

Automated Modeling and Robotic Grasping of Unknown Three-Dimensional Objects

Gary M. Bone, Andrew Lambert and Mark Edwards

Abstract – This paper describes the development of a novel vision-based modeling and grasping system for three-dimensional (3D) objects whose shape and location are unknown *a priori*. Our approach integrates online computer vision-based 3D object modeling with online 3D grasp planning and execution. A single wrist-mounted video camera is moved around the stationary object to obtain images from multiple viewpoints. Object silhouettes are extracted from these images and used to form a 3D solid model of the object. To refine the model, the object's top surface is modeled by scanning with a wrist-mounted line laser while recording images. The laser line in each image is used to form a 3D surface model that is combined with the silhouette result. The grasp planning algorithm is designed for the parallel-jaw grippers that are commonly used in industry. The algorithm analyses the solid model, generates a robust force closure grasp, and outputs the required gripper position and orientation for grasping the object. The robot then automatically picks up the object. Experiments are performed with two real-world 3D objects, a metal bracket and a hex nut. The shape, position and orientation of the objects are not known by the system *a priori*. The time required to compute an object model and plan a grasp was less than 4 s for each object. The experimental results demonstrate that the automated grasping system can obtain suitable models and generate successful grasps, even when the objects are not lying parallel to the supporting table.

I. INTRODUCTION

Typically, for robotic grasping to be successful the shape and location of the object must be known in advance. This lack of adaptability has two main disadvantages. First, expensive fixtures or other devices must be used to position and orient the object prior to grasping. Second, the robot is limited to objects whose shape is already known. A vision-guided grasping system that could adjust to the position, orientation and shape of the object would provide the needed adaptability. Such a system would be useful in both manufacturing and service applications of robots. Robots equipped with vision systems capable of grasping objects whose planar shape and location are unknown have existed for many years (e.g. [1]). However, the automated grasping of three-dimensional (3D) objects whose shape and location are unknown *a priori* remains a challenging and unsolved problem, and is the subject of this paper. Its solution necessitates online object modeling, grasp planning and

grasp execution. Note that this is distinct from the problem of recognizing the object from a predefined set or database of known objects and then grasping it. Stansfield [2] presented a system for grasping 3D objects with unknown geometry using a Salisbury robotic hand. Each object was first placed on a motorized table. The object was then rotated and translated under a laser scanner to generate a set of 3D points. These were combined to form a 3D model. The model formed the input to an expert system that planned the grasp. Experimental results were presented for several objects. A system using a parallel-jaw gripper and machine vision intended for picking an object from the top of a pile was described by Taylor, Blake and Cox [3]. They used a wrist-mounted camera to first scan the pile for the highest object. They described a method for planning a grasp for this object based on images from several viewpoints. However, the only experimental grasping results they included were for a single object (a potato) located on a table. The system presented by Trobina and Leonardis [4] used two range sensors to model groups of objects. The models, consisting of planar patches, were used to plan grasps for a parallel-jaw gripper. The tallest object was picked up and removed first, a new 3D model constructed, and the process repeated until all objects had been removed. Only one experimental result for a set of household objects (milk carton, coffee cup, etc.) was included. Namiki *et al.* [5] presented a system for high speed grasping using visual and force feedback with a 1 ms sampling rate. Their system was able to grasp a moving object in about 0.5 s, but the robotic hand they used had to be pre-shaped according to the expected shape of the object. Only rectangular block and spherical shapes were considered. The object motion was also limited to a plane and the final position of the object within the hand could not be determined. A system combining a grasping simulator with a real-time visual tracking system was described by Kragic, Miller and Allen in [6]. The tracking system could accurately estimate the 3D pose of the object while the simulator was used to generate a suitable grasp. However, both the tracking system and grasping simulator required a predefined CAD model of the object in order to function. A humanoid robot equipped with a stereoscopic light stripe scanner and a prosthetic hand was presented by Taylor and Kleeman [7]. Their robot could reliably track objects through clutter by employing color, texture and edge information. It successfully located and picked up a yellow box, given only the information that the object was yellow

Manuscript received September 14, 2007.

