GRASPING UNKNOWN NOVEL OBJECTS FROM SINGLE VIEW USING OCTANT ANALYSIS

by

AARON A CHLEBORAD

B.S., Missouri Western State University, 2001

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computing and Information Sciences
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2010

Approved by:

Major Professor
David Gustafson

Abstract

Octant analysis, when combined with properties of the multivariate central limit theorem and multivariate normal distribution, allows finding a reasonable grasping point on an unknown novel object possible. This thesis's original contribution is the ability to find progressively improving grasp points in a poor and/or sparse point cloud. It is shown how octant analysis was implemented using common consumer grade electronics to demonstrate the applicability to home and office robotics. Tests were carried out on three novel objects in multiple poses to determine the algorithm's consistency and effectiveness at finding a grasp point on those objects. Results from the experiments bolster the idea that the application of octant analysis to the grasping point problem seems promising and deserving of further investigation. Other applications of the technique are also briefly considered.

Table of Contents

List of Figures

List of Tables

Acknowledgements

First and foremost I'd like to thank God for the gift of life and the abundance of graces given to me without which this paper wouldn't have been possible. Secondly, I owe an infinite amount of gratitude to my lovely wife Devree for her amazing patience, understanding and support for the last two and a half years. I'm very thankful to my major professor Dr. David Gustafson for challenging me to take on this topic and his support and guidance over the development of this thesis. I'm also grateful to Dr. Scott DeLoach for giving me the opportunity to work with the HuRT group during the summer of 2008 as it furthered my understanding of robotics (and cockroaches) in many ways. I further thank Dr. David Gustafson, Dr. Scott DeLoach and Dr. Mitchell Neilsen for agreeing to be on my committee. I give a nod to both Tim Weninger and Andrew King for their encouragement to take the thesis challenge and another nod to Andrew for the numerous engaging robotics discussions. Lastly, I thank my family and friends for their support and patience over the course of this growth filled journey.

CHAPTER 1 - Introduction

When the majority of people envision robotics, they see a robot that is an anthropomorphic being, highly autonomous and somewhat capable of intelligent interaction with humans. The reality of today's consumer robotics, however, is that of inefficient vacuum cleaners, overpriced dinosaurs and brainless alarm clocks. The consumer robotics industry is still in its infancy as of 2010 due to the complexity of problems that must be solved in order to provide the public with practical, cost-effective robots capable of performing meaningful tasks around the home and beyond. One of the most essential tasks that robots must perform is the successful grasping of objects. Further still, robots must be able to grasp objects that they have never yet seen in order to be truly useful in our ever-changing world.

Robotic grasping has been used in manufacturing since 1961 with the introduction of the Unimate [6] robot and has largely been utilized for very monotonous and repeatable tasks such as welding and material handling. Robots in this context perform much more consistently and faster than their human counterparts. The downside to these systems is their inability to adapt to changing requirements quickly. In recent years, strides have been taken to make industrial robots configurable to a varying array of tasks. Expensive, cutting-edge robotics systems allow technicians to 'teach' robots new skills and have increased their return on investment. Still, they are largely static machines with strict environmental conditions requirements and need regular maintenance.

Outside the industrial context, groups in the academic and open source communities are progressively bringing the grasping capability of robots closer to reality. The following examples of this exciting research cover a span of objectives. Wheelchairs equipped with robotic arms are being researched at South Florida University to assist severely disabled people perform basic object manipulation functions [7], Willow Garage's open source PR2 robot can intelligently navigate throughout an office and plug itself into the socket in numerous rooms [8] and the STAIR robot from Stanford is able to grasp objects out of a dishwasher with a high degree of accuracy [9][28]. The previous two robots are especially interesting since they aim to work in a changing environment autonomously. While their progress is impressive, they are still 'research' robots and come with hefty price tags. The STAIR Katana 6m arm kit alone costs

about $25,000 and the undisclosed price of the PR2 arm with similar functionality is probably in the same ballpark.

In the consumer market, grasping capable robots currently seem to be geared toward developmental robot platforms that can be extended using software APIs to perform the required functions. Roboware's 3E, Speecys's NNR-1 and Mecanno's Spykee, shown in Figure 1-1, are three platforms that promise open APIs and hardware platforms capable of performing lightweight grasping among other capabilities. Still yet, other platforms such as the iRobot Create® are being engineered to perform grasping with a high degree of success [10]. These platforms currently cost anywhere from $200 to $3000 which begins to put these robots within reach for enthusiasts. If these robots are to be put to work in a useful way, grasping algorithms have to be developed.

Octant analysis can be used to find reliable a grasping point on a novel object by taking advantage of statistical properties associated with the multivariate central limit theorem and multivariate normal distribution. The algorithm developed was developed in a closed loop system using stereo vision. No prior object knowledge was assumed. Objects with non-elementary shapes (like balls or blocks) were targeted in a non-cluttered scene for simple grasps.

This thesis is organized into six chapters. Chapter two introduces the background of robotic grasping. Chapter three will describe an algorithm for grasping novel objects using data from a single viewpoint with the Lynx 6 robotic arm. Experiments using the algorithm are conducted in chapter four and results are outlined in chapter five. Conclusions and future work are discussed in chapter six.

Figure 1-1 Roboware E3, Mecanno Spykee and Speecys NNR-1 robots

CHAPTER 2 - Background

A survey of robotic grasping in the research literature yields a myriad of differing approaches to solving the problem. Robotic grasping has a number of issues that must be addressed in order to engineer a base platform in which to develop an overall solution. These issues include.

- Is the system open or closed?
- How is data obtained about the object to be grasped?
- Is prior object knowledge known or not?
- What type of objects will be grasped?
- Will the scene be cluttered?
- Is the grasp simple or complex?
- If cameras are used, how are they positioned?

Multiple answers for each of these questions have been proposed. The combinations of varying approaches to all these questions lead to a wide array of grasping algorithms. The following will be a brief overview of techniques used to answer each question above.

2.1 Open and Closed Systems

Control theory is an interdisciplinary branch of engineering and mathematics, which deals with the behavior of dynamical systems [11]. In the context of robotic grasping, control theory plays a large role in how one approaches the task of picking up an object. During the process of the gripper approaching and picking up the object, the gripper itself becomes part of the environment which can either be used to further guide the system in its decision processes or not. A system that is able to use itself as an aid to solving a problem is said to contain a feedback loop. To be more formal, a feedback loop is "The section of a control system that allows for feedback and self-correction and that adjusts its operation according to differences between the actual output and the desired output [12]."

The term closed loop describes a system containing a feedback loop and conversely an open loop system is one in which the feedback loop is absent. In the literature, open loop grasping systems are synonymous with 'blind' grasping. Blind grasping systems gather and process all information about a scene before making any changes to the environment, including

its own movements. Once calculations have been made, the gripper simply makes a blind grasp at the object without performing any further error correcting adjustments. Naturally, the sensor system and gripper must be calibrated very accurately for this approach to be successful. Closed loop systems benefit from their ability to adjust against miscalculations and therefore tend to be more robust. However, closed loop systems require more time calculating during the grasping process, which could preclude its use in real-time systems. The method chosen for a particular implementation is based on a number of factors that range from real-time constraints to hardware capabilities.

2.2 Scene data gathering techniques

Many sensors have been created over the years to obtain scene data pertinent to the grasping problem. Sensors are generally broken down into two categories: passive and active. Passive sensors can be thought of as observers in that they make no effort to disturb the environment and only provide data that they are able to sense. Common passive sensors in robotic grasping are those that 'listen', 'watch' or 'feel' and are known as passive sonar, vision and touch sensors[1]. Active sensors take the alternative approach in that they emit some form of energy into the environment and then use the energy's echo to calculate distances. Common active sensors are sonar and laser sensors with each type of sensor having its benefits and drawbacks. Passive sensors usually have lower associated costs while active sensors usually provide better quality distance measurements. In addition to the sensors themselves, a variety of techniques are used to gather scene data from those sensors. The following sections will describe these techniques in further detail.

2.2.1 Lasers

Lasers are often employed to gather high quality 3D data of a scene. Typically, a laser sensor is placed on a type of pan-tilt device and the laser is used to scan the workspace. The resulting data is called range data, also known as a point cloud, from which further processing can be used to remove non-essential data[2] and either create a model of the object of interest or

[1] Touch in this instance does not imply object manipulation. These sensors are generally acted upon by the environment and use resistance or capacitance changes to 'detect' the touch or pressure applied to the sensor.

[2] The background and/or other objects

the cloud itself can be used to make decisions in the grasping procedure. The accuracy of laser range scanners can be very accurate as long as the objects being scanned aren't highly reflective or highly absorptive[3]. Similar to sonar devices in that they measure distance by time of flight calculations (using light instead of sound), specular and diffuse reflections can alter the return path of the emitted laser light such that it cannot be detected by the sensor. This results in a loss of fidelity in the environment readings. Consideration must also be given to the cases where objects are located farther away from the laser causing the granularity of the data to decrease due the laser's angular resolution limitations. Furthermore, laser range readings also lack color[4], which may be considered a negative aspect depending on the application. Despite these drawbacks as well as generally high costs, lasers enjoy popularity in research and commercial contexts due to their high quality data. Examples of 3D range data created from laser scans can be found in numerous papers [13] [14] [15] to list but a few.

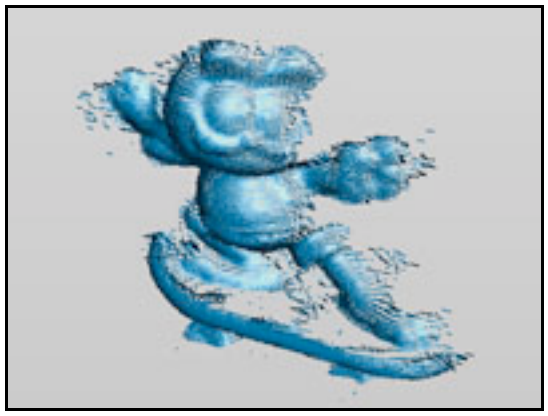

Figure 2-1 3D Laser Scan Example [1]

2.2.2 Structured Lighting

Structured lighting is based on active triangulation. Active triangulation is based on the law of sines which is an equation[5] relating the lengths of the sides of an arbitrary triangle to the sines of its angle. The process works by having an emitter project a form of structured lighting onto the scene while a camera at a different position captures images. The term "structured

[3] This of course depends on the type of laser being used and that the scene is stationary.

[4] Without the aid of a camera to assist in this regard

[5] $\frac{a}{\sin A} = \frac{b}{\sin B} = \frac{c}{\sin C}$, where a, b and c are sides of the triangle and A,B,C are the opposite angles

lighting" can apply to visible lasers, light patterns from a projector or invisible light [16]. Using the relative positions of the emitter and camera, triangulation is used to determine depth. The core principle at work is that when a known pattern of light is projected onto an object's surface, the pattern becomes distorted in other perspectives (the camera(s)) by the object's geometry.

