Second Formal Report: SoccerFan

EEL 5666 Intelligent Machines Design Laboratory

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

Hao (Hardy) He

Nov. 12, 2009

Table of Contents

Abstract

This report describes the current progress of my robot – SoccerFan. The focus is the configuration of the system and the use of the special sensor – CMU camera.

Introduction

As said in the proposal, the goal of SoccerFan is to automatically locate a colored ball and “kick” the ball into a colored goal. Besides the main controller (ATxmega128A) and the PV board, the peripherals include motors, servos, a camera, a LCD screen, IR range detectors, bumper switches and so on. The kernel component of SoccerFan is a color-sensitive digital camera which is required to be able to communicate with the microcontroller. In my implementation, I choose the CMUcam2 system. SoccerFan uses this camera system to locate colored objects, which is the crucial step of the whole task.

Integrated System

The system configuration is outlined in Figure 1.

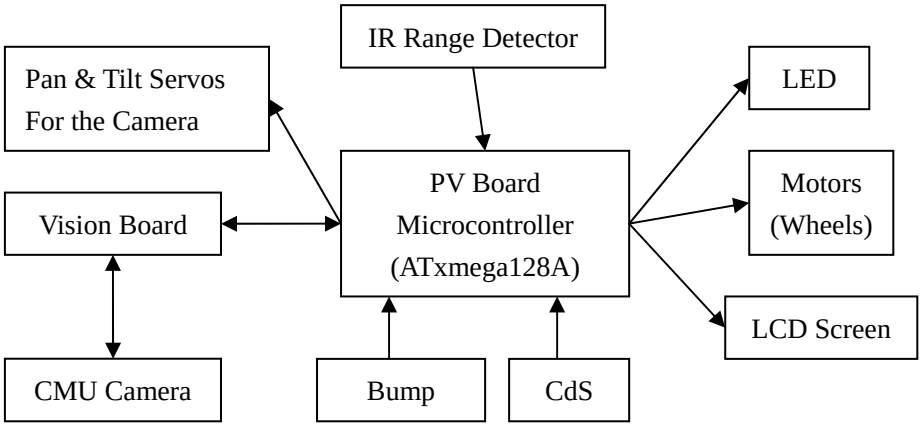

Figure 1

An ATxmega128A 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 with a proper distance. The bumper

switch and the CdS light sensor are used to signal SoccerFan to do certain tasks. For example, after powered on, SoccerFan will first calibrate the system (Camera and IR) to the surrounding environment and SoccerFan will not start searching for the ball until the bumper switched is pressed. An LCD screen shows the current status of the robot and the readings from the camera, mostly for debug convenience.

By analyzing the information from sensors, the microcontroller drives the pan/tilt servos to direct the camera looking for the colored ball or goal. The microcontroller also drives the motors so that the robot can move to the ball and “kick” it to the goal.

Mobile Platform

The platform of SoccerFan is cut out of a wood board using the T-tech machine. Figure 2 shows the platform where most parts have been mounted.

