Rapid 3D Model Acquisition from Images of Small Objects

Wee Kheng Leow, Zhiyong Huang, Yong Zhang, and Rudy Setiono
School of Computing, National University of Singapore
Lower Kent Ridge Road, Singapore 119260
leowwk, huangzy, zhangyon, rudys@comp.nus.edu.sg

Abstract

This paper describes a system for rapid acquisition of 3D models of objects for use in applications such as CAD and VR. The system consists of an image capturing platform, which comprises a PC-controlled turntable and CCD camera, and associated computer vision and graphics algorithms for 3D model acquisition. An algorithm for accurately recovering 3D points from image sequences is described. Experiments performed on synthetic and real data show that the algorithm is accurate in recovering the coordinates of 3D points and is robust against noise in 2D feature location and 3D object rotation. Results of applying the system to acquiring 3D models of real objects are illustrated.

1. Introduction

3D models of objects and scenes are essential components of CAD and VR systems. However, the acquisition of the geometry of 3D models is a long-standing difficult problem in computer graphics. In the past, forward methods such as CSG, B-rep, NURBS, and sweeping have been successful in modeling regular shapes such as mechanical and architectural parts. These methods allow a user to directly and interactively define the shape of an object. On the other hand, natural objects and scenes have irregular shapes and complex surface structures. It is very difficult to use forward methods to interactively define the details of the complex shapes and surfaces.

With the availability of vision equipment and the maturity of computer vision technology, there is now a trend to reverse engineer the acquisition process by recovering the geometric and topological information from measurements of real scenes and objects. Examples of this approach include the acquisition of human faces from photographs [3, 10], human cornea from specular reflection patterns [11], and sculpture models from range images [4]. While some of these methods are designed to acquire specific object models, others (e.g., [4]) are applicable to modeling general, natural objects.

There are two main approaches to the rapid acquisition of the models of natural objects and scenes: structured light and multiple views. The first approach projects a structured light pattern, usually a stripe of laser beam, onto the surface of an object and uses a camera to capture the surface contour as revealed by the reflected laser light. The contours along different surface patches are then combined to compute the 3D geometry of the object. Such laser scanning products are currently available in two versions: hand-held and automated. The hand-held version (e.g., [16]) requires the user to manually move the laser stripe to various parts of the object's surface. It, therefore, takes a lot of manual work to capture all the surface details of an object. The automated version (e.g., [18, 20]) uses a motorized system to move the laser in a manner similar to a 2D image scanner. 3D laser scanners typically produce accurate results but are expensive, especially the automated version and the color scanners.

The second approach uses one or more cameras to capture multiple views of an object and uses the multiple views to recover 3D coordinates of the feature points on the object's surface. The single-camera system (e.g., [14, 15, 17]) either moves the camera or the object so as to capture various views of the object. The multiple-camera system (e.g., [19]) typically employs static cameras to simultaneously capture multiple views of the object. However, it would require many cameras to adequately capture various facets of a complex object, and would therefore be more complex and expensive than the single-camera system.

This article describes an inexpensive 3D model acquisition system that is affordable to the home users. It consists of an image capturing platform, which comprises a PC-controlled turntable and a CCD camera, and associated computer vision and graphics algorithms for 3D model reconstruction. In the following sections, a comparative review of related single-camera systems is presented in Section 2. Next, our system for rapid acquisition of 3D models is described Section 3. Results of applying the system to

To appear in *Proc. Geometric Modeling & Processing 2000.*

acquiring 3D models of real objects are illustrated in Section 4. In addition, Section 4 also reports experiments conducted to measure the system's reconstruction accuracy and robustness against noise. Finally, Section 5 summarizes and concludes our research work.

2. Related Works

The systems proposed by Matsumoto et al. [14], Fitzgibbon et al. [9], Mehren and Rodehorst [15], and Szeliski and co-workers [13, 21] are most similar to ours. They also use a turntable to capture multiple views of an object. In particular, the methods of Matsumoto et al. and Fitzgibbon et al. reconstruct 3D model from image silhouette. These methods have the disadvantage that concave parts of the object may not appear on the silhouette and, thus, cannot be recovered. In contrast, our method recovers 3D structures based on the feature points on the surface of the object and it does not have the limitation of the silhouette method.

