A Low Cost Visual Fiducial Control Mechanism for Multiple Agents Working Towards the Same Goal Demonstrated through Robotic Soccer

Austin Oltmanns, Goutham Miryala, Haiming Lou

Abstract:

Many systems must be controlled remotely and an important part of control is localization. Being able to do this at a low cost would provide opportunities for a plethora of useful systems to be developed. This study focuses on an ultra-low cost agent consisting of a Bluetooth module interfaced with a drive system and an ApriTag [6] for identification. Applications such as warehouse organization, where a line-of-sight between system and agent is present, would be benefited.

Introduction

Robot soccer is an important research tool for developing new AI techniques and optimizing existing AI techniques. [3]

Several techniques have been used and explored in biomimetic robotics which can be applied to this problem [4]. Although these techniques have been applied to robot soccer, most of them treat each robot as a separate entity and require a localization process for each robot. To conceptualize the soccer robots as a single entity we need to treat an n-robot system as a cluster and set the desired motions as cluster attributes. This includes position, orientation and group geometry [5]. Such a framework can be implemented in different ways to express some of the most common multi-robot formation control methods from the cluster space perspective.

In this paper, a method for building soccer robots with minimal or no on-board sensors is presented. Further details about the implementation of the algorithm for object avoidance and path planning, recognizing and localizing robots with AprilTag, and soccer ball identification with OpenCV is presented in the following sections.

Background

Early low-cost visual fiducial systems such as ARToolkit and ARTag were not fast enough to be used in real-time control applications [6]. With the introduction of AprilTag2, this is now a possibility. Also, advances in real time path planning algorithms such as PSO [7] have opened new doors for real time control algorithms. Combining these methods has been explored some by H.Y.H et al. but the implementation was not delivered. Similarly, Pratomo, in “Position and Obstacle Avoidance Algorithm in Robot Soccer” explored these concepts. These previous ventures relied on costly equipment, and a large amount of testing.

Wang, L.et al [7] used PSO for obstacle avoidance path planning in soccer robot system. They found that PSO algorithm has several advantages compared to other methods. For example, it is easy to implement and can escape a local optima solution to achieve the global optimal solution; this is a common problem with other obstacle avoidance and optimization techniques.

Wireless technology has been stable for several years and is able to provide quick serial connections between many devices around the world. When close enough, these connections can facilitate real-time control systems. Bluetooth technology has reached a point of becoming very cheap; for this research, very low cost Bluetooth Serial Modules were used to control the system in real-time.

Finite state machines are a simple and effective method for implementing robot control systems and decisions trees when complicated behaviors are necessary. This system attempts to perform the complex behavior of playing soccer. The tactics and strategies outlined in this publication will be of use [9].

AprilTag2 is an “Efficient and robust fiducial detection” [6]system which is necessary for this application. Use of this system will enable the robust detection of the location of all robotic players.

System Overview

Figure 2 depicts the overview of the system. The system consists of five components. The largest of these components are the Analysis and Command Control unit. Each component is detailed below.

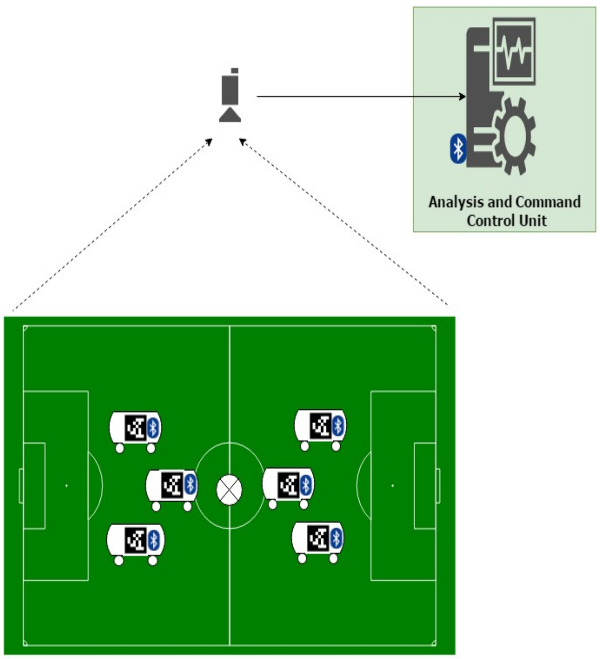

Figure 1: Soccer Robot System Overview

1. Soccer field

The soccer field dimensions are the same as those specified by the robocup regulations [1] shown in figure 2.