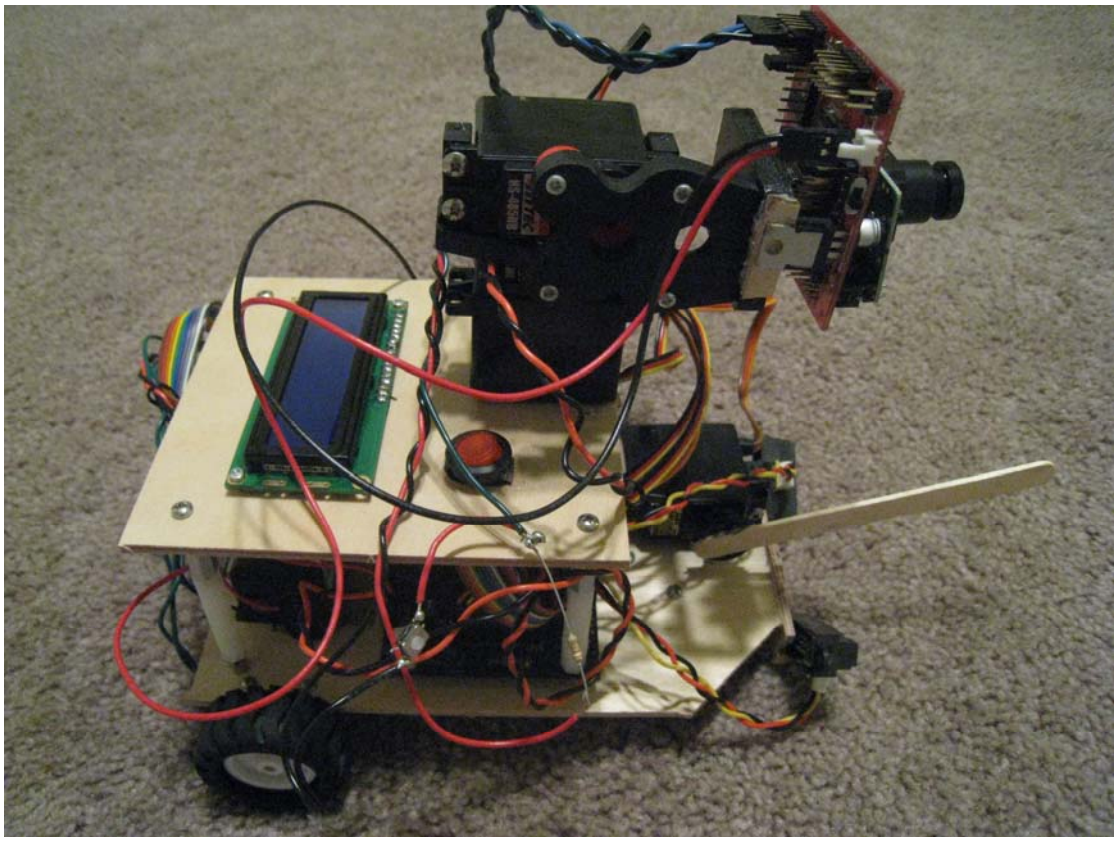

Figure 2

The platform has two layers. The main board, battery pack, two motors (with wheels), a servo (for the mechanical arm) and a caster wheel (for balancing) are mounted on the bottom layer. A switch, an LCD screen, pan/tilt servos and the camera are mounted on the top layer. A bumper switch is loosely connected. There will probably be a few more components added, such as the CdS light sensor, the buzzer and LED lights.

The CMU camera

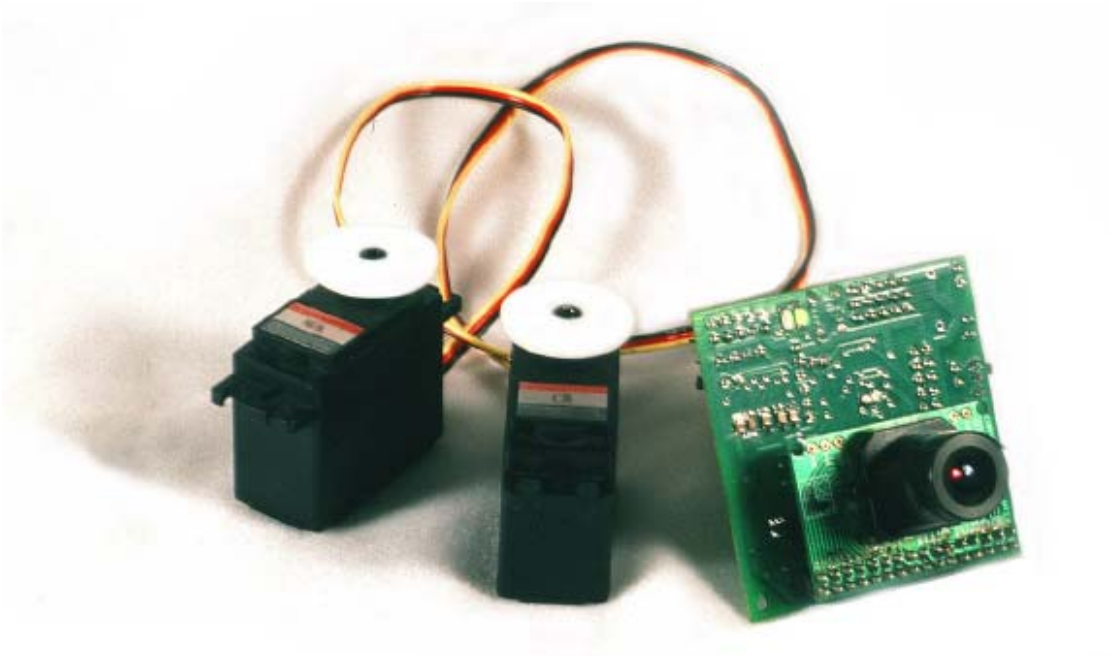

Figure 3

The CMUcam2 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 3 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. Without interfacing with any computer or other microcontrollers, CMUcam2 can track colored objects by directing pan/tilt servos automatically – called demo mode. This is achieved as follows. Connect power to the camera and connect two servos (one pan and the other tilt) to the servo port 0 and 1 respectively. Place a colored object in front of the camera. Hold the push button on CMUcam2 and turn on the power switch. Then immediately release the push button. After a few seconds (a status LED blinks and goes off) CMUcam2 will enter the demo mode. If we move the colored object slowly to the left/right/up/down, CMUcam2 will drive the pan and tilt servos to follow the object.

The demo mode is an easy way to test that the camera is working. To test all other functions and especially how to use various commands, it is better 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 4 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 4, 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.

