Grasped Object Orientation Detection for Robust Peg-In-Hole Tasks

NSF Summer Undergraduate Fellowship in Sensor Technologies
Kendall, Queen, Sunfest Fellow (Computer Engineering)
University of Maryland Baltimore County
Advisor: Kostas Daniilidis, PhD (SEAS)

ABSTRACT

This paper approaches the problem of detecting the orientation of an object while the object is grasped by a robot. Manipulator robots are learning new and exciting ways to interact with their environment every day. Object recognition and grasping algorithms allow a robot to identify, target, and grasped objects of importance. However, a common problem in grasping is maintaining knowledge of the orientation of the object while it is being grasped or manipulated. We have developed an algorithm that estimates the orientation of painted plastic test tubes without assistance from external markers. By initially using line fitting to localize the center axis of the tube to assist in the tube's orientation detection, we then compare the orientation of the tube to the gripper's approach to the test tube array. Based on the angle difference between the +Z-axis of the gripper and the center axis of the tube, the movements necessary for the robot to reposition its arm to properly approach the test tube array can be calculated leading to success in a Peg-In-Hole task.

Table of Contents

1. INTRODUCTION

The world is now approaching the use of robotic technology in ubiquity. From autonomous vehicles to disaster relief, robots are being utilized and integrated into everyday tasks. Humans assisted by robotic systems are starting to eliminate risk and danger from hazardous situations. For example, with the help of a micro drone swarm that can respond to a scene of a hostile situation, first responders can be more knowledgeable and better equipped to protect themselves and efficiently resolve the situation.

Another example is related to the recent emergence of infectious disease incidents. Robotics could be used to evaluate and test human samples for infections while reducing the risk of contamination or unwanted exposure. In order for a robot to engage in these complicated and meticulous tasks, the robot must be able learn about its environment and manipulate the tools at its disposal to increase efficiency. Among a number of obstacles to overcome before a robot is equipped to participate in a laboratory setting, **Figure 1**, we found that grasped object orientation detection would be an impactful tool to increase the accuracy of peg-in-hole tasks as well as increase safety and success during tasks that include peg-in-hole.