Mehren and Rodehorst's method adopts an affine camera model which is less accurate then the perspective model used in our system. The methods of Szeliski and co-workers adopt approximations of the perspective camera models and recover a quasi-Euclidean structure of the object. Our method, on the other hand, recovers the true 3D model of the object.

Whereas the above systems focus on acquiring 3D models of objects, the systems of Teller [22] and Debevec et al. [6] aim at acquiring buildings and architectural structures from multiple perspective views and aerial images. Teller's system consists of a camera mounted on a pan-tilt head which is, in turn, mounted on a mobile platform. The platform also includes instrumentation for maintaining estimates of global position (GPS), heading information (IMU), and for dead-reckoning (mechanical wheel encoders). These devices are controlled by a PC which is also mounted on the mobile platform. Debevec et al. apply a hybrid approach consisting of image-based and geometry-based modeling. The approach consists of two components. The first component is a photogrammetric modeling method which facilitates the recovery of the basic geometry from scene images. The second component is a model-based stereo algorithm which recovers depth from widely-spaced image pairs. Human intervention is required to interactively indicate to the system block structures, feature correspondences, and un-occluded texture regions in the scene. In contrast, our system is automatic and does not require human intervention. It is also simpler than Teller's system because our goal is to acquire 3D models of small objects instead of buildings.

Figure 1. The acquisition hardware platform consisting of a PC, a PC-controlled turntable, and a CCD camera.

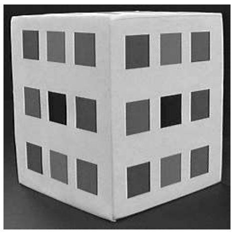


Figure 3. Object with known corner points used for calibrating the camera.

3. 3D Model Acquisition System

3.1. The Acquisition Framework

Our 3D model acquisition system consists of three hardware components (Fig. 1): a PC, a PC-controlled turntable, and a CCD camera. The turntable is rotated by a high-precision stepper-motor with a smallest step size of $0.036°$ and a maximum error of 1% (i.e., $0.01°$ for $1°$ rotation). Therefore, the amount of rotation can be controlled by the PC accurately.

The software framework of the system is illustrated in Fig. 2. It consists of four main modules: Camera Calibration, Feature Tracking, 3D Recovery, and Triangulation. The first module, Camera Calibration, is performed only once. A calibration object (Fig. 3) is placed on the turntable and its image is captured by the CCD camera. A refined version of the Beaudet's corner detector [2] is applied to accurately locate the 2D coordinates of the corners in the image. The known 3D positions of the corners and their corresponding 2D image coordinates are input into Tsai's calibration algorithm [24] which computes the intrinsic and extrinsic parameters of the camera.

After calibration, the object to be acquired is placed on

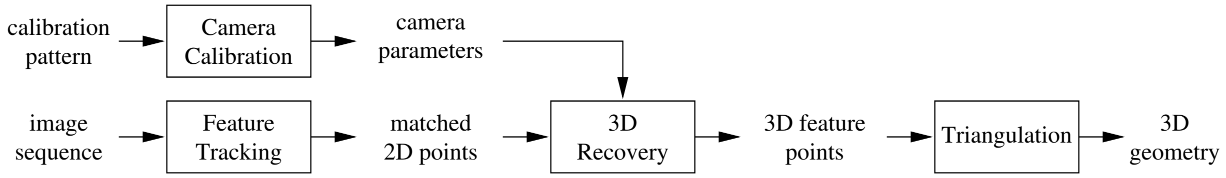


Figure 2. The acquisition platform: software functional modules.

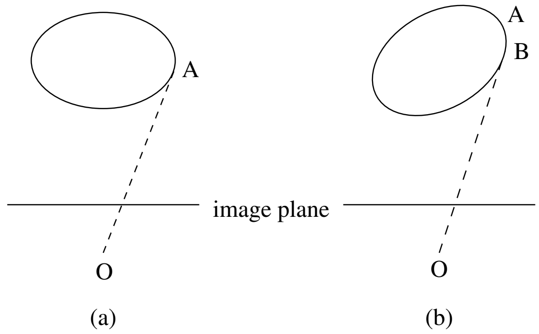


Figure 4. The dashed lines denote the lines of projection from the object's boundary through the image plane (viewed from the top) to the focal point O. As the object is rotated, its boundary point changes from point A to point B.

the turntable and rotated through $360°$ to capture different views of the object. The current implementation of the system rotates the object at a step size of $1.8°$ thus capturing a total of 200 images. Next, the Kanade-Lucas-Tomasi (KLT) tracker [23] is applied to the 200 images to match the corresponding feature points in consecutive images. It is observed that as the object is rotated, smooth curve surfaces of the object present different feature points on the 2D object boundaries at different time (Fig. 4). The feature tracking algorithm may misinterpret the boundary points in consecutive images as arising from the same 3D feature point. To overcome this problem, the original KLT tracker is modified to ignore feature points that are located near object boundaries.