To date, many forms of structured lighting have been studied. Binary light stripes were some of the first techniques to be used, followed by grayscale, colored stripes, grid patterns, and random patterns successively. A survey of recent techniques can be found in [17]. Figure 2-2 provides a view of the geometry and Figure 2-3 demonstrates the light stripe technique discussed earlier.

$$[x\ y\ z] = \frac{b}{f\cot\theta - u}[u\ v\ f]$$

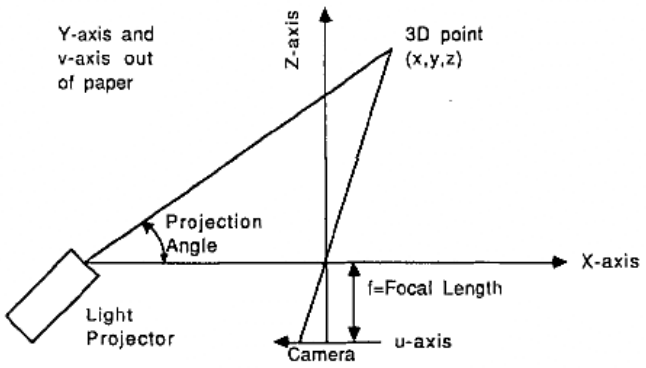

Figure 2-2 Active Triangulation [2]

Figure 2-3 Light stripes [3]

2.2.3 Optical Flow

The goal of optical flow is to determine the velocity of each pixel in an image given a pair of images taken at differing times. These velocities can help us determine an approximate depth of a scene since 3D objects, when projected orthographically onto a 2D plane (an image), appear to 'move' faster in the foreground than in the background. The process of determining the change from two consecutive images is known formally as differential optical flow. Two procedures that must be completed in this process [18]:

1. Measure the spatio-temporal intensity derivatives (which is equivalent to measuring the velocities normal to the local intensity structures) and
2. Integrate normal velocities into full velocities, for example, either locally via a least squares calculation or globally via a regularization.

In the first step, spatio-temporal intensity derivatives are changes in the pixel color values at consecutive points in time. The second step is a consequence of the aperture problem, which results in motion being detected only perpendicular to the orientation of the contour that is moving. An excellent example of the phenomenon can be found in [19]. Essentially, at the local image region level, it may be impossible to find the true direction of motion. Other local regions' data may need to be integrated together to determine the true velocity.

Since determining optical flow is based largely on image intensity values, irregular changes in intensities are likely to result in calculation errors unless these are modeled in the system. To make calculating optical flow more accurate, three constraints are ideal:

1. Minimal occlusions
2. Minimal specular reflections
3. Objects should have rigid bodies

Even with these limitations, determining accurate depth for various scenes can still be quite challenging. Scenes with large regions of constant intensity values or those with multiple light sources are prime difficult examples. Even so, the optical flow technique has been used in many applications ranging from robot navigation to determining impact times for missiles. Figure 2-4 shows an example of optical flow being used to determine the depth of a scene. Two consecutive images on a moving car (driving forward) were used to determine the depth map in Figure 2-5. Empty regions in Figure 2-5 were due to unreliable points being removed.

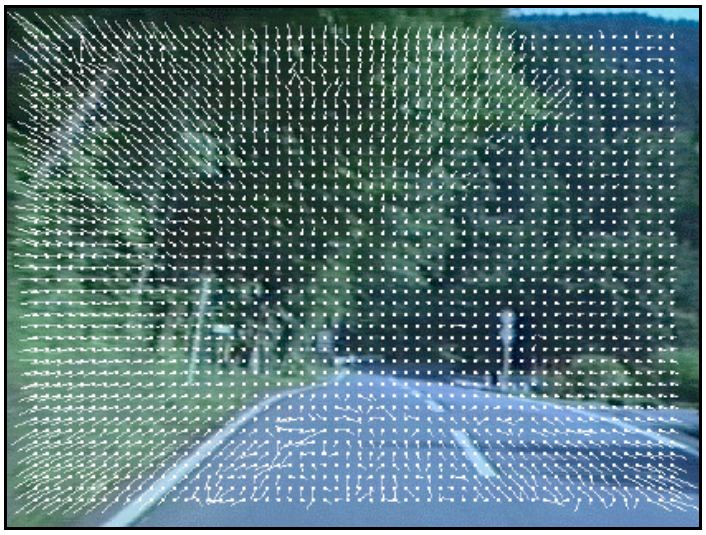

Figure 2-4 Optical Flow Example [4]

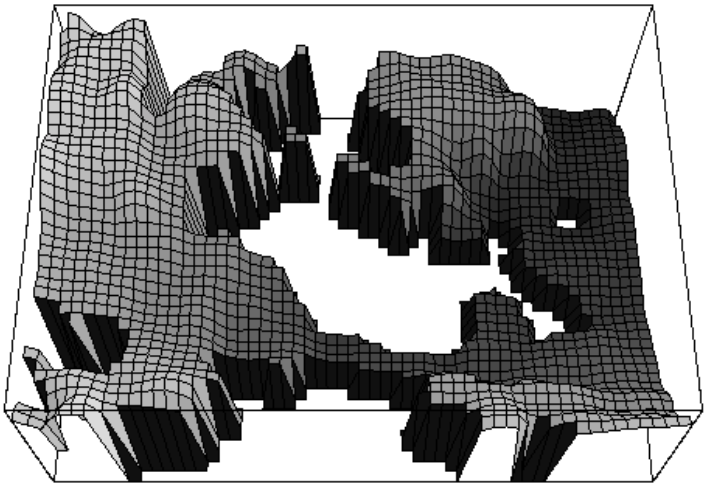

Figure 2-5 Reconstructed Depth from Optical Flow in Figure 2-4 [4]

2.2.4 Stereo Vision

Stereopsis is the process by which humans are able to perceive depth using two slightly different projections of the same scene (one from each eye). Stereo vision is built upon stereopsis but using cameras instead of eyes to collect projections of the world and calculate depth based on the disparities in the two images. In order to obtain depth from two images a number of topics must be addressed.

Before beginning on the actual depth calculations, it is necessary to first understand how digital cameras function. Light passes through a lens that focuses the light onto an array of photosensitive cells. A property called photoconductivity causes the photosensor array's

conductance to vary with changes in light intensity and the signals read by the host computer reflect those intensity values. A simple way of modeling the process of capturing an image is with the pinhole camera model. The pinhole camera model works like illustrated in Figure 2-6: Imagine an opaque surface s1 with a single hole in the middle of the page punched open to allow light to pass through. The light passes through the hole in s1 and is cast onto surface s2. The image passing through s1 is upside down and generally, much smaller when projected onto s2. The distance between s1 and s2 is called the focal length and the intersection of the optical axis with s2 is called the principal point.

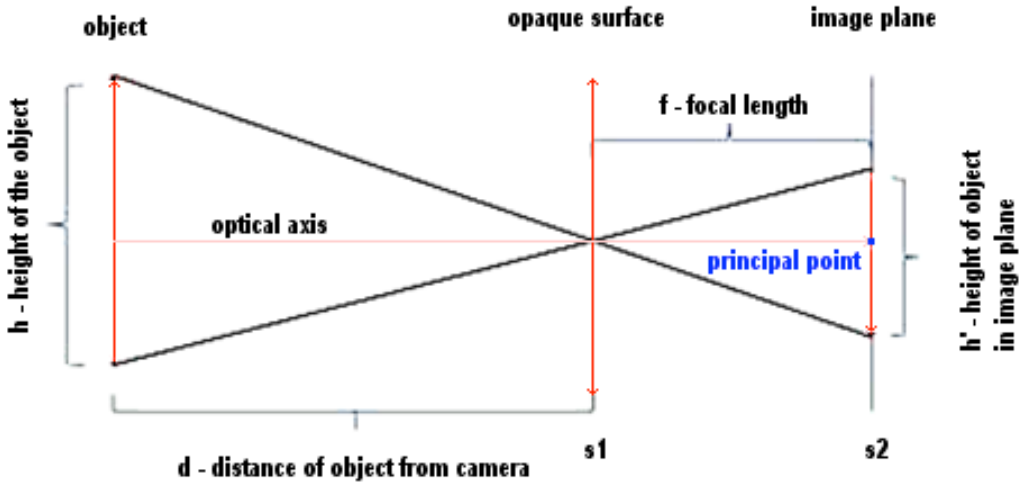

Figure 2-6 Pinhole Camera Model

A pinhole image quality is poor due to the small amount of light entering through the hole. To improve the quality of and speed of obtaining an image, optics are placed in front of the pinhole to focus more light onto s2. While this provides a boost in image quality, lenses introduce a phenomenon known as radial distortion in all but the highest quality cameras that generally have corrective optics. Another distortion known as tangential distortion is related to how the camera is assembled. During the assembly of a camera, it is extremely difficult to align the lens with the imaging surface. Not only does this give error to the center of the image but also the angle at which the light hits the image plane i.e. s2.

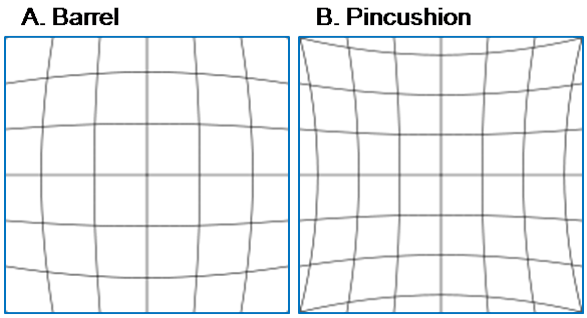

Figure 2-7 Lens Distortions

To correct for the aforementioned distortions, cameras must be calibrated to minimize the errors. Calibration is the process of relating pixels on an image array to 3D points in the real world. Two parameter sets for the camera(s) are required for the conversion calculations and they are named intrinsic and extrinsic parameters. Intrinsic camera parameters are those that describe the camera's internal measurements that include principal point, pixel scale factors, focal length and all distortion factors. Extrinsic parameters describe the position and orientation of the camera(s) in the real world, which are defined in terms of a translation and rotation matrices. Obtaining the values for all parameters in the intrinsic and extrinsic sets involves using either 3D calibration objects which are difficult to deal with or using a regular pattern such as a chessboard. In either case, the geometry of calibration object is known when the calibration procedure is performed and using its 3D measurements, all camera parameters can be calculated. Some popular calibration methods can be found in [20] and [21].

Once camera images have been calibrated, they must undergo rectification. During the rectification procedure, the images are aligned so that corresponding points in both pictures are made parallel to each other using the images' epipolar geometry. Using epipolar geometry reduces the processing time required to match points in both images by taking searches for points from the 2D search space (image) to the 1d search space (single scan line in the image). In practice, even though we can reduce the search space by an order of magnitude, the correspondence problem is still quite difficult due to the many pixel ambiguities that can occur. Large groups of similarly colored pixels or pixels for which there is no match (or more than one) in the other image are common hurdles to overcome during correspondence matching. In Figure 2-8, the epipolar constraint defines the line on which the 3D points X, X_1, X_2 and X_3 are projected on to the left and right image planes. Note that $O_L O_R X$ defines a 3D plane (epipolar plane) in which X_1, X_2 and X_3 lie. The red line in the right view, and in particular the line $X_R e_R$,

is the epipolar line in which X, X_1, X_2 and X_3 must be found[6] and is the place where the epipolar plane intersects the right image plane. X_L is the point in the left view where all Xs are projected. O_L and O_R are the centers of projection or focal points in both images and e_L and e_R are the epipoles, the image of the center of projection of the other camera. The line $e_L e_R$ denoted by B is known as the baseline of the epipolar plane. In general, epipolar geometry assures that every 3D point in view of the cameras is contained on an epipolar plane, which intersects the image planes at an epipolar line [22].