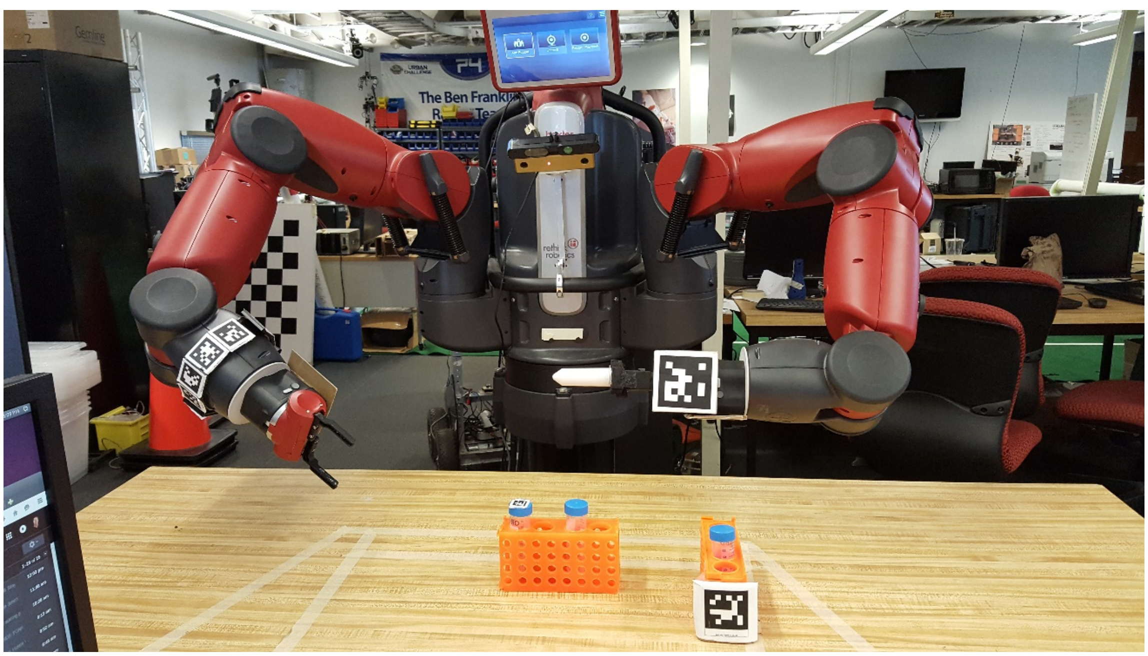

Figure 1: Grasp Lab's Baxter robot at a work bench with laboratory test tubes and test tube arrays

Assuming the robot has knowledge about the objects in the testing environment and how to manipulate and interact with those objects, there are a number of object qualities that are taken for granted that can affect change in the environment. In previous experiments, we found that our robot is able to identify and grasp objects, but in some situations the object's orientation can be altered while the robot has the object in its grasp. We noticed and decided to focused our research on painted test tubes, **Figure 2**.

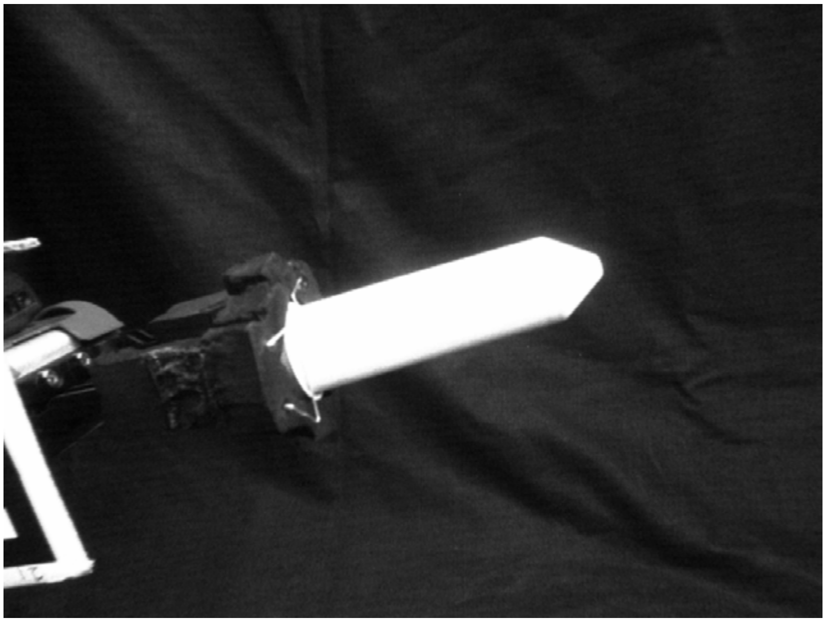

Figure 2: Robot gripper holding experimental test tube

This introduces the novelty of orientation detection. The robot will need to place down a test tube in a specific position in the test tube array. If the robot is unaware that the tube's orientation has altered due to slippage caused by weight or lack of grip, when the object is finally placed down it will most likely be in the wrong final position. By developing an algorithm that can identify the orientation of the tube in the robot's gripper, we can improve operational accuracy and precision while ensuring success for peg-in-hole tasks.

2. BACKGROUND

A number of research scientists and roboticists have approached the problematic aspects of pose and orientation estimation of objects. First the robot must be able to detect the object of interest. Rabbani, et all [1], and Thomas, et all [2], both used

Hough transforms to detect cylindrical objects of interest in order to infer cylindrical edges and object axis. Due to our use of cylindrical objects, Hough transforms and other edge detectors are extremely important in cylindrical object detection.

Cylindrical object detection is important, but our objects are also made of glass or plastic, which means that they are transparent. Lysenkov, et all [3] managed to recognize and estimate the pose of transparent objects with a Kinect Sensor. This is a novel approach because it is popular belief that the depth sensor would penetrate any type of transparent object. Nevertheless, they focused on the areas where the Kinect failed to produce a depth map, thereby using the disadvantage of lack of detection as an advantage for detection. Phillips, et all [4], used learning algorithms to assist in detection of edges and shapes of the transparent objects.

