Robot Manipulation Grasping of Recognized Objects for Assistive Technology Support Using Stereo Vision

Sutono Effendi, Ray Jarvis and David Suter
Monash University, Clayton, Victoria 3800, Australia
{sutono.effendi, ray.jarvis }@eng.monash.edu.au
david.suter@adelaide.edu.au

Abstract

This paper demonstrates a "Grasp X" robot arm manipulation using a 2.5D disparity map and a Scale Invariant Feature Transform (SIFT) algorithm for recognizing the target object. The SIFT algorithm is applied after stereo segmentation, which reduces the matching time. Due to the high inaccuracy of the stereo disparity map and the maneuverability limitation of a two-finger gripper of the robot, RT 100, a cube model is used in order to calculate the object centroid as a grasp point.

1 Introduction

More and more attention has been given to human centric robotics research world-wide. One typical increasingly important application is robots as domestic helpers. The ultimate goal is to achieve a robot, which is able to execute human commands such as "Please, bring me my blue cup!" The robot should then be able to search for the blue cup in possible places such as a table in the kitchen, a table in the living room, etc.

Despite a robot's success in a controlled environment such as a factory floor, it is still very difficult for a robot to perform in a domestic setting [Kemp *et al.*, 2007], which is uncontrolled and dynamic. In a controlled factory setting, a limited number of known objects are used; therefore their shape, position and orientation (pose) can be easily estimated. By contrast, in a domestic environment, various different kinds of unknown objects are to be manipulated.

In domestic environments, in order for a robot to manipulate an object, it needs to know the object's 3D pose. There are a number of techniques in order to recover 3D information from 2D images (to reduce visual ambiguity) such as structured light, laser range finder, tactile sensor, etc. They all may be found inappropriate. This is because a domestic robot's domain is surrounded by people and thus, applying a laser range finder on human beings is certainly undesirable. Hence, this research explores the possibilities of non-intrusive passive methods such as stereo vision.

The ultimate objective is that robots can manipulate recognized objects in unknown environments without disturbing the surrounding objects. However, those surrounding objects are not necessarily recognized. Yet, their pose should be known in order to prevent collisions during manipulation. For instance, in order for a robot to reach a recognized target, it may encounter many unknown obstacles on the way. Hence, it is sufficient for a robot to know they are NOT the target yet be able to avoid them.

Within the constraints of non-intrusiveness and no prior knowledge of the 3D object's shape, we have developed our research of human centric assistive robot support for domestic environments.

This paper is organized as follows: section 2 describes previous related works done, followed by our hardware framework in section 3, section 4 shows our proposed techniques to use stereo segmentation for object recognition and stereo reconstruction; section 5 discusses experimental results, section 6 gives our conclusion and, finally, section 7 states some future research directions.

2 Related Work

The Visual information for a robot is very important in order to know where and how to manipulate an object. In [Sanz *et al.*, 2005], several industrial applications of a visually guided system for robot grasping using an inexpensive two-finger gripper are reported. The capabilities of this methodology to deal with more complex shapes, even 'a priori' unknown, are studied. However, that work is limited to planar objects. Unfortunately, this is not realistic for domestic robot applications, as lots of non-planar unknown objects are involved.

The assumption is made that everyday objects that are prone to be handled by service robots tend to consist of primitive convex geometrical shapes, especially rectangular boxes and cylinders and can therefore be described by a superquadric model as in [Taylor and Kleeman, 2006; Schlemmer *et al.*, 2007].

[Jafari and Jarvis, 2005] use a tactile sensor on the robot hand to detect the object, table, etc and thereby measures the distance of the table from the camera using the manipulator configuration data. They then perform relative visual servoing of the end-effector by monocular color camera to reach and grasp objects on the table.

[Taylor and Kleeman, 2006] use a laser range scanner and stereo cameras to detect geometrically primitive objects (boxes, bowls, cylinders). They used scene segmentation using surface curvatures and later built a 3D model of the object. After the shape of the object is obtained, it is combined with appearance properties such as color and texture in order to be able to track the object pose. This method works even in the presence of occlusion. This system is similar to the setup in [Schlemmer *et al.*, 2007].

[Kragic, 2001] implemented visual servoing with a given wireframe model of the object. However, the initialization part to estimate the object pose with the object model is still performed manually by clicking on corner points. Later in [Kyrki and Kragic, 2005], it is improved by automatically initialization by employing the SIFT algorithm [David GL, 2004] compared with initial known pose object. Objects model from a database of known objects are matched with stereo B-Rep in order to perform object recognition in [Sumi *et al.*, 1997]. In addition to "eye-in-hand", and stereo cameras at a fixed location, another camera mounted on the ceiling looking at planar objects on the table is used [Bennet and DeJong, 1996]. This method is avoided for two reasons. The first one is that, in order to grasp a recognized object, there may be obstacles or unknown objects along the way. Yet, the robot should be able to manipulate or move them away so that its arm can reach the desired object. The second one is that, since the robot should be competent in manipulating unknown objects, it is then also able to estimate the recognized object's pose without prior knowledge.

A no prior knowledge grasp algorithm using vision was reported recently in [Ashutosh S, *et al.*, 2008]. Their work argued that many objects share the same sub-part for grasping. Hence, the algorithm is trained via supervised learning of 2D images to identify a few points for grasping. Next, only those few points are triangulated to extract their 3D locations for actual grasping. This method has been successfully tested in a domestic setting such as unloading items from a dishwasher, etc.

