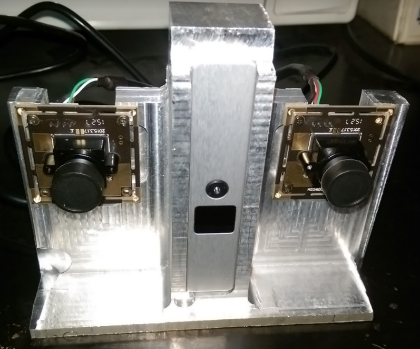
|  |  |
| --- | --- |
| **ADEV Image Correlation** | **2016-2017** |
| **Tim McDaniel**  **Dustin George**  **Drew Maione** | **Sponsored by**  **https://lh6.googleusercontent.com/MDjqszAwrVpDqF3solBLLV6UkoT5-gl-Pf1IojnEQ8zz9ccShE0DmyKaPprDtLpCGW_wRFYehc53gkeNwYssA7TEFK0vKj66PsvDWCJKg6ZhPuj4Z5L2kdBUfMBZTYWm0brupeD7** |

**Objective**

The client for our senior design project is ADEV Automation, a robotics startup company that is developing a robot capable of navigating a strawberry field and picking strawberries using a cutter arm. We were tasked by ADEV to correlate the images generated by three cameras; two webcams and a depth camera. In other words, our project was to give a robot depth perception. ADEV wanted to be able to select a specific pixel in an image and be told the location of the image in the pixel relative to the robot arm. ADEV also needed a mount to hold their cameras on the end of the robot’s arm.

**Cameras**

Our project uses three cameras.  The first is an infrared time-of-flight camera from PMD that measures the amount of time it takes for infrared light to “echo” off of its surroundings.  We used this camera in this project to find distance readings from our sensor’s location to any point of interest in the area.  The second and third cameras are standard RGB webcams.  With the three of them arranged in a line, we are able to select a pixel in one of the webcams and then have the program calculate the location and distance to that pixel relative to the depth camera.



Correlation

C++ to access depth camera

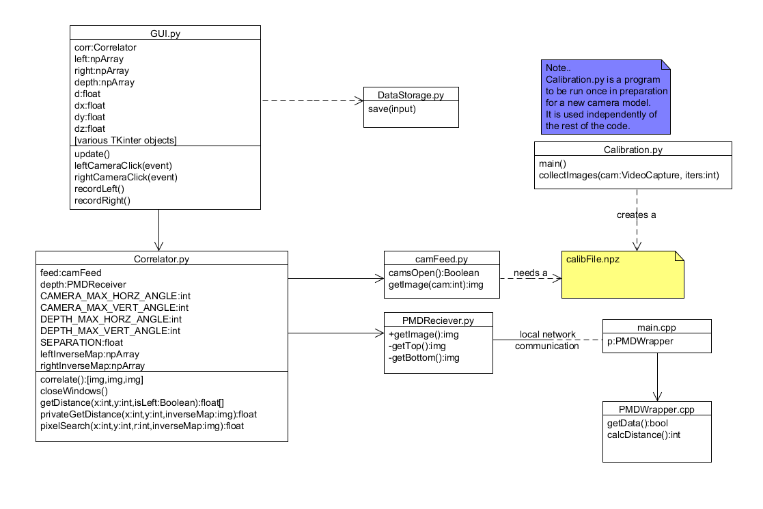
Main project and GUI is in Python

Webcam calibration

Use trig to correlate

Describe GUI interaction + storage

UML diagram



Mount

|  |  |
| --- | --- |
| https://docs.google.com/drawings/d/s1IySHhxA33ZA7Xp2-4wSVw/image?w=294&h=242&rev=221&ac=1 | https://lh3.googleusercontent.com/2JXZQERSZLVTyl2TD-7H74_3qbjV_fhclOSIODmxL-wmresEcDtlIx-OyZmLdi7TPNmA06dmONDhSoe6t6bt7HPZLmrTdfCtesOvl0GWnIPyXQKEYm6pFlG2kopx7pPsKs5aEdPS |

**Needs**

·       Software must correlate the webcam’s pixels to each depth camera pixel.

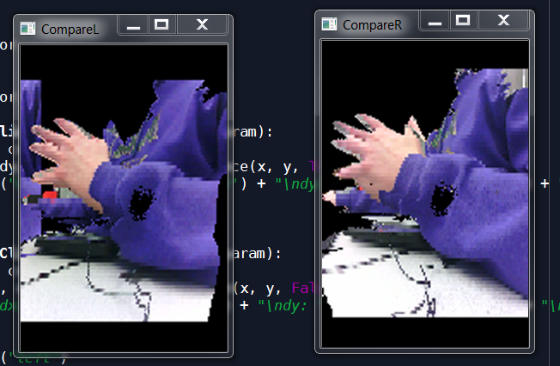
·       Software must return x, y and z displacement values relative to cameras.

·       The mount must weigh 0.5 lbs or less in order for the robot arm to carry it.

·       The mount must be able to dissipate the heat from the depth camera.

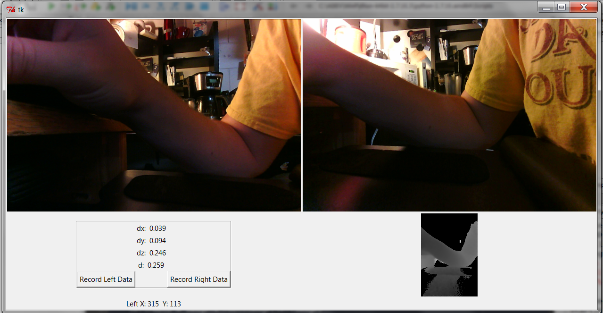
**Evidence**

**Software must correlate the webcam’s pixels to each depth camera pixel.**



This is an image of the correlated depth and webcam images with the left and right webcams corresponding to the left and right images above. The black space in the background are pixels that were out of the depth camera’s range.

**Software must return x, y and z displacement values relative to cameras.**

****

The GUI we designed for this software is shown above. The left and right color images represent the left and right webcams and, when these images are clicked on, the software returns the location and distance of the pixel selected relative to the depth camera. These values are returned in the bottom left-hand corner as the dx, dy, dz, and d values.

**The mount must weigh 0.5 lbs or less in order for the robot arm to carry it.**

We were able to reduce the weight of the mount to 0.69 lbs which ADEV stated as satisfactory. We also informed ADEV that, if in the future there was a need to reduce the weight of the mount, there was still material that could be removed.

**The mount must be able to dissipate the heat from the depth camera.**

When attached to the aluminum mount the depth camera shows a temperature of 80.33℉. However, when removed from the mount, the temperature very quickly increases to 116℉.