SDP Group 1

Individual Report:
Milestone 1 1/2/12

3/9

03 FEB 2012

[redacted]

Due to circumstances outside of university I was absent for the first week of this project. This meant I was playing catch up initially as I missed several important meetings, and communications in emails.
Once I was back in Edinburgh, I spoke to the group and began work on strategy. I worked mainly on dissecting the previous years strategies.
The main goal in this was to research which of last years was not only successful overall in the tournament results wise, but also how accessible the code was for use within our own system. Looking at these codes and planning ahead for what strategies at both the overall level, as well as at the basic level of individual movements.

After our first meeting with our tutor, as a group we decided to firm up our roles in the project. At this point it became quite clear that there were to many people, more then half, that are working on strategy/simulation. I offered to get involved with vision and have been working on this since. However due to not being initially involved with this I have been playing catch up.

I began by again breaking down past systems and seeing the methods they had used. I then began trying to run previous years code. This was fairly difficult as I was working from home at first trying to use sample images of the pitch. I tested a handful of systems in both Java and python. These systems utilised video capture set-ups v4l4j in Java or opencv and simplecv with python. After playing around with both of these I spoke to Alex who informed me both himself and Altenbeck had been focusing on group sdp10's system from last year. Their vision system though not incredibly accurate was fairly accessible to understand and run.
From here I started trying to work with sdp10's code and get some basic knowledge of how opencv was used.
I did research into the preprocessing of the image feed and normalisations involved as well as edge detection. We used some of these in previous subjects like IVR and AILP. I looked at the methods used in the previous years

Summary:
This milestone:
For milestone 1 I feel I didn't contribute a great deal of work because
All the goals of the milestones did not require any of the areas I was involved in.
As a group we have a working vision system. I researched methods on normalisations and edge detection.

For next milestone:
I Intend to work on the normalisations used in the vision system and the threshold settings and image processing files in the vision system.
Improve accuracy and speed of the vision system.

9/9

SYSTEM DESIGN PROJECT 2012, GROUP 1

Milestone 1 Individual Report

1. Introduction

This report describes my contributions for the Milestone 1 of SDP 2012, Group 1.

2. Team and Organization

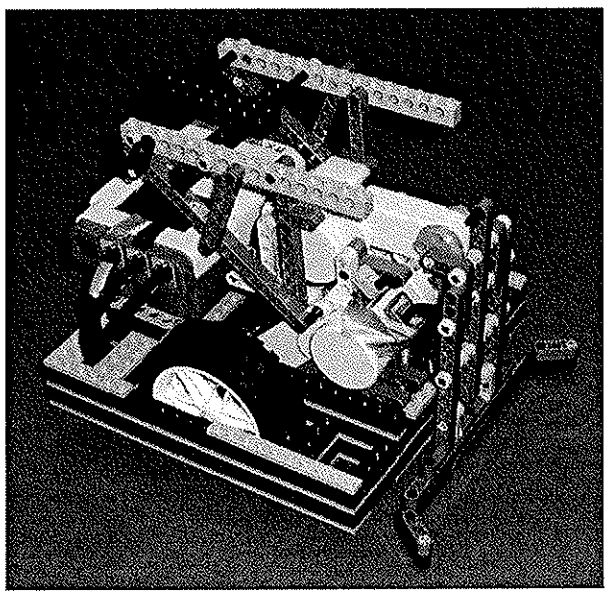
After organizing the first meeting of our team, I was delegated of being their leader. Until now, this role involved:

- Organizing meetings
- Setting up a Facebook group for communication.
- Setting up a BaseCamp project to keep track of the things we need to do.
- Make sure each member has something to do and make sure everyone knows what we aim to achieve.
- Make sure we get what we need: more batteries, Lego parts for the robot, keys for our locker, etc.

3. Designing and Building the Robot

After studying successful robots from the previous years I decided a robot with 2 powered and 2 free caster ball wheels would be suitable for our team. I proceeded in building such a robot. Together with another team member I built a suitable squared shaped chassis that allows the robot to spin in place. The powered wheels are placed in the center of two opposite sides and each of them is connected to a NXT motor without any additional gearing. This seems to perform well in practice.

We also equipped our robot with a kicker, powered by one NXT motor placed on top of the ones that are powering the wheels. This allowed us to put the NXT Brick on the other side of the chassis and balance the weight of the robot.



4. Vision

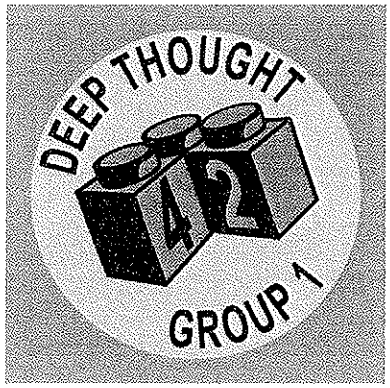
After trying to get more vision systems from the previous year to run we decided that we will write our own system from scratch. I started working on this and managed to write a fairly basic vision system that can detect the position of the robots, their orientation and the position of the ball at a rate of 30 frames per second. This system is still quite sensitive to changes of light but it gives us a very good starting point as I also integrated it with the rest of our code.

I used the SimpleCV python library (a wrapper on top of PyGame and OpenCV) to get the camera feed and process the frames. The communication between the Strategy code and Vision is done via a TCP socket.

SDP

02 FEB 2012

Group 1: Deep Thought
Performance Review Summary: Milestone 1



[redacted] **3/9 points**
Researched strategies from previous year's code and identified key algorithms which successful groups used (e.g A* search)

[redacted] **6/9 points**
Designed the team logo, gave some advice on strategy based on reading past year's code and performance videos.

[redacted] **- 6/9 points**
Installed openCV/simpleCV. Got basic camera feed and fixed issue with feed not displaying in certain situations. Fixed "barrel effect" caused by camera lens distortion by writing calibration code.

2

[redacted] **(Team leader) - 9/9 points**
Arranged meetings, project tracker etc. Built majority of the physical robot. Helped to get last year's code running. Wrote large parts of vision system and a client socket to send information to strategy

[redacted] **- 9/9 points**
Involved in the physical construction of robot. Made significant improvements to vision system such as implementing a background removal algorithm as well as better thresholding.

1

[redacted] **3/9 points**
Attempted to get a few previous year's code base to run, read some of their code and identified the groups who had the easiest to understand code.

[redacted] **- 6/9**
Created parts of simulation environment and wrote some low level commands for simulated robot. Researched possible architectures for strategy such as the "subsumption architecture".

[redacted] **6/9**
Got correct measurements for simulation walls, pitch, ball etc. Also calibrated simulation robot's wheels and added additional functionality such as a GPS, compass and receiver.

[redacted] **9/9**
Extensively refactored base code. Created a GUI commander, simplifying the running of basic commands and scripts. Rewrote server files to allow communication with vision system. Created robot and ball classes to store information about positions, orientations etc.

[redacted] **– 9/9**
Set up code repository. Got several past year's code bases to work on DICE. Figured out how to send files/commands to robot brick. Rewrote robot controller to accept abstract commands such as kick and move rather than numbers. Built program to perform robot operations from the command line.

[redacted] **6/9**
Created part of the simulation environment and robot. Wrote a large amount of the control code for simulation including basic commands, accessor methods and top level strategy.