Netz, et all [5], approached object pose estimation via invariant descriptors from specular highlights. This approach permitted very high averages of success in estimating pose for a number of objects, but the pose estimation algorithm depended solely on specular highlights, which are not guaranteed in our circumstances. Zhu, et all [6], however, emphasized a robust pose estimation of trained objects in cluttered spaces. With the use of model silhouettes of the objects, the system could recognize the pose of the object in preparation for a robot to grasp.

Peg-in-hole [7] is a problem tests position and control capabilities. **Figure 3** shows an example of a peg-in-hole task. Most would remember peg-in-hole as playing with baby blocks where the triangle fits in the triangular hole and the ball fits in the circular hole. This task is a crucial process for almost every element at a work bench.

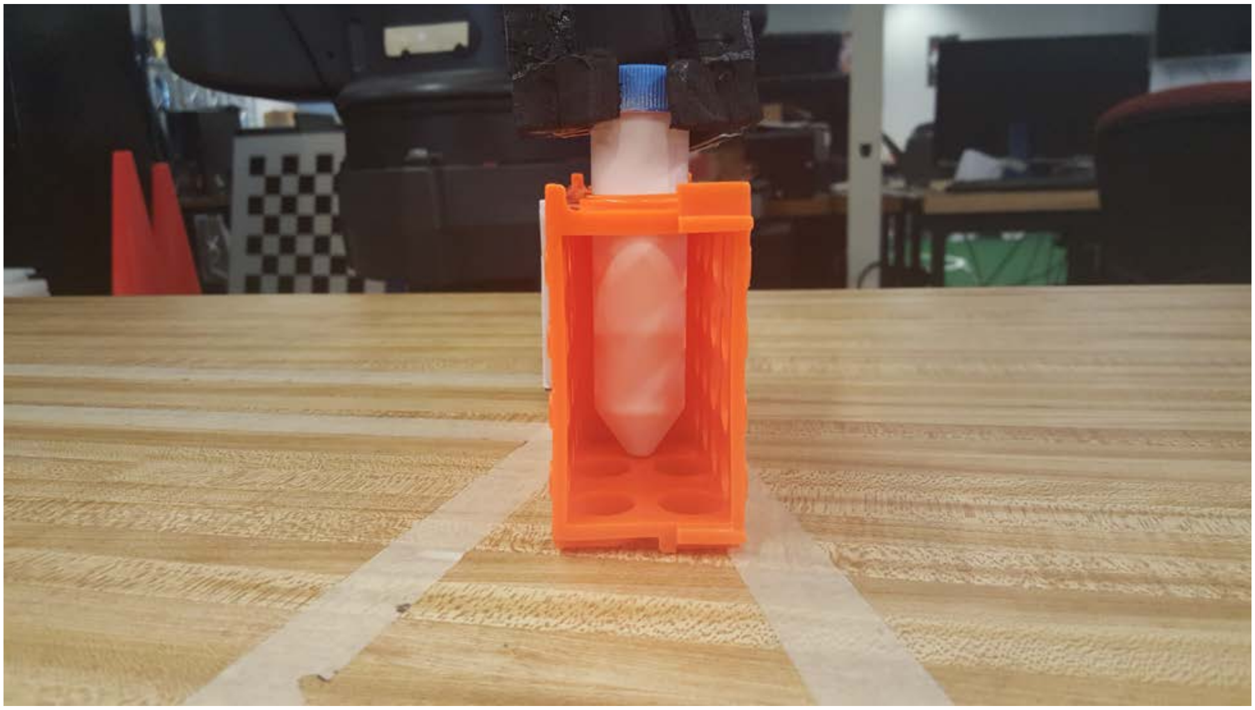

Figure 3: Robot placing a test tube in the array demonstration of peg-in-hole

3. TECHNICAL COMPONENTS

3.1 Visual Transformations

Transformations are a computer vision technique that includes translations and rotations. In this situation, transformations were used to properly orient real world objects from the world frame or perspective to the perspective of another coordinate frame. A transformation can contain a translation(T) or a rotation(R) or both. A translation is the movement to a point in the world frame with respect to the target frame, the camera. For example, if the object in the world has an origin of P(3, 5, -6), that corresponds to 3 units in the +X direction, 5 units in the +Y direction, and 6 units in the -Z direction. The translation connects the origin of the world to the origin of the camera. A rotation is a bit more difficult.

