Final Project Progress Report

Dakota Ward, Sam Burkhart, Khyati Sinha
ECE 544
Spring 2017

Mechanical Design:

Completed:

- Base robot assembly and motors
- Gripper arm
- Camera

ToDo:

- Attachment of the PYNQ board
- Attachment of the battery pack(s) for the PYNQ/Camera/motors

IP Design:

Completed:

- Identified WIFI connection options for PSU
  - Can connect to PSU Secure for network but not control right now.(Possible port blocking)
  - Can use hotspot from either laptop or phone to control the robot.
- Generated Vivado project for PYNQ board and identified IP insertion points and tasks.
  - Can “gut” current PMOD controls, and use our custom IP blocks.
  - Or we can use the IOP blocks in GPIO mode to control the motors with a minimum IP design. (Will test both methods)

ToDo:

- Test the IOP as GPIO control for PMOD DHB1 device and gripper arms.
- Push the custom DHB1 IP to a custom Overlay and interface with Python driver

Motor Control:

Completed:

- Motors fully assembled and connections to the DHB1 identified
- Identified interface of dual h-bridge parts (DHB1) and began design of custom IP based on HB3 design from Project ⅔
- Plan to implement motor control without feedback

ToDo:

- Digilent motors with Quadrature encoding built-in are available, and may be used if time permits, as this would require changing the robot design as well.

- As stated above: We can leverage the IOP blocks for the PMOD in GPIO mode to directly control the motors to start, and create Python drivers for the motor interface/control. For full functionality we would want to incorporate the custom DHB1 controller design though.

Servo Arm:

Completed:

- Gripper assembly completed and plan for servo control identified

ToDo:

- Connect servo control to Arduino GPIO ports and use IOP for Arduino in GPIO mode to control the gripper arm.
- Design a top level python driver for the servo arm

Integration/Control:

ToDo:

- Complete the algorithm design for manual and automatic robot movement
- Design layout of “terrain” where tests will be performed
- Design control interface and user feedback while tests are being performed
- Test with various cans/colors

Computer Vision:

Can detection: (in progress) Training a haar-cascade classifier and Deep Learning Network.

- Sample sizes (positives, negatives) need to be fine tuned
- Integration with web camera needs to be tested
- LBP (Local binary pattern) can be used instead of haar-like features (faster to train) This method will be investigated and detection rate compared to haar.

Distance Measurement: (in progress):

- Python script using similarity of triangles completed but needs fine tuning
- Integration with can detection algorithm needs to be implemented

Color detection: (in progress):

- Will be developed once basic can detection algorithm/method is fully implemented

Deep Learning network: (in progress):

- Steep learning curve and difficulty setting up toolchain has slowed progress.
- Once classic CV approach is implemented, will switch over efforts to utilize the DL for possible brand label recognition