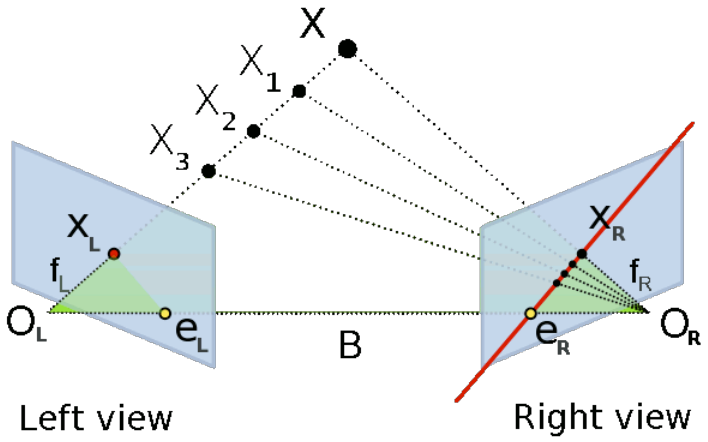

Figure 2-8 Epipolar Geometry [5]

At this point, we are nearly able to calculate the distance from the camera for each pixel. The last ingredient to calculating the depth is to measure the triangulated disparity[7] of a matched point between the two images. Depth is inversely proportional to disparity. That is to say, that objects further away have smaller disparities and closer objects have larger disparities. This makes intuitive sense if we imagine looking out a window in a moving vehicle and notice the objects closest to the vehicle appear to be moving faster than those far away. Specifically, the distance Z from O_R to X in Figure 2-6 is calculated using the formula $Z = f_R \frac{B}{d}$ where B is the baseline distance of the centers of projection of the two cameras, f_R is the right camera's focal length and d is the disparity of the pixel in the two images. By using the above calculation for every pixel in the image, we obtain a point cloud that represents the 3D space. In Figure 2-9 the

[6] since the geometry of the stereo rig is know at this point

[7] $d = x^l - x^r$ where d is disparity and x^l and x^r represent a representative pixel in both the right and left image

object being scanned can be discerned but there are some depths that are incorrect do to correspondence errors and/or image irregularities due to various forms of distortion.

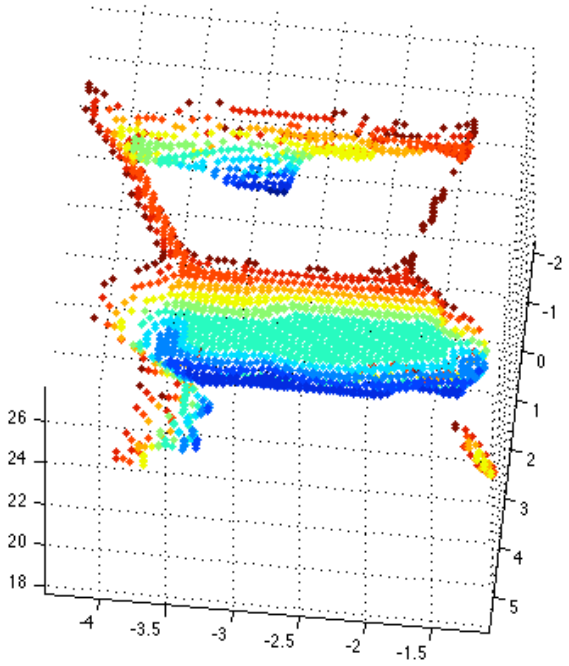

Figure 2-9 Point cloud of toy chair from stereo depth calculations.

2.3 Prior object knowledge

When searching a workspace for objects to grasp it can be beneficial to have a library of object models with which it is possible to refine the calculations for the grasping procedure. The benefits come from the fact of knowing the workspace object's dimensions, which have been stored in the library of model objects beforehand. There are a couple of drawbacks to using this approach. The time and effort required to obtain all the data for a grasping system can be daunting if not infeasible for systems requiring real world functionality. However, for systems not aspiring to completely robust grasping[8], the data gathering required can be relaxed. Another drawback is that it requires object recognition to be implemented. In spite of the drawbacks, the

[8] Grasp robustness is the capability of keeping hold of manipulated objects in spite of all possible disturbances (unexpected forces, erroneous estimates of the object characteristics, etc) while maintaining a "gentle" enough grip not to cause any damage. [34]

approach is widely used. Papers [23], [24], [25] and [26] are but a tiny fraction of approaches that have used object libraries to aid in grasping. The three main approaches used consistently in the literature can be categorized in the following groups:

1. Storing images of the object in as many poses and from as many angles as possible
2. Store geometric models of objects
3. A combination of one and two

Alternatives to using an object library do exist. Fuentes [27] uses the idea that objects can be modeled as tetrahedrons by using the actual object contact points (contact tetrahedral) and commanded[9] grasp-points (grasp tetrahedral) of the fingers on the object being grasped and linking them using a virtual spring. The authors state the grasp is well conditioned when the spring energy function, defined by the rigid displacements of the grasp tetrahedron with respect to the contact tetrahedron, is minimized. Assuming the contact points are fixed points on the object allows the object to be manipulated by performing transformations on the grasp tetrahedra. Another alternative that has generated interest lately is found in [28]. The authors used a library of synthetic images[10] of non-deformable objects, labeled with correct grasping positions, to train the system and then had the system infer grasping points on previously unseen objects using the training objects as guides. These two methods aim to grasp previously unseen objects by various approaches and have their own drawbacks. For example, the first approach suffers if the grasped object has a very unbalanced shape that causes it to slip in the grasp hand. Attempts would have to be made many times in order to find a suitable grasp. Worse yet, it is possible that there is no suitable grasp point on the object, which would render all grasping attempts moot. The AI approach used in [28] could suffer if the previously unseen object was truly a novel object for which there was no suitable analog in the training set. In this case, a best 'guess' could be attempted but with variable degrees of success.

2.4 Types of objects to be grasped

The complexity of the grasping problem varies greatly depending on the shapes of the

---

[9] These are the points given/executed by the robot hand but not necessarily actual contact points on the object.

[10] Generated from computer graphics

objects to be grasped. Using simple shapes, such as balls, negates the need to calculate grasp angles as the ball can be approached from any angle and makes the problem much easier. Objects with regular geometries can make the problem simpler by allowing the assumption of object symmetries and lessens the issue of an inability to see a part of the object from the depth sensor(s)[11] thus helping reduce the number of possible bad grasp angles. The last objects are those with irregular/deformable geometries and are considered novel objects. These are some of the most challenging to grasp since they reduce the number of assumptions we can make. They add numerous variables to define the objects' characteristics thereby requiring complex algorithms to produce satisfactory approaches, grasp angles and real-time adjustments.

2.5 Clear or cluttered scene

The environment in which robotic grasping is to take place dictates much of the complexity of the grasping procedure. Intuitively, performing a grasp of a ball in a clear and isolated work area is much easier than grasping a specific piece of clothing partially covered on a messy bedroom floor. The example above highlights some of the overarching environmental challenges. The first challenge is identifying the object of interest. In a simple environment, it is easier to filter out the 'background' from the target, as it is regular and constant. A dynamic environment affords no such assumptions making the identification task quite difficult and nearly impossible if prior object knowledge is not known. If the object is partially hidden, detecting it even with prior knowledge can be a challenge. Assuming these hurdles can be overcome, the object still must be grasped. In highly cluttered spaces, the path the arm takes to the object plays critical importance. This adds to the difficulty, as path planning is in itself a research topic. In real living spaces, random events happen all the time. If an assistant robot is working on cleaning a room and reaches for an item and an animal/human (which may have never been previously seen by the robot) runs into its path it must not only recognize the new object but rework its grasping plan, possibly around the obstruction.

[11] This assumes the sensors are set up to view from a single viewpoint.

2.6 Simple or complex grasping

The position of objects in the environment can greatly affect the difficulty in determining the best method to grasp them. A classic example is picking up a book laying flat on a table. If the book is not at the edge of the table and the gripper is not wide enough to grasp the whole book, a combination of maneuvers must be performed --- `{move book to the edge of the table, grasp edge of book protruding from table}`. This type of grasping is considered complex for the high degree of sophistication and 'knowledge' required. Complex grasping is the holy grail of robotic grasping as it enables true achievement for the robot. On a side note, speculations as to when autonomous robots, capable of complex grasping, will be a reality have put estimates sometime in the mid 2030s [29].

2.7 Camera positioning

While not all approaches rely on the use of cameras as sensors, many do. Once the decision to use cameras has been made, the next decision is to decide on where to place the cameras in the system. Camera positioning is broken down into two major categories [30]:

1. Eye-in-hand – In this configuration, the camera is placed on the robotic arm itself, typically near the end effector. A camera in this position provides a number of benefits such as mitigating the need for complex calibration or the need for other depth sensors. There are costs to be paid however. Placing the camera on the robot introduces camera motion issues relating to depth and lighting.
2. Eye-to-hand – The camera(s) sits apart from the robot arm in this configuration. The system can ignore the moving camera issues named above but has to deal with calibration issues and camera to arm coordinate transformations.

CHAPTER 3 - Grasping with Lynx 6 and webcams

Using the background information from the previous chapter as a foundation, an approach to grasping novel objects using a Lynx 6 robotic arm, standard webcams and general-purpose pc is now presented. This thesis shows that it is possible to grasp a previously unknown novel object, such as a dollhouse chair, in various poses utilizing workspace data gathered from a single viewpoint using consumer grade electronics. This chapter will proceed by first describing the hardware in which the grasping was performed followed by sections on assumptions and implementation details.

3.1 Grasping System Hardware

The Lynx 6 robotic arm (see Figure 3-1) is a hobbyist category robot (retails for about $400) with all the base functionality of higher priced robot arms. The robot has five degrees of freedom meaning it has a joint for every degree giving five total joints. Each joint is enabled by an independent servo which when combined are controlled by an SSC-32 servo microcontroller. The arm's detailed specs are laid out in Table 3-1.

No of axis	5 + Gripper
Distance between axis	4.75"
Servo motion control	Local closed loop
Height (arm parked)	5"
Height (reaching up)	17.5"
Reach (forward)	13"
Gripper opening	2"
Lift weight (arm extended)	Approx. 3 oz
Weight (without batteries)	21 oz
Range of motion per axis	180 degrees
Accuracy of motion per axis	Servo controller dependant (SSC32 .09 degrees)
Servo voltage	6 VDC

Table 3-1 Lynx 6 Arm Specifications

It must be noted that although the arm possesses the basic functionality of a robotic arm it suffers from several deficiencies when compared to high-end robots. Servos on the robot are not powerful enough to negate the effects of gravity. As a result, we are left with a nonlinear system meaning the output does not equal the sum of the inputs. In a linear system, the angles of each joint could be calculated exactly based on the inputs given to each servo leading to a precise

knowledge of where the end effector rests. In the Lynx 6 case however, the effects of gravity become greater on the joint angle calculations the further removed they are from the base of the robot. Another complication is the presence of hysteresis in the system. Commanding the robot to go to the same end position from differing start positions leads to slight variances in the final resting position. This is true even if we could negate the effects of gravity completely. The final obstacle lies in the fact that the arm has no servo feedback from which to adjust joint calculations. Without servo feedback it is impossible to completely ensure that the arm or objects in the workspace will not suffer damage if calculated servo positions aren't correct[12]. Using feedback, it is possible to adjust the joints in near real-time to prevent damage to the robot or objects around the robot.

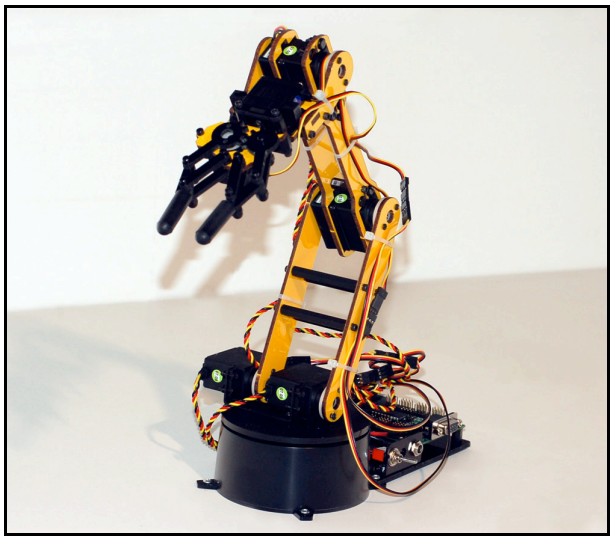

Figure 3-1 Lynx 6 robotic arm

In order to view and measure the robot's workspace two Logitech QuickCam® Pro 4000 webcams (see Figure 3-2) were used. These webcams are targeted for the home and small office market segment (about $100 each). Each camera's CCD sensor offers up to 640x480 pixel video resolution at 30 fps[13] and 1280x960 pixel resolution for still image capture through a USB 2.0 connection. The camera is manual focus capable. The drawback of a lower end camera is that it

[12] The use of other environmental sensors such as sonar, infrared or bump sensors could be employed to reduce this risk but were not in this case.

