear-ratio motors (see Figure 3) bought from www.sparkfun.com.

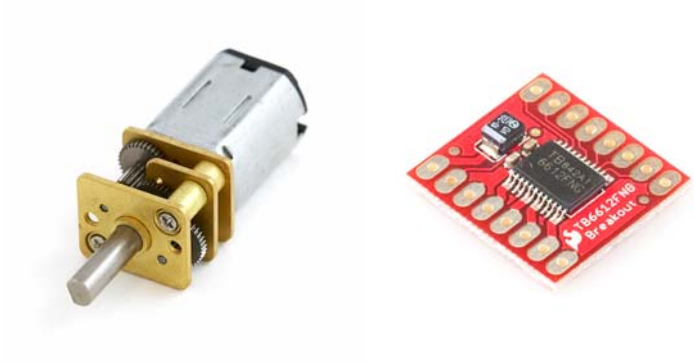

Figure 3

This motor driver can control two motors simultaneously. For each motor, there are three input signals: PWM (100 kHz width-modulated pulse), A1 and A2. The combination of (A1, A2) indicates the mode of the motor. For example, (A1, A2) = (1, 0) makes the motor spin clockwise while (A1, A2) = (1, 1) gives the motor a brake. PWM controls the speed of the motor. A full pulse leads to the full speed which is 160rpm.

Two HS-485HB servos from www.servocity.com (see Figure 4) are used as the pan and tilt servo to steer the CMU camera. The PWM signal period is 20ms. A 1.5ms pulse width corresponds to the neutral position while 0.6/2.4ms corresponds to the left/right 90 degree rotation.

Sensors

Three short-range SHARP GP2D120XJ00F IR proximity sensors are used to detect obstacles/objects. The input varies from 3.1V at 3cm to 0.3V at 40cm with a supply voltage between 4.5 and 5.5V. In my experiment, they work well with 3.3V supply.

Figure 4

The main sensor in SoccerFan is the CMUcam2 which consists of a SX52 microcontroller interfaced with an OV6620 Omnivision CMOS camera. The board can communicate via a RS-232 port with the computer or with another microcontroller, such as ATxmega128A. Figure 4 shows the CMUcam2 system along with two servos.

CMUcam2 is powered by 6 to 15 volts DC, so a normal battery pack (with 6 AA batteries) is sufficient. To test the camera, an easy way is to resort to a PC. By connecting CMUcam2's serial port to a USB port in a PC using a serial-to-USB adaptor, we can conveniently use a terminal (such as the HyperTerminal in Windows XP) to send and receive information from the camera. Figure 5 shows the received packets from the camera after sending the command "GM\r" from the terminal. "GM" stands for "Get Mean" and "\r" is the carriage return to end a command. In Figure 5, each line is a so-called S packet, where the six numbers after the letter S are the mean value of red, green, blue and the deviation of red, green, blue in the current frame, respectively. If we place a red object in front of the camera, the first number after S will increase significantly.

