

SDP Individual Report

Mark Nemec - s1140740

1 Introduction

The purpose of this report is to describe my contribution to the SDP project for team 2, The Underdogs. As the team leader and most active contributor [1] to the project, I believe that my contribution to the project was substantial. It ranged from managing the team to working on all parts of the code base and robot design.

2 Team Responsibilities

When it comes to team responsibilities, I was active from the start. I set up version control on GitHub [2], task-management on Trello and a discussion group on Facebook for the team. Moreover, I wrote a guide for setting up the project on DiCE [3].

I attended vast majority of the team meetings. Throughout the semester, I tried organising meetings regularly even though the turnout was usually not great. I created tasks on Trello for team members so that they would know what parts of the project to work on.

Even though our team had a conflict at the last performance review, I tried to include everyone - posting to our Facebook group when I was going to work on the project - and was always willing to help. For example, our mentor contacted me after one of our members stopped showing up to talk it through with him.

3 Vision

I started working on the vision system in week 2. I was suggesting to use a vision library because our implementation at the time was needlessly complicated and slow - running at about 10 frames per second (fps). I also implemented retrieving the orientation of the robots from the vision system before milestone 3.

After our failure to field well-performing robots for milestone 3 and the assessed friendlies the team decided to change the vision system to use OpenCV library [4]. The core of the new system was implemented by me with assistance from Julien in a span of a week. The system was based on the vision system of group 9 from 2012 [5] who also used OpenCV.

The new vision system correctly detected the position of the ball, and the position and orientation of robots. It ran at optimal 25 fps and was much simpler to parametrise. Moreover, I worked on including camera lens distortion correction and perspective correction. These improvements allowed for a more precise representation of objects on the pitch. For example after the perspective correction, the x-coordinate of the ball was within 1 cm from the x-coordinate of the robot.

Additionally, I worked on the main GUI of our program. For example, I re-wrote the thresholding tools so we would get immediate feedback after we change thresholds and helped Gordon implement saving thresholds for each pc in a separate JSON file. This is something we should have done earlier because it greatly reduced the time required for thresholding.

4 Object Representation and Geometry

I created classes for object representation and geometry in week 3. These were incrementally updated, as needed. After the friendlies, I had added unit tests for object representation and geometry so we could be sure that our code is not broken at a fundamental level when we started working on strategy.

I also kept on adding library-like methods to the `MovableObject` and `Robot` classes which were fundamental to our strategy system.

5 Strategy

I started working on strategy only after the assessed friendlies as I was focusing on other parts of the project. At that point we only had a working defender shot-blocking strategy. Thus, I added strategy for passing the ball up to the attacker and trying to score with the attacker.

I also refactored the defender strategy to use the previously mentioned library-like methods from object representation. This provided better transparency of the strategy code.

6 Robot Design

After multiple unsuccessful designs the team has decided that we would try to ‘borrow’ robot design ideas from other more successful teams. Julien and I therefore created a new chassis and grabber/kicker for the robots. This design was later expanded upon by other members of the team into what ended up being our final robot design.

With the new grabber design the robot was able to reliably grab a ball from a distance of 13 cm from the centre of the robot. Moreover, the kicker was able to kick the ball with a very good speed - being easily able to score across the whole pitch.

7 Appendix

- [1] Project Contributions: <https://github.com/sdpgroup2/sdp/graphs/contributors>
- [2] Group 2 SDP GitHub Repository: <http://github.com/sdpgroup2/sdp/>
- [3] DiCE Set-up README: https://github.com/sdpgroup2/sdp/blob/master/documentation/howto/README_DICE_SETUP.md
- [4] OpenCV: <http://opencv.org>
- [5] Group 9 (2012) Google Code Repository: <https://code.google.com/p/group-9-sdp2012/>