The coordinates of the matched 2D feature points and their corresponding rotation angles, together with the camera's parameters, are input into the 3D Recovery module to recover the 3D coordinates of the feature points. An algorithm, called the *Ray Intersecting* (RI) algorithm, is developed to accurately compute the 3D coordinates. Details about RI will be described in Section 3.2.

In the final stage, the 3D coordinates of feature points are input to the Triangulation module to reconstruct the surfaces of the object. Currently, the 3D Alpha Shapes algorithm introduced by Edelsbrunner and Mucke [8] is used. The algorithm outputs a 3D model represented by a polygon mesh. Polygonization from scattered points is an important problem in computer graphics and CAGD. Well known polygonization methods include crust algorithm [1], zero-set algorithm [12], and Alpha Shapes algorithm [7, 8]. In the Alpha Shapes method, a simplex (edge, triangle, or tetrahedron) is included in an Alpha Shape if it contains some circumspheres with no interior sample points. A circumsphere is a sphere of radius α whose surface touches at least three sample points.

We adapted the implementation of Alpha Shapes algorithm from CGAL library [5]. It works well for most cases. However, if the feature points' sampling rate varies a lot on different parts of an object, it is impossible to have a unique α value that polygonizes the whole model. This situation can happen because the 2D feature points tracked by KLT algorithm may not be evenly distributed. We are now working on a new polygonization algorithm that can adapt to different sampling rates on different parts of the object.

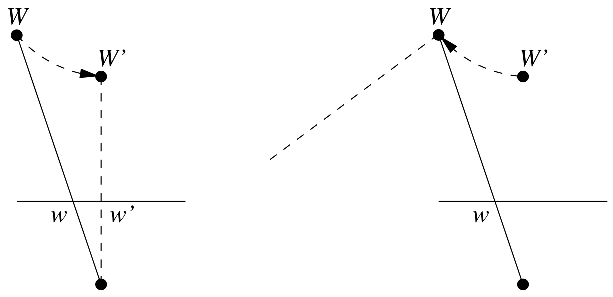
3.2. Ray Intersecting Algorithm

The Ray Intersecting (RI) algorithm is a variation of the triangulation method. It recovers 3D coordinates of feature points as follows. A 3D point W at world coordinate $\mathbf{X} = (X, Y, Z)^\top$ is projected to a 2D point w at coordinates $\mathbf{x} = (x, y)^\top$ in the image plane through a perspective transformation:

$$\begin{aligned} x &= \frac{f\xi(\mathbf{R}_1 \cdot \mathbf{X} + D_X) + s\,(\mathbf{R}_2 \cdot \mathbf{X} + D_Y)}{\mathbf{R}_3 \cdot \mathbf{X} + D_z} + o_x \\ y &= \frac{f(\mathbf{R}_2 \cdot \mathbf{X} + D_y)}{\mathbf{R}_3 \cdot \mathbf{X} + D_z} + o_y \end{aligned} \tag{1}$$

where f is the focal length, ξ the aspect ratio, s the skew, (o_x, o_y) the principal point, $\mathbf{R}$ the rotation matrix, and $\mathbf{D}$ the translation matrix of the camera. By rearranging terms in Eq. 1, we obtain a parametric equation that describes the projection line from a 3D world point to a 2D image point:

$$\begin{aligned} X_0 &= (B_1D_2 - D_1B_2)/(A_1B_2 - B_1A_2) \\ Y_0 &= (D_1A_2 - A_1D_2)/(A_1B_2 - B_1A_2) \\ u &= (B_1C_2 - C_1B_2)\,/(A_1B_2 - B_1A_2) \\ v &= (C_1A_2 - A_1C_2)\,/(A_1B_2 - B_1A_2) \end{aligned} \tag{2}$$



(a) (b)