Recently, in [Yamazaki K, *et al.*, 2008], a two gripper vision guided mobile robot implementation was described. Their work demanded that the object location is known, however, no constraint is placed on the object shape. The work also assumed that the object has enough texture so that its 3D model can be built using Structure From Motion (SFM) as in [Yamazaki K, *et al.*, 2004]. Hence, the object's features are easily extracted and tracked by employing a Kanade-Lucas-Tomasi (KLT) - tracker to create dense 3D points. In addition to that, the object's features are also useful for camera pose estimation. After the 3D object modeling has been completed, the grasp is then carried out and its algorithm is as exemplified in [Yamazaki K, *et al.*, 2006]. The grasping method is computed by minimizing three cost functions, namely the contact area, the object gravity balance, and the grasping pose.

Our framework setup is similar to that of [Jafari and Jarvis, 2005; Yamazaki K, *et al.*, 2008] for achieving human-centric solution. The difference is that our system utilizes stereo instead of monocular vision and uses 2.5D instead of 3D object modeling as in [Yamazaki K, *et al.*, 2004]. The self-occluded object area from the camera views during manipulation is assumed to be flat as a cube model. This approach is much simpler than the object convex polygon approach in [Harada, *et al.*, 2008]. In addition to that, our approach performs object recognition before the grasp planning. This process, however, is skipped in [Yamazaki K, *et al.*, 2008]. Hence, their work requires the object location. In comparison to the work in [Ashutosh S, *et al.*, 2008], ours use 2.5D while theirs employed only 2D information.

Unlike the works in [Sanz *et al.*, 2005; Katz and Brock, 2008] and many others that used planar objects, we emphasis our work on non planar objects. The reason lies in the application of the assistive technology such as the domestic robot application domain where one is likely to encounter non-planar objects more often than planar objects.

3 Robot Hardware Framework

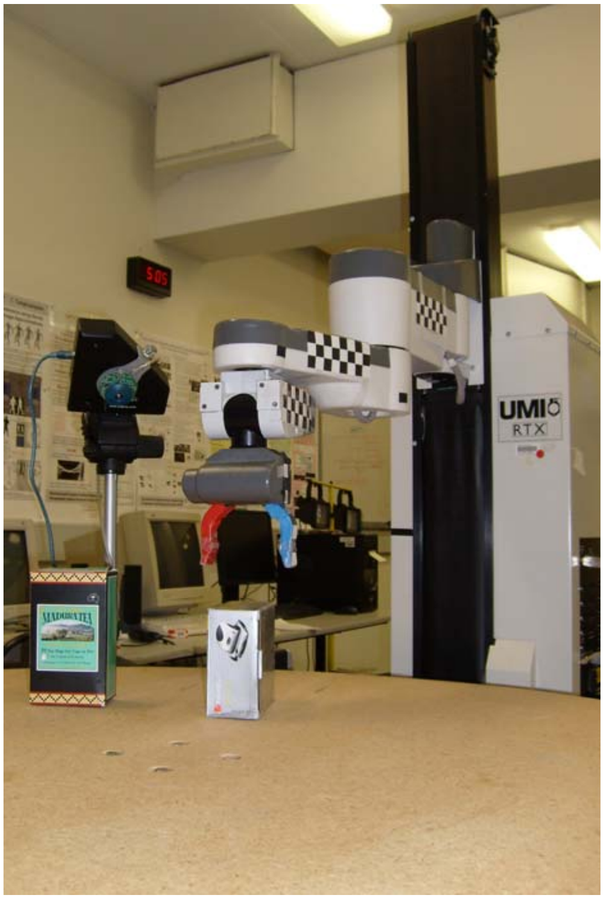


Figure 1: Human Centric Robot Framework

Figure 1 shows our hardware configuration, consisting of a robot arm (UMI RT100) and a Triclops camera. The robot arm is a SCARA robot type with 6 DoF. In order to establish a humanoid environment, the camera is placed at the right hand side of the robot. This robot arm driver is available online [Knight, 1999]. The Triclops camera consists of three B/W digital CCDs included in one package. The advantage of using this camera is that it has built-in functions for stereo matching, image rectifying and generating disparity maps.

The camera is configured for 8 bit depth resolution with an image resolution of 320x240 pixels, and a frequency of about 4 Hz (after being processed through Triclops SDK). The raw data from the camera is transferred to a notebook via IEEE – 1394 while the robot is controlled via RS 232.

The above is our initial hardware setup. However, our long-term objective would be a human centric robot mounted on a mobile platform with a notebook on-board as the control system.