[13] Whereby meeting the appropriate computing hardware requirements

lacks high quality optics and a high precision CCD sensor leading to more distortions in the final image.

Figure 3-2 Logitech QuickCam® Pro 4000

The last component of the system is a standard PC running Windows XP Professional with an Intel Pentium 4 processor at 2.93 GHz and 2 Gigabytes of RAM.

3.2 Assumptions

Due to the complexity of the grasping problem, a number of assumptions were made in order to facilitate the testing of the hypothesis. The list below gives the assumptions and conditions under which the hypothesis was carried out along with a short description of the reasoning behind the decision.

- Controlled multi-source lighting was employed to reduce the complexity of dealing with shadows in the workspace and color variances on the objects of interest due to shading from poor, single-source and ambient lighting conditions. Fluorescent lighting from the ceiling bulbs in the research area was found to be inconsistent and too weak to provide adequate coverage of the work area and added an unnecessary complexity to the hypothesis test.
- The workspace floor and wall surfaces have been covered using matte, dark cloth in order to reduce glare from the lighting sources and simplify the process of detecting the object of interest for grasping.
- It was assumed that there were no obstructions in workspace in order to circumvent the need for advanced inverse kinematics and path planning logic in the algorithm. Grasping an object from above was assumed possible without the need for obstacle avoidance except for the object on the ground plane.

- Only one object of interest was allowed in the workspace during each trial. Since this approach did not use object recognition in the solution, having more than one object in the workspace would have added further technicalities.
- The weight of grasped objects was expected to be within the lift capabilities of the robotic arm. Since the servos on the robotic arm provide no feedback to the computer system, it makes it virtually impossible to detect when the robotic arm is under strain. Knowingly taxing the robot beyond its capabilities would not have provided any usefulness of the hypothesis test.
- Objects used during the experiment were expected to have a grasping point or be enveloping grasp capable. That is to say that the object had at least one location in which the arm was able grasp it even if it meant grasping the whole exterior of the object. The hypothesis did not attempt to address objects that were too big for the gripper's capabilities.

3.3 Implementation Details

As previously seen in chapter two, it is necessary to address seven basic questions when developing a solution for the grasping problem. Each of these questions will be answered here with further details following.

Is the system open or closed? The grasping procedure was implemented as a closed loop system. Since this system has no real-time constraints and is primarily aimed to show that we can successfully grasp novel objects with consumer electronics, the accuracy of the grasp was deemed more important than the speed at which the grasp was carried out. The vision system was used to provide feedback on the arm's location and adjust calculations based on the robot end effector's location in relation to the object of interest's grasp point.

How is data obtained about the object to be grasped? In order to gain understanding of the workspace, stereo vision was chosen as the main method of 3D data gathering. Stereo was chosen for numerous reasons. Given the weight of the cameras, it would have been impossible to mount them on the robot arm thereby leaving the eye-to-hand configuration as the only other option. The use of a laser range finder would have gone against the spirit of the hypothesis due to its being relatively expensive for a consumer product. The use of optical flow was rejected due to its need to move the camera in some fashion. A pan-tilt device is rather expensive and

cameras can't be mounted on the arm as discussed earlier. Stereo allows us to gather 3D data for a tiny fraction of the cost of a laser albeit at the expense of some complexity. The complexity is diminished somewhat by the availability of computer vision libraries which can be used to aid the gathering of required scene data. To facilitate vision data processing, the computer vision library used was OpenCV. OpenCV provides many basic and advanced image operations as well as a plethora of statistical and machine learning methods. The library was used extensively during the raw image processing and depth calculation phases of the algorithm.

Is prior object knowledge known or not? It was decided not to incorporate prior object knowledge into the algorithm. One of the goals of the thesis was to show that it is possible to determine a grasp point without the use of prior object models or similar knowledge. It was thought that this would reduce processing time as well.

What type of objects will be grasped? The types of objects to be grasped were chosen for their interesting shapes and sizes in addition to their weight. Due to the limited lifting ability of the robotic arm, objects had to be lightweight but also provide an interesting shape with which the grasping system would contend. Three objects were chosen to present the grasping system with a set of tests.

Will the scene be cluttered? As mentioned earlier, without the use of object recognition it becomes extremely difficult to decipher the object of interest from the rest of the scene in a cluttered environment. It was decided to focus on the robotic grasp primarily rather than object recognition problems.

Is the grasp simple or complex? Complex grasping requires the use of many techniques (some of which are still in active research) to perform successfully. Given the general constraints on the hardware used in the experiment, the simple grasping method was chosen.

If cameras are used, how are they positioned? Since cameras were selected to gather the data, their placement was determined partly on the capability of the robotic arm as discussed previously. The decision to keep the cameras together (i.e. stereo) was made in order to make the system more practical from a mobile robot's perspective. For instance, if the grasping system developed in this thesis was placed onto a mobile robot, one could argue it would be easier and less obtrusive than having to build an apparatus that allows the cameras to be separated and having each camera look at the workspace from two orthogonal positions.

Now that the basic questions have been answered, details of the algorithm are now given. It is beneficial to start with the overarching algorithm in diagram form. Descriptions of each phase of processing will follow in the order shown below.

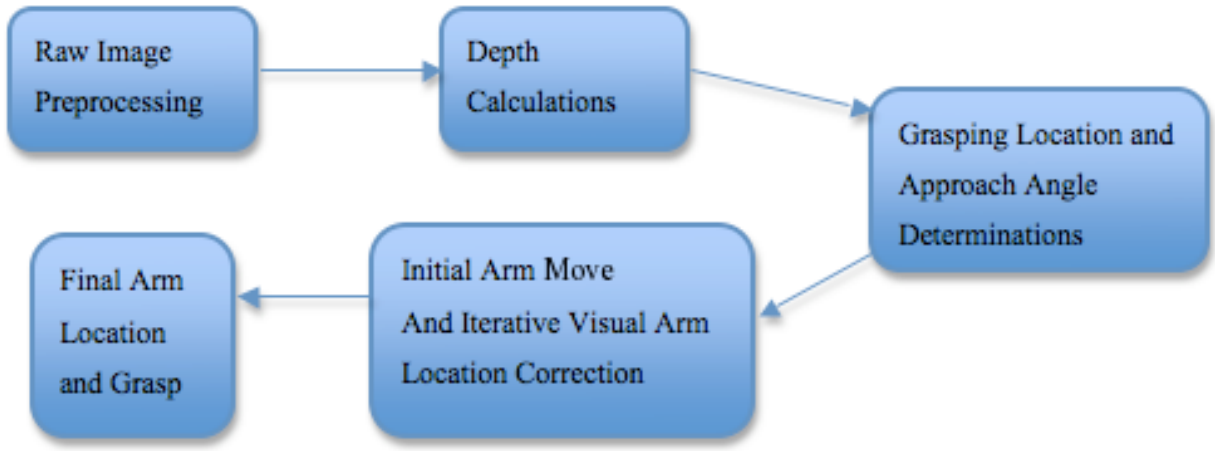

Figure 3-3 Grasp Algorithm

3.3.1 Raw Image Processing

To begin, images must be taken of the workspace. The idea is simple and yet there are a number of caveats. The manufacturing process of lower grade cameras makes it highly likely that individual cameras will not be the same in every aspect. This means that the images taken from each camera will likely have different focus, brightness, contrast, color intensity, white balance, exposure and gain values. Therefore, the first step of raw processing is to set each of those attributes equal on both cameras in so much as they appear the same in each image since they may have different relative values in each camera due to minor electronics-related differences. Illumination is also an item of concern. To be detected consistently, objects of interest must have adequate lighting. For this reason, three lamps were used to provide lighting at three separate angles to provide light to all surfaces of the objects facing the cameras. Care was taken to reduce the amount of directed lighting that would cause reflections off the objects in the workspace. Use of the black cloth further reduced complications with glare and reflections. Cameras were placed approximately 10cm above the workspace and were positioned to face toward the ground plane in order to provide a slight top-down view of the workspace. Doing this reduced the amount of image clutter that had to be dealt with in the background subtraction phase, which comes next.

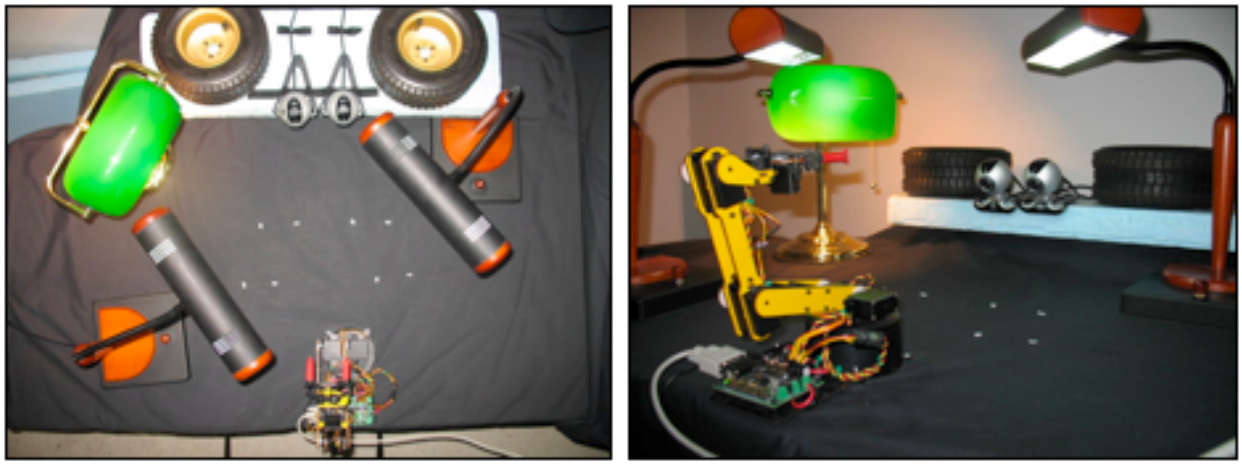

Figure 3-4 Overhead and side view of workspace

The first major task is detecting which object to grasp. As was stated earlier, previous object knowledge was not used so some other method had to be devised. A well-known technique for detecting an object in the workspace is background subtraction also known as background differencing. The idea behind the method is to 'train' the imaging system to recognize an empty workspace. After training, the object of interest is placed inside the workspace and with the object now in view, the background, which was 'learned' during training, is subtracted (i.e. removed) from the current image leaving only the object of interest. During training, a range of colors for each pixel in the image is recorded. Color images are comprised of three channels, which are red, green and blue[14]. Calculations performed on color images must work with the data for each individual channel in most cases since values for a single pixel are rarely the same across all three channels. For background differencing, ranges of values for each channel are recorded. Suppose an object is placed in view. If the value of a particular pixel's channel falls within in the trained color range for that channel, that pixel's channels are set to 255(white) and all others are set to zero (black). The purpose of this is to build an image mask. The binary mask is then applied to the original color image's channels and only non-zero mask pixels are allowed to keep their color. Others are set to black leaving only the object of interest. This technique was performed for both cameras.

[14] Many other color models do exist but RGB was chosen due to its widespread use in computer vision applications.