Figure 5. Ray Intersecting method. (a) World point W casts a projection line (solid line) onto the image point w on the image plane. As W rotates to its new position W', it casts a new projection line (dashed line) onto the image plane. (b) By rotating the new projection line backward, it now intersects the original line at the world point W.

where

$$
\begin{array}{rcl}
A_1 & = & f\xi R_{11} + sR_{21} + (o_x - x)R_{31} \\
B_1 & = & f\xi R_{12} + sR_{22} + (o_x - x)R_{32} \\
C_1 & = & f\xi R_{13} + sR_{23} + (o_x - x)R_{33} \\
D_1 & = & f\xi D_X + sD_Y + (o_x - x)D_Z \\
A_2 & = & fR_{21} + (o_y - y)R_{31} \\
B_2 & = & fR_{22} + (o_y - y)R_{32} \\
C_2 & = & fR_{23} + (o_y - y)R_{33} \\
D_2 & = & fD_Y + (o_y - y)D_Z \ .
\end{array}
\tag{3}
$$

Equation 2 can be written in the vector form

$$\mathbf{X} = \mathbf{X}_0 + t\mathbf{u} \tag{4}$$

where $\mathbf{X}_0 = (X_0, Y_0, 0)^\top$ and $\mathbf{u} = (u, v, 1)^\top$.

The world coordinate system is fixed onto the turntable such that the turntable rotates about the Y-axis. As the world point W is rotated about the Y-axis, it projects several lines

$$\mathbf{X} = \mathbf{X}_{0i} + t_i\mathbf{u}_i \tag{5}$$

onto the image plane at various rotation angles θ_i, where $\mathbf{X}_{0i} = (X_{0i}, Y_{0i}, 0)^\top$ and $\mathbf{u}_i = (u_i, v_i, 1)^\top$. By rotating the new projection lines backward about the Y-axis by the corresponding angles, the new world points would now coincide with the original world point W (Fig. 5).

The backward rotated projection lines are given by

$$\mathbf{X} = \mathbf{M}_i(\mathbf{X}_{0i} + t_i\mathbf{u}_i) \tag{6}$$

where

$$\mathbf{M}_i = \begin{bmatrix} \cos\theta_i & 0 & \sin\theta_i \\ 0 & 1 & 0 \\ -\sin\theta_i & 0 & \cos\theta_i \end{bmatrix} . \tag{7}$$

A closed-form solution of the 3D world point can now be obtained by solving for the intersection of any two backward rotated projection lines, for example, line 0 (Eq. 4) and line i (Eq. 6). For notational convenience, denote $\mathbf{X}'_0 = \mathbf{M}_i\mathbf{X}_{0i}$ and $\mathbf{u}' = \mathbf{M}_i\mathbf{u}_i$. Then, the 3D world point is given by Eq. 4 at the following t:

$$t = \frac{(X'_0 - X_0)v' + (Y_0 - Y'_0)u'}{uv' - u'v} . \tag{8}$$

In practice, due to the limited resolution of the camera's digitizer and the presence of noise, the backward rotated projection lines may not intersect at exactly one point. A good approximation of the intersection can be obtained by finding the point $\mathbf{Q}$ nearest to the lines. That is, find the $\mathbf{Q}$ that minimizes the total distance to the lines

$$D_T = \sum_i \left\| (\mathbf{Q} - \mathbf{X}_{0i}) - \frac{(\mathbf{Q} - \mathbf{X}_{0i}) \cdot \mathbf{u}_i}{\|\mathbf{u}_i\|^2}\,\mathbf{u}_i \right\| \tag{9}$$

This minimization can be performed efficiently and reliably given a good initial guess of the actual solution, which can be obtained from the intersection of any two projection lines using the closed-form solution (Eq. 8).

Note that, in contrast to most existing algorithms, RI recovers each 3D point independently of other 3D points. Consequently, if an object has mostly simple surfaces and a small number of complex, convoluted surfaces, RI's error in localizing the 3D points on complex surfaces does not affect its accuracy in recovering those on simple surfaces. Thus, RI is robust against localized acquisition inaccuracy.

4. Experiments and Results

Two sets of experiments were conducted. The first set of experiments measured the accuracy and robustness of the RI algorithm using both synthetic data and a real object. The second set of experiments applied the system to acquire the 3D models of three real objects: a calibration cube, a duck and a mouse.