Gary M. Bone, Andrew Lambert and Mark Edwards are with the Mechanical Engineering Department, McMaster University, Hamilton, Canada, L8S 4L7. (e-mail: gary@mcmaster.ca, lamberaj@mcmaster.ca and edwardmj@mcmaster.ca)

and had a box shape. Jang et al. [8] described a novel approach to accessibility analysis for grasping and manipulation tasks. The workspace was modeled using a stereo camera and objects to be grasped were recognized using their plane features and pre-existing object models. Obstacles were modeled by multi-resolution octrees. Their vision system required the environment to be strongly textured to obtain the 3D information. Collision-free grasps were planned for a parallel-jaw gripper using a novel visibility-based approach. The grasping of a drink container inside a refrigerator was used as a demonstration. Sanz *et al.* [9] presented a vision-based object handling system for industrial applications. A 2D model of each unknown object was generated from a single overhead image. The minor axis of a fitted ellipse was used to plan the grasp for a parallel-jaw gripper. Their system was limited to the case where the gripper jaws were parallel to the world Z axis (*i.e.* the jaws are perpendicular to the table). Results were included for small tools (*e.g.* pliers) and for food items (*e.g.* lettuce). Another vision-guided robotic arm was developed by Kragic and Björkman [10]. They integrated monocular and binocular visual cues from five cameras to provide robust 3D object information. Their model-free approach was applicable to well-textured, unknown objects. A three-fingered hand equipped with tactile sensors was used to perform the grasp in an interactive manner.

This paper describes the development of a novel vision-guided grasping system for grasping 3D objects whose shape and location are unknown *a priori*. The system employs online 3D object modeling, grasp planning and grasp execution. A robotic arm equipped with a single wrist-mounted video camera, a wrist mounted line laser, and a parallel-jaw gripper is used. The method for generating a 3D object model online is described first. Next, the grasp planning algorithm is presented. Following a description of the experimental setup, the results of automated grasping experiments performed with two real-world objects are discussed, and conclusions are drawn.

II. ONLINE 3D OBJECT MODELING USING COMPUTER VISION

A. Introduction

The 3D object model will be obtained using both silhouette and structured-light based methods. Silhouette-based methods (also known as the “method of occluding contours”) were first introduced by Martin and Aggarwal [11]. With these methods, a single calibrated camera and silhouettes of the object (*i.e.* binary classifications of the image into object and background) from several viewpoints are used to generate the object model. They have the advantages of: simplicity, speed, being guaranteed to produce a closed surface, and robustness. They have the disadvantage that at best they can only model the “visual hull” of the object. The volume derived from the visual hull is less than or equal to the volume of the convex hull of the

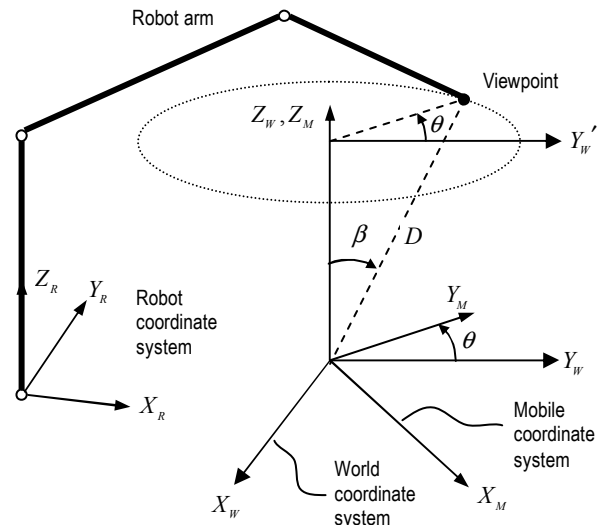


Fig. 1. Definitions of the robot, mobile and world coordinate systems (Note: Y_W and $Y_{W'}$ are parallel).

object and greater than or equal to the true shape of the object. In structured-light methods, a light pattern of known geometry is projected onto the object and an image is acquired with a calibrated camera. The deformation of the light pattern on the surface of the object allows one to determine the object’s shape. These methods have the advantage of being able to simply and quickly model 3D shape from single 2D images. They have the disadvantage of only being able to model sections of the object where the light pattern reaches.

B. System Calibration

In order to be grasped, the object must be located within the field of view on a worktable. The system must also be calibrated once before it may be used. The calibration procedure produces mappings between the three coordinate systems defined in Fig. 1 using a calibration pattern and Tsai’s method [12]. A similar calibration procedure is used to produce a mapping between the 2D image sensor and the vertical plane that will be illuminated by the wrist-mounted line laser.