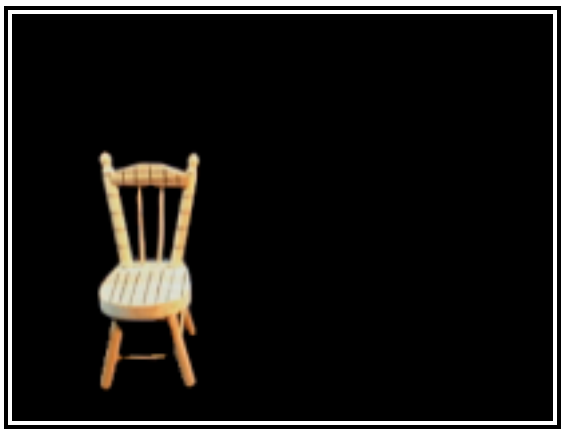

Figure 3-5 Single Camera - Chair with background removed

3.3.2 Depth Calculations

Having obtained a pair of images with the background removed we are left with the object of interest alone. The process to obtain the depth of the object of interest follows the stereo imaging approach described in section 2.2.4. OpenCV provided the many useful methods to help perform the calculations necessary. Referencing section 2.2.4, there are three major steps that must be performed to obtain depth values for a particular pixel. These are calibration, rectification and correspondence, in that order.

Calibration is performed by the following method. Chessboard images are taken from each camera simultaneously then each image is analyzed to find the corners of all inner squares. These provide a set of points for the planar object, i.e. the chessboard. Points are further refined to subpixel accuracy since otherwise they would only be accurate to one pixel eventually leading to higher error rates in correspondences later. A homography of the chessboard corner points onto the camera imager is then performed. Formally, the homography defines the relation between the viewed point Q and the point q on the camera's imager. Defining $\tilde{Q} = [X \quad Y \quad Z \quad 1]^T$ and $\tilde{q} = [x \quad y \quad 1]^T$, the homography H provides the correlation between the two planes as seen as $\tilde{q} = sH\tilde{Q}$ where s is an arbitrary scale factor. Using multiple homographies from a single camera, it is possible to determine the extrinsic and intrinsic parameters for each camera [22]. Recalling from section 2.2.4, the extrinsic and intrinsic parameters give the data necessary to correct image distortion and begin the rectification stage.

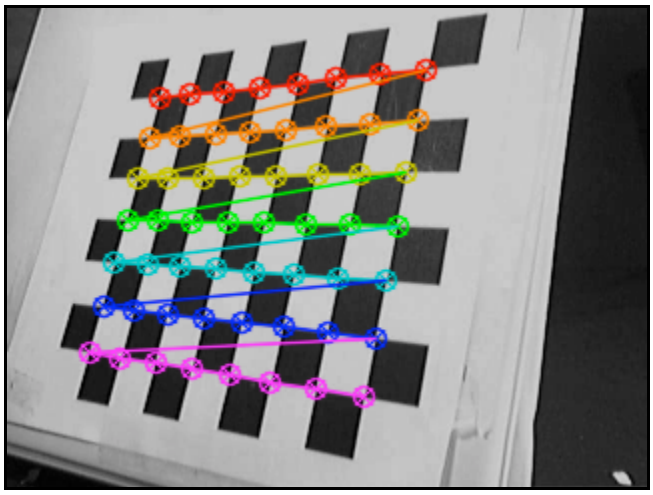

Figure 3-6 Finding chessboard corners

The goal of rectification is to align the image pairs such that corresponding points in each image lie along the same horizontal line. To attain that goal, images must first have their distortions corrected. Then, epipolar geometry is used to relate the two cameras horizontally by enabling the calculation of the fundamental matrix which provides the rotation and translation parameters needed to match the left image pixels to the right image pixels and vice versa. Note the distortions (the outside edges) in the left image below, which actually correct the horizontal alignment of the images on the interior of the image. The outside edges won't be rectified well so it is important to keep this in mind during experiments.

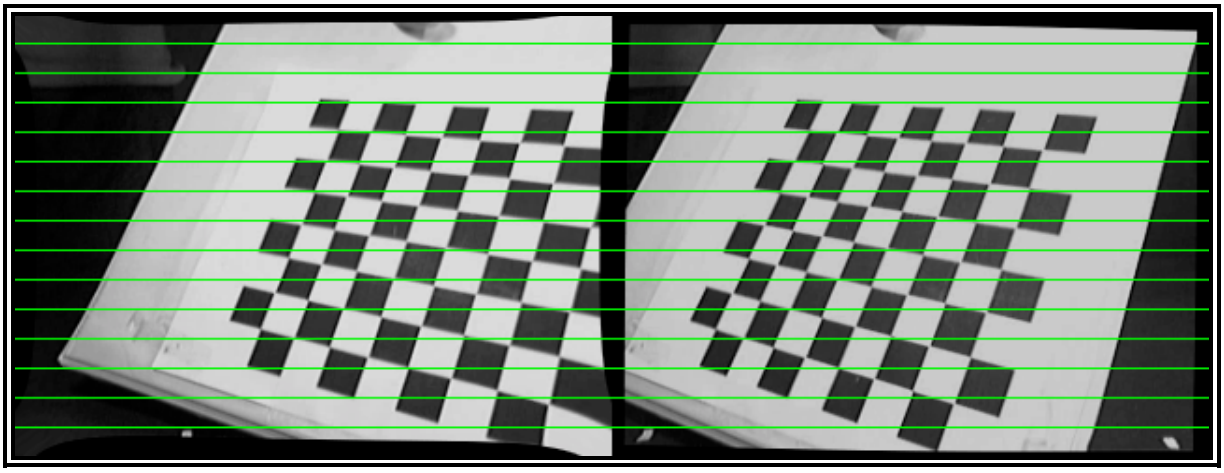

Figure 3-7 Rectified stereo chessboards

The final step is to determine the depth value for each pixel in the image. Correspondence attempts to find the exact pixel in the opposite image that 'corresponds' to the pixel under examination. OpenCV provides two correspondence methods:

`cvFindStereoCorrespondenceBM` for fast matching at the cost of less accuracy and `cvFindStereoCorrespondenceGC` for slower matching but more accurate results. The second was chosen for trials due to the precision required for the grasping task as graph cut methods have been shown to be more accurate than other methods [31]. The result of the correspondence method is a disparity map, which is a grayscale image where lighter shades represent points that are closer to cameras and darker shades represent points that are farther away from the cameras. The final step in determining the depth values is to perform a reprojection of the pixels in the correspondence image into 3D space. A reprojection matrix C, which was calculated during the rectification stage, is used to give each pixel a depth value as shown in the following equation: $C[x \quad y \quad d \quad 1]^T = [X \quad Y \quad Z \quad W]^T$ with the 3D coordinates being X/W, Y/W and Z/W. Figure 3-8 shows the disparity and reprojection images. The reprojection image on the right is difficult to interpret at this level since several severe outliers (the blue dots at the bottom) are constraining the view to a large distance. What is distinguishable however is the large rectangular patch of red is the background of the image, and the small patch of dots above it which is the object. When zoomed in on the object and the background is removed, a point cloud similar to Figure 2-7 becomes apparent. The disparity image on the left is not precise since the correspondences are incorrect in some locations and fail altogether in other locations. Incorrect correspondences are a cause of errors in depth values and failed correspondences lead to omissions in depth values at some image locations.

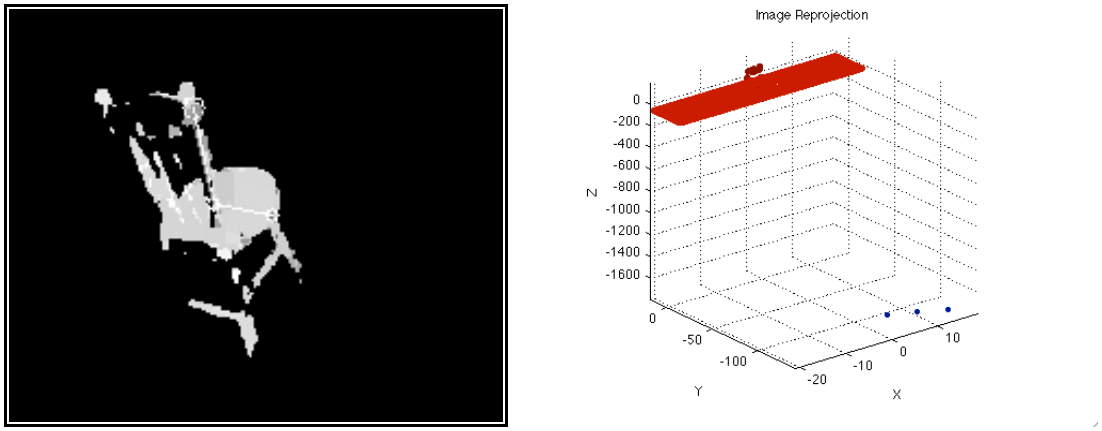

Figure 3-8 Disparity / Correspondence and Reprojection

3.3.3 Grasping Location and Approach Angle Determinations

Clearly, the determination of a grasping point plays a pivotal role in the success of a grasp. A premise of this thesis is that the high point(s) on an object generally provide a reasonable grasping point. This does not necessarily imply the best or an optimal grasp point. Reasonable in this case means that the object is less likely to slip from the gripper upon arm lift if the centers of mass of the end effector and object are not vertically aligned and the object is top-heavy. These perilous conditions can cause objects to twist and spin out of the end effector if the end effector is not positioned accurately. The premise suggests that there are more grasp-points for which a grasp from above provides a smaller difference in vertical, center of mass distances than there are with other side grasp approaches. Thus using a grasping approach from above should result in a higher success rate by virtue of having less object slippage. An ancillary benefit is a reduced possibility for the arm to nudge or push the object away from the end effector during grasping which otherwise would increase failure rates. By minimizing the amount of end effector movement on the same plane as the object we minimize errors related to their interactions on that plane.

As seen in Figure 3-8, the image reprojection contains range readings that are extremely inaccurate. These were dealt with by using a simple filter to remove these points from the data set. In addition to outliers, the raw range data tended to be quite coarse in that range readings appear only at sizeable intervals instead of a steady progression of range values, as one would expect. In order to smooth the range data both median and bilateral filters were applied to the disparity image, which is in turn reflected in the pixel reprojection values. Median filters are good at ignoring outliers by taking the median[15] value in the evaluation neighborhood and bilateral[16] filters are good at edge-preserving smoothing.

With the range data's extreme outliers removed and point distances smoothed to better reflect the object's actual measurements, it is possible to begin searching for a grasping point. If we assume the x, y and z components of each reprojected point to the cameras is random and occurs with equal probability, then it is possible to build a multivariate probability distribution

---

[15] As opposed to an averaging or mean filter that can be greatly affected by outlier points.

[16] Unlike Gaussian filters which smooth without regard to edges

using the components of all reprojected points. The key to showing this is a multivariate normal distribution lies in the Multivariate Central Limit Theorem.

Theorem 3.1 (MCLT) Suppose $X = (x_{1,\ldots,}x_n)^T$ is a random vector with covariance $\sum$ and mean vector u. If $X_1,\ldots X_n$ is a sequence of independent and identically distributed random vectors then $S_n = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}(X_i - \mu_i) \xrightarrow{d} N(0,\Sigma)$ where $\xrightarrow{d}$ is convergence in distribution and $N(0,\Sigma)$ is the multivariate normal distribution defined as $\frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{(x-\mu)^T\Sigma^{-1}(x-u)}{2}}$.