4.1. Accuracy and Robustness Measurement

To measure the accuracy and robustness of the RI algorithm, twenty synthetic world points were randomly selected from a sphere of radius 200mm. These world points were projected onto the image plane through a perspective camera model to generate the synthetic 2D image points. The values of the camera parameters were set according to a real calibrated camera. In particular, the focal length was set at 8mm and the effective resolution was set at 100 pixels/mm instead of infinity. In contrast, published tests on synthetic data typically assume infinite camera resolution. The world points were rotated about the Y-axis through an

Table 1. Reconstruction error in the presence of noise in image pixel location. Noise is measured in pixels and mean error in mm.

noise	0	1	2	3	4	5
error	0.38	0.73	1.33	1.89	2.65	3.13

Table 2. Reconstruction error due to inaccuracy in rotation. Noise is measured in degrees of rotation and mean error in mm.

noise	0.0	0.1	0.2	0.5	1.0	2.0
error	0.38	0.33	0.35	0.47	0.73	1.31

angle of 90° in steps of 10° to generate the multiple views required for the reconstruction. Moreover, random noise was added to the coordinates of the 2D image points and to the rotation angles of the object.

In the test, both calibration and reconstruction errors were measured. Calibration error was computed as the mean distance from a recovered 3D point to the projection line (Eq. 4) through its corresponding 2D image point. The reconstruction error was calculated as the mean absolute error between a recovered 3D point and the corresponding actual 3D point. Tables 1 and 2 show that, in the absence of noise, RI has a mean error of 0.38mm, which is 0.19% of the size of the object. The test results also show that, despite the addition of noise, RI's reconstruction error does not increase significantly. Therefore, RI is a robust algorithm for recovering the 3D coordinates of feature points.

An experiment was also conducted to measure the accuracy of RI and Tsai's calibration algorithm in combination. In this test, the real calibration object (Fig. 3) was used. The object has a size of about $100 \times 100 \times 100$ mm^3. 3D coordinates of 72 corner points on the cube were measured manually. These coordinates were used to calibrate a real camera using Tsai's algorithm. Next, the cube was rotated and the images at different views were captured. 3D coordinates of the corner points were computed using the RI algorithm. This experiment was performed three times and the mean error was averaged over the three tests. Test results in Table 3 show that in recovering the 3D coordinates of the corner points, RI and Tsai's algorithm in combination have a mean error of about 1% of the object's size. This amount of error is about 6 times larger than that of RI alone under noise-free condition (compare with Table 1). The main source of the error is attributed to calibration error, which accounts for about 70% of the error. If calibration error is removed, then the reconstruction error amounts to 0.34% of the object's size, which is only 1.8 times that of RI under noise-free condition. This experiment again shows that RI is robust against noise in recovering 3D coordinates of real objects. More accurate calibration, such as precise measurement of the actual 3D points, would improve the overall reconstruction accuracy.

Table 3. Mean errors in calibrating and recovering the 3D coordinates of corner points on a real calibration object.

error source	error	object size	% object size
calibration	0.77	100	0.77%
calibration & reconstruction	1.11	100	1.11%
reconstruction (noise-free)	0.38	200	0.19%

4.2. Model Acquisition

Figure 3 shows an image of the calibration object. The corners of the squares on the object's surfaces are extracted and their 3D coordinates are recovered. Fig. 6 shows three perspective views of the squares acquired. As expected, the squares in the acquired model lie on flat surfaces.

Figure 7 shows two different views of a duck figurine. Figure 8 illustrates four perspective views of the acquired duck model in polygon mesh. These figures show that the general shape of the duck is correctly acquired. In particular, the duck's head, bill, neck, body, and tail are correctly acquired.

Figure 9 illustrates two views of a mouse figurine and Fig. 10 illustrates three perspective views of the acquired mouse model in polygon mesh. The general shape of the mouse is well acquired. The mouse's ears, body, forelimbs, and hindlimbs are apparent in the perspective views.

Most of the computation time was spent in tracking 2D features and recovering 3D coordinates of feature points. For the duck model, tracking of the 9921 feature points in 200 images took about 2.5 hours. For the mouse model, tracking of the 3848 feature points in 200 images took about 1 hour. The RI algorithm took about 1/10 the amount of time to compute.