C. Centering Object within Camera Field of View

As the object may be located anywhere on the table, the first task is to locate the object’s position in order to center the camera overtop. This ensures that when images are captured, the object is entirely visible in the image and not cut off. The steps for locating the object are as follows:

- 1) Move the camera above the world origin, with $D \approx 450\text{mm}$ and $\theta \approx \beta \approx 0$ (ref. Fig. 1).
- 2) Capture the image from the camera.
- 3) Detect the edges of the object in the image using a mean filter followed by a Sobel filter and thresholding.
- 4) Compute the mean values of the X and Y edge coordinates in world coordinates, and move the camera above this location.

D. 2D Image Acquisition and Silhouette Generation

The first step in generating a model online is to acquire and process 2D images from several viewpoints. In general, acquiring more images produces a more accurate model at the cost of longer execution time. We found that nine viewpoints were an effective compromise. The first viewpoint places the camera above the mobile coordinate system origin, with $D \approx 300\text{mm}$ and $\theta \approx \beta \approx 0$. For the remaining viewpoints the camera is moved incrementally around a circle with $\theta \approx -110^\circ, -80^\circ, -50^\circ, -20^\circ, 10^\circ, 40^\circ, 70^\circ$ and 100° . The steps of the 2D image acquisition and processing are as follows:

- 1) Move the camera to the first viewpoint and pause the motion.
- 2) Capture the image from the camera.
- 3) Detect the edges in the image using a mean filter followed by a Sobel filter and thresholding.
- 4) Obtain the contour of the object using the boundary-following algorithm from [13].
- 5) Fill in the object contour to complete the silhouette for the current viewpoint.
- 6) If at the last viewpoint then stop, otherwise move to the next viewpoint and continue with step 2.

Note that in steps 3-5, edge detection, boundary-following and pixel filling are used to separate the image pixels belonging to the object from those belonging to the background. This is reliable since significant pixel intensity changes normally exist near the edges of the object and the background.

E. Generation of the 3D Silhouette Model

With silhouette-based modeling the information from several image silhouettes is combined to form a solid model of the object. For the case of a single silhouette, the object must lie within the polyhedral cone formed by the silhouette and the camera focal point. The process of intersecting a series of these cones forms a model that is progressively closer to the real object. An example for an ellipsoid object and four viewpoints is given in Fig. 2. A key issue is how to represent the volume elements of the model. In this paper we extend the beam-based approach presented in [14]. The beam-based approach has the advantage of a simpler and

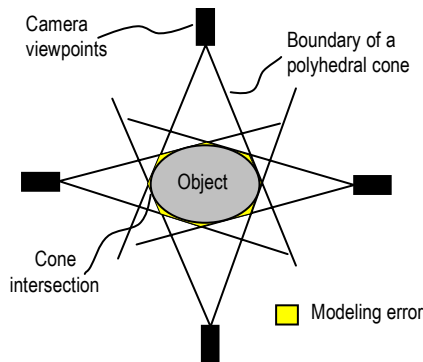


Fig. 2. Modeling example (top view is shown).

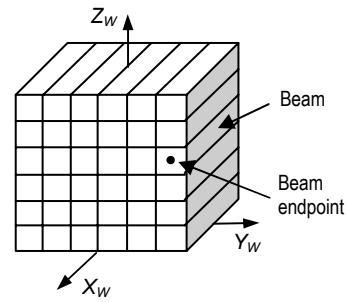


Fig. 3. Example of a stack of X axis aligned beams used in the volume representation.

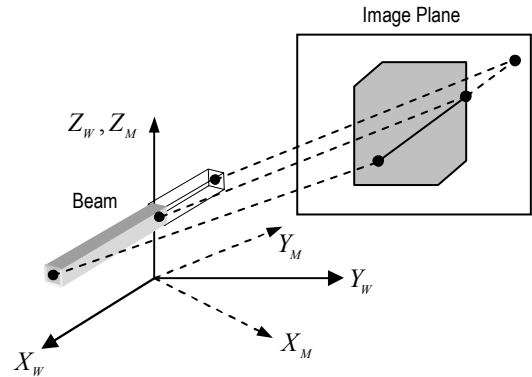


Fig. 4. Shortening of a beam using an image silhouette.

easier to compute intersection test compared with the more common octree method. The steps of the 3D modeling algorithm are as follows:

- 1) The object is first represented by two stacks of square cross-section beams. The first stack is aligned with the X_w axis (see Fig. 3), while the second is aligned with the Y_w axis.
- 2) For each object silhouette, starting with the first beam:
 - a. Transform the world coordinates of the beam endpoints into the current mobile coordinate system.
 - b. Project the beam endpoints onto the image plane of the camera.
 - c. If either of the projected endpoints lies outside the silhouette then shorten the beam incrementally and repeat (a) and (b) until both projected endpoints lie on the silhouette or the beam length equals zero.
 - d. Apply steps (a) to (c) for the remaining beams.