Similar to a translation, a rotation is determined by a matrix, but this matrix is multiplied by the world frame in order to be properly orient the object with the camera frame. The rotation matrix is a 3x3 matrix whose determinant is 1.

$$\mathbf{{}^{c}P = {}^{c}R_{w} * {}^{w}P + {}^{c}T_{w}}$$

For example, **Figure 4** displays the object's coordinate system in the world view. The camera(c) coordinate system is presented on the right. is the translation between the camera origin(^{c}P) and the object's origin(^{w}P) in the world. In order for the camera to properly determine the position and orientation of the object in the camera's frame, the object must be represented in the frame of the camera, thereby transforming the camera coordinate system to that of the world [8].

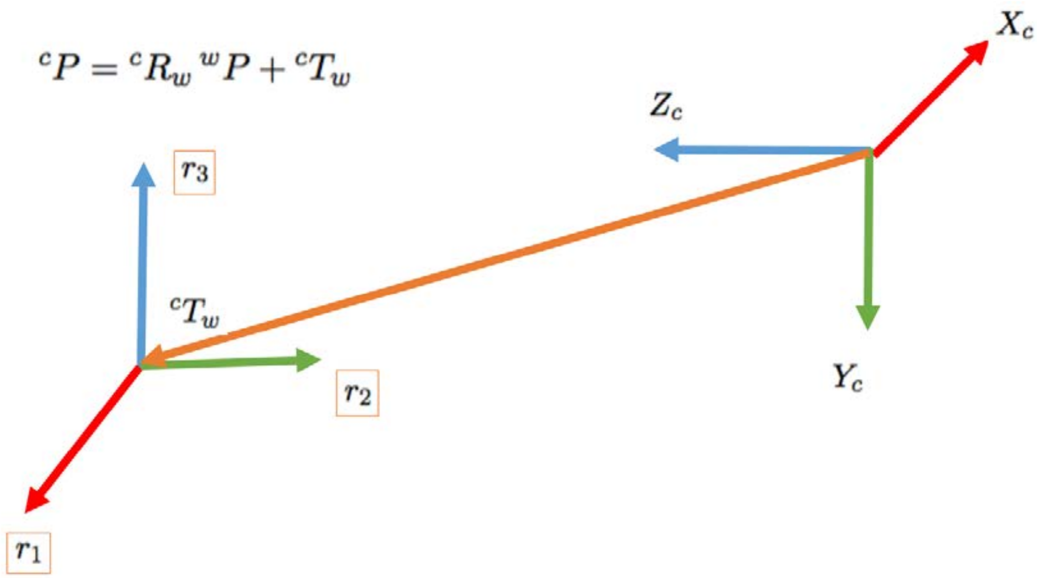

Figure 4: 3D transformation from the camera's origin to the world's origin

With a synced universal coordinate system, we can use the data from the perspective a single device, a camera, and make control decisions with accuracy.

3.2 April Tags and Visual Servoing

The robot understands its own position and the position of the test tube array via april tag detection. April tags are fiducial markers that, when read and computed by the detection software, can display 3D position, orientation, and identity information of the tags relative to the camera [9]. The april tags we used were part of the Tag36h11 family. With the assistance of april tags, we were able to accurately determine the positions of each test tube array index with respect to the april tag attached to the array. For our purposes, the april tags are being detected through the right perspective of a stereo bumblebee camera.