4 Structure from Stereo

4.1 System Overview

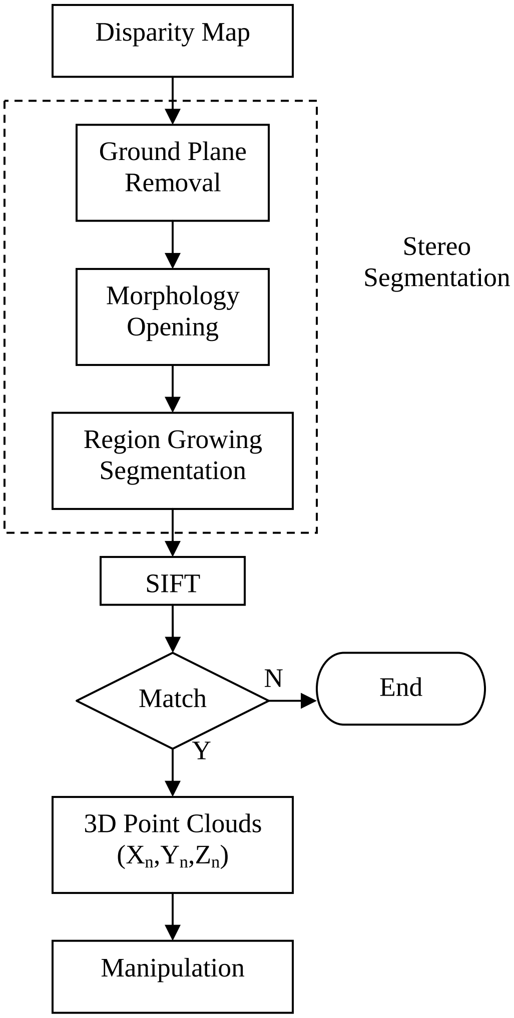


Figure 2: Stereo Manipulation Overview

Unlike [Taylor and Kleeman, 2006] who used laser scanning for 3D object reconstruction, our method uses a disparity map from stereo to produce 3D object reconstruction for manipulation as shown in figure 2.

Since a Triclops camera can produce a disparity map, we utilize this capability to speed up the processing time. There is an assumption made that all objects rest on a flat surface. Hence, ground plane removal can be used prior to object extraction. However, due to noise, false matching, etc; there may be few small blobs left. These blobs can be cleaned up according to their size by morphology operations.

These objects can occlude one another; therefore their disparity blobs may be also joined together. To separate each individual object, region growing is used which assumes that an object's disparity variation should be smooth. The segmentation result from region growing is then matched with images from a pre-built database using the Scale Invariant Feature Transform (SIFT) algorithm. If a match is found, 3D point clouds are calculated to obtain a box model and grasp manipulation is performed from above as our initial experiment. This is the simplest way for a grasp task as there may be many obstacles to avoid if the robot arm reaches the desired object horizontally. However, there are many cases that the robot will need to perform the grasping task horizontally, such as getting a bottle out of a fridge, etc. This will be included in our future directions.

4.2 Stereo Segmentation

In order to process a disparity map, a simple pin hole camera model in [Trucco, 1998] is used. Thus, a relation between a pixel $\left(x,y\right)^T$ in the acquired image and a point $\left(x^w,y^w,z^w\right)^T$ in 3D space world coordinate can be defined as follows

$$x-o_x=-f_x\frac{r_{11}x^w+r_{12}y^w+r_{13}z^w+T_x}{r_{31}x^w+r_{32}y^w+r_{33}z^w+T_z} \quad (1)$$

$$y-o_y=-f_y\frac{r_{21}x^w+r_{22}y^w+r_{23}z^w+T_y}{r_{31}x^w+r_{32}y^w+r_{33}z^w+T_z} \quad (2)$$

where o_x,o_y are principle points; f_x, f_y are focal length in the *x* and *y*-axes respectively, r_{mn} is the rotation from the camera to object, and T_x,T_y,T_z are the translation in the *x*, *y*, and *z*-axis respectively.

Since the Triclops software development kit (SDK) is able to provides rectified images, the disparity is defined as follows

$$d=x_l-x_r \quad (3)$$

where x_l,x_r are a pair of matched points in left and right image respectively. Hence, the depth of a 3D point in space can be written as follows

$$z=f\frac{B}{d} \quad (4)$$

where *B* is baseline or distance between stereo cameras, and f is the focal length.

Having defined the disparity related to a depth of a 3D point in space, we can further process this disparity map to extract the objects. Since our framework targets humanoid robotic domestic applications, we may assume that objects rest on a flat surface such as a kitchen table, floor, etc. This problem is named the ground plane removal problem, which can be frequently found in the automotive industry.

Our method is based on Franke's [Franke, 1996] assuming flat surfaces and parallel axes cameras, which are suitable for the Point Grey cameras used. The coordinate of the *x* and *y*-axis above follows image coordinate as described in [Franke, 1996]. For more detailed mathematical derivation, one can refer to Labayrade's work in [Labayrade, 2002]. Hence, the disparity of the flat surface at a certain coordinate in an image viewed by a tilted camera is described as follows

$$d(x,y) = \frac{B}{H} f_x \left(\frac{y}{f_y} \cos\alpha + \sin\alpha \right) \qquad (5)$$

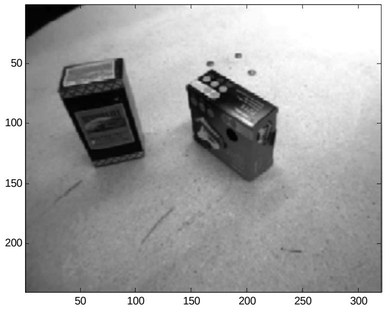
where *H* is height of the camera above the table; α is tilt angle; $d(x,y)$ is the disparity at a position (x,y). Equation (1) can be rewritten in term of P_1 and P_2 as

$$d(x,y) = P_1 y + P_2 \qquad (6)$$

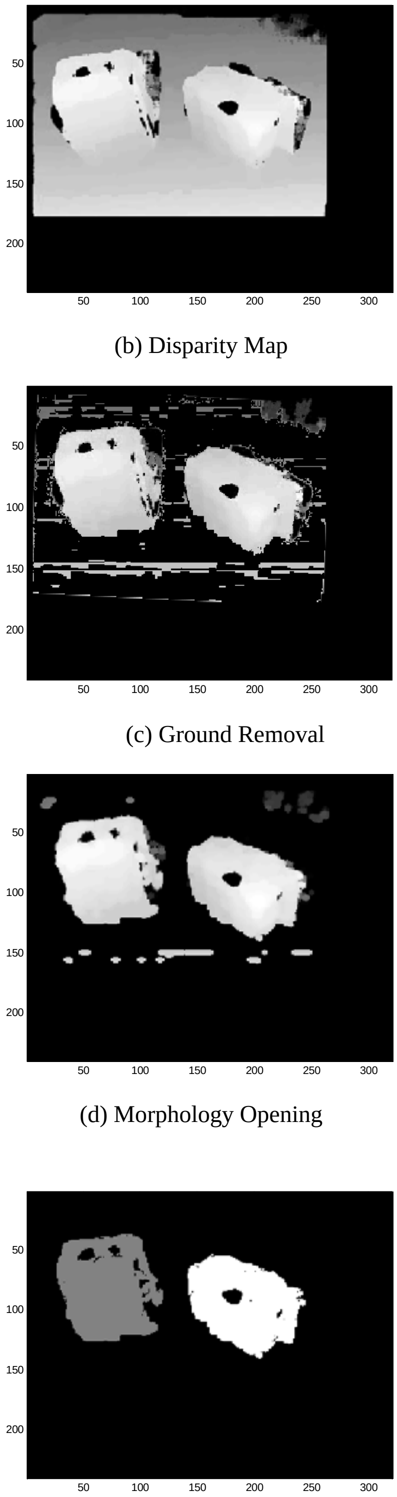
where $P_1 = \frac{B}{H}\frac{f_x}{f_y}\cos\alpha,$ and $P_2 = \frac{B}{H} f_x \sin\alpha$

Each pixel in an image, whose disparity satisfies equation (5) or (6) within a certain threshold, is categorized as ground plane, thereby needs to be removed. Whereas, those pixels that do not fulfill the condition, are classified as objects. This can be seen in figure 3a whose disparity map is shown in figure 3b.

In our Triclops camera, there are thee individual cameras. Due to internal characteristics, each camera generates slightly different image from the others (even for zero disparity). In addition to that the ambient illumination condition varies, hence the disparity map produced is affected. Hence, prior to ground removal to in order to reduce noise power in this disparity image, the disparity map is averaged over a specified *M* frames.



(a) Original image



(b) Disparity Map

(c) Ground Removal

(d) Morphology Opening

(e) Region Growing.
Figure 3: Stereo Segmentation.

Unfortunately, due to imperfect estimation of the tilt angle, the camera height from the flat surface, the roll angle that is ignored, different illumination between cameras and noise; there are still few small regions left after ground plane removal. Hence, a conventional morphology opening filter is engaged to clean up noisy small regions as depicted in figure 3c. This issue can be also removed in the subsequent step by eliminating areas that are smaller than a certain threshold. However, this process is necessary in order to speed up the region growing process timing.

The morphology opening implemented is taken from openCV library [Intel, 2008] with an ellipse shape as a kernel of axis 3 and 5 as shown in figure 3d. This shape is chosen because it best suits the noise left from flat surface, which is elongated in horizontal direction and not sharp at the same time.

After an almost clean image is obtained, it still needs to be processed further to separate each individual object blob. Our region growing uses 8-neighborhood connectivity. A pixel whose disparity differs from that of its neighbor pixel by less than a certain threshold, t_d , belongs to the same region of its neighbor pixel. In our framework, a threshold value of 2 is used. At this step, another threshold is also applied to eliminate those regions that are smaller than a certain size. Eventually, the regions left are identified as objects as shown in figure 3e.

Objects that occlude one another will have their disparity map, touching one another also. However, if their locations in 3D space are far enough separated, their disparity values are easily separated. Nevertheless, this also depends on the object texture, and the accuracy of the stereo matching.

4.3 Object Recognition

The ultimate objective from the previous steps is to recognize and localize objects of interest before manipulation. Since we target a specific object, the SIFT algorithm [David GL, 2004] can be used to match the image with the database. SIFT can identify an object of interest, although the outcome of segmentation from the disparity map is not clean. This is because it recognizes specific, local features in the object and not the whole area as in cross correlation pattern matching.

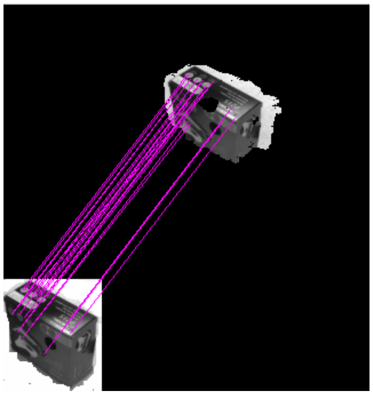


Figure 4: Stereo Segmentation SIFT

The SIFT algorithm is applied after the stereo segmentation has been completed. This reduces the matching time in the SIFT operation and at the same time the object volume can be used as a searching criteria. In order to search for a desired object, we use several training images of the same object from different angles. In this experiment, six of training images are stored in the database for each object.

4.4 3D Point Cloud Analysis

Finally, if an object searched for is successfully found, its pose and grasp point have to be estimated. Hence, that object's disparity has to be converted to a 3D point cloud using equation (1) and (2). The result is shown in figure 5.

Those 3D point clouds are used to estimate their enclosing box occupancy, namely: the object height, the object width, and the object length. A 3D object has its position and orientation in space as clearly shown in figure 5 that the first box has different orientation from the other box's. Its orientation can be calculated from the 2nd order statistics method [Castleman, 1996].

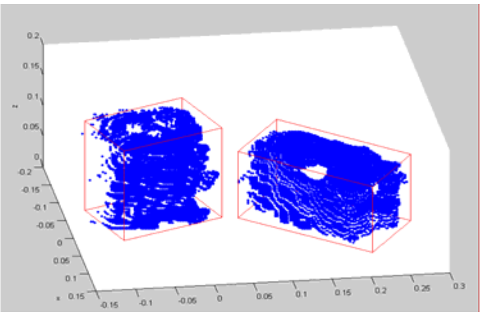


Figure 5: Box Reconstruction

The orientation angle (θ) of a region is defined as

$$\tan(2\theta) = \frac{2\mu_{11}}{\mu_{20} - \mu_{02}} \qquad (7)$$

where μ_{jk} is called as central moments. Its equation is given as

$$\mu_{jk} = \iint (x - \bar{x})^j (y - \bar{y})^k f(x, y)\, dxdy \qquad (8)$$

where $\bar{x}, \bar{y}$ are center of gravity in x and y axis, $f(x, y) = 1$ is a weight factor that is implemented.

The center of gravity in *x* and *y* axis are defined as

$$\bar{x} = \frac{M_{10}}{M_{00}} \qquad (9)$$

$$\bar{y} = \frac{M_{01}}{M_{00}} \qquad (10)$$

where M_{jk} is moments given as

$$M_{jk} = \iint x^j y^k f(x, y)\, dxdy \qquad (11)$$

The object orientation angle is not only important for 3D reconstruction but also for manipulation of the gripper of robot RT 100. For stability purposes, the gripper has to rotate in order to align its angle with the smaller principle axis of the object. In other words, the gripper has to open as wide as the shorter object principle axis.

4.5 Grasp Point

Since the 3D object modeling used is a cube, the most stable point would be the cube centroid. Hence, once the

3D object modeling has been successfully reconstructed, the grasp point can be computed as the centroid of that object. The *x* and *y* coordinate can be computed according to equation (9) and (10), while its *z* coordinate is the midpoint of the object height.

For simplicity, grasp planning from above is carried out in these experiments. However, certain precautions have to be taken care of if the object is tall or the centroid distance from above is longer than the gripper length as shown in figure 6, where the grasp point should be shifted vertically higher to avoid collision of the robot gripper and the object itself. Therefore, the grasp point should be re-adjusted if the centroid is beyond the reach of the gripper (i.e. deeper than 6 cm). Hence, the new grasp point is written as follows

$$z' = \begin{cases} z & \text{if } (h - z) < 6 \\ h - 6 & \text{otherwise} \end{cases} \tag{12}$$

where z is the height of the object centroid from the surface, h is the object height, and z' is the new grasp point.

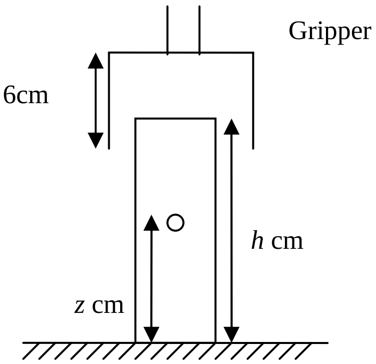


Figure 6: Collision Prevention

4.5 Error Analysis

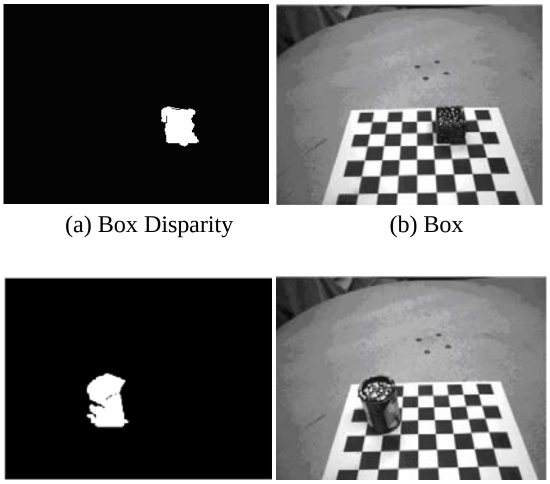
The grasping location is determined from the 3D point cloud process. That process relies heavily on the stereo matching. Hence, overall, our method's performance depends on the accuracy of the stereo matching algorithm, the objects' textures, surrounding ground planes, etc.

For the purpose of error analysis of our method to estimate the object location, two objects with known dimensions are used. The first object is a cube of edge length 63 mm and the second one is a cylinder of radius about 40 mm and of height 120 mm. The advantage of using these types of object is their symmetry. Hence, their centroid locations on the checkerboard of size 40 mm are easily estimated as shown in figure 7.

These two objects are placed at several locations on the checkerboard and then their centroids are estimated. These are then compared with the ground truth by observing their location on the checkerboard.

The coordinates of the object centroid are represented in 3D world Cartesian coordinate. Hence, there will be three error analysis components namely, the *x*-axis, *y*-axis and *z*-axis. The error analysis is performed on 21 discrete locations on the checkerboard, (0,0), (0,40), … (0,240), …, (40,0),…, (80, 240) mm.

The *z*-axis error is not that critical compared to the *x*-axis and the *y*-axis error. This is because the object's centroid height is around the robot's gripper length as depicted in figure 6. Hence, 1 cm ~ 2 cm error does not cause the failure of the gripping manner. Also, the *z*-axis error is small within ± 5 mm as shown in figure 8. Hence, in our method, the object centroid in *z*-axis is not compensated for.