An example of a beam being shortened using a single silhouette is given in Fig. 4. The silhouette modeling procedure introduced here has the limitation of not accurately modeling the top surface of objects. Since the images are acquired from elevated positions, the object's top surface is contained within the silhouette. This results in excess, cone-shaped volumes present on the top surfaces of the models. To correct this, a structured-light model is used to accurately model the top surface, as discussed in section F.

F. Structured-Light Image Acquisition and Model Generation

Structured-light modeling utilizes a projected pattern of light, in this case a red line laser that is mounted on the robot end effector. In order for the entire surface to be modeled, the robot arm is used to scan the laser across the object while the camera acquires images. The robot wrist is reoriented such that a vertical plane of laser light is cast on to the object subject, utilizing the vertical plane calibration to relate the image co-ordinates to world coordinates. The end effector speed and the camera frame rate are known, allowing determination of the laser plane separation between each image. The steps of the structured-light image acquisition and model generation are as follows:

- 1) The object's X_W coordinate boundary is determined from the silhouette model, and the laser begins at this point oriented parallel to the Y_W axis.
- 2) The robot then scans a set distance along the X_W direction while the camera acquires images.
- 3) The red laser line is extracted from the images using thresholding, and the portions of the line that contact the table are eliminated.
- 4) For each image, the vertical calibration data is used to relate the laser line's image coordinates to Y_W and Z_W coordinates.
- 5) The world X_W coordinate of each image's laser plane is found from the computed laser plane separation between consecutive images.

The structured-light model of the object's surface is stored as a set of points.

G. Combination of Silhouette and Structured Light Models

The silhouette model contains excess volume that will be removed using the structured-light representation of the object's top surface. The approach is to use the structured-light surface representation like a cookie cutter, to cut the desired shape out of the silhouette model. The procedure to combine the two models is as follows:

- 1) Project both models into the $Y_W - Z_W$ plane.
- 2) Starting with the first silhouette model beam, if either endpoint lies above (ie. greater Z_W position) the corresponding structured-light model point, shorten the beam incrementally until both projected endpoints agree with the structured-light model or the beam length becomes zero.
- 3) Apply step 2 to the remaining beams.
- 4) Project both models into the world X-Z plane and repeat steps 2 and 3.

Note that repeating the cutting process in additional planes produces greater detail at the cost of longer execution time. We found using two planes to be sufficient.

III. ONLINE GRASP PLANNING

A. Finding Potential Grasping Surfaces

A parallel-jaw gripper with flat jaws can be used to grasp many objects, including those without flat surfaces. However, it is best suited to objects with approximately flat parallel surfaces. Our grasp planning algorithm is currently limited to this case. The solid model consisting of the beam endpoints forms the input to the first planning phase. This phase involves finding nearly flat surface patches that could be used for grasping. The algorithm steps are as follows:

- 1) Form triangular facets from the surface points.
- 2) Calculate the outwardly pointing normal vectors for the set of facets.
- 3) Calculate the distances to the origin of the planes containing each facet.
- 4) Form a subset containing the first facet.
- 5) Determine the angle between the normal for this facet and the normal for another facet from the set.
- 6) If the angle is less than a flatness tolerance, then calculate the distance between this facet and the first facet (using the values from step 2). If this distance is less than a distance tolerance then add this facet to the subset.
- 7) Repeat steps 5 and 6 until no more facets satisfy the conditions of step 6.
- 8) The subset of facets forms a nearly flat surface patch. Assign a distinct label to the members of this subset.
- 9) Starting with the first unlabeled facet, repeat steps 4-8.
- 10) Repeat step 9 until all facets are labeled.