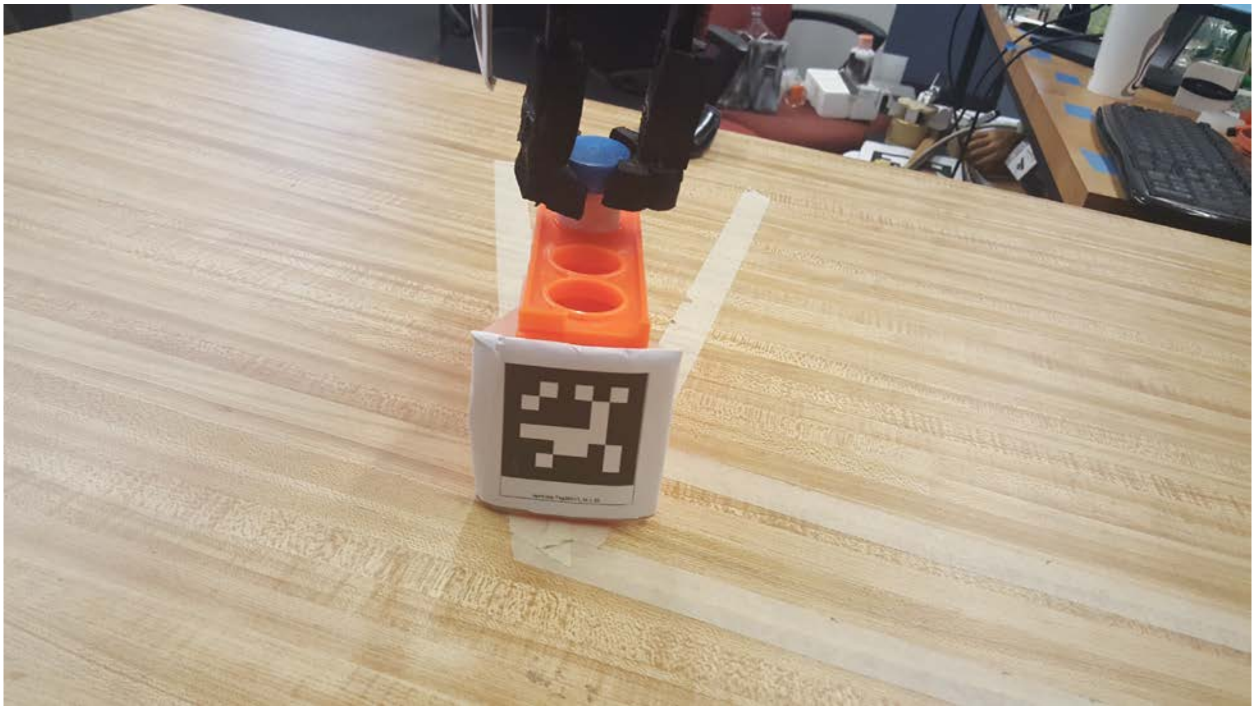

Figure 5: Experimental test tube array with april tag attached

We are also using april tags to assist in the robot's arm movements. This is a technique called visual servoing, or vision-based robot control. With the information gathered by the april tags placed on the robot's gripper, we are able to track and control the gripper position. Visual servoing also contributes to direct movements. Movements through the baxter robot API, may be planned and executed in a different orientation than expected. Visual servoing calculates movements from point to point, meaning in a straight line with regards to position. The system corrects the difference between current position and the desired position within a small threshold.

3.3 Test Tube Orientation Detection

In our experiments, we decided it best to use a painted test tube with a black background. This isolates and focuses the attention of the Canny edge detector and the Probabilistic Hough transform used through the OpenCV Library on the test tube. Our algorithm first detects the edges of the test tube in frames that come from the camera. The algorithm runs the detector over several frames as seen in **Figure 6**.