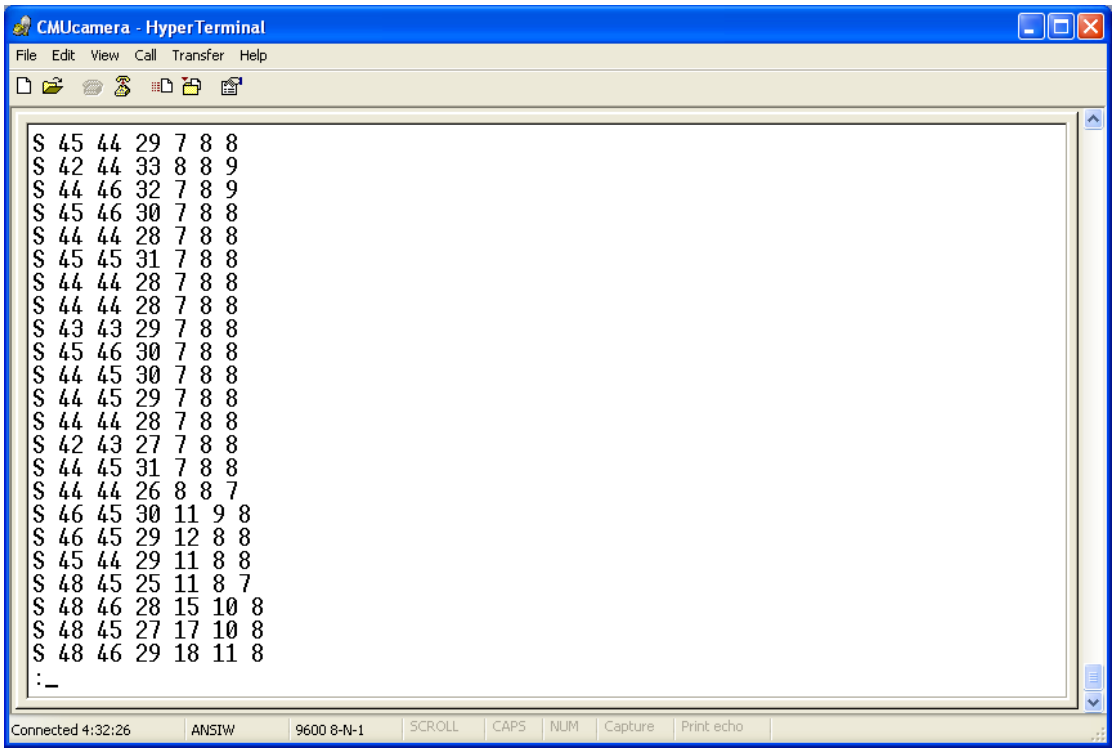

Figure 4

It is the same principle if we connect CMUcam2 to the serial port on the PV board. ATxmega128A sends command to the camera and the received the corresponding packets from the camera. In this case, it is better to 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

As shown in Figure 5, the expected behavior of SoccerFan is to hit the green balls into the red goal.

Since we put the balls randomly on the field, the first thing that SoccerFan needs to do is to locate a ball. Straightforwardly, SoccerFan roams on the field randomly until the IR sensors detect an obstacle, which must be a ball in our case (assume that the goal is relatively far away). Two IR sensors can give SoccerFan a general idea whether this ball is on the left or right. Then SoccerFan will use the camera to tell the color of this ball. If not green, SoccerFan will move away and continue searching; if green,

SoccerFan will combine the camera and IR inputs to move properly so that the ball is in front of itself. This function has been largely achieved.

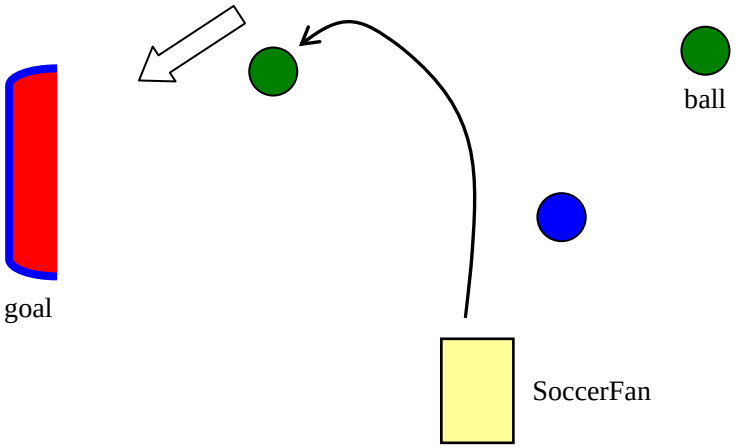

Figure 5

After locating a ball, SoccerFan should drive the camera in all possible directions to find the red goal. The goal should be large and in bright red so that the camera can find it even from a relatively long distance. With the knowledge of the goal's position, SoccerFan needs to maneuver to the back side of the ball with respect to the goal and hit the ball into the goal. This step has not been completed and is under heavy testing and trial.

Conclusion

Almost all components of SoccerFan have been mounted and tested. SoccerFan is already functional in terms of obstacle avoidance and locating colored objects. More careful testing and more delicate algorithms are under progress to achieve the final goal: hitting a ball into a goal.