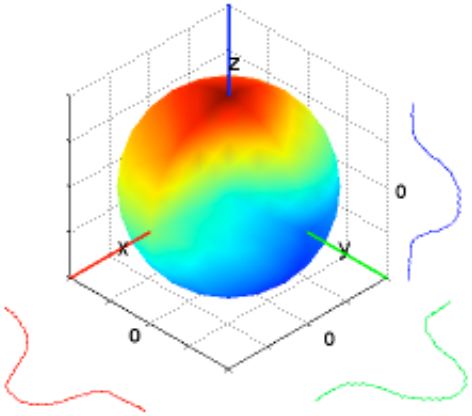

Figure 3-9 Multivariate Normal Distribution

One could argue that the reprojected points are not precisely independent and identically distributed. While the possibility of a degree of correlation may exist among the vectors in X, it is impossible to determine the value of a vector based on previous vectors in the set with any certainty. Thus, in the worst case, the reprojected points are only approximately modeled by the multivariate normal distribution yet still yield acceptable results. In this thesis, $x_1,\ldots,x_n$ = $[x_1,y_1,z_1],\ldots, [x_n,y_n,z_n]$ (each reprojected point's components), $X = (x_1,\ldots,x_n)^T$ and $u = [\ \bar{x},\bar{y},\bar{z}\ ]$, the means for each component. Figure 3-9 shows pictorially how an object with the same component normal distributions would look in 3D. The mean of all the distributions is 0 and the mean vector is [0,0,0]. Note [0,0,0] is the CoG of the sphere. The MCLT is foundational since it can be said with confidence that the reprojected 3D points can be reasonably represented as a whole by the mean vector of the multivariate probability distribution which incidentally happens to coincide with the CoG of the object in question. An important point to be made is that these results are the same regardless of the object and its distance from the cameras. The shape and

distance of the object will only affect the shape of the density function for each component but they will remain normal. The MCLT also shows that the more points there are, the better the distribution will modeled by a multivariate normal distribution. Empirical measurements taken over many trials confirmed the MCLT principles. In general, the CoG may not be the most accurate point calculated in every trial but it does provide a reasonably accurate place to start the grasp point calculations for every trial. Few, if any other points have the same property[17].

Earlier it was stated the high point of an object provides a reasonable grasp point. Unfortunately, the measured distance to the high point tends to be unreliable since correspondence matches are often incorrect at object boundaries meaning that value should not be relied on directly. Instead, a technique called octant analysis was devised to leverage the stability (by the MCLT) of the object CoG in calculating progressively higher and more refined grasp points until a confidence threshold[18] has been reached. Any point in the grasp point set can be used to attempt a grasp but grasp points later in processing generally have a higher likelihood of success. (Inspiration to use octants stems from octrees, which are commonly used in computer graphics to divide three-dimensional spaces into eight partitions.) Once the disparity points have been reprojected into 3D space, the points are placed in an imaginary cube centered on the global CoG. It is at this point where the measured high point on the object is used a guide. The octant in which the high point resides is chosen for a further iteration. All other points outside this octant are discarded. The CoG for the new octant is calculated and the process begins again. The recursion continues until a threshold of confidence in the accuracy of the measurements is reached. Generally, the fewer points used to calculate the CoG in the current octant, the higher the error rate. This reflects back to the normal distribution where the smaller the population the more unreliable the statistics are considered. Eventually, the analysis ends with a single octant CoG that is influenced by the high point of the object calculated earlier. The degree to which the final CoG matches to the high point depends heavily on the number of correspondences found during the vision processing. If the correspondences are high, especially

---

[17] That is unless one considers the median and mode vectors, which should be similar for a normal distribution.

[18] The threshold is essentially the place where the MCLT breaks down and is no longer reliable.

in the octants surrounding the high point, the odds are favorable that the chosen grasp point, i.e. the local octant CoG, corresponds closely to the high point because there are enough data points to keep the confidence high. Conversely, poor correspondences in the octants around the high point can lead to grasp-points quite distant from the high point. In the worst case, no correspondences are found for the entire object and the grasp fails. More commonly, correspondences are found but are mediocre. In these cases, the CoG of the entire object can be calculated and octant analysis can occur at least once. Therefore, while the grasp point chosen is not the high point, we have some confidence that the grasp point chosen resides on or near the object itself and not in empty space rebutting an assured grasp failure. Indeed, the purpose in using octant analysis is to incorporate the multivariate normal distribution's properties in all iterations in such a way that we end with a reasonable grasp point. Figure 3-10 conveys the idea behind octant analysis although it's depiction is not precise in that after each octant is chosen, the new octant is centered on the remaining points and not centered on empty space as shown. With the image's limitations in mind, first the upper-right octant is chosen, followed by the lower-left then upper-left octants.

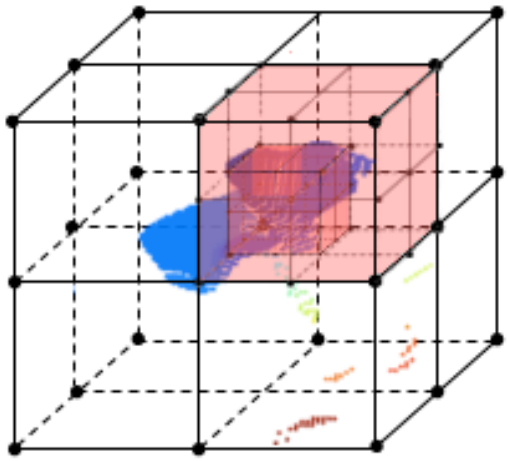

Figure 3-10 Toy bucket data points and octant analysis after four iterations

Figures 3-11 and 3-12 are examples of calculated grasp-points of the bucket and chair found in Figure 4-1 in various poses. The images contain three trials of the bucket and chair respectively. Most of the objects have had their images smoothed but the chair on the far right is the data without smoothing applied. The white circle in the center of the object is the CoG. The large circle is the calculated high point on the object, the # is the calculated grasp point and the smaller circle is the estimated low point (not used). The different shades of gray represent

different calculated depths. It is difficult from these images alone to understand how the grasp point was calculated since it can be hard to discriminate between the grayscale shades and get a clear understanding of the underlying point cloud. Without the help of another dimension, it is difficult to see how all these points relate to each other. As can be seen, the high point and grasp point don't always match closely since the analysis stopped short due to crossing the confidence threshold. Plots, like those in Figure 2-9 and Figure 3-10, were used to further investigate the underlying point clouds during development and testing of the algorithm.

Figure 3-11 Bucket grasp-points

Figure 3-12 Chair grasp-points

Before moving on to the grasp angle procedure, errors in the grasp point distance are addressed. It is known that as space increases between an object and the cameras the depth measurements degrade along the way. Furthermore, the error rate differs depending on the vertical and horizontal components. For these experiments, a linear error model was approximated by plotting measured distances with actual distances and fitting a line to the measurements. The resulting function was used to determine actual distance from measured distance. Since the true error in the system cannot be modeled linearly this approach does have its shortcomings and will fail to give an exact corrected distance in all cases.

With the grasp point now determined, all that is left to find is the grasp angle. Before starting, it should be noted that during vision processing z represents the depth component, y represents the height component and x is the horizontal component. In order to determine a reasonable angle the following process was carried out.

1. The points contained within a sphere around the grasp point were used in the calculations. Points outside the sphere were ignored. This space around the grasp point is termed the Sphere of Angular Influence, or SAI, as these are the only points that will affect the grasp angle. The sphere diameter was chosen to be five centimeters based on experimental trials.
2. All points had their y component set to zero essentially turning the 3D problem into a 2D problem. The height component is not required for simply determining the grasp angle.
3. Conceptually, the view was changed to turn the z component into the y component in a traditional 2D plane. Thus z is now the height component and x is still the horizontal component. This can also be imagined as looking down at the workspace from directly above.
4. Each point had its x component and z component added to separate component sets. The sets were allowed to contain only unique items.
5. After all points were processed in step 4, the largest and smallest points were used to define the length vectors $x\hat{i}$ and $z\hat{j}$ where i hat and j hat are unit vectors.
6. The grasp angle theta was then determined by $\theta = \tan^{-1}\frac{z\hat{j}}{x\hat{i}}$

The intent of the algorithm was to prevent obvious collisions with the object where the gripper approaches parallel to the object's x or z component. Considering both components prevents either from dominating completely and places the gripper's angle in between the components. Still, it is recognized that if one of the components overwhelms the opposing component this approach can fail.

3.3.4 Arm movement and iterative location correction

Having determined the grasp point and angle, the next step is to calculate the servo positions on the robot arm to get the end effector to that point with the appropriate angle. This problem is known as the inverse kinematics problem. The problem is that given a point in space,

there may be many sets of servo positions that place the end effector in the correct position. Unfortunately, some of those solution sets contain servo angles outside the operating bounds of the servo(s) and must be discarded. Others must be discarded if the servo angles place the arm below the work surface at any point during execution. The choice from the remaining sets still must be executed with care since simply setting all the joints to their desired locations may swing the arm through the object to be grasped. Choosing the optimal path and set from the group determined from the IK calculations is a complex problem. As stated in the assumptions earlier, this thesis chose not to focus on these particular issues. Therefore, the method implemented to determine a suitable set of angles for the arm was found on the arm manufacturer's website [32]. The spreadsheet on the website defines the formulas needed to calculate angles based on a given x and y coordinate that defines the end effector height and length from the robot's base. Further servo range limitations were placed on the calculations to reduce the solution set to one by allowing a top-down end effector approach only.

3.3.5 Final Grasp

A background subtraction image, containing the workspace and object of interest, for both cameras is calculated. Then, with the grasp point, grasp angle and servo values prepared the algorithm executes an arm move to the desired location but positions end effector slightly above the target point. The program then attempts to find the range of the robot's end effector by using the stereo approach described earlier. In Figure 3-4, it can be seen that the gripper ends are covered with red tape. This was used to help identify the gripper from the rest of the workspace objects since object recognition routines were not implemented. The distance from the CoG of the gripper and grasp point is calculated. New servo positions are determined based on the difference in the expected values and the actual values. This becomes the final grasp position. The arm is moved to the new location slightly above the target then lowered into grasping position. The end effector's pincers are closed and the arm lifts with the object being grasped.

CHAPTER 4 - Experiments

In this chapter, the description of the full system setup and experiments grasping various objects in different poses are described and executed.

4.1 System preparation

The workspace is prepared by setting up the lighting in such a way as to minimize shadows when objects are placed in the workspace and making sure the cameras are aligned as close to frontal parallel as possible. Markers are placed on the floor of the workspace to designate areas visible to both cameras. The left image of Figure 3-4 shows the markers in the center of the workspace. The middle trapezoid represents the space where objects will be in full view of both cameras. Next, camera calibration is performed by capturing twenty-five pairs of chessboard images and processing them via the calibration method. The outputs of the calibration method are the intrinsic and extrinsic matrices along with the reprojection matrix. These matrices provide the heavy lifting required to rectify the camera images and reproject the 2D image points into real 3D space. Importantly, they are read into the grasping program used in the following section to make the majority of vision related calculations. Lastly, physical measurements of the workspace are taken and used in the program to transform camera coordinates to arm coordinates.

4.2 Experiments

The objects chosen for the trials were:

- A dollhouse chair
- A small toy bucket
- A power adapter end

The objects were chosen based on their size, color, weight and novelty factor. Objects with simple shapes such as balls, blocks and cylinders were avoided in order to test the hypothesis that objects with novel shapes can be grasped. Due to the assumptions placed on the implementation, certain colors were avoided. Dark colors resembling the black cloth covering the workspace as well as objects with highly reflective surfaces had to be avoided. Often times, dark objects weren't detected at all by the vision system and reflective surfaces not only caused image distortions but correspondences resulting from the captured images were extremely poor

and virtually unusable. Reasoning for the size and weight requirements was described in section 3-2. The novelty factor is highly subjective but the idea was to give the algorithm a relatively eclectic set of objects to grasp.

Figure 4-1 Objects of interest

The algorithm starts by learning the initial workspace background then the chosen object is placed in the workspace. Care is taken not to allow movement in the workspace or directly behind the workspace as it can disturb the background learning and grasping methods likely resulting in a failed trial. Stereo images are captured and processed resulting in a 3D point cloud of data points representing the object in the workspace. It should be noted that computer vision accounts for the bulk of the algorithm's processing time. Continuing, the point cloud is then run through the octant analysis and SAI methods. The program then performs the initial grasp and uses visual feedback to make a final set of adjustments before performing the final grasp.

