Bi-Weekly Report

13 March 2015

Team Information

This document contains a summary of all that has been achieved over the last two weeks by team 19 on the CTSNet Robotics project as a part of the module Systems Engineering II of the University College London Computer Science course. Team 19 is formed of three students, Edward Collins (team leader), Kirthi Muralikrishnan (website and documentation lead) and Tom Page (research and programming lead).

Client Information

Team 19 has three clients:

~Dr Shabnam Parkar, paediatric surgeon at Great Ormond Street Hospital.
~Dr Joel Dunning, cardiothoracic surgeon at James Cook University Hospital.
~Dr Lourdes Agapito, computer graphics and vision expert at UCL and primary supervisor.

Summary of Progress

Over the last two weeks the team has made good progress on the actual construction of the system. Our aims for these two weeks included deciding what aspect of the system we would focus on implementing, and then use the Kinect to extract depth values of green pixels. Far more progress than just this was made.

First off a phone call was held with our client Dr Joel Dunning, to determine exactly what he would like to see demonstrated in April and to generate ideas for test scenarios. Dr Dunning approved of our designs for presenting depth feedback by colouring the tool in the camera feed according to depth information, so this is the primary goal for implementation. To test the system, we will need to design and build several scenarios based on the ideas that Dr Dunning has given us to make sure that the tests are realistic, albeit scaled up from the scale of actual surgery.

The coordinate mapping function has now been fully implemented, yielding a colour image where each pixel also has a depth value. This means that our props can be identified by their green colour, and then their depth values can be read. To test whether this was working, a small test scene was constructed. The Kinect was placed in a marked position, and then markers were placed at 30cm, 60cm and 90cm away. A green prop was then placed on these markers, and the program outputted the depth values of green pixels. The experiment yielded the same depth as was physically measured, which assured the team that the coordinate mapping function was working properly.

Coordinate mapping was then expanded on - so that the system will now present an augmented reality image - where the green tool is coloured when displayed on the computer screen according to how far away it is from the camera, turning red when close to the camera, then moving through a spectrum of orange, yellow and green as it moves further away from the camera. Furthermore the “General Depth” augmentation has now been created - that colours every pixel of the image based on how far away it is from the camera - again red for close and green for further away. Screenshots of this are shown below.

Perhaps the greatest achievement of the past two weeks however is the realisation of the first iteration of the “Tool Proximity” algorithm, which colours the tool according to how far away it is from the background, which would be the body in surgery. This augmentation is so important because it would inform the surgeon when tools are getting close to the body, so they should move more slowly. The algorithm the team has so far developed identifies the edges of the tool. and then measures the depth of pixels surrounding the edges of the tool, which would correspond to body tissue and which yields the distance between the tool and the body. Currently, the edges of the tool are then coloured according to this depth information, but this will be extended so that the entire tool is coloured according to how close it is to the body tissue.

Furthermore, the team has begun to produce the final documentation of the project that details the whole design and implementation process, has made sure our GitHub repository is fully up-to-date. Finally we have researched into how to make our software run faster; as reliable, quick image processing is a must-have requirement for surgery. Some of the methods we researched include using a Fast Fourier Transformation and using the graphics processing unit (GPU) as well as the CPU to perform some of the image analysis with a software called OpenMP. The team may not be able to implement this in this project due to lack of time, but will be able to suggest it to future developers as a way to improve the speed of the system.

Augmentation Based on Depth

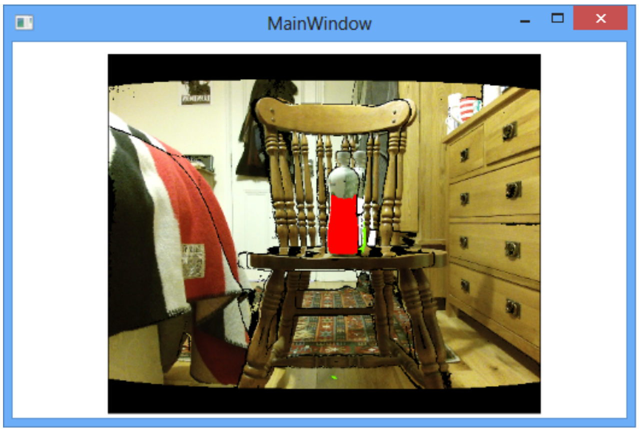

Figure 1 - Bottle coloured red based on depth.

