System Design Project, Group 11, Report 4

The following report describes our progress towards creating a Lego robot that can play robocup football. It also summarises our attainment of Milestone 4, which required our current robot to be able to intercept a moving ball. Reaching this milestone involved developing several aspects of our program, while making further improvements to our robot's design.

Progress Summary

Area	Target Aim for Milestone 4	Status at Milestone 4	Assessment	For Further Development
Vision	Convert first pitch thresholds to HSV colour space.	First pitch thresholds in HSV colour space. Can calculate robot direction without black spot.	On target	Improve performance from present ~15 frames per second.
Planning/ Strategy	Calculate plans to intercept a moving ball.	Planner can intercept a moving ball and can dribble around an opponent. Goalkeeper plan implemented for defending penalties.	On target	Optimise paths further. Alter planner to include backwards routes. Implement kicking to improve situation, not just to score.
Movement	No changes required for this milestone.	Internal position improved with vision information.	On target	Improve speed while maintaining control. Add backward movement.
Robot Design	No changes required for this milestone.	Kicker now powered by two motors. Strengthened design. More central centre of gravity. Two rear touch sensors.	Development complete	No further significant changes planned.
Integration	No changes required for this milestone.	Vision GUI implemented. Thresholds stored in separate file.	Ahead of target	Develop more unit tests. Investigate measures to improve latency/ping time.

Programming

A considerable amount of work has gone into improving the reliability of our vision system, after it suddenly failed to locate our robot during our last demonstration. The robot's direction can now be calculated without using the black spot, which is particularly useful in situations where the spot is within dark areas of the pitch. The algorithm used to achieve this is:

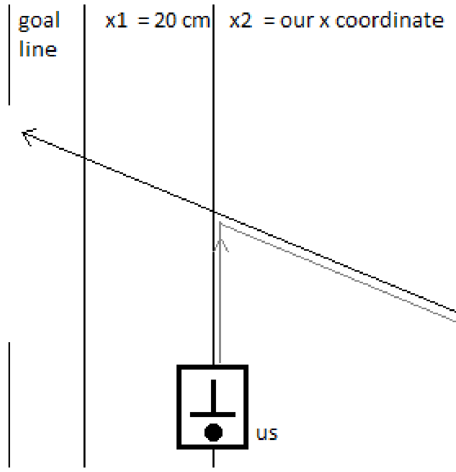
1. Threshold the plate using a range of green HSV values.
2. *Find the minimum enclosing rectangle.*
3. *Compare the euclidean distance from the centre of the 'T' to each of the plate's corners.*
4. *Choose the closest two corners to be the corners at the back of the plate and the others to be the corners at the front of the plate.*
5. *Look for the largest distance between a corner at the back to the two corners at the front of the plate. The furthest corner at the front can be paired with the other corner at the back and vice versa.*
6. *Finally, find the direction the robot is facing using a pair of back and front corners.*

As explained in the previous report, using the HSV colour space has the advantage of storing hue, saturation, and value (for brightness) in a single range of threshold values, whereas using RGB requires multiple sets to achieve the same amount of detail. Using HSV counters the effects of changing light at acceptable computational cost. The conversion of threshold values to HSV has now been completed.

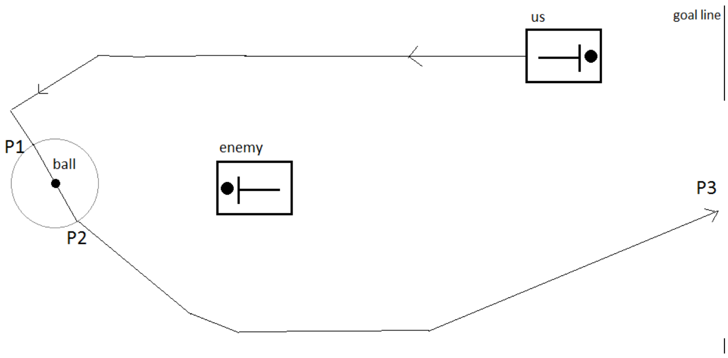
In addition to these changes, a Graphical User Interface has now been developed for the vision system. We found editing text files to be a clunky method for making minor changes, and our new GUI allows us to, for example, change threshold values and see the effect on performance in real-time. The GUI also facilitates the processes of changing sides, defending penalties, and troubleshooting problems.

The thresholds have been moved to a separate configuration file. The GUI can load and save these, which facilitates simple switching between different configurations.

The planner calculates a vertical path to an interception point on the path of the ball. The plan then follows the path of the ball to achieve interception.