5. Conclusions

3D model acquisition from images of existing scenes and objects is a promising geometric modeling approach. This reverse engineering approach, which can be fully automated, is complementary to forward approaches such as

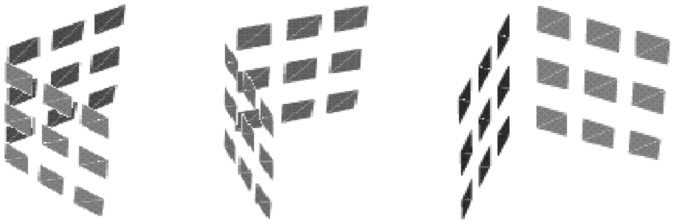


Figure 6. Perspective views of the acquired cube's faces.

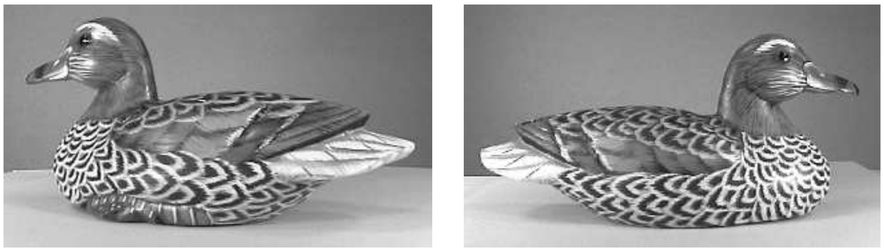


Figure 7. Two views of a duck figurine.

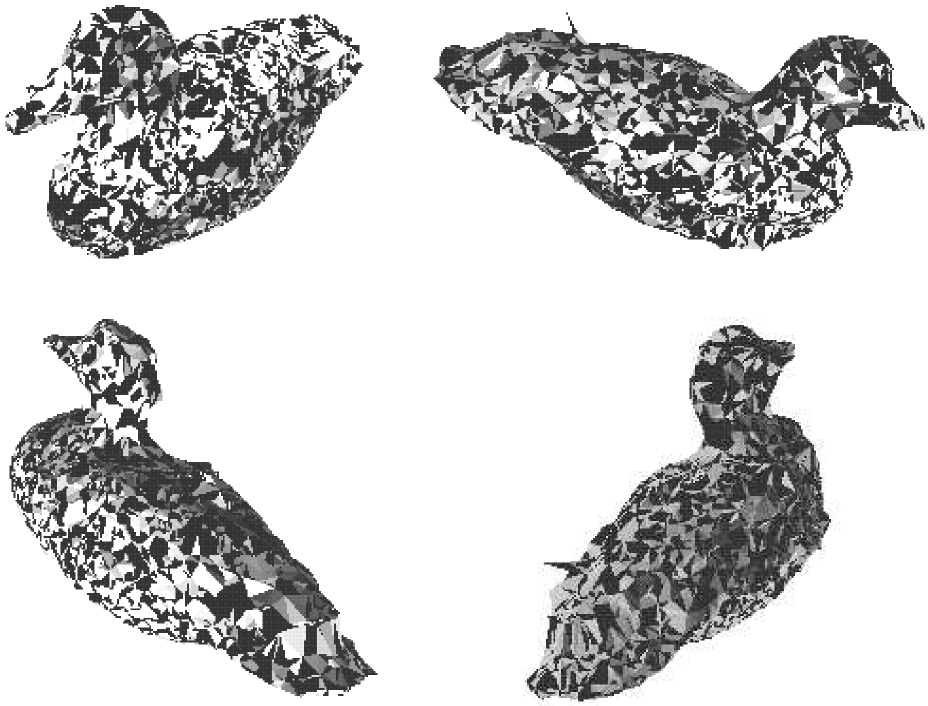


Figure 8. Four perspective views of the acquired model of the duck.

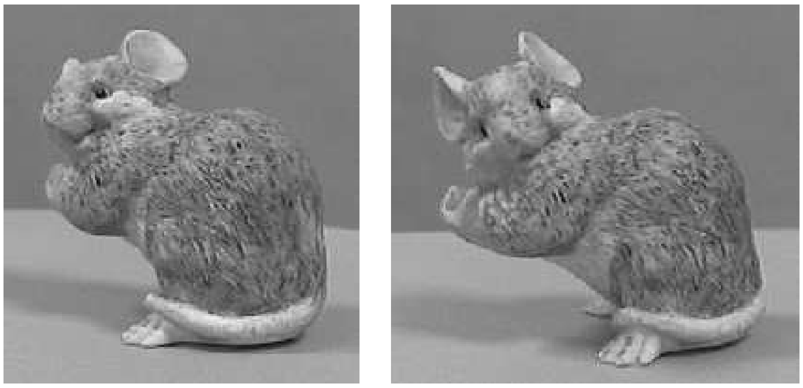


Figure 9. Two views of a mouse figurine.