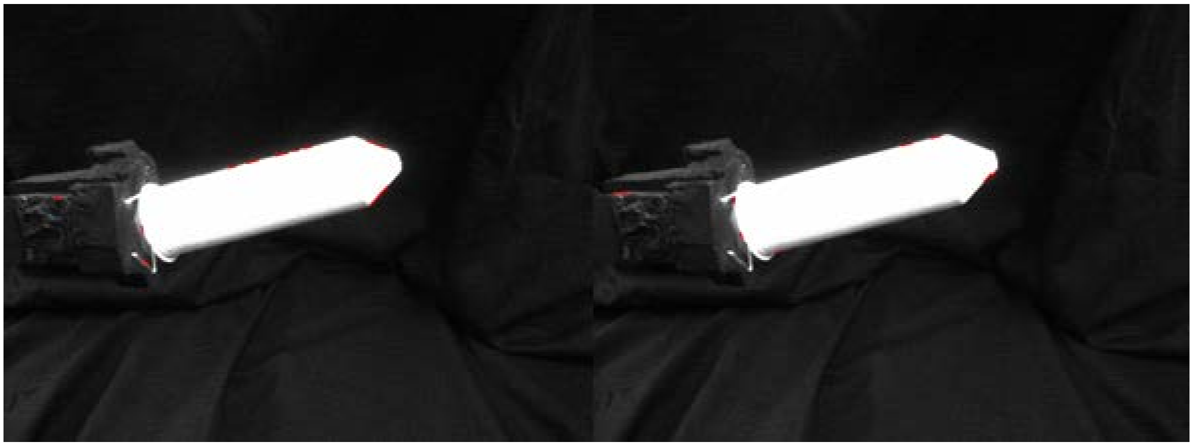

Figure 6: Segment detector iterations on two different camera frames.

The segments are collected over each iteration and republished as a separate image similar to **Figure 7**. When the new image of collected points is finished, another Hough transform commences, to fit lines to the edges of the tube. This transform usually detects several lines, the algorithm averages the two lines with the maximum distance from each other and uses their coordinates to continue the process**, Figure 8**. Due to this being a separate image, the only characteristics are the accumulated line segments thereby eliminating unwanted noise in the image.