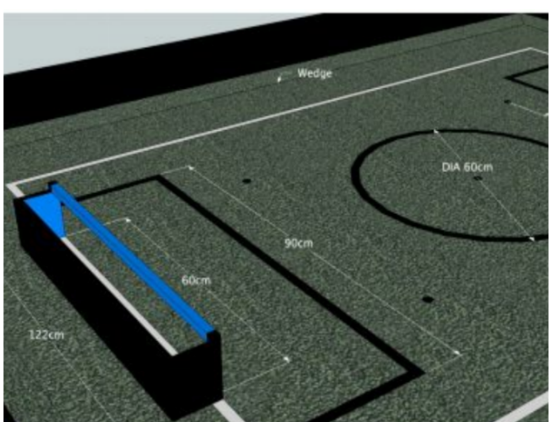

Figure 2: Soccer Robot Field

2. Soccer Robots

The soccer robots are relatively simple in design. The job of the robots is to follow the commands relayed from the command center. No localization or analysis is performed on the agents, but rather; information is processed by software running on the Analysis and Command Control Unit (ACCU) (the implementation of this software is detailed in section 5). The robot consists of a simple chassis with two wheels, an H-bridge, 2 DC motors, a bluetooth module with General Purpose Input and Output (GPIO) and an AprilTag.

To control the robots, commands are sent to the Bluetooth module (via bluetooth from the ACCU) to toggle the GPIO pins between high and low. These outputs feed directly to the H-bridge which controls how the motors spin and turn the wheels. This allows the robot to be controlled. Also, a unique AprilTag is attached to each of the agents to identify and track the agents in the soccer field from the ACCU.

3. Soccer Ball

The soccer ball is 2 cm in radius and orange in color. The color of the ball is chosen to facilitate easy detection by the ACCU. Figure 3 depicts the ball.

Figure 3: The Ball

4. Camera

A basic webcam is used to capture the stream of images processed by the ACCU. The webcam is placed directly above the field in order to capture the entire soccer field.

5. Analysis and Command Control Unit (ACCU)

The Analysis and Command Control Unit oversees the entire operation of the system, controlling the robots to achieve the goal of winning the game (the rules of which are not detailed in this document). This includes processing the image stream, analysis of positions of the robots and the ball, and relaying commands to the robots. The ACCU employs the following subroutines named in the flowchart in figure 4 to perform its task.

i. *Initialization* First, the system must initialize. This includes turning on the camera as well as connecting with the bluetooth modules.
ii. *Capture Image* An image is captured from the webcam via OpenCV for analysis.
iii. *Extract AprilTag Data* The AprilTag library is given the image and in return yields the positions of the robots relative to the camera.
iv. *Extract Ball Position* OpenCV is used to extract the position of the ball relative to the camera.
v. *Determine Velocity Vector* For each robot, agent software representing the robot is given the positions extracted from the image. These agents then coordinate to choose velocity vectors calculated to be optimal.
vi. *Convert Vector to Motor Commands* The velocity vectors chosen in the previous step are converted to motor commands, i.e. how much to spin the right and left wheels.
vii. *Send Commands* The commands are sent via bluetooth to the robots to be executed.
viii. The loop repeats with capturing another image.

Testing and Implementation

We have decided to test and implement the system in several phases: building the robots, simulating the system virtually and building the physical system.

a. ***Building the Robot:***
In this phase, the Robot is assembled as per the requirements outlined in the systems overview section. The Robot is made to move via instructions sent via the Bluetooth module as a basic test of functionality.

b. ***Simulate the Environment:***
This phase simulates the system to determine the best possible parameters for our algorithm to win the game. The simulation will be done using ROS Kinetic, Rviz and C++ code. ROS is used to allow the code to communicate with Rviz, which is used for visualization of the simulation. The idea is to use all the soccer robot system characteristics such as soccer field dimensions, number of soccer robots/agents (both teams), the physics of the system and soccer ball so that the algorithm can be exactly replicated to the actual soccer robot system.

