The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Development and Evaluation of a Vision Algorithm for 3D Reconstruction of Novel Objects from Three Camera Views

Steven C. Colbert*, Redwan Alqasemi, and
Rajiv V. Dubey
Department of Mechanical Engineering
University of South Florida
Tampa, Florida 33620
Email: sccolber@mail.usf.edu
alqasemi@eng.usf.ed
dubey@eng.usf.edu

Gregor Franz and Konrad Wöllhaf University
of Applied Sciences
Ravensburg-Weingarten
Weingarten, Germany
Email: franzg@hs-weingarten.de
woellhaf@hs-weingarten.de

Abstract— **When planning robotic grasping and manipulation maneuvers, knowledge of the shape and pose of the object of interest is critical information. In order for an autonomous or semi-autonomous system to operate intelligently in an unstructured environment and interact with novel objects, it must have the ability to recover this information at run time, even when no *a priori* information of the object is available. In this paper, we describe the development and testing of an algorithm that can reconstruct the full 3D geometry of a novel object from just three images. A variant of shape from silhouettes, the algorithm first generates a rough surface approximation in the form of a point cloud. This approximation is then refined by fitting an eleven parameter geometric surface to the points in such a manner that the surface ignores noise and perspective projection shadows. We test the algorithm in both simulation and on several real world objects. We show that the algorithm provides accurate reconstructions that can be directly used to plan grasping maneuvers. Compared to other attempts in the literature, the proposed algorithm is faster, requires fewer images, is more accurate, and degrades gracefully in the presence of bad data. A real world test case is included that shows that the algorithm still yields usable results when the form of the object is amorphous or otherwise non-geometric.**

I. Introduction

Accurate knowledge of the position, pose, and shape of objects of interest is critical information in the process of planning robotic grasping and manipulation maneuvers. In unstructured environments, the ability to recover this information at run time is a crucial behavior for an autonomous or semi-autonomous system. Up until this point, many of the systems described in the robotics and machine vision literature have focused on recognizing objects in which the system has some form of *a priori* knowledge. This knowledge usually takes the form of a 3D model of an object and/or a corresponding set of images or feature vectors of the object. Once the object is recognized by matching the visual input to the patterns in the database, various techniques such as visual servoing are employed to grasp the object according to a predefined metric. For some recent examples of work in this area, see [1], [2], [3], [4], [5].

In contrast, there have been relatively few attempts at recovering the geometry of novel objects for which the system has no prior knowledge. For robots that are tasked with operating in a fully unstructured environment, such as a domestic service robot, such behavior is critical since it would be a herculean effort, and wholly impractical, to fully program the system with the plethora of objects it could be asked to manipulate.

Perhaps the most complete example(s) is [6], [7], [8], where the authors have developed a mobile robot that can reconstruct novel objects for the purposes of grasping by capturing a sequence of images as the mobile base drives around the object and subsequently performing a dense structure from motion reconstruction via SIFT key point matching. The location of the object of interest, however, must be given beforehand. An alternative to dense stereo reconstructions and key point matching is the well-known shape from silhouettes. Though this method has been well researched, e.g. [9], [10], [11], [12], [13], a novel shape from silhouettes algorithm was recently developed in [14] which, we believe, is both conceptually and computationally more efficient than previous voxel coloring methods, and far more efficient than dense reconstructions. Further, this method does not suffer from the limitation of key point matching algorithms where the reconstruction will fail if the object has little to no texture; a case commonly encountered with household objects. However, an issue with both of these approaches is the large number of images required for the reconstruction and the fact that images are required from around the entire periphery of the object. In the case of [8], 134 images were captured around the periphery of the object leading to an offline reconstruction time of around 100 seconds with a 2 GHz Intel CPU. The authors of [14] reduce the number of captured images to 12 and though they do not report the execution time, they claim real-time performance. But in both cases, the requirement of the system to capture images from around the entire periphery severely limits the application of the algorithm in unstructured environments

*This work was sponsored in part by the Baden-Württemberg-STIPENDIUM. The Landesstiftung Baden-Württemberg has established the Baded-Württemberg-STIPENDIUM in order to support the international and reciprocal exchange of qualified young people. Further information: **www.bw-stipendium.de** and **www.landesstiftung-bw.de**

978-1-4244-6676-4/10/$25.00 ©2010 IEEE

where complete 360 degree access to the object is unlikely to be available.