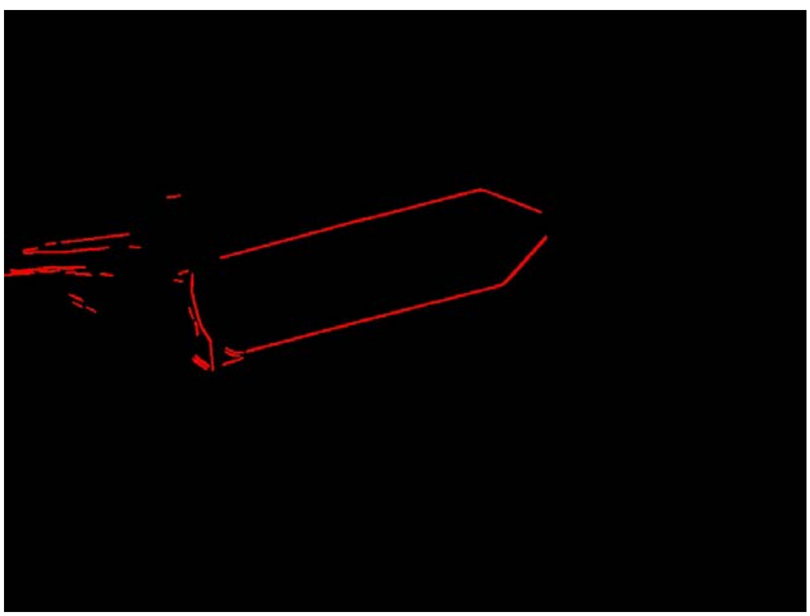

Figure 7: Image of accumulated line segments from Probablistic Hough transform

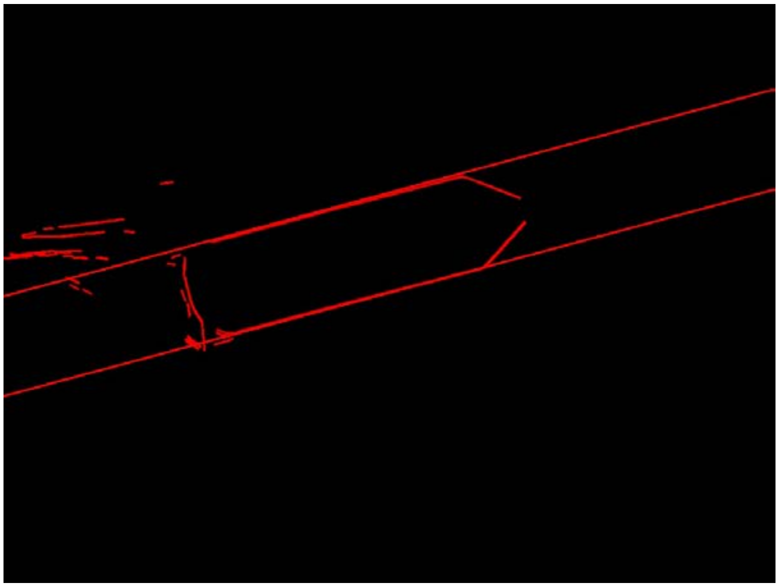

Figure 8: Hough Transform of the accumulated line segment image

The algorithm then averages the two similar lines that represent the edges of the test tube, creating the objects center axis. The center axis provides the angle at which the test tube is oriented in 2D space. A transformation from the april tag on the robot’s wrist to where the tube is grasped to find a starting point. Using the center axis angle, we project a point along the axis to estimated tip of the tube, shown in **Figure 9**.

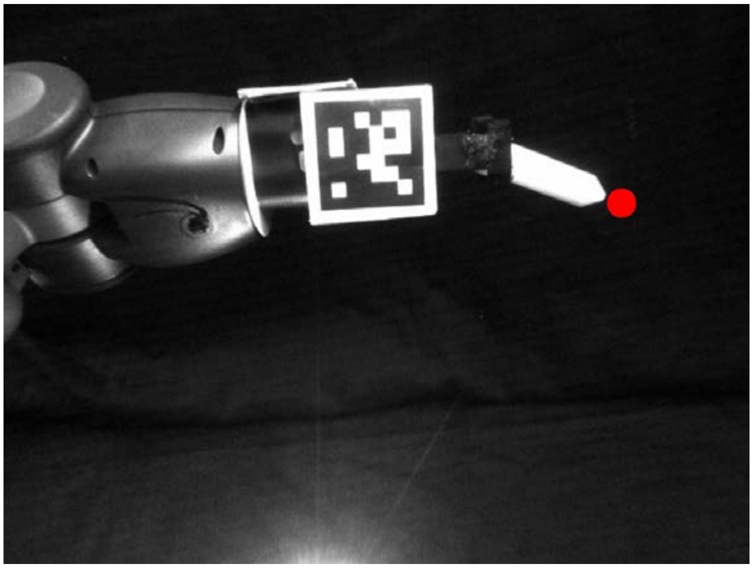

Figure 9: 2D projection of where the algorithm estimates the tip of the test tube

In order to correct the angle of approach for the peg-in-hole task, we treat the projected point as an origin for the world with the +Z axis normal to the tip of the test tube, **Figure 10**. With this new world origin, we compare the angular difference between the gripper’s +Z axis and the new origin’s +Z axis. The difference represents the change necessary to complete the peg-in-hole task successfully.

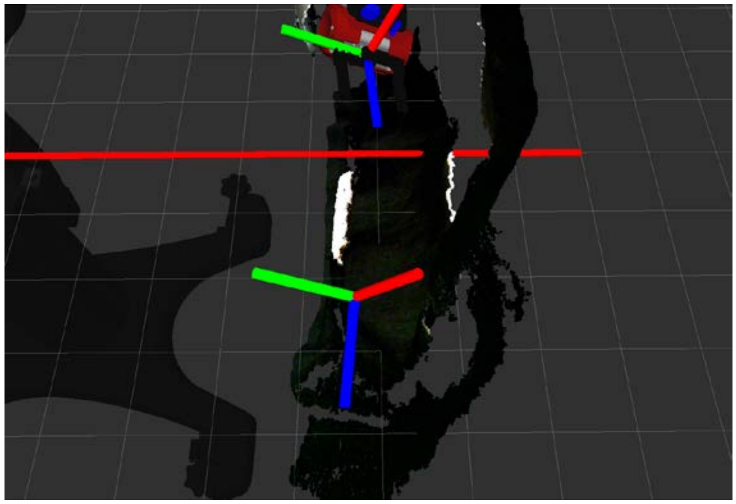

Figure 10: 3D and point cloud showing the estimated orientation of the test tube in relation to the robot's gripper