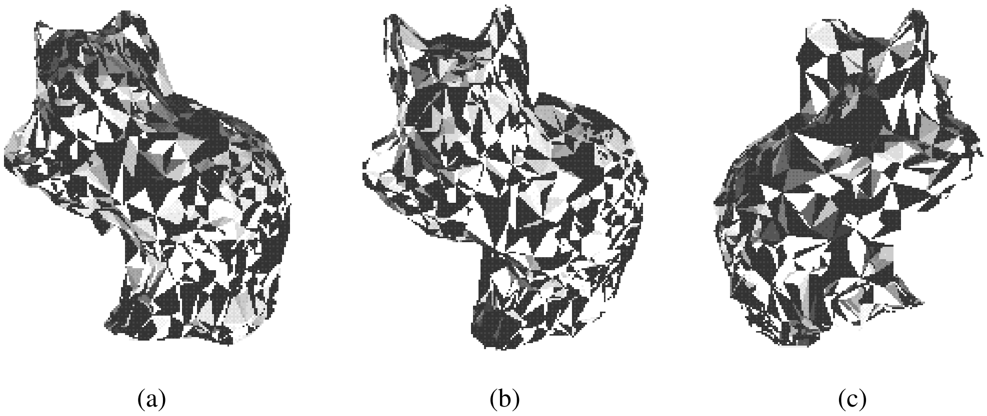


Figure 10. Three perspective views of the acquired model of the mouse: (a) front, (b) front-top, (c) back.

CSG, B-rep, NURBS, and sweeping. With the availability of inexpensive equipments such as digital camera and computer-controlled turntable, 3D scanning technology is becoming more affordable to home users. In the near future, we expect automated 3D scanners to be widely available to home users, just like 2D image scanners.

In this paper, a rapid 3D model acquisition framework is presented. The framework consists of an image capturing platform and associated algorithms of computer vision and graphics. Experimental results show that it is appropriate for acquiring the geometric models of small objects. Moreover, the algorithm for recovering 3D feature points from 2D image points is accurate and robust against noise in pixel location and object rotation. Further research is being performed to develop better polygonization algorithms for reconstructing complex surfaces such as convoluted surfaces, discontinuous surfaces, and surfaces on a thin volume.

Acknowledgment

This research is supported by NUS Academic Research Fund RP3982704. We also like to thank Indriyati Atmosukarto for her help in touching up the displayed images.

References

[1] N. Amenta, M. Bern, and M. Kamvysselis. A new Voronoi-based surface reconstruction algorithm. In *Proc. ACM SIGGRAPH 98*, pages 415–421, 1998.

[2] P. R. Beaudet. Rotational invariant image operators. In *Proc. 4th Int. Conf. Pattern Recognition*, pages 579–583, 1978.

[3] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Proc. ACM SIGGRAPH 99*, pages 187-194, 1999.

[4] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proc. ACM SIGGRAPH 96*, pages 303-312, 1996.

[5] S. Schirra, R. Veltkamp, and M. Yvinec eds. *CGAL release 2.0 Reference Manual.* http://www.cs.uu.nl/CGAL/, 1999.

[6] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *Proc. of ACM SIGGRAPH 96*, pages 11–20, 1996.

[7] H. Edelsbrunner, D. G. Kirkpatrick, and and R. Seidel. On the shape of a set of points in the plane. In *IEEE Trans. on Information Theory*, 29, pages 551–559, 1983.

[8] H. Edelsbrunner and E. P. Mucke. Three-dimensional Alpha Shapes. In *ACM Trans. on Graphics*, 13, pages 43–72, 1994.

[9] A. W. Fitzgibbon, G. Cross, and A. Zisserman. Automatic 3D model construction from turn-table sequences. In L. Van Gool and R. Koch, eds., *Structure and Motion from Multiple Images in Large-Scale Environments*. Springer, 1998.

[10] B. Guenter, C. Grimm, D. Wood, H. Malