To: Dr. Satici, Nardos Ashenafi, Brian Higgins

From: Robotic Vision Team: Daniel Pullicar, Mason Cannon, Roscoe Ambrose, Haston LaGrone

Subject: Biweekly Update 3/12/2023 – 4/2/2023

Date: April 2, 2023

Hello again professors. We hope you had a good break and a great start to April. Here is a short update about progress over the last few weeks.

# Work Completed

Daniel Pullicar:

* Completed image-to-world point conversion (Woo hoo!)
* Edited prototype video

Mason Cannon:

* Assisted in testing and compiling the programs for calibration and detection.

Roscoe Ambrose:

* Reinstalled Linux (Ubuntu 20.04)
  + Installed ROS on Ubuntu
* Compiled Calibration and GUI code into our class implementation of HoughCircles.

Haston LaGrone:

* Tested tandem camera operation.
* Tested calibration, GUI, and detection operating concurrently.
* Determined that further attempts at image refinement are best done with the GUI rather than trying to preset intrinsic values. Each camera interprets the configurations slightly differently.

# Areas That Need More Attention

# Work Pending

The team continues to try and put some finishing touches on the program. As the operation moves to a Linux base, further testing will be required. Emphasis on the API backend and camera setting interpretation. We are also planning to add default configurations for the cameras available as well as a method to save camera settings for future use.

Daniel Pullicar:

* Give Roscoe GUI requirements
* Help Roscoe with loading variables
* Edit poster

Mason Cannon:

* Edit poster for the showcase
* Resolve problems that come from downloading the files from GitHub
* Assist with the file loading system for cameras already calibrated

Roscoe Ambrose:

* Improve sizing capabilities of GUI to fit on smaller/different resolution screens.
* Add a save variables button to the GUI.
* Load all variables from file when starting program.
* Add extra parameters to Caller.py to properly initialize variables.

Haston LaGrone:

* Investigate multiple camera capture on a more powerful computer. FPS is bottle necked by CPU with multiple cameras operating at more than 30 FPS.
* Configuration save option.
* Investigate ROS node implementation. (Ongoing)

# Conclusion:

The three critical elements of the project are capable of working together and accurately providing the real-world coordinates of the puck, which is exciting to the entire team. We continue to try and refine and improve the overall design, as well as modify it to be accessible on the Linux based device. We are quite optimistic the progress will continue, and we will be able to provide the desired end product.