(a) Box Disparity (b) Box

(c) Cylinder Disparity (d) Box

Figure 7: Error Analysis Method

However, the *x*-axis and *y*-axis errors really determine the successfulness of the gripping, as the gripper will hit the object's surface. The *y*-axis error of the cylinder and the cube are shown in figure 10 and 12 respectively. Since, there are only 21 numbers of discrete data available on each error analysis, it is sufficient that we then model the error in *y*-axis direction as below

$$\hat{y} = y + 5 \tag{13}$$

where $\hat{y}$ and y are the estimated and the true value of *y*-axis coordinate respectively. The *x*-axis error of the cylinder and the cube are shown on figure 9 and 11 respectively are rather complicated. It seems that the error increases along the *x*-axis error and *y*-axis. The *x*-axis error is given as follows

$$\hat{x} = x + e \tag{14}$$

where $\hat{x}$ and x are the estimated and the true value of *x*-axis coordinate respectively. And, its error is approximated as

$$e = \frac{r}{8} \tag{15}$$

where $r = \sqrt{x^2 + y^2}$. Substitute equation (15) into (14) then solve for *x*, thus equation (14) can be approximated as

$$x \approx \sqrt{\frac{64\hat{x}^2 - y^2}{81}} \tag{16}$$

In order to reduce the error, in our code the object centroid which is the grasp point in x-direction and y-direction equation (9) and (10) are offset with the equations (16) and (13) respectively. The result is that the estimation error is less as plotted in stem mode in figure 9 - 12.