The typical processing time was two minutes per trial. The objective was to test the repeatability of the grasping as well as showing the algorithm's ability to grasp objects in differing poses and in slightly different locations. Trials were run for each pose five times. In each pose, the object remained in the same pose but might have been placed in slightly different locations within the common camera-viewing trapezoid. Due to the shapes of different objects, not all were evaluated in the same number of configurations. Objects with higher degrees of symmetry necessarily have fewer unique viewing angles. During tests, each object was evaluated based on its particular set of unique characteristics. Thus, some objects have relatively few poses and others have substantially more. The dollhouse chair was tested the most comprehensively due to the many possible configurations. The toy bucket was tested in fewer positions due its symmetry. The plug was tested in fewer positions for a number of reasons. The metal ends of the plug were essentially invisible to the system and played no active role in the

object's shape. The metal ends were used to place the object in standing poses however. Poses, where the plug is laying flat and either face up or face down, were eliminated since these poses are basically seen as a generic block or ball shape.

Figures 4-2 and 4-3 show the ending location of the robot end effector after each trial. As can be seen, the chair underwent 30 trials, and the bucket and plug end underwent 15 trials each. Upon inspection, it is possible to tell which trials succeeded and those that didn't.

Figure 4-2 Chair Trials

Figure 4-3 Bucket and Adapter Trials

CHAPTER 5 - Results Analysis

In this chapter the results of the trials for each object class captured during the trials are discussed and analyzed. The chapter begins by discussing some specifics with the correspondence method used during the trials then looks at each object individually. The chapter concludes by summarizing the data of all the objects combined and discusses several pertinent metrics.

5.1 OpenCV Graph Cut Correspondence Implementation Analysis

Before beginning in earnest, a few item specifics as to how the correspondences were calculated (Section 3.3.2) must be discussed as they may have implications for all trials. The OpenCV implementation of the graph cut algorithm is based on Kolmogorov [33]. It is defined as $E(f,g) = E_{data}(g) + E^{(1)}_{smooth}(f) + E^{(2)}_{smooth}(g) + E_{vis}(f) + E_{consistency}(f,g)$ with the smoothness factor $E^{(1)}_{smooth}(f)$ defining how well the discontinuities are preserved in the depth measurements and $E_{data}(g)$ defining which interactions between pixels in both images are visible. Results from [31] showed that the accuracy of the depth measurements are highly dependent on the balance between the data and smoothness factors and that finding a good smoothness value is image specific. Since the precise implementation of the OpenCV GC algorithm was not known, there exists a non-negligible possibility that depending on how the $E^{(1)}_{smooth}(f)$ value was determined, gradual depth changes will be calculated better than locations where the disparity/depth changes significantly (see convex smoothness) [33]. In the case where the disparity changes significantly, depth values can 'bleed' over resulting in incorrect depth calculations for pixels around the depth change.

5.2 Chair Results and Analysis

The chair was tested in six positions and the results are captured in Table 2. The table records eight distinct pieces of data. Column 'Pose Description' is self-explanatory. '+ Grasps' indicate successful grasps, '- Grasps' indicate failed grasps and '+ %' gives the success percentage out of the five trials for that pose. 'Error (cm)' describes the distance by which the gripper missed the intended grasp point, 'Good ∠' and 'Bad ∠' summarize the number of trials

with good and bad grasp angles and the final column '+/- Grasps' gives the number of trials where the grasp was considered mediocre. Mediocre grasps did succeed in grasping the object but could have occurred as a result of grasping in the wrong location or the object was moved or bumped during the procedure but ended in a pose that was beneficial or at least non-detrimental.

Poses were chosen in order to present the imagers with a unique view of the chair. Referring to Figure 4-2, each major part of the chair was faced towards the cameras and presented the system with a slightly different problem. As can be seen in Table 2, the success rates vary among the poses.

#	Pose Description	+ Grasps	- Grasps	+ %	Error Δ (cm)	Good ∠	Bad ∠	+/- Grasps
1	Chair back facing cameras	4	1	80%	3	5	0	1
2	Chair seat and back facing ground	3	2	60%	1,2	5	0	1
3	Chair on back, seat facing cam	1	4	20%	1,1,2,1	5	0	0
4	Chair normal, facing cam	3	2	60%	1,1	5	0	0
5	Feet facing cams	4	1	80%	3,.5	5	0	2
6	Chair profile	2	3	40%	1,1,0	3	2	1
	Raw	17	13	56.67%	~1.32 (1.29)	28	2	5
	≤ 1cm error w/ good ∠	24	6	80%	NA	NA	NA	NA

Table 5-1 Trial chair results

Poses one and five had the highest success rates. The chair legs, having good texture and color differences in relation to the rest of the chair and background, were a prominent high feature, the flat seat was a secondary foreground feature of the pose and the depth progression of the chair was fairly smooth. This combination of features likely played a dominating factor on the success for these poses. Interestingly, these poses also had the highest error deltas. In the two trials where the error was the greatest, the arm collided with the chair causing it to shift pose resulting in a significant grasp error.

Pose three had the worst results yet appears to only have missed by a small margin on many of the trials. The chair back and seat play the dominating features in this pose. The chair legs, although the highest points of the object are set behind the seat from the cameras' view.

Recalling section 5.1, it is possible that the sudden depth change from the chair seat to the legs might have influenced depth measurements enough to cause missed grasps. In addition, it is known that as the distance between the imagers and the object increase, the depth accuracy diminishes. Although the error rate was corrected using a linear model it is possible that the error will not have been completely negated. Since the chair legs were the farthest away from the camera and were chosen to be the grasp point, the error correction explanation seems the most likely.

The three remaining poses faired average in their success. In each of these poses, the chair is standing upright with only the placement of the chair back differing. Again, many of the errors were within only one centimeter. In pose six, the chair is sitting profile to the cameras. In this configuration, the correspondence would be more difficult as the chair top is directed away from the cameras. The closer the chair top aligns with a camera, that camera's view of the chair top becomes compressed resulting in more correspondence mismatches. Pose six also had two instances where the grasp angle was incorrect. The shortcomings of the grasp angle calculation became known in these instances. Investigation into an improved grasp angle calculation was undertaken but was done after these trials. Trials two and four made up the last of the average success trials. In pose two, the chair back was closer to the cameras and in pose four the back was further away. This didn't seem to make too much difference in the results however. With the results being so similar, the errors could be related to lighting and/or texture issues, which again would affect correspondence validity.

Overall, the raw success of grasping the chair was 56.6% with an average error of 1.32cm. The error when the best and worst outliers were thrown away was 1.29cm. The vast majority of grasp attempts had good grasping angles and roughly a third of the successful grasps were mediocre.

5.3 Bucket Analysis

The bucket has a fairly simple shape yet it poses a few challenges. The sides of the bucket are angled inward, the edges are quite thin, and the color and texture remain the same throughout the whole bucket. The side angles can affect the grasp causing it to slip and the thin edges, texture and color issues can all affect the correspondence matching. Conversely, in some poses the bucket provides numerous ways in which the end effector can grasp the object, be it

with an enveloping grasp or more specific such as a point on the rim. The three bucket poses chosen were right side up, upside down and on its side. The results of the bucket trials are found in Table 3.

#	Pose Description	+ Grasps	- Grasps	+ %	Error Δ (cm)	Good ∠	Bad ∠	+/- Grasps
1	Bucket, upside-down	3	2	60%	2,6	5	0	1
2	Bucket right side up	5	0	100%		5	0	4
3	Bucket on side	2	3	40%	2,2,2	3	2	1
	Raw	10	5	66.6%	~2.8 (2)	13	2	6
	≤ 1cm error w/ good ∠	10	5	66.6%	NA	NA	NA	NA

Table 5-2 Trial bucket results

Again, starting with the most successful pose we find that it had is successfully grasped the bucket in all five trials. Four of these successes were mediocre however. During four of the five trials one of the end effector's digits ended up inside the bucket, allowing the inside and outside digits to pinch the edge of the bucket. In a few cases, it looks as though the grasp point depth was slightly off but near enough for the end effector to collide with the bucket and maintain a grasp. The two remaining trials fared similarly in their success. Both pose one and three consistently missed by at least two centimeters. The extreme miss of six centimeters in pose one can only be explained by very bad correspondence matches. The surface of the bucket is rather reflective and in pose one the light is reflected more toward the cameras than in the other two poses. The resulting images had large areas of white pixels where the reflection occurred and made matching those pixels virtually impossible.

The average error for the bucket turned out to be about 2.8cm and 2cm when outliers were thrown out. The success rate was 66% however, which was also higher than the chair. Of the successful grasps, 60% were mediocre. The grasp angle suffered the most in pose three. Depending on the lighting conditions and resulting point cloud, the z component might have weighed in much less on the angle determination resulting in a grasp that was largely parallel to the bucket face for the two trials with bad angles.

5.4 Plug Analysis

The final object proved to be the most difficult to grasp during the experiments. The plug was very challenging on a number of points. The texture of the plug was very smooth, the surface was glossy, it only had two colors, and the metal ends of the plug were extremely reflective. The only positive factor was that it did have a shape that was interesting in several poses. The three poses chosen all had the plug leaning on the metal edges but varied on which end was touching the work surface. Poses where the plug was laying flat on the ground were avoided since the grasp point found in such a position might have caused the gripper to collide with the work surface. Table 4 shows the outcome of the plug trials.

The two 'successful' poses had success rates below fifty percent. As expected, the vision system had a difficult time with the aforementioned challenges. In poses one and two however, it appears that there were enough points in the images that matched. This was likely due to shadows creating artificial texture on the surface of the plug so enough correspondences could be found. While the trials were being carried out for the plug, it was noticed that the metal parts of the plug were not 'seen' in any trial. Chrome proved to be beyond the system's capabilities thus the remaining part of the plug was used to find a grasp point. Even with all the challenges missed grasps were not that severe.

#	Pose Description	+ Grasps	- Grasps	+ %	Error Δ (cm)	Good ∠	Bad ∠	+/- Grasps
1	Plug profile, leaning on metal ends	2	3	40%	2,1,1	5	0	1
2	Plug resting on small end, leaning on metal	2	3	40%	1,2,1	5	0	0
3	Plug large flat face facing cameras	0	5	0%	1,1,2,2,1	5	0	0
	Raw	4	11	26.6%	~1.36 (1.33)	15	0	1
	≤ 1cm error w/ good ∠	11	4	73.3%	NA	NA	NA	NA

Table 5-3 Trial plug results

Last and least of all the trials was plug pose three. With a zero percent success rate it deserves a bit of scrutiny. In this pose, the flat surface of the plug was faced toward the cameras. It was very similar to bucket pose one which incidentally had the worst miss of all the trials. It is

almost certain that there are similar factors causing the missed grasps in both poses. Aside from the reflected light, lack of features and color for the vision system to utilize, the depth change in the plug pose was severe. Lack of texture around the edge of the face of the plug also probably affected the results. In four of the trials, the depth was calculated to be too far. Mismatched correspondences could have been the cause as well as the depth correction function.

All in all, the average error was 1.36cm and 1.33cm if outliers are thrown out. There were no bad grasp angles during these trials but the plug was small enough that just about any angle would have probably worked. The raw success raw was 26.6%.

5.5 Summary

The results for each object lead to an interesting comparison among them. A few observations are listed below.

- The bucket ended with the highest grasp rate of all the objects yet missed by greater distances on average than the other two objects.
- When considering trials with an error less than or equal to 1cm the chair ends up with the highest grasp rate with a solid 80%. Surprisingly, the plug is in second place with a success rate of 73.3% and the bucket ends in third with the same success as it had with the actual grasp rate, 66.6%.
- There were three poses with actual grasp success rates equal to or greater than 80%. Of these, two included the chair and one included the bucket. In all cases, a mediocre grasp accounted for at least part of the successes.
- The two worst trials included one from the plug and one from the chair. In the plug pose, the gripper reached too far 80% of the time. Oppositely, in the chair pose, the gripper didn't reach far enough 100% of the time.
- On a more comprehensive scale, the majority of all chair misses were missed short. The majority of plug misses were missed long. The bucket seems to have missed predominantly long. Generally, it appeared that as the grasp point moved farther away from the cameras the gripper failed to move out far enough. Also, when the grasp point was close to the cameras the gripper went too far.

The summarized results of all objects are found in Table 5. The accumulated raw success of the system in all trials was 51.67%. On average, the total error for all trials was about 1.82cm

and 1.54cm when outliers for each object are thrown out. Of the thirty misses, 56.6% percent were less than or equal to one centimeter. About one third of the successful grasps ended with a mediocre grasp. Grasp angles were found to be good 93% of the time.

	+ Grasps	***- Grasps***	***+ %***	***Error Δ (cm)***	***Good ∠***	***Bad ∠***	***+/- Grasps***
Raw	31	19	51.67%	~1.82 (1.54)	56	4	10
≤ 1cm error w/ good ∠	45	15	75%	NA	NA	NA	NA

Table 5-4 Summarized results

CHAPTER 6 - Conclusion

The ability to find a grasping point for novel objects is a key task in bringing utility robots into our everyday lives. The ability to perform grasping using cheap components also plays a critical role in making this technology affordable to the masses. To take a step toward that goal this thesis presented octant analysis as a method to find a grasp point on an unknown novel object from a single view using consumer grade electronics. The algorithm succeeded in picking up various novel objects in various poses using data acquired from a single view source over half of the time on average and even better for half of the object poses under test. In all cases, a reasonable grasp point was found even in the presence of a high degree of correspondence errors. Experiments revealed that the basis of finding a grasping point using octant analysis performed quite well even if the physical grasp failed due to other complications. Further investigation into this approach seems to be justified. Examples might be where octant analysis is used to determine how close or far away parts of an object lie. It could also be extended to possibly recognize parts/surfaces of objects and use that information in determining a good localized grasp point on that particular part of the object. Lastly, the uses and extensions of the successive CoGs found during OA (i.e. CoG point path) haven't been fully considered.

Experiments also revealed some issues that could be addressed to make the octant analysis approach more applicable to various problem domains. With each trial approaching two minutes, it makes using this system impractical for real-time systems uses. The bulk of the limitation comes from the vision subsystem, which could be improved drastically with dedicated hardware. The variety of the items grasped with this system is limited due to the weight constraints on the arm itself. Clearly, the current system cannot be used in any safety critical environments, like a home, since no safety features were incorporated to prevent or avoid arm collisions. Putting this algorithm in a real home environment would require extensive development efforts as the lack of any feedback from the arm makes this very difficult. On a similar vein, lack of force feedback from gripper prevents completely robust grips. There is no way to tell the quality of the grasp on the object or if the object is slipping when a lift is attempted. With no way to compensate for failing grasps, this system wouldn't be ideal situations where complete robustness was required. Lastly, objects with highly reflective

surfaces and/or lack of texture and/or monochrome styled coloring are likely to experience a lower grasp success rate than objects without those traits. Various methods for dealing with these difficulties could be incorporated to improve the overall grasp success rate of the system.

6.1 Summary

The bulk of research in robotic grasping has historically relied on high-grade sensors and robotic arms. While the reasoning behind using such materials is warranted and important, this thesis targeted novel object grasping using lower-grade components in order to show it is possible to perform a complex robotic task using affordable consumer components with the hopes of bringing this technology closer into the lives of consumers in some fashion. To compensate for the correspondence difficulties, i.e. poor data, a new technique in regards to robotic grasping was developed. The combination of octants, the multivariate central limit theorem and the multivariate normal distribution applied to the grasping problem was not found in a search of the grasping literature. Octant analysis itself is easily comprehensible, easy to implement and results in giving back a 'reasonable' grasp point by using the properties of the multivariate normal distribution as the foundation. It is believed that the missed attempts during the trials are more of a reflection of the computer vision difficulties than that of the grasp point return. If the trials that failed by one centimeter or less could be improved the success rate for all trials jumps dramatically to 75%. Isolating the specific error points and the amount of error for each part of the system would reveal where the most benefit could be gleaned with further work.

The total cost of the entire grasping system, which included the pc, robotic arm and two webcams was about $1400. This price point is very attractive in terms of affordability to consumers and lends credence to the idea that challenging robotic grasping can be performed with commonplace electronic components.

6.2 Future Work

The most intriguing possibility for immediate future work would be in improving the speed and quality of the vision processing system. Micro projectors are now approaching sizes able to fit in cell phones. Additionally, these projectors are becoming more affordable by the day. The use of such a projector in combination with a single upgraded camera utilizing structured lighting would reduce processing time when compared to processing stereo images

while still maintaining or possibly improving the quality of the point cloud generated from the system.

Frankly, the arm used for the trials was too weak to be of any practical use. A more robust arm would have to be used and ideally pressure, weight and proximity sensors would be incorporated to add stability, functionality and failsafe features to the system. Using such an upgraded arm along with an improved kinematics engine nearly puts this in the living room of consumers.

References

[1] G. Tarawneh, "Black [X] truder - First 3D Scans," Jul. 18, 2009. [Online]. Available: http://black-extruder.net/blog/first-3d-scans.htm. [Accessed: Oct. 19, 2009].

[2] P. J. Besl, "Active optical range imaging sensors," in Advances in Machine Vision (J. Sanz, Ed.). New York: Springer-Verlag, 1989; see also Machine Vision and Applications, vol. 1, pp. 127-152, 1989.

[3] C. Rocchini, P. Cignoni, C. Montani, P. Pingi and R. Scopigno, *A low cost 3D scanner based on structured light,* Computer Graphics Forum (Eurographics 2001 Conference Proc.), vol. 20 (3), 2001, pp. 299-308, Manchester, 4-7 September 2001.

[4] N. Ohta. "Optical flow and 3-D shape," (n.d.). [Online] Available: http://www.ail.cs.gunma-u.ac.jp/~ohta/3-D.html [Accessed: Nov. 8, 2009].

[5] Wikipedia contributors, "Epipolar geometry," *wikipedia.com*, Nov. 28, 2009. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Epipolar_geometry&oldid=328414056 [Accessed: Dec. 4, 2009].

[6] Wikipedia contributors, "Unimate," *wikipedia.com*, Sep. 16, 2009. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Unimate&oldid=314361708 [Accessed: Oct. 11, 2009].

[7] R. M. Alqasemi, E. J. McCaffrey, K. D. Edwards and R. V. Dubey, "Wheelchair-mounted robotic arms: Analysis, evaluation and development," in *Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on,* 2005, pp. 1164-1169.

[8] E. Guizzo. "IEEE spectrum: Willow garage PR2 robot navigates through office, plugs itself into electrical outlet," Jun. 9, 2009 [Online]. Available: http://spectrum.ieee.org/automaton/robotics/robotics-software/willow-garage-pr2-robot-navigates-through-office [Accessed: Sept. 17, 2009]

[9] A. Saxena, "STAIR, the STanford AI Robot," (n.d.). [Online]. Available: http://ai.stanford.edu/~asaxena/stairmanipulation/ [Accessed Oct. 11, 2009].

[10] Z. Xu, T. Deyle and C. Kemp, *1000 trials: an empirically validated end effector that robustly grasps objects from the floor.* In Proceedings of the 2009 IEEE international Conference on Robotics and Automation (Kobe, Japan, May 12 - 17, 2009). IEEE Press, Piscataway, NJ, 3971-3978.

[11] Wikipedia contributors, "Control theory," *wikipedia.com*, Apr. 22, 2010. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Control_theory&oldid=357607830 [Accessed: Jan 31, 2010].

[12] "Feedback loop | Define Feedback loop at Dictionary.com", *dictionary.com*. 2009. [Online]. Available: http://dictionary.reference.com/browse/feedback+loop [Accessed Oct. 18, 2009].

[13] G. Taylor and L. Kleeman, *Grasping unknown objects with a humanoid robot*. In Proc. 2002 Australiasian Conf. on Robotics and Automation, 2002, pp. 191–196.

[14] H. Nguyen, C. Anderson, A. Trevor, A. Jain, Z. Xu and C. C. Kemp, "El-E: An assistive robot that fetches objects from flat surfaces," in *HRI Workshop on Robotic Helpers,* 2008,

[15] B. Wang, L. Jiang, J. W. LI and H. G. Cai, "Grasping unknown objects based on 3d model reconstruction," in *Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on,* 2005, pp. 461-466.

[16] D. Fofi, T. Sliwa and Y. Voisin, "A comparative survey on invisible structured light," *SPIE Electronic Imaging – Machine Vision Applications in Industrial Inspection XII*, San José, USA, pp. 90-98, January 2004.

[17] J. Batlle, E. Mouaddib and J. Salvi, "Recent progress in coded structured light as a technique to solve the correspondence problem: A survey," *Pattern Recognition,* vol. 31(7), pp. 963-982, 1998.
[18] J. L. Barron, N. A. Thacker, "Tutorial: Computing 2D and 3D Optical Flow," Imaging Science and Biomedical Engineering Division, Medical School, University of Manchester, Jan. 2005.
[19] Wikipedia contributors, "Motion perception," *wikipedia.com*, Mar. 25, 2010. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Motion_perception&oldid=351949181 [Accessed: Apr. 20, 2010].
[20] R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses," *IEEE Journal of Robotics and Automation,* vol. RA-3, 1987.
[21] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 22, pp. 1330-1334, 2000.
[22] G. Bradski and A. Kaehler, *Learning OpenCV*, First ed. O'Reilly Media, 2008,
[23] A. T. Miller and P. K. Allen, "Graspit! A versatile simulator for robotic grasping," *Robotics & Automation Magazine, IEEE,* vol. 11, pp. 110-122, Dec. 2004.
[24] D. Kragic and H. I. Christensen. "Model based techniques for robotic servoing and grasping." Presented at *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 299-304, 2002.
[25] V. Kyrki and D. Kragic, "Integration of Model-based and Model-free Cues for Visual Object Tracking in 3D," *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on,* pp. 1554-1560, 2005.
[26] G. Taylor and L. Kleeman. "Fusion of multimodal visual cues for model- based object tracking." In *Australasian Conference on Robotics and Automation (ACRA2003),* Brisbane, Australia, Dec. 2003.
[27] O. Fuentes and R.C. Nelson. "Experiments on dextrous manipulation without prior object models." Technical Report 606, Computer Science Department, University of Rochester, Rochester, New York, Feb. 1996.
[28] A. Saxena, J. Driemeyer and A. Y. Ng, "Robotic Grasping of Novel Objects using Vision," *Int.J.Rob.Res.,* vol. 27, pp. 157-173, 2008.
[29] D. Pelletier. "Robots will surpass human intelligence by 2030, scientists say." May 18, 2008. [Online] Available: http://memebox.com/futureblogger/show/541-robots-will-surpass-human-intelligence-by-2030-scientists-say [Accessed: Jan. 26, 2010].
[30] C. E. Smith and N. P. Papanikolopoulos, "Vision-guided robotic grasping: Issues and experiments," in *In Proc. of the 1996 IEEE International Conference on Robotics and Automation,* pp. 3203-3208, 1996.
[31] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *International Journal of Computer Vision,* vol. 47, pp. 7-42, Apr. 2002.
[32] L. Gay. "Lynxmotion arms and inverse kinematics," (n.d.). [Online]. Available: http://www.lynxmotion.com/images/html/proj073.htm [Accessed: Mar. 10, 2010]
[33] V. Kolmogorov. (2004), *Graph based algorithms for scene reconstruction from two or more views*. PhD thesis, Cornell University, Jan. 2004.
[34] A. Bicchi, "Hands for dexterous manipulation and robust grasping: a difficult road toward simplicity," *IEEE Transactions on Robotics and Automation,* vol. 16, pp. 652, 2000.