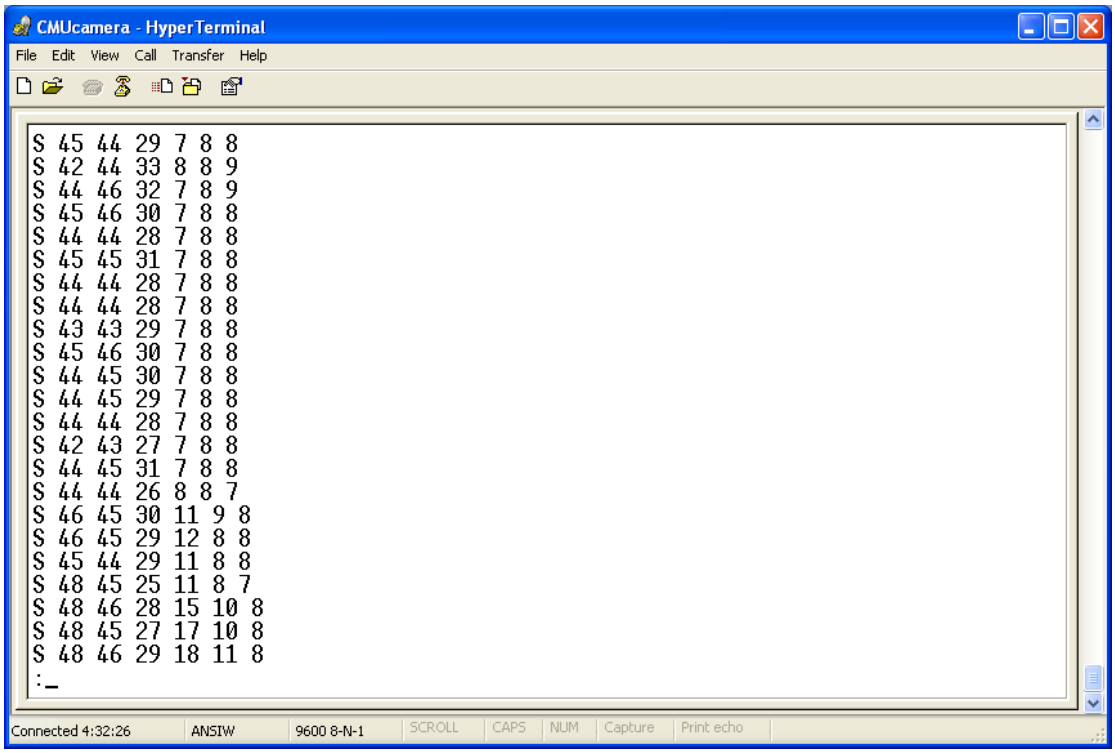

Figure 5

It is in the same principle if we connect the camera to the serial port on the PV board. ATxmega128A sends command to the camera and the received the corresponding packets from the camera. In my experiment, I set the camera in the "poll" mode, which means that only one packet is returned when a command is called. It facilitates synchronization between the central microcontroller and the camera.

Behaviors

The behaviors/algorithms of SoccerFan after the power is turned on are outlined below.

System initialization (microcontroller, servo, A/D, LCD and CMUcam2)

The LCD shows the IR and Camera readings until the bumper switch is pressed.

While (true)

1. Randomly moves/turns until it senses an obstacle in front

2. Steer the Camera in 15 directions (pan: -50, -25, 0, 25, 50 and tilt: 10, 25, 40 degrees) and examine the green-color component value in these directions

If the biggest value > threshold (background green value + 20)

The green LED lights up. Go to Step 3.

Otherwise

Go back to Step 1.

3. Steer the Camera in 7 directions (pan: -90, -60, -30, 0, 30, 60, 90 and tilt: 0 degree) and examine the red-color component value in these directions

If the biggest value > threshold (background red value + 30)

The red LED lights up. Go to Step 4

Otherwise

Circle clockwise 90 degrees around the ball and then go back to Step 3

4. Based on the rough location of the goal, move towards the goal (the ball being pushed during this movement), track the goal continuously to adjust the direction and finally stop when getting very close to the goal. Buzzer beeps.

5. Wait unitl the CdS light sensor reading is below 15 (i.e. CdS is covered by something, say, my hand)

END while

The color threshold parameters and pan/tilt servo rotating positions are carefully chosen after many trials so that the expected behaviors are relatively robust under different lighting conditions.

One problem is that when SoccerFan pushes the ball, the ball may roll out of the two antenna-sticks and become out of control. Another problem is the “circling clockwise 90 degrees around the ball” because of the motor performance variability. The same amount of motor running time does not always lead to the same distance.

Experimental layout and results

The experiment is carried out in an open field, e.g. 8 feet by 8 feet. There is a red goal and several balls (both green and red) on the field, as shown in Figure 6.