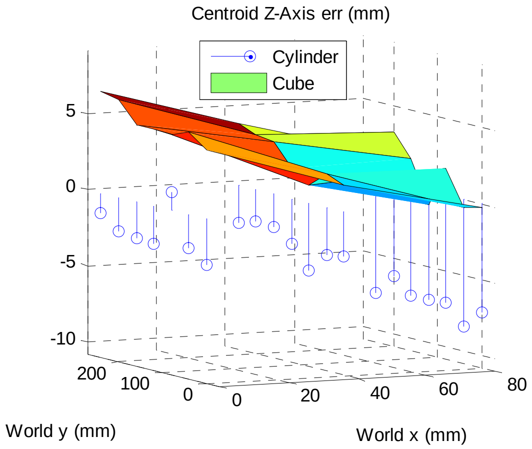


Figure 8: The *z*-axis error for the cube and the cylinder

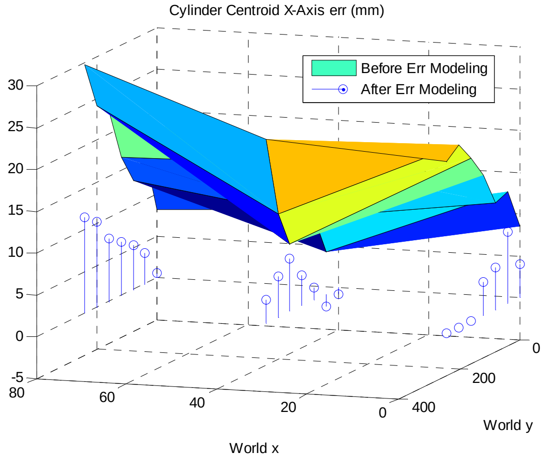


Figure 9: The cylinder *x*-axis error before and after compensation

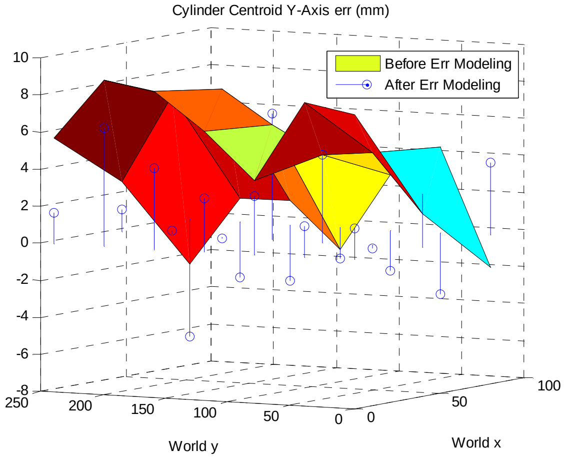


Figure 10: The cylinder *y*-axis error before and after compensation

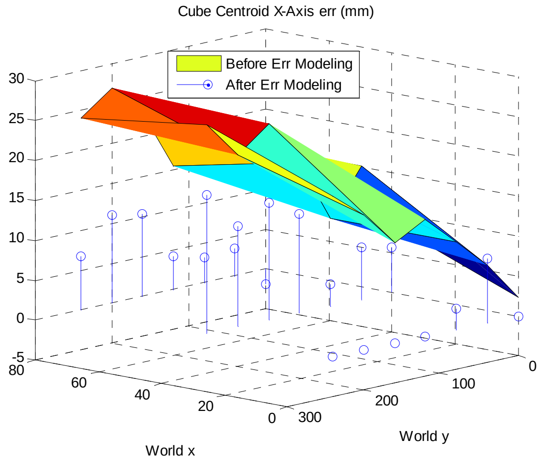


Figure 11: The cube *x*-axis error before and after compensation

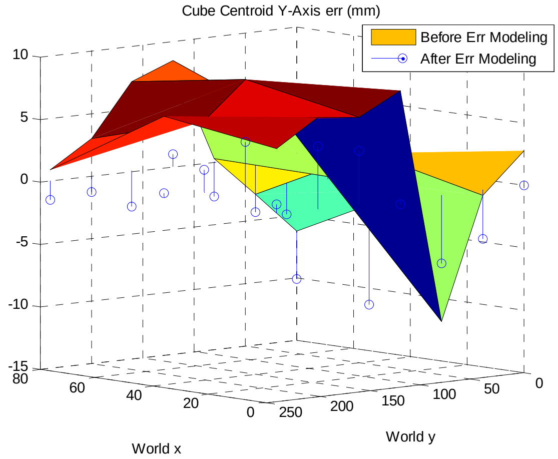


Figure 12: The cube *y*-axis error before and after compensation

5 Experiments

Figure 13 shows a sequence of frames recording the robot picking up an object from above. Since our method uses pure vision, the performance depends on the capability of the camera, the correctness of the stereo matching to produce disparity map, the 3D reconstruction algorithm and slight difference of intrinsic conditions from each individual camera. The table texture also plays important role in our approach. The success of ground plane removal will determine the successfulness of objects extraction and thereby the grasp location. Lastly, the objects should have enough texture for stereo algorithm to work successfully.

During our experiments, there were some cases where the robot fails to pick up the object due to error in the disparity map. So, the gripper may collide with the object itself when it comes down to pick it up. Hence, in our future work we will implement closed loop visual servoing to fine tune the estimated pose of the object relative to the gripper. Consequently, the robot should be able to distinguish its own arm from other objects and simultaneously update its arm path planning.

6 Conclusion

In this paper, we have demonstrated that a 2.5D disparity map instead of a 3D object reconstruction with a cube modeling for simple objects is sufficient to provide information for the grasping manipulation.

Full 3D object reconstruction as developed in [Yamazaki K, *et al.*, 2008] takes longer time and more-over the invisible object shape at the other side of the camera's view is sufficiently estimated as a cube. Therefore, 3D object reconstruction for grasping purposes is rather redundant.

7 Future Works

Occlusion certainly will contribute to the quality of the disparity map, as certain object's area is visible in one camera but invisible in the others. Since we focus on the grasp manipulation tasks, the camera working area is relatively close which is within one to two meters away.

The SIFT algorithm performs well in recognizing a specific object. However, a general object recognition is of greater importance for assistive technology.

We will try to explore how to manipulate objects with a humanoid setup using vision as it has considerable importance in many emerging applications; for instance rescuer robot in the field, domestic robots in human environments, etc.

Lastly, a closed loop visual servoing system and horizontal grasp will be implemented in our future work to improve the robustness of the grasp planning task for human centric robots.

References

[Ashutosh S, Justin D and Andrew YN, 2008] Robotic Grasping of Novel Objects using Vision. *International Journal of Robotics Research* 27:157-173, 2008.

[Bennet and DeJong, 1996] Bennett S.W.1; DeJong G.F. Real-World Robotics: Learning to Plan for Robust Execution. *Machine Learning* archive Volume 23 , Issue 2-3 (May/June 1996).

[Castleman, 1996] Kenneth R. Castleman. *Digital Image Processing*. Prentice Hall, 1996.

[David GL, 2004] Distinctive Image Features from Scale-Invariant Keypoints. 60:91-110, 2004.

[Franke and Kutzbach, 1996] U.Franke, 1.Kutzbach, "Fast Stereo based Object Detection for Stop&Go," IEEE Intelligent Vehicles '96, Tokyo, 19.120.Sept.1996, *S.* 339-344.

[Harada K, Kaneko K and Kanehiro F, 2008] Fast grasp planning for hand/arm systems based on convex model. *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*2008, pp. 1162-1168.

[Intel, 2008] http://opencvlibrary.sourceforge.net/

[Jafari and Jarvis, 2005] Jafari, S. and Jarvis, R. Robotic hand-eye coordination: From observation to manipulation. *In Proceedings - HIS'04: 4th International Conference on Hybrid Intelligent Systems*, 2005, pp. 20-25.

[Katz D and Brock O, 2008] Manipulating articulated objects with interactive perception. *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on* 2008, pp. 272-277.

[Kemp et al., 2007] Kemp, Charles C., Edsinger A, Jara E.D. Challenges for Robot Maniuplation in Human Environments. *IEEE Robotics & Automation magazine*, 2007.

[Knight, 1999] UMI Robot User and Programmer's Manual, for the UMI RT100+ Win32 Library, www.eng.uts.edu.au/~carlo/pdf/Hitsquad_Robot_Manual.pdf.

[Kragic, 2001] D. Kragic. Visual Servoing for Manipulation: Robustness and Integration Issues," doctoral dissertation, Dept. Numerical Analysis and Computer Science, Royal Institute of Technology, KTH, Stockholm, 2001.

[Kyrki and Kragic, 2005] Kyrki, V., Kragic, D. Integration of model-based and model-free cues for visual object tracking in 3D. *In Int Conf on Robotics and Automation, ICRA'05*.

[Labayrade et al., 2002] R. Labayrade, D. Aubert, and J.-P. Tarel. Real time obstacle detection on non flat road geometry through "v-disparity" representation. *In IEEE Intelligent Vehicle Symposium (IV'2002)*, June 2002.

[PGR, 2008] http://www.ptgrey.com.

[Sanz et al., 2005] Sanz, P.J. Requena, A. Inesta, J.M. Del Pobil, A.P. Grasping the Not-So-Obvious Grasping the Not-So-Obvious. *IEEE Robotics & Automoation Magazine*, 2005.