B. Planning a High Quality Force Closure Grasp

In the second planning phase, the objective is to find the pair of nearly flat surface patches where the gripper jaws should squeeze the object to produce a high quality grasp. An ideal grasp should be insensitive (or robust) to position and orientation errors prior to grasping, and to forces and moments applied to the object after it has been grasped. A low sensitivity to the coefficient of friction is also desirable. Consider the contact between one gripper jaw and the object. Jaw contact with a nearly flat surface patch with a large area will be better able to resist applied moments, and less sensitive to position errors, than contact with a smaller patch. However, requiring that both grasping surface patches have a large area does not guarantee a robust grasp. An example is illustrated in Fig. 5. Cross-sections of a prismatic object and the gripper jaws are shown. F_N is the normal force and μ is the coefficient of friction. The object will slip from its initial orientation in Fig. 5a to the orientation shown in Fig. 5b if the moment created by the normal forces is greater than the moment due to the friction forces. If this condition is not true the object will keep its initial orientation (in 5a). This grasp is therefore not robust to orientation errors, applied moments or the coefficient of friction. In Fig. 6 a robust grasp is shown for the same object. With this grasp, if the object is tilted (due to an

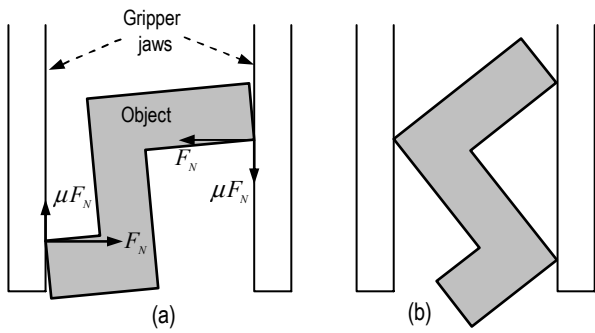


Fig. 5. Example of a non-robust grasp.

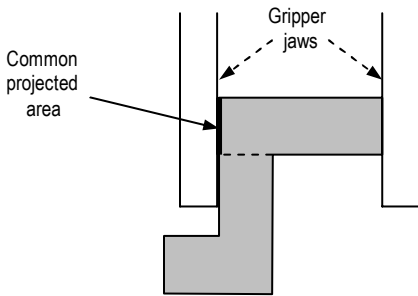


Fig. 6. Robust grasp for the object shown in Fig. 5.

applied moment or an initial orientation error) the normal forces will tend to return it to the orientation shown. The difference between the grasps of Figs. 5 and 6 lies in the “common projected area” of their grasping surface patches. With the Fig. 5 grasp, the patches have no area in common when projected onto each other, whereas the common projected area for the Fig. 6 grasp is as indicated in the figure.

Although we would prefer that the contact between the jaw and object surfaces to be a plane-plane frictional contact (that could be modeled by three frictional point contacts) this cannot be guaranteed in practice. Since the gripper jaws will be covered by a layer of rubber, in the worst case of a single point of contact between the jaw and object the contact can be modeled as a soft finger contact. Then a grasp of the type shown in Fig. 6 will be (in the worst case) a two opposing soft-finger grasp that Nguyen [15] has proven to be a 3D force closure grasp.

Another important issue is robustness to robot positioning error in the tool Z (or approach) direction. To avoid collisions with the worktable or other features of the object, the gripper jaws should not be placed too low on the grasping surface patches. At the other extreme, the object could be dropped if the jaws are placed too high. Placing the tips of the jaws close to the middle of the patches will make the grasp robust to large errors in robot tool Z positioning.

The preceding observations led to the following algorithm for the second phase of the grasp planning:

- 1) Calculate the areas of the patches in the set obtained in the first planning phase.

- 2) Delete all patches whose area is less than a threshold.
- 3) Calculate a normal vector for each remaining patch by averaging the normals of the facets belonging to it.
- 4) Select a patch from the set.
- 5) Determine the angles between the normal for this patch and the normals of the other patches in the set.
- 6) Record all pairs of patches from step 5 that are within the parallelism tolerance.
- 7) Repeat steps 4-6 for the other members of the set.
- 8) Calculate the common projected area for each of the pairs of patches recorded in step 6.
- 9) Select the pair of patches with the largest common projected area to be the desired grasping surface patches (DGSP).
- 10) Compute the centroids of the DGSP. Compute the midpoint of the line segment joining the two centroids.
- 11) Compute the transformation required to make the jaw surfaces parallel to the DGSP, and the jaw tip center (defined in Fig. 7a) coincident with the midpoint from step 10. An example is shown in Fig. 7b. This transformation will be in world coordinates.
- 12) Transform the result from step 11 into robot coordinates, and calculate the commanded robot locations for approaching, grasping and picking up the object.

