EE/CprE/SE 491 - sdmay19-31:
Multi-Purpose Automated Robotic Mixer (mpARM)
Week 10 Report
December 16 – January 25
Client: Alexander Stoytchev/Brett Altena
Faculty Advisor: Alexander Stoytchev

Team Members

Drew Caneff — *3D Printer Specialist/CAD Designer/Accountant*
Amos Hunter — *Electromechanical Specialist/Meeting Scribe*
Brett Altena — *Meeting Facilitator/ Computer Vision Developer*
Kristian Wadolowski — *Report manager/Front-end programmer/Computer vision Developer*
Jase Grant — *Embedded Systems/ Assignment Manager*

Summary of Progress this Report

- **Frame work -** Found a place to store materials and tools which would not otherwise fit in my apartment. Drafted two new documents which highlight how the joints of the frame are held together. I decided to use wood blocks and screws with washers as a fastening solution. The sketches also show how to orient the strut channel. The original plans did not account for the specific kind of steel channel available, and so it was necessary to be careful of how the channel was facing. I took measurements from the robotic arm base to align the holes in the base with the steel bars of the chassis. It was necessary to make these tweaks after shopping for parts and materials and it was discovered how the options available for purchase differed from initial design requirements.
- **Computer vision -** The approach to the computer vision program is being more thoroughly defined by determining the best method of determining pancake readiness. Both approaches use the same method in the beginning by taking in a real-time image of the pancake cooking, and using a specific threshold that eliminates lighter colors such as the pancake batter color and darker colors such as the griddle background. This provides an image where the black groupings represent the bubbles that popped on the pancake. The first approach is to count all the groupings ensuring that the same position is not counted more than one time within 10 secs, and the second approach is to count the amount of bubbles at a given time and if it passes a specific number it will signal its readiness. The first approach I believe will be more consistent and measurable (the best pancake made took a total of 34 well-seen bubbles to appear), whereas, the second approach is left more to random chances (the best pancake was flipped when 6/7 bubbles were present). Both approaches have the risk of under or over cooking the pancakes.
- **3D part cleaning** - Upon cleaning the 3D printed parts, a concern arose associated with the quality of some of the 3D printed parts. As a result of some of the parts complexity and the limitations of the home printer some aspects of the 3D printed parts were low in quality which could in turn result in complications when constructing the robotic arm. Some of the problems which may arise could include; difficulties assembling the robotic arm as parts will be difficult to fit together, a slightly malfunctioning arm from 3D printed parts not properly working in unison, and even a completely broken arm as a result of a 3D printed part breaking during arm operations.
- **Arduino code** - When I attempted to upload the code to the arduino I notices that the code had very sparse documentation. The code itself is a modded version of grbl, which itself is an open source project. This likely led the modders to neglect properly documenting how to set up the code, and how to use the provided functions. There is a wiki for this modded version, but it contains minimal useful information. So in order to use the provided code I need to analyze the key components to learn how to operate the

robotic arm. In the main method I found a reference to the main loop, which is stored in protocols.c In the main loop I found that the arm can be operated in real time by using a serial input, and found the function for executing serial inputs in system.c I am currently deconstructing the serial inputs so that we can get a list of all functions that can be used via serial input.

Pending Issues

- Batter issues (Splashing, feeding, mixing, type, etc...)
- Complete Computer vision code
- Assemble the arm

Individual Contributions

Team Member	Contribution	Weekly Hours	Total Hours
Drew Caneff	➢ Received 3D printed parts from providers ➢ Completed 3D part clean up ➢ Performed inventory check ➢ Began working on PCB ➢ Purchased additional components ➢ Participated in group meetings	24	156
Amos Hunter	➢ Found workshop location/materials for frame ➢ Produced detailed frame sketches ➢ Participated in group meetings ➢ Wrote and posted meeting documentation	14	117.5
Brett Altena	➢ Reviewed computer vision progress ➢ Outlined new computer vision strategy ➢ Created spreadsheet of pancake data ➢ Created group attendance sheet ➢ Participated in group meetings ➢ Set up advisor meeting	18	132
Kristian Wadolowski	➢ Completed tasks: ➢ Attended all meetings ➢ Received arduino to begin development ➢ Began uploading code to arduino ➢ Began analysis and deconstruction Arduino Code	12	88
Jase Grant	➢		37

Plans for Upcoming Reporting Period

Team Member	Plans
Drew Caneff	➢ Finish PCB ➢ Send off new buy list ➢ Collect remaining Thor Arm parts ➢ Begin construction of Thor Arm

Amos Hunter	➢ Purchase materials ➢ Attend meetings ➢ Produce update emails ➢ Continue scribe duties
Brett Altena	➢ Have advisor meeting ➢ Research new approach for computer vision
Kristian Wadolowski	➢ Continue to analyze the arduino code ➢ Figure out the serial inputs ➢ find a way to test the arduino without the robot arm if possible ➢ Write up the weekly report
Jase Grant	➢

Gitlab Activity Summary

Action: joined, Tue Sep 04 2018
Author: dvcaneff