[Schlemmer et al., 2007] Schlemmer, M.J., Biegelbauer, G., Vincze, M. Rethinking robot vision - Combining shape and appearance. *International Journal of Advanced Robotic Systems* 4 (3), pp. 259-270, 2007.

[Sumi et al., 1997] Y. Sumi, Y. Kawai, F. Tomita. Three-Dimensional Object Recognition Using Stereo Vision. *Systems and Computers in Japan*, Vol. 28, No. 13, 1997.

[Taylor and Kleeman, 2006] Geoffrey Taylor and Lindsay Kleeman. Visual Perception and Robotic Manipulation: 3D Object Recognition, Tracking and Hand-Eye Coordination. *Springer*, 2006.

[Trucco and Verri, 1998] Emanuele Trucco, Alessandro Verri, "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998

[Yamazaki K, Tomono M, Tsubouchi T et al., 2004] 3-D object modeling by a camera equipped on a mobile robot. *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*2004, pp. 1399-1405 Vol.2

[Yamazaki K, Tomono M, Tsubouchi T et al., 2006] A grasp planning for picking up an unknown object for a mobile manipulator. *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*2006, pp. 2143-2149

[Yamazaki K, Tomono M and Tsubouchi T, 2008] *Picking up an Unknown Object through Autonomous Modeling and Grasp Planning by a Mobile Manipulator* Vol. 42. Springer Berlin / Heidelberg, 2008.

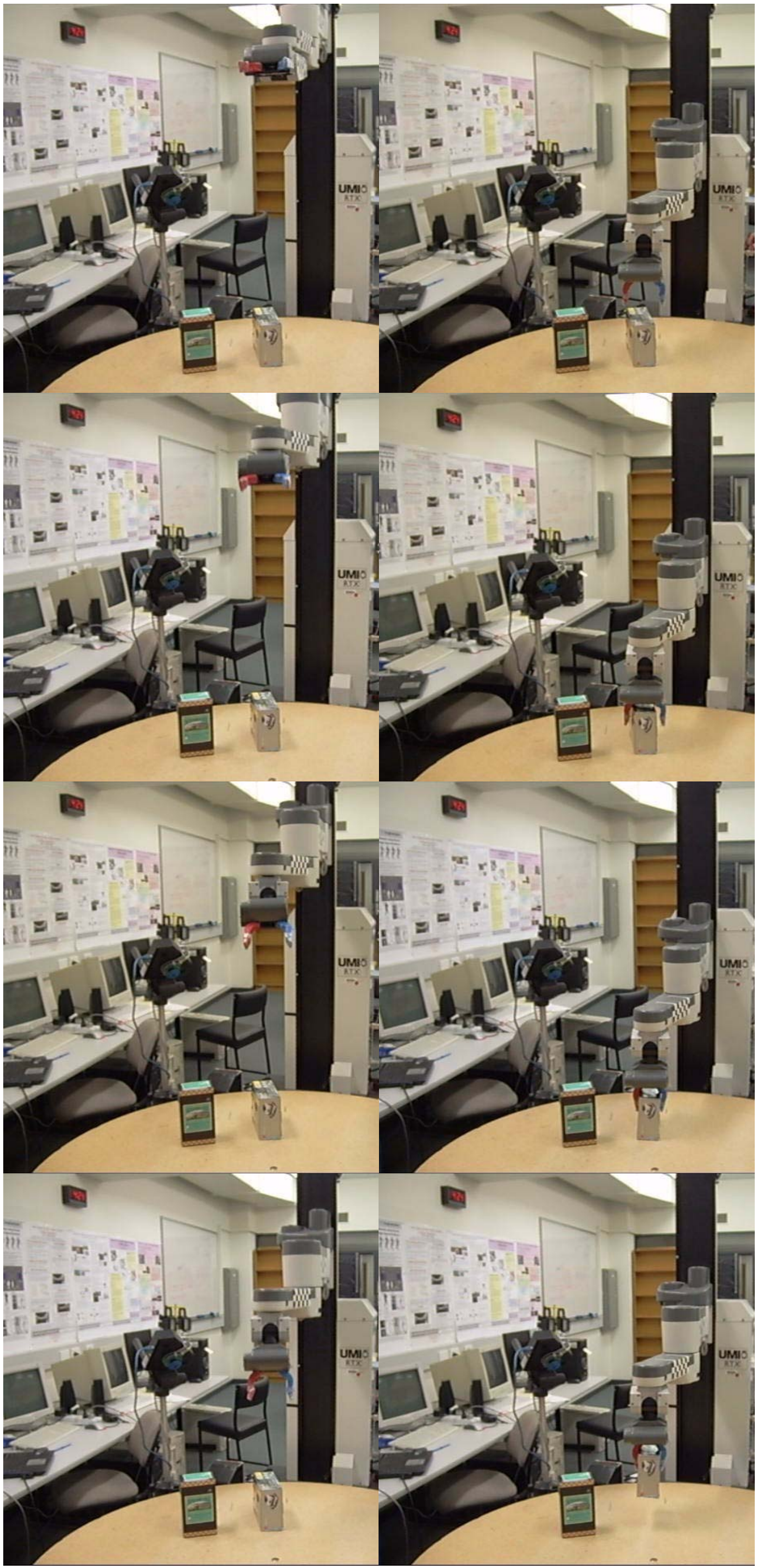


Figure 13: Robot Manipulation sequences captured in different time, from top to bottom and from left to right.