Rather than reconstruct the full 3D geometry of the novel object, various approaches have been proposed that either make simplifying assumptions of the general nature of the objects, or seek an alternative means of determining grasping positions. The authors in [15] have developed a special purpose robot, 'El-E', that uses a multitude of sensors and cameras to manipulate unknown objects. They assume that the object is oriented vertically on a horizontal surface with respect to gravity and rely on the horizontal 2D cross sectional geometry of the object and an overhead approach in order to perform the grasp. The authors in [16] have developed a novel system for manipulating unknown objects that predicts appropriate grasping locations without needing to reconstruct the full 3D geometry of the object. Given multiple 2D images of a novel object, the system seeks to predict and triangulate the location of an appropriate grasping position. What constitutes an appropriate position is defined by a synthetically generated training set of various objects which are unrelated to the real world objects. Appropriate grasping positions are marked on objects in the training set and various feature vectors of these locations are stored for later run time comparison. The training phase is a one-time operation, and the system is capable of calculating an appropriate grasp position for objects that vary widely in form and appearance from those in the training set. The advantage of these approaches are that they do not require many images of the objects in order to successfully grasp the object and, provided the algorithmic assumptions hold, are robust in their capabilities. The disadvantages become apparent when the assumptions break down, or when the geometric information of the object becomes required. In the case of [15], limitations become apparent when the object does not lie on a horizontal surface, does not have a relatively constant cross section geometry, or when an overhead grasping approach is untenable.

This paper describes the development and evaluation (with a focus on the latter) of a new shape from silhouettes algorithm that we have developed which attempts to provide the benefits of both of the above approaches while eliminating most of the drawbacks. That is, our algorithm is capable of reconstructing, to a sufficiently accurate approximation, the full 3D geometry of a completely novel object using substantially fewer images than is typically required. As direct result, our algorithm is extremely efficient and capable of performing the reconstructions on a time scale suitable for most uses. Further, the proposed algorithm is robust in the sense that it degrades gracefully when presented with non-optimal data. Under optimal conditions, the algorithm yields a reconstruction that is accurate to within a few percentage points of ground truth. Under non-optimal conditions however, the algorithm still generates useful and plausible results. While the images required for the reconstruction must be obtained from disparate locations, 360 degree access to the object is not required; frontal and overhead access is sufficient. The assumptions made by our algorithm concerning the object is that the object is a) the object of interest, b) present somewhere within the workspace, and c) segmentable from the background. Understanding that for any autonomous system there must be some method for the system to recognize that a given object is the object of interest, we feel that these assumptions are reasonable and that system still operates with no *a priori* knowledge of the shape, pose, or position of the object. Further, since the purpose of our algorithm is not to solve the notoriously difficult problem of image segmentation, some liberties were taken with the test objects to make the segmentation criteria more tractable.

The balance of this paper progresses as follows: Section II describes the reconstruction algorithm. Section III explains the performance of the algorithm within an ideal simulated environment. Section IV describes the hardware test setup and the real objects that were used for testing. Section V discusses each test case in detail, focusing on both the strengths and weaknesses of the algorithm. The paper is rounded out by Section VI in which we discuss our conclusions and future work.

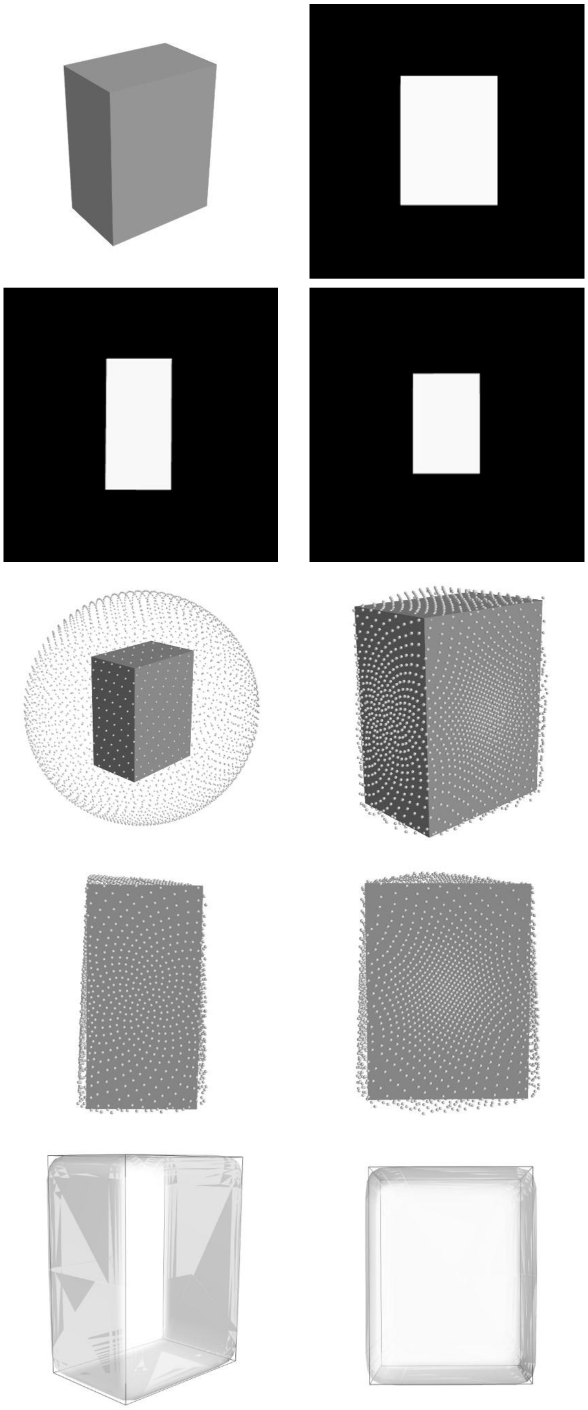
II. Reconstruction Algorithm

Our reconstruction algorithm, which is developed in detail in [17], is presented here in a shorter overview. The reconstruction process is broken down into three main portions: image capture and silhouette calculation, approximation of the object's surface by a three dimensional point cloud, and finally refining the approximation through surface parametrization. The entire process is captured in Figure 1 which shows every step of the reconstruction of a simulated prismatic object.