We have developed the ability to dribble around an opponent. To accomplish this, the planner calculates the point P3 as close to the centre of the goal as possible and not blocked by the opponent. Then it calculates the point P2 that is as close to P3 as possible and not blocked by the opponent or the wall. Then the respective point P1 on the other side of the ball is calculated and also adjusted to not be blocked.



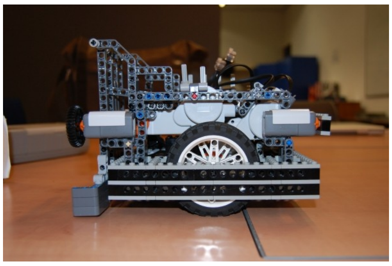
Our movement module, running on the NXT brick using Lejos, now improves its internally-calculated position and orientation using vision data with timestamps. This allows our robot to navigate quickly between waypoints without having to wait for vision to catch up and without losing track of its pose.

It is important to have the ability to defend penalties, as conceding a penalty is an easy way to concede a goal. We have implemented a strategy that quickly responds to the taking of a penalty and attempts to intercept the ball, which could make an important difference in match situations.

Robot Design

As we may move backwards, we are required to be able to detect collisions while moving in this direction. For this reason, we have installed two rear-facing touch sensors, complementing the two forward-facing touch sensors we already had.

While redesigning the robot we encountered problems with space. To solve this, we incorporated triangular shapes wherever possible. Lego parts are quite limited in how they can be combined to form triangles. However, by using Pythagorean triples, triangles can be made to fit whole Lego units. This also improved robustness



Group Organisation

Overall, the group continues to work efficiently and productively. There have been no specific challenges in this area since the last milestone. The group co-ordinates through email and our Trac ticketing system. We continue to program to no particular paradigms, not having felt the need to adopt one. However, we have adopted a general process of building the simplest solution, then refining and adding complexity. This ensures we achieve the milestone before trying to get ahead in certain areas. We also investigate libraries that are available instead of 're-inventing the wheel'. All group members choose their tasks and work in areas they feel comfortable with, which currently works well, owing to our wide range of talents. We remain aware that working this way may not continue to be appropriate.

First Friendlies

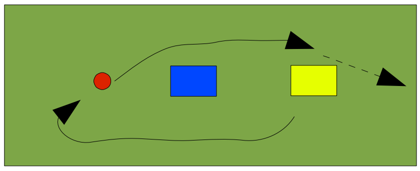
In the first friendlies, we won one match before losing the next. We experienced two problems relating to the kicker. The first was that our module which coordinates communication between the other modules did not initiate subprocesses correctly under Linux, and as a result the kicker was non-functional. We were not aware of this problem because we had mainly run our system on Windows up to this point, but matches require the system to operate on DICE machines. This problem was fixed in time for the second match, but then we found the planner to be too strict in deciding when to kick. We have since relaxed the tolerance.

Milestone 4 Demonstration

During the Milestone 3 demonstration, there was an unexpected and sudden failure of the vision system. Reliability has fortunately improved greatly since then; this time the demonstration proceeded as planned.

The demonstration began with our robot intercepting the ball, propelled at varying speeds. The robot was able to intercept the ball properly at the fastest speed, though it was generally unable to score a goal resulting from the interceptions. The ball was propelled towards a corner of the pitch, and once we had intercepted it the ball often went into the side of the pitch. Currently, our planner cannot devise a path for our robot to retrieve the ball from the side of the pitch. However, from watching other demonstrations, our robot's performance was clearly among the best.

We then demonstrated our robot navigating around a static opponent to retrieve the ball, then subsequently taking the ball back around this opponent and shooting this ball into the goal. The robot performed well in this task, apart from one incident when it nudged the ball into the side of the pitch. The ball was moved two inches back into the pitch and the robot continued without further problems. We concluded with a presentation of our vision GUI and the changes made to our robot's design.



Overall, we were pleased with our performance during the demonstration. There was very little dead time and our work was clearly exhibited. The reaction from the markers was positive throughout. The group agreed things had gone well; they felt the demonstration had proceeded smoothly and had given a comprehensive account of our system.

Conclusion

Our project has met Milestone 4. We have developed clear, achievable goals for the next two weeks. The group has bounced back from the significant failure of the Milestone 3 demonstration to build on the work we had done and ensure the success of our demonstration for Milestone 4. We have maintained our standing among the most competitive of the groups, therefore we believe we will perform well at the friendlies and final tournament.