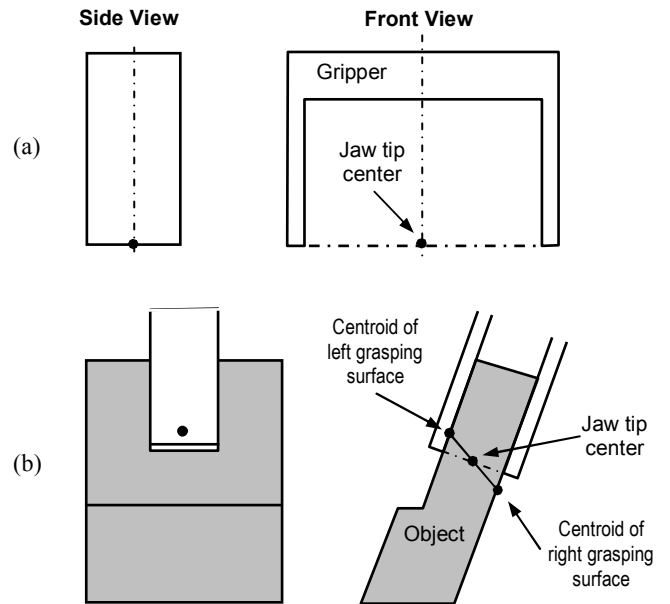


Fig. 7. (a) Definition of the jaw tip center. (b) Jaw placement example for an asymmetric object.

IV. AUTOMATED GRASPING EXPERIMENTS

A. Experimental Setup and Procedure

The hardware used for the experiments consists of a 1.8GHz P4 PC; and a CRS F3 robot equipped with a Robohand RPL-4 parallel-jaw gripper, a Point Grey Research Dragonfly2 video camera with 8.5mm lens, a 5 mW red line laser, and a LED-based light (see Fig. 8). The motion commands are transmitted from the PC to the robot controller via an RS-232 serial connection. The video camera is mounted parallel to the gripper. The object to be grasped is placed on a black pad to improve the reliability of the edge detection. Two objects were tested: a metal bracket and a hex nut (see Fig. 9). The volume representation used for the online modeling consisted of a 120 mm cube made up of 7,200 pillars with end dimensions of $2 \times 2 \text{ mm}^2$. Note that a larger volume representation could be used if warranted by object size. In the experiments the test objects were placed at various positions and orientations on the table.

B. Results and Discussion

Typical experimental results are given in Figs. 10-11 and Table 1. To demonstrate that the system works when the grasping surfaces are not vertical, the nut was deliberately tilted out of the plane of the worktable by placing it on top of a screw (see Fig.11a), and a tilted orientation was used for the bracket (see Fig.11d). The online generated model of the bracket (Fig. 10) demonstrates that the system can model it effectively for the grasping application. It also shows that our modeling method does not properly model holes, as the silhouettes are created using the object's outer contour. This limitation is irrelevant for our application since the object will be grasped by its outer sides. Sequences of photographs depicting the successful grasping and picking up of the nut and bracket are shown in Figs.11a-11c and 11d-11f, respectively. The duration required for each experiment was about 48 s. The movements of the robot consumed most of this time. This duration could be reduced significantly in future by optimizing these movements. The execution times for the silhouette model generation, structured-light model generation, model combination, and grasp planning are listed in Table 1. The model of the nut consisted of 995 surface points, while the bracket model included 3044 surface points. The modeling and grasp planning execution times reflect the complexity of the object shape.

TABLE 1
EXECUTION TIMES

Object	Execution Time (s)				Total
	Silhouette Model Generation	Structured-Light Model Generation	Model Combination	Grasp Planning	
Hex nut	1.37	0.17	0.59	0.05	2.18
Bracket	2.29	0.25	1.04	0.16	3.74