A. Image Capture and Silhouette Calculation

This initial step is the simplest portion of the algorithm. We capture three images of the object from three disparate viewing locations. Typically, we choose two frontal positions that are separated by 90 degrees and one overhead position. This viewing configuration lends itself to wide coverage of the object and thus good reconstruction accuracy, as will be shown in the in the simulation results. In practice however, this constraint must be relaxed due to the kinematic constraints of the manipulator. Thus our viewing locations in real world experiments are still widely disparate, though not purely orthogonal. It will be seen however, that this does not have a large effect on the reconstruction accuracy.

Once the images of the object are captured, the object is segmented from the background and a binary silhouette image is generated. The method chosen for segmentation is highly dependent on the environment and nature of the the objects. It is beyond the scope of this paper, and indeed our algorithm, to address the problem of segmentation. Instead, we assume that a reasonably decent segmentation of the object is available, and we insure this in our testing by using uniformly colored objects.



a	b
c	d
e	f
g	h
i	j

Fig. 1. The reconstruction process as a step by step simulation. (a) The original shape. (b)-(d) The generated silhouettes. (e) The encompassing sphere of points. (f)-(h) The point cloud after the points have been shrunk to the silhouette boundaries. Error due to perspective projection is clearly seen. (i)-(j) The superquadric that was fit to the point cloud. Original shape shown as a wire frame. Notice the ability of the superquadric to ignore the perspective projection error.

B. Surface Approximation

We chose the algorithm presented in [14] as the basis for our initial surface approximation as it represents a simpler and more efficient method of reconstruction compared to the more traditional voxel based methods. We note that the authors of that work required at least 12 images using this algorithm for an accurate reconstruction. And indeed, with just three images, the result of this algorithm is a very rough approximation of the object's surface (the next phase of our algorithm refines this approximation). Even though this algorithm is efficient, we were able to further improve its performance by removing the iteration step that was present in the original version.

We use the three silhouette images to derive the approximate three dimensional centroid and radius of a bounding sphere that fully encompasses the object. Then, a set of 3000 points is evenly generated across the surface of the sphere. Finally, the position of these points are modified so that the resulting set of points approximates the surface of the object. This is accomplished with the following procedure:

1) Let the center of the camera be $\mathbf{c}_0$.
2) Let the center of the sphere be $\mathbf{x}_0$.
3) Let $\mathbf{x}_i$ be any point in the sphere other than $\mathbf{x}_0$.
4) Let $\mathbf{x}_{i_{new}}$ be the updated position of point $\mathbf{x}_i$.
5) Let the projection of the center of the sphere into the image be $\mathbf{x}'_0$.
6) Then, for each point $\mathbf{x}_i$:
   a) Project $\mathbf{x}_i$ into the silhouette image to get $\mathbf{x}'_i$.
   b) If $\mathbf{x}'_i$ does not intersect the silhouette:
      i) Find the pixel point $\mathbf{p}'$ that lies on the edge of the silhouette along the line segment $\mathbf{x}'_i\mathbf{x}'_0$.
      ii) Reproject $\mathbf{p}'$ into $\mathbb{R}^3$ to get the point $\mathbf{p}$.
      iii) Let the line $\mathbf{c}_0\mathbf{p}$ be $\mathbf{L}_1$.
      iv) Let the line $\mathbf{x}_0\mathbf{x}_i$ be $\mathbf{L}_2$.
      v) Let $\mathbf{x}_{i_{new}}$ be the point of intersection of lines $\mathbf{L}_1$ and $\mathbf{L}_2$
7) Repeat steps 2-6 for each silhouette image.

In the original algorithm [14], the authors accomplish Step 6b in an iterative fashion. Rather than treat each point individually, the authors shrink the entire radius of the sphere at once, for all $\mathbf{x}_i$, by an amount that is dynamically determined based on a point $\mathbf{x}'_j$ that lies closest to, but does not intersect, at least one of the silhouettes. This process is repeated until $\mathbf{x}'_j$ intersects every silhouette. When this happens, point $\mathbf{x}_j$ is removed from computation and the process is repeated for all remaining $\mathbf{x}_i$. The authors state "the step is variable because, when a point is back projected near the silhouette contours, the step is reduced to reach a better approximation of the object model". Step 6b shows that such an approximation is unnecessary because the position of the point can be calculated exactly, in a single step. The geometry of this result is shown in Figure 2. Since the number of points in the cloud is large, eliminating the iteration step amounts to a significant computational savings. The proposed algorithm must visit each point only once for each image, and thus executes in a single pass.