Figure 6 also illustrates a simple situation: the green ball and the red goal are in front

of SoccerFan. The task is performed as follows. SoccerFan starts moving forward and stops in front of an obstacle -- a green ball. Then the pan and tilt servos steer the camera to examine the color in all fifteen directions. Because the ball is green, in a certain direction the green-color component increases sharply and then SoccerFan knows that it has found a green ball. After this, the camera is tilted horizontally (to see far away objects) and steered in seven directions trying to locate a red goal. Successfully it finds a red object about 30 degrees on the right. According to this angle, it turns right properly so that it faces the goal approximately. Then it moves towards the goal, during which it micro-adjusts its moving direction so that it can hit head-in onto the goal. Because the ball was already in front of and very close to SoccerFan, when SoccerFan is moving to the goal, the ball is being pushed to the goal also, with the help of two antenna sticks.

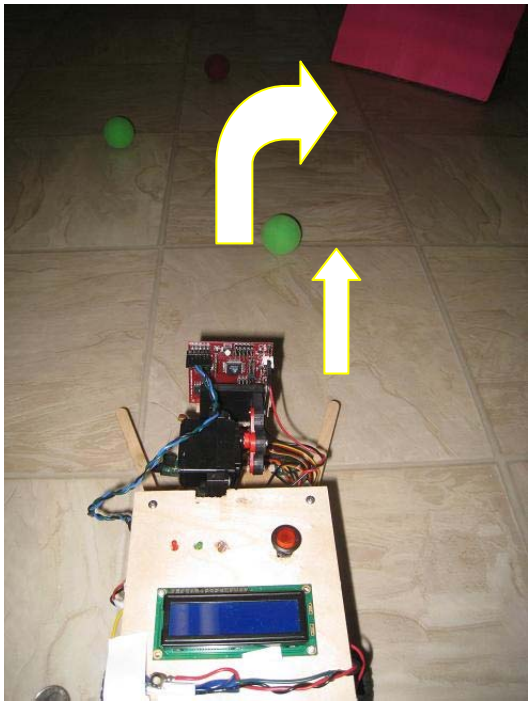

Figure 6

A little more involved situation is shown in Figure 7. SoccerFan moves to point A and finds a green ball. But now because the red goal is in the back and the pan servo can only steer the camera in plus/minus 90 degrees, SoccerFan will not be able to find the red goal. In this situation, SoccerFan will circle clockwise 90 degrees around the ball, i.e., it moves from point A to point B. In point B, SoccerFan can locate the red goal because the goal is now in its front. This task is not always successfully fulfilled because “circling clockwise 90 degrees around the ball” requires careful calibration of the motors. Sometimes after doing a “90 degree circle turn”, SoccerFan is no longer facing the ball and even using the camera to relocate the ball does not help much.

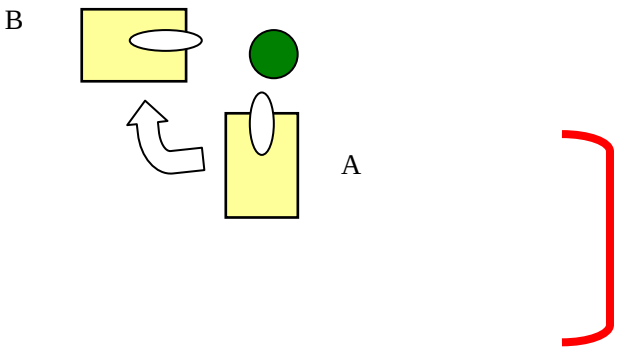

Figure 7

In the most general situations where balls are randomly placed, it takes quite some time for SoccerFan to find a green ball (because it random moves and turns and stops whenever it detects an obstacle). During this process, it may have already stricken balls away from the side and thus ruined the whole setup.

Conclusion

It has been a wonderful experience to build my robot SoccerFan. The function of “kicking a ball into a goal” is successful for certain positions of the ball/goal. It is far from fully adapting to any random ball/goal positions, but even the one success out of ten trials encourages me to perfect the robot and brings me so much joy. To make SoccerFan better, 360 degree camera view, motor calibration and how to bring the ball with the robot would be the next emphasis.