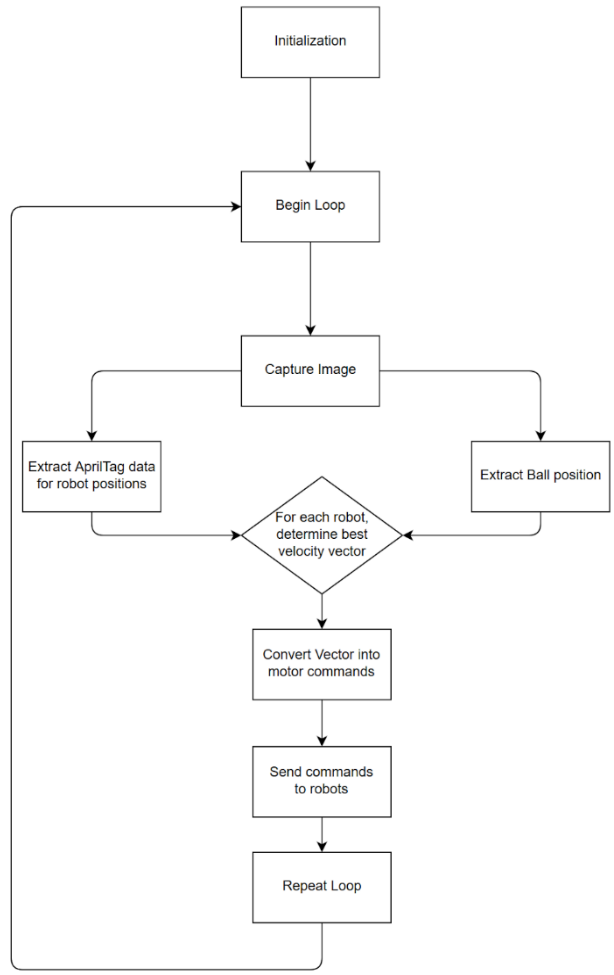

Figure 4 Analysis and Command Control Unit

The algorithm will initially, reset the positions of the soccer agents to their original formations as shown in figure 1. Once the game starts, the algorithm finds the soccer robot closest to the ball. Once the agent is identified, it moves towards the ball using the obstacle avoidance algorithm trail-blazed by Awang Hendrianto, et. Al [10]. A curved path to the goal is expected to be obtained to avoid obstacles and is updated continuously (at a frequency of 10 times per second) to accommodate parameters changing in real-time. Once the soccer agent successfully reaches the ball, the goal becomes the opposing goal post in the soccer field. Again, the same process is applied to find the best path to reach the goal post and to make a goal.

c. ***Build Physical System:***
The Physical system as mentioned in the systems overview is built. The server runs a GNU/Linux Operating System. The algorithm extracted from simulation is used here based on the positions of ball and soccer robots obtained from the OpenCV and AprilTag code.

Conclusion

This document and experiment is a work in progress and as such, a conclusion has not yet been made.

Future Work

See above.

References

[1] M. A. H. Kitano, "The RoboCup humanoid challenge as the millennium challenge for advanced robotics," in *Advanced Robotics*, 1998, p. vol. 13, nos. 8, 723–736.

[2] "A Brief History of RoboCup," *RoboCup Federation*, 1997. [Online]. Available: http://www.robocup.org/a_brief_history_of_robocup.

[3] M. Moradi, Morteza; Ardestani, Mahdi A.; Moradi, "Learning decision making for Soccer Robots: A crowdsourcing-based approach," in *Artificial Intelligence and Robotics (IRANOPEN)*, 2016.

[4] A. S. Kumar, V.; Leonard, N. E.; Morse, *Cooperative Control*. Block Island Workshop on Cooperative Control, 2003.

[5] C. Mas, Ignacio; Kitts, "Centralized and decentralized multi-robot control methods using the cluster space control framework," in *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2010.

[6] E. Wang, John; Olson, "AprilTag 2: Efficient and robust fiducial detection," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[7] H. D. and Y. X. Li Wang, Y. Liu, "Obstacle-avoidance Path Planning for Soccer Robots Using Particle Swarm Optimization," in *2006 IEEE International Conference on Robotics and Biomimetics*, 2006.

[8] H. Y. H. and W. M. F. A. Z. Mohamed, S. H. Lee, M. Aziz, "A proposal on development of intelligent PSO based path planning and image based obstacle avoidance for real multi agents robotics system application," in *2nd International Conference on Electronic Computer Technology*, 2010.

[9] C. S. Michalis Foukarakis, Asterios Leonidis, Margherita Antona, "Combining Finite State Machine and Decision-Making Tools for Adaptable Robot Behavior," in *8th International Conference, UAHCI 2014*, 2014.

[10] Pratomo, "Position and Obstacle Avoidance Algorithm in Robot Soccer," *J. Comput. Sci.*, vol. 6, no. 2, pp. 173–179, Feb. 2010.