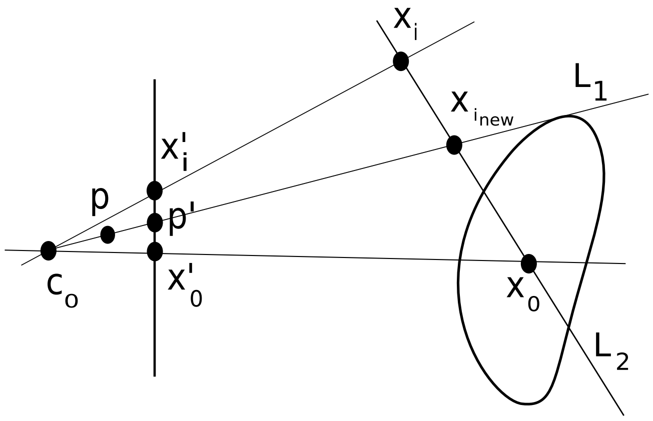


Fig. 2. The geometry of point $\mathbf{x}_{i_{new}}$, which is the intersection of lines $\mathbf{L}_1$ and $\mathbf{L}_2$. The line $\mathbf{L}_2$ is defined by known points $\mathbf{x}_i$ and $\mathbf{x}_0$. The line $\mathbf{L}_1$ is defined by point $\mathbf{c}_0$, which is the camera center, and point $\mathbf{p}$, which is the reprojection of the image point $\mathbf{p}'$ into $\mathbb{R}^3$.

We note that in the perfectly theoretical case, the lines $\mathbf{L}_1$ and $\mathbf{L}_2$ will have an intersection. However, since the point $\mathbf{p}'$ is not sub-pixel accurate, the lines will typically not intersect. Instead, we find the point of nearest intersection of the two lines. This turns out to be the midpoint of the line segment that is the perpendicular distance between the two lines, and therefore has a closed form solution.

C. Surface Parametrization

Since point cloud only very roughly approximates the surface of the object (see Figure 1 (f-h)), we need a way to refine the approximation such that it accurately reflects the geometry of the object. We accomplish this by fitting a superquadric to the point cloud using non-linear least squares minimization. Superquadrics are three dimensional solid models that are capable of modelling a wide variety of shapes with a relatively simple parametrization. A thorough treatment of superquadrics, their derivation, and minimization function can be found in [18]. The motivation to parametrize the object with a superquadric is threefold:

1) It was shown in [3] that superquadrics can accurately model a wide variety of objects typically found in a domestic setting.
2) The 11 parameters of the superquadric immediately yield the shape, size, orientation, and position of the object, and can also be used to quickly find volume and moments of inertia. Thus, the superquadric parameters are ideal for planning grasping maneuvers.
3) The structure of a superquadric and the nature of the minimization routine lends the shape to ignoring localized noise.

We modify the standard superquadric minimization function derived in [18] by adding a weighting factor which has the effect of forcing the superquadric to ignore those points that likely represent a perspective projection artifact. Our modified fitting function is:

$$\begin{aligned} \min_{\Lambda} \Bigg[& w \sum_{i=1}^{n} (\sqrt{\lambda_1\lambda_2\lambda_3}(F^{\epsilon_1}-1))^2 + \\ & \left((1-w) \sum_{i=1}^{n} (\sqrt{\lambda_1\lambda_2\lambda_3}(F^{\epsilon_1}-1))^2 \in F^{\epsilon_1} < 1 \right) \Bigg] \end{aligned} \quad (1)$$

where F is defined as:

$$\begin{aligned} & F(x_w,y_w,z_w) = \\ & \left[\left(\frac{n_x x_w + n_y y_w + n_z z_w - p_x n_x - p_y n_y - p_z n_z}{a_1} \right)^{\frac{2}{\epsilon_2}} + \right. \\ & \left. \left(\frac{o_x x_w + o_y y_w + o_z z_w - p_x o_x - p_y o_y - p_z o_z}{a_2} \right)^{\frac{2}{\epsilon_2}} \right]^{\frac{\epsilon_2}{\epsilon_1}} + \\ & \left(\frac{a_x x_w + a_y y_w + a_z z_w - p_x a_x - p_y a_y - p_z a_z}{a_3} \right)^{\frac{2}{\epsilon_1}} \end{aligned} \quad (2)$$

and 9 of the variables are reduced with the following ZYZ-Euler Angle transformation:

$$\begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \left[\begin{array}{ccc|c} & & & p_x \\ \multicolumn{3}{c|}{R_z(\phi)R_y(\theta)R_z(\psi)} & p_y \\ \hline 0 & 0 & 0 & p_z \\ & & & 1 \end{array} \right] \quad (3)$$

yielding a total of 11 parameters:

$$\begin{aligned} \Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_{11}\} = & \\ & \{a_1, a_2, a_3, \epsilon_1, \epsilon_2, \phi, \theta, \psi, p_x, p_y, p_z\} \end{aligned}$$

The parameters a_1, a_2, a_3 are the width, height, and depth of the object in an object-centered coordinate system. The parameters ϵ_1, ϵ_2 define the shape of the object, and the parameters $\phi, \theta, \psi, p_x, p_y, p_z$ are the 6 independent elements of the transformation of the object-centered coordinate system with respect to the world. The variable ω in Equation 1 is the weighting factor we have added to aid in projection shadow rejection. It works by placing a penalty on points that lie inside the superquadric surface. We use an empirically determined value $w = 0.2$, thus placing an 80% weight on the error of points that lie within boundary of the superquadric surface. In effect we force the superquadric to be as large as possible while minimizing any extension beyond the boundary of the points. Since the point cloud will never be smaller than the object, this is a valid and effective operation.