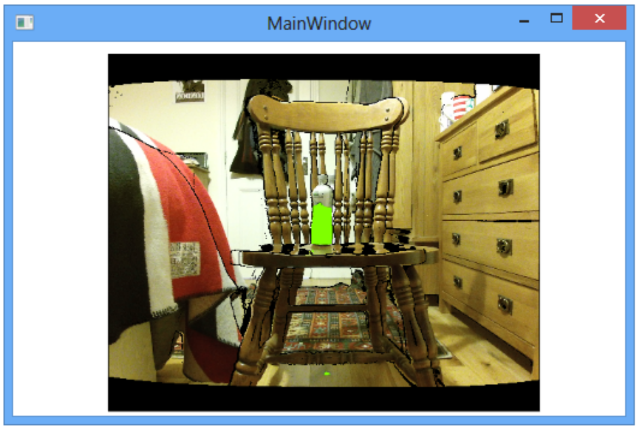

Figure 2 - Bottle coloured green based on depth.

Summary of Team Meetings

Date	Topics Discussed	Time
4 March 2015	Phone call with Dr Dunning. Discussed state of project, final system design and test scenarios.	20 minutes
6 March 2015	Discussion with PhD student Aron Monszpart about technical aspects of the project and work assignment.	1 hour
10 March 2015	Meeting with Dr Agapito. Demonstrated work with Kinect and discussed test scenarios.	45 minutes
13 March 2015	Discussion of progress, next work targets and technical discussion with Aron Monszpart.	1 hour 30 minutes

Difficulties Encountered

The primary difficulty encountered was, once again, with the coordinate mapping. It seems that the mapping is not completely accurate, and the image produced when the depth is mapped to the colour feed occasionally shows doubles of object overlaid on one another but slightly offset, as shown in Figures one and two above. This could present an issue for the Tool Proximity augmentation - as when colouring the tools it is essential that the right pixels are coloured. There was also an issue with correctly identifying the edge of a tool, as this was not always accurate. That issue was solved by taking a wider sample of pixels to improve accuracy of the edge detection.

Progress Target

Over the next two weeks, the team hopes to:

- Put all algorithms so far developed such as the Tool Proximity and General Depth into a single piece of software, that will provide several different views and have controls to switch between augmentations.
- Optimise and perfect the algorithms for the Tool Proximity augmentation and for identifying a green tool.
- Design test scenarios based on Dr Dunning's recommendations.
- Begin research into voice control.

This work will take us to the end of term, by which time we aim to have a first iteration of the demonstrable system, which will be further developed over the Easter break for presentation in April.

Individual Description of Tasks

Ed Collins

The last two weeks have been productive. The first thing I worked on was redesigning the team's work assignment, setting more realistic targets and timelines and determined which members of the team should do which tasks. I then created a substantial piece of documentation which details how the green tool identification idea was generated and ensured that all of our code was synchronised on GitHub. I also created a detailed UML diagram to add to our design section, that shows the software architecture of our final system, making use of the "Strategy" design pattern.

I then worked on the creation of the General Depth augmentation in conjunction with Kirthi, which colours the image displayed on the screen based on depth information. Kirthi completed the first iteration of this algorithm while working on identifying tools by depth, which I then expanded on. Doing this involved using the coordinate mapper function, and then using the mapped image to create two different augmentations: one that colours the whole image based on the depth values of each pixel, and one that colours only the green pixels belonging to the tool. Screenshots are above.

By Friday 27 March, we hope to have perfected the "Tool Proximity" algorithm and implemented the system according to the architecture that we have designed.

Tom Page

Over the last two weeks as a group I believe we have made fantastic progress. We have met our goals for completion and furthered them. I personally have created an edge finding program using the colour feed however this met issues with mapping. Therefore we decided to do edge detection from the depth feed. At this point I furthered the algorithm Kirthi made, meaning the edges were colour coded depending on distance to background.

Our goals for the next few weeks are to fully augment this feed, meaning the tool will be coloured fully depending on distance to background. Following this we can implement a framework for the entire system and at that point we should be able to present our solution. We also intend to optimise the algorithms in order to improve accuracy as well as speed up the process since currently the feed runs at approximately 5-10 frames per second which is simply too slow.

Kirthi Muralikrishnan

The last two weeks we have been programming with the Kinect and we have achieved a decent amount of progress. We have also split up the main programming tasks, so that we could all contribute and help achieve the goals faster. This week I worked on understanding the depth based segmentation of the tool. I devised two main algorithms. The first one is to differentiate the different segments of the feed based on the distance from the Kinect, using the grey scale. This meant that we colour coded the feed based on the depth data received from the Kinect. The next algorithm I developed was to find edges based on the depth, this helped us identify the edges of the tool distinctly. This was achieved by comparing the depth of the current pixel with its adjacent pixels. Hence, we decided to go along with the depth segmentation.

The next few steps would be to efficiently achieve augmentation of the tool based on its distance from the background. I also intend to work on the website.