4. DISCUSSION AND CONCLUSIONS

At this point, we are able to perform all the necessary actions to successfully complete a peg-in-hole task. This includes identifying each index in relation to the test tube array april tag. The gripper is able to move to each index and grasp a test tube. The algorithm will then run and the orientation of the test tube would be detected. We have not finished the procedure to replace the test tube to a specified index. Nevertheless, how we plan to implement that transition has been explained above. We do not foresee any more fundamental issues that would keep us from finalizing a full demonstration. It is more of a matter of combining the pieces of code.

In conclusion, we are very excited about what we have accomplished so far with the software. We have new directions in mind and will be working towards some of the recommendations below.

5. RECOMMENDATIONS

We made a number of assumptions and there are a few new challenges we could overcome to make this algorithm more robust.

By using a black background, we eliminated any background noise that could alter the Hough transform and Canny detector results. It would be best if we could find a way to make the background invariant to the detection algorithm. There are several computer vision techniques that could possibly work to eliminate the background from each of the frames in the algorithm processes.

At this moment, the object orientation detection algorithm works with a painted test tube. The test tubes are naturally transparent plastic. The detection needs to be able to identify the dimensions of a test tube that is transparent and unaltered. Our initial attempts to run the preliminary detection processes were unsuccessful on transparent tubes, which was why we elected to paint the test tube white for the sake of time.

We still have not fully integrated our algorithm with a peg-in-hole demonstration. A demo would be the most direct way to prove that the algorithm works with a high efficiency rate.

We also want to collect metrics of how efficient and accurate the algorithm is working. Primarily, it is important to recognize the error in the tip estimation, because a slight error can result in an unsuccessful peg-in-hole attempt. We also want to see if it is more accurate to continue using the stereo camera or the robots arm camera for identifying the test tube array and its indices.

6. ACKNOWLEDGMENTS

I want to thank the University of Pennsylvania’s SUNFEST REU for providing me the opportunity to work with some of the brightest minds in computer vision for robotics. Thank you to Alex Zhu, my research mentor, for guiding me along this journey. Dr. Kostas, thank you for trusting in my abilities and allowing me to work with your group by assisting in discovering new knowledge in computer vision. Special thanks to the University of Marlyand, Baltimore County and the Meyerhoff Scholars Program.

7. REFERENCES

[1] T. Rabbani and F. Van Den Heuvel, "Efficient Hough Transform for Automatic Detection of Cylinders in Point Clouds," *ISPRS Work. Laser Scanning*, vol. 3, pp. 60–65, 2005.

[2] J. Thomas and G. Loianno, "Visual Servoing of Quadrotors for Perching by Hanging From Cylindrical Objects Visual Servoing of Quadrotors for Perching by Hanging From Cylindrical," 2016.

[3] I. Lysenkov, V. Eruhimov, and G. Bradski, "Recognition and Pose Estimation of Rigid Transparent Objects with a Kinect Sensor," *Proc. Robot. Sci. Syst.*, p. 8, 2012.

[4] C. J. Phillips, M. Lecce, and K. Daniilidis, "Seeing Glassware : from Edge Detection to Pose Estimation and Shape Recovery," *Robot. Sci. Syst.*, 2016.

[5] A. Netz and M. Osadchy, "Using specular highlights as pose invariant features for 2D-3D pose estimation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 721–728, 2011.

[6] M. Zhu, K. G. Derpanis, Y. Yang, S. Brahmbhatt, M. Zhang, C. Phillips, M. Lecce, and K. Daniilidis, "Single image 3D object detection and pose estimation for grasping," *2014 IEEE Int. Conf. Robot. Autom.*, pp. 3936–3943, 2014.

[7] S. K. Yun, "Compliant manipulation for peg-in-hole: Is passive compliance a key to learn contact motion?," *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 1647–1652, 2008.

[8] K. Kitani, "2D Image Transforms," 2016.

[9] E. Olson, "AprilTag: A robust and flexible visual fiducial system," *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 3400–3407, 2011.