The result of fitting a superquadric to the point cloud approximation is illustrated in Figure 1 (i-j). Notice that the fitted shape accurately represents the original object and fully ignores the perspective projection shadows that are present in the point cloud. Furthermore, the superquadric ignores all localized noise, though in this simulated case the only noise is due to quantization error.

III. Simulation

We developed a simulation environment which allows us to test the reconstruction algorithm under ideal conditions i.e. no noise, perfect segmentation, perfect camera calibration. We can simulate a wide variety of simple geometric shapes and capture images of the object from any arbitrary positions. The captured images are then used in the reconstruction routine and the results overlayed on the ground truth as seen in Figure 1. In order to quantify the reconstruction accuracy, we chose three orthogonal viewing directions: two frontal separated by 90 degrees and one from overhead. We then compared the 11 recovered superquadric parameters to the known ground truth. We also compare the volume of the recovered superquadric to the known volume of the shape. We define the volume fraction as

$$v_f = \frac{Volume_{recovered}}{Volume_{truth}}$$

Table I lists the results for a few simulated shapes. When comparing the values in the table, care should be taken when interpreting the values for orientation ϕ, θ, ψ. Since the objects are symmetric about certain axes, there is more than one equivalent orientation. It is readily seen from the table that the algorithm is capable of exceedingly accurate reconstructions. Though the algorithm has tendency to overestimate the size of the object (as seen by the volume fraction), most parameters are off by only a few percent of ground truth.

Furthermore, the execution time of the reconstruction algorithm is typically between 0.25 and 0.4 seconds on a 2.53 GHz CPU and is dependent on the time taken for the non-linear solver to converge. This is a huge savings compared to the ~100s in [8] and is negligible when compared to the time it would take a robotic manipulator to capture the images of the object.

IV. Experimental Setup

A. Hardware

For real world testing, our hardware configuration consists of an Axis-207MW wireless network camera mounted in an eye-in-hand configuration at the end effector of a Kuka KR6-2 six axis industrial manipulator. The robot and camera platform is shown in Figure 3. The test object is placed in a random location on a table which is in the robot's workspace. The robot is programmed to observe the scene from three locations. Due to kinematic constraints, these locations are not mutually orthogonal but they approach such a condition. The three images captured by the robot during one of the test runs are shown in Figure 4. From these images, one can see the nature of the disparate viewing locations; the frontal views are not perfectly horizontal nor is the overhead view perfectly vertical. We note that during reconstruction, the robot is not informed of the location of the object on the table. Rather, it is merely assumed that the object is visible in all three images of the scene; the location of the object in the scene is determined as part of the reconstruction (the p_x, p_y, p_z parameters of the superquadric).

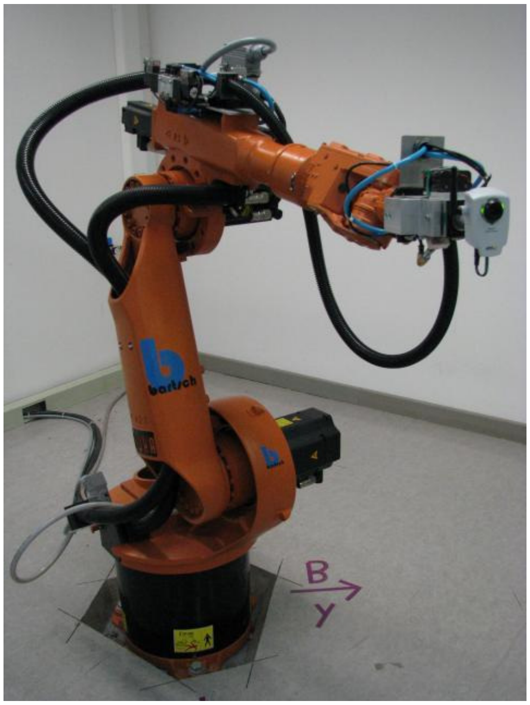


Fig. 3. The robot and camera platform.

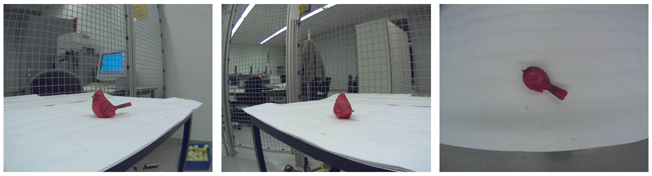


Fig. 4. Three images captured by the robot during a test run. The nature of the disparate viewing locations can be inferred from these images.

B. Test Objects

We tested the algorithm on four different objects: a prismatic battery box, an elongated cylinder composed of two stacked cups, a ball of yarn, and a small cardinal statue. The first three objects represent the range of geometric shapes frequently encountered in domestic settings: cylindrical, prismatic, and ellipsoidal. It is expected that the algorithm will achieve accurate reconstructions for these shapes. The last object is amorphous and is included to test the robustness of the algorithm when presented with data that is incapable of being accurately described by the model. In all cases, the test objects are red in color to ease the task of segmentation and facilitate reliable silhouette generation. Again, it is not our aim to solve the broader machine vision problem of segmentation. The four objects tested are shown in Figure 5.

V. Experimental Results

This sections discusses the reconstruction results of each of the test objects mentioned in Section IV-B. Each of the cases (with the exception of the cardinal) is accompanied by a rendered figure which shows the ground truth overlayed by the calculated reconstruction. The ground truth is shown

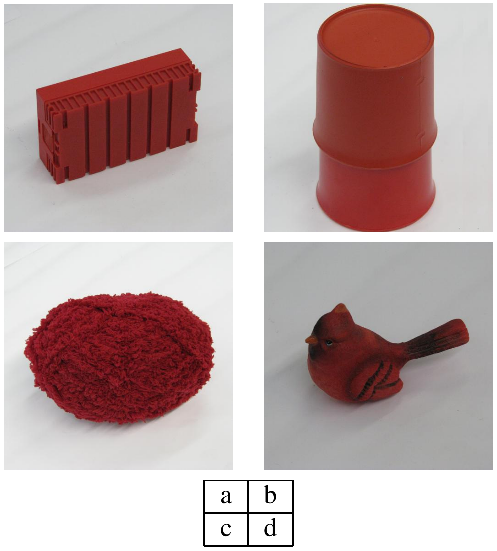


Fig. 5. The four real-world test objects. (a) A prismatic battery box. (b) A stack of cups. (c) A ball of yarn. (d) A cardinal statue.

as a wire frame and the reconstruction as an opaque surface. The accuracy is discussed from a qualitative perspective in the frame of whether or not the reconstructed shape could be used to plan a grasping maneuver. The numerical results, presented in same fashion as the simulated reconstructions in Table I, are given in Table II.

When interpreting the accuracy of the results, it must be kept in mind that there are several sources of error that are compounded into these results which are not present in the simulation:

- Uncertain camera calibration: intrinsics and extrinsics
- Robot kinematic uncertainty
- Imperfect segmentation
- Ground truth measurement uncertainty

The last bullet is particularly noteworthy. Since the object is placed randomly in the robot's workspace the only practical way of measuring the ground truth position and orientation is to use a measuring device attached to the end effector of the robot. Though more accurate than attempting to manually measure from the robot base, the error is compounded by both machine inaccuracy and human error.

We must point out, that despite all of these sources of error, the accuracy of most reconstructions is within a couple millimeters of ground truth. Compare this with the results in [6], where a reconstruction with over 200 images resulted in an error of 10 millimeters.

A. Battery Box

The reconstruction of the battery box, shown in Figure 6, was overall the most accurate of all the reconstructions. It is clearly seen that the model correctly captures the height, width, depth, and shape of the battery box with only a slight deviation in position and orientation. The numerical values of the results in Table II confirm this. Though this reconstruction has the largest deviation from unity for the volume fraction, there is no question that the resultant model can be used as a model for grasp planning. Furthermore, the accuracy of the shape representation opens the door for other possibilities such as task inference based on shape and/or appearance.

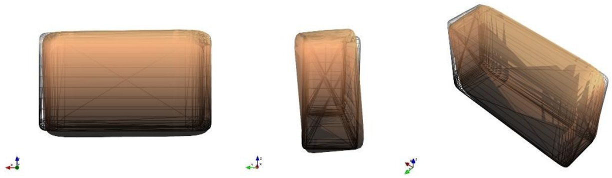


Fig. 6. The reconstruction of the battery box. Ground truth is shown as a wire frame.

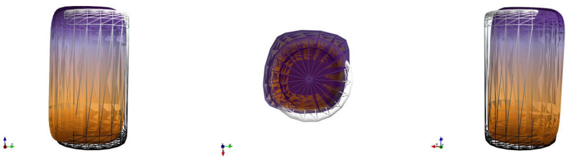


Fig. 7. The reconstruction of the stack of two cups. Ground truth is shown as a wire frame.

B. Cup Stack

The reconstruction of the stack of cups, which would be accurately approximated as a cylinder, did not achieve high accuracy in all parameters. Namely, the shape parameters ϵ_1, ϵ_2 were inaccurate with respect to ground truth. Shown in Figure 7, it is seen that the reconstructed shape is bordering on prismatic rather than cylindrical. This is a byproduct that stems from the nature of perspective projection shadows and can be eliminated by either more views, or a view perfectly in line with the major axis. The rest of the reconstruction parameters (height, width, depth, position) however, are all accurate, with only the orientation deviating slightly. We note again that this error stems from a combination of the many compounded error sources mentioned in the beginning of this section.

Despite the non-cylindrical shape of the object, we believe that the overall size and position are still accurate enough to attempt a grasping maneuver based on the model parameters. A robot designed to operate in a domestic setting should have no problem with the margin of error present in this reconstruction.

C. Yarn Ball

The yarn barn reconstruction, Figure 8, is nearly as accurate as the battery box. There is slight deviation in the orientation similar to the two previous cases. The yarn ball was the largest of all objects tested at 150mm in length and 100mm in diameter. And though such an object is likely too large to be grasped by most domestic sized manipulators, the accuracy is sufficient to plan the maneuver provided the manipulator has sufficient capacity. We note that the size parameters a_1, a_2, a_3 can be directly used as a criteria to determine if an object is within capability limits of the manipulator.

TABLE I

SIMULATION RESULTS

Shape		a_1	a_2	a_3	ϵ_1	ϵ_2	ϕ	θ	ψ	p_x	p_y	p_z	v_f
Prism	Truth	0.4	0.5	1.25	0.0	0.0	1.571	1.571	-1.571	0.0	0.0	0.0	
	Reconstr.	0.422	0.518	1.265	0.1	0.172	-1.571	1.571	-1.571	-0.009	0.007	0.003	1.087
Cylinder	Truth	1.0	1.0	1.5	0.0	1.0	1.571	1.571	0.0	0.0	0.0	0.0	
	Reconstr.	0.987	0.993	1.544	0.186	0.724	-1.575	1.573	0.013	-0.01	0.0	0.007	1.088
Sphere	Truth	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
	Reconstr.	0.96	0.96	0.968	0.793	0.781	0.49	-0.005	-0.188	-0.012	0.005	0.004	1.092

TABLE II

EXPERIMENTAL RESULTS

Shape	units-mm	a_1	a_2	a_3	ϵ_1	ϵ_2	ϕ	θ	ψ	p_x	p_y	p_z	v_f
Battery Box	Truth	30	15	52.5	0.1	0.1	0.0	1.57	0.0	880	-924	865	
	Reconstr.	32.9	16.9	51.6	0.2	0.2	3.12	1.56	0.10	878.4	-924.6	864.9	1.18
Cup Stack	Truth	34	34	60	0.1	1.0	0.0	0.0	0.0	898	-915	892	
	Reconstr.	41.0	37.8	61.2	0.3	1.4	-0.30	3.10	-2.60	893.8	-917.5	894.8	1.13
Yarn Ball	Truth	50	50	75	0.7	1.0	-0.17	1.53	0.0	898	-915	855	
	Reconstr.	57.1	51.5	74.4	0.6	1.1	3.07	1.52	0.86	893.9	-912.7	854.1	1.14
Cardinal Statue	Truth	25^1	25^1	30^1	*	*	*	*	*	898	-915	862	
	Reconstr.	24.0	18.4	29.5	0.1	0.4	-9.35	-0.82	6.01	892.0	-908.9	867.3	*

[1] Approximation based on the bounding box that would encompass the bulk of mass.

*The value has no meaning in the context of this shape.

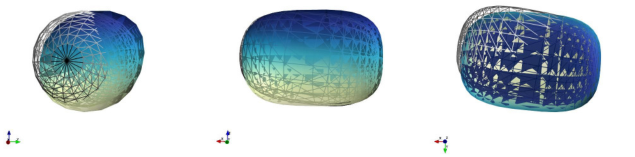


Fig. 8. The reconstruction of the yarn ball. Ground truth is shown as a wire frame.

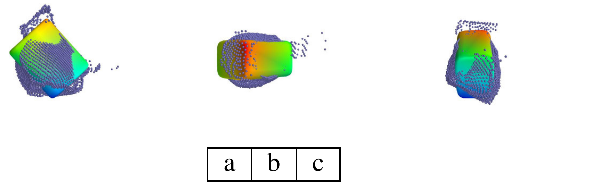


Fig. 9. The reconstruction of the cardinal statue. (a) Side view. (b) Top view. (c) Rear view. The points are the results of the surface approximation phase. The opaque surface is the fitted superquadric. A perspective projection shadow is clearly evident in the bottom right corner of the point cloud in (c).

D. Cardinal Statue

We included the figurine of the cardinal to test how our algorithm performs when provided with data that does not fit well with our reconstruction model and assumptions. This test case is shown in Figure 9. Since it would be difficult to model the ground truth as a wire frame, the results of the surface approximation phase of the algorithm are used instead. From the figure, it is clear that there would be no way to infer from the box shape that is the final reconstruction that the original object was a bird. However, it is interesting to note that the reconstruction is very close to what a human would likely provide if asked to select a bounding box that best describes the object. That is, the reconstructed shape does an excellent job of capturing the bulk form of the statue despite the fact that the data is ill formed with respect to our modelling assumptions. It is not a stretch of the imagination to think that a grasp could be accurately planned for this object using the reconstructed shape.

This example shows that, even when the object does not take a form that can be accurately modelled by a single superquadric, our proposed algorithm still generates useful results.

VI. CONCLUSIONS AND FUTURE WORK

We have shown that by using three images of a novel object taken from disparate locations, our algorithm can calculate a parametrized model of that object with sufficient accuracy to allow for the planning of grasping and manipulation maneuvers. In contrast to other efforts in the literature, the proposed algorithm requires fewer images, significantly less computation time, and yields an overall higher reconstruction accuracy. Furthermore, the parameters of the reconstructed model can be directly used for grasp planning. No further analysis of the shape or time consuming statistical methods are necessary. We feel that the results presented here merit further investigation and research into this approach of novel object recognition.

Our future plans include integrating a grasping algorithm based on the reconstructed superquadric parameters and testing how the algorithm behaves when the viewing locations become less and less disparate. We also plan to investigate what can be done to increase the accuracy to an acceptable level when such a condition arises, such as incorporating the appearance data that is discarded by using only silhouettes. That is, we will attempt to incorporate structure that can be inferred from the raster images with the structure of the superquadric to improve the accuracy and overall robustness. We also plan to investigate incorporating other sensory information, such as laser range finder, to augment the abilities of the optical reconstruction by providing depth information that cannot be recovered due to projection shadows.

REFERENCES

[1] D. Kim, R. Lovelett, and A. Behal, "Eye-in-Hand Stereo Visual Servoing of an Assistive Robot Arm in Unstructured Environments," *International Conference on Robotics and Automation*, pp. 2326–2331, May 2009.

[2] S. Effendi, R. Jarvis, and D. Suter, "Robot Manipulation Grasping of Recognized Objects for Assistive Technology Support Using Stereo Vision," *Australasian Conference on Robotics and Automation*, 2008.

[3] M.J. Schlemmer, G. Biegelbauer, and M. Vincze, "Rethinking Robot Vision - Combining Shape and Appearance," *International Journal of Advanced Robotic Systems*, vol. 4, no. 3, pp. 259–270, 2007.

[4] F. Liefhebber and J. Sijs, "Vision-based control of the Manus using SIFT," *International Conference on Rehabilitation Robotics*, June 2007.

[5] D. Kragic, M. Bjorkman, H. I. Christensen, and J. Eklundh, "Vision for robotic object manipulation in domestic settings," *Robotics and Autonomous Systems*, vol. 52, pp. 85–100, 2005.

[6] K. Yamazaki, M. Tomono, T. Tsubouchi, and S. Yuta, "3-D Object Modelling by a Camera Equipped on a Mobile Robot," *International Conference on Robotics and Automation*, Apr. 2004.

[7] K. Yamazaki, M. Tomono, T. Tsubouchi, and S. Yuta, "A Grasp Planning for Picking up an Unknown Object for a Mobile Manipulator," *International Conference on Robotics and Automation*, 2006.

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

[9] C. R. Dyer, *Volumetric Scene Reconstruction From Multiple Views*, pp. 469–489. Foundations of Image Understanding, Boston: Kluwer, 2001.

[10] A. Laurentini, "The Visual Hull Concept for Silhouette-Based Image Understanding," *Transactions of Pattern Analysis and Machine Intelligence*, vol. 16, Feb. 1994.

[11] R. Szeliski, "Rapid Octree Construction from Image Sequences," *CVGIP: Image Understanding*, vol. 58, July 1993.

[12] K. Shanmukh and A. Pujari, "Volume Intersection with Optimal Set of Direction," *Pattern Recognition Letters*, vol. 12, pp. 165–170, 1991.

[13] H. Noborio, S. Fukuda, and S. Arimoto, "Construction of the Octree Approximating Three-Dimensional Objects by Using Multiple Views," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, Nov. 1988.

[14] V. Lippiello and F. Ruggiero, "Surface Model Reconstruction of 3D Objects From Multiple Views," *International Conference on Robotics and Automation*, pp. 2400–2405, May 2009.

[15] H. Nguyen, C. Anderson, A. Trevor, *et al.*, "El-E: An Assistive Robot that Fetches Objects from Flat Surfaces," *HRI Workshop on Robotic Helpers: User Interaction Interfaces and Companions in Assistive and Therapy Robots*, 2008.

[16] A. Saxena, J. Driemeyer, and A. Ng, "Robotic Grasping of Novel Objects using Vision," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.

[17] S. C. Colbert, R. Alqasemi, and R. V. Dubey, "Efficient Shape and Pose Recovery of Unknown Objects from Three Camera Views," *In Press: International Symposium on Mechatronics and its Applications (ISMA) 2010.*, May 2010.

[18] A. Jaklic, A. Leonardis, and F. Solina, *Segmentation and Recovery of Superquadrics*, vol. 20 of *Computational Imaging and Vision*. Kluwer Academic Publishers, 2000.