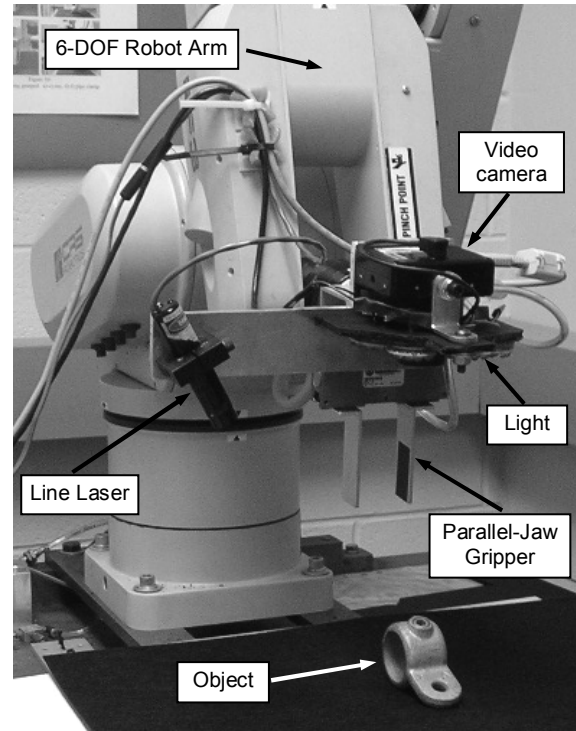


Fig. 8. Experimental setup.

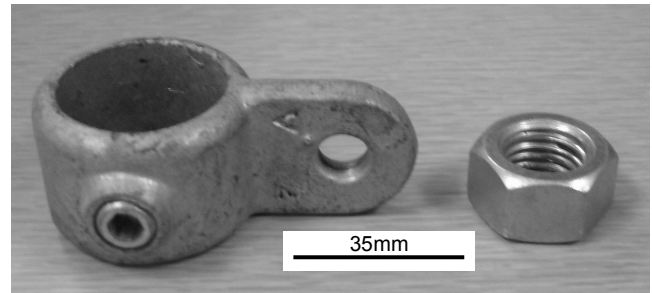


Fig. 9. Objects used for the tests.

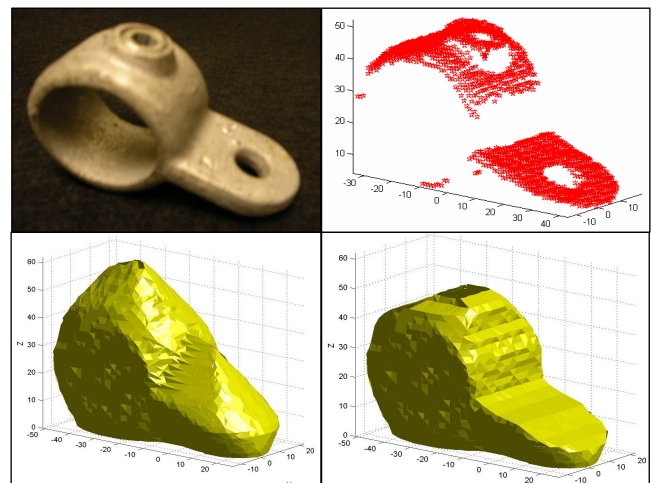


Fig. 10. Clockwise from top left: object, structured-light model, silhouette model, and combined model.

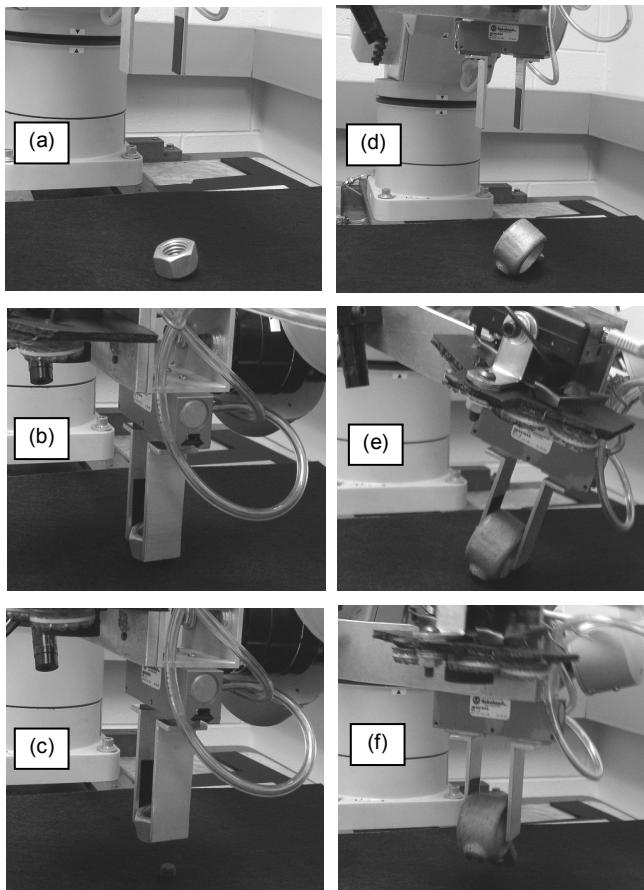


Fig. 11. Photographs from two grasping experiments.

V. CONCLUSIONS

In this paper a novel approach for grasping unknown 3D objects was presented. Our approach combines online silhouette and structured-light 3D object modeling with online grasp planning and execution. Compared to previous work, the silhouette and structured-light based modeling combination has the advantages of simplicity and robustness. It only requires the use of a single wrist-mounted video camera and a line laser. Unlike previous systems, neither the object nor the environment needs to be strongly textured. The silhouettes and laser points are obtained using simple and reliable image processing techniques. After a onetime calibration, an object model may be generated from these silhouettes and laser points. The modeling method does not properly model a hole through an object. However, this limitation is irrelevant for the grasping application. The grasp planner generates robust force closure grasps. The experimental results demonstrate that the automated grasping system works with real-world objects whose 3D shape and location are unknown by the system *a priori*. The objects were deliberately tilted (*i.e.* their sides were not perpendicular to the table) to make the task more challenging. The time required to compute an object model and plan a grasp was less than 4 s for each object.

REFERENCES

- [1] M. A. Rodrigues, Y. F. Li, M. H. Lee and J. J. Rowland, "Robotic grasping of complex shapes: is full geometrical knowledge of the shape really necessary?", *Robotica*, Vol.13, pp.499-506, 1995.
- [2] S. A. Stansfield, "Robotic grasping of unknown objects: a knowledge-based approach", *Int. J. Rob. Res.*, Vol.10, No.4, pp.314-326, 1991.
- [3] M. Taylor, A. Blake and A. Cox, "Visually guided grasping in 3D", *Proc. 1994 IEEE Int. Conf. on Rob. & Auto.*, pp.761-766, 1994.
- [4] M. Trobina and A. Leonardis. "Grasping arbitrarily shaped 3-D objects from a pile", *Proc. 1995 IEEE Int. Conf. Rob. & Auto.*, pp.241-246, 1995.
- [5] A. Namiki, Y. Nakabo, I. Ishii, and M. Ishikawa, "High speed grasping using visual and force feedback", *Proc. 1999 IEEE Int. Conf. Rob. & Auto.*, pp.3195-3200, 1999.
- [6] D. Kragic, A. T. Miller and P. K. Allen, "Realtime tracking meets online grasp planning", *Proc. 2001 IEEE Int. Conf. Rob. & Auto.*, pp.2460-2465, 2001.
- [7] G. Taylor and L. Kleeman, "Integration of robust visual perception and control for a domestic humanoid robot", *Proc. 2004 IEEE/RSJ Int. Conf. Intel. Rob. & Syst.*, pp.1010-1015, 2004.
- [8] H.-Y. Jang, H. Moradi, S. Lee and J. Han, "A visibility-based accessibility analysis of the grasp points for real-time manipulation," *Proc. 2005 IEEE IEEE/RSJ Int. Conf. Intel. Rob. & Syst.*, pp. 3111-3116, 2005.
- [9] P.J. Sanz, A. Requena, J.M. Inesta and , A.P. Del Pobil, "Grasping the not-so-obvious: vision-based object handling for industrial applications," *IEEE Rob. & Auto. Mag.*, Vol.12, No.4, pp.44-52, 2005.
- [10] D. Kragic and M. Björkman, "Strategies for Object Manipulation using Foveal and Peripheral Vision", *Proc. IEEE Int. Conf. on Computer Vision Systems*, pp. 50-55, 2006.
- [11] W. N. Martin and J. K. Aggarwal, "Volumetric descriptions of objects from multiple views", *IEEE Trans. PAMI*, Vol.5, No.2, pp.150-158, 1983.
- [12] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses", *IEEE J. Rob. & Auto.*, Vol. RA-3, No. 4, August 1987
- [13] R. Jain, R. Kasturi and B. G. Schunk, *Machine Vision*, McGraw-Hill, 1995.
- [14] W. Niem, "Robust and fast modeling of 3D natural objects from multiple views", *Proc. of SPIE*, Vol. 2182, pp.388-397, 1994.
- [15] V.D. Nguyen, "Constructing force-closure grasps," *Int. J. of Robotics Research*, Vol.7, No.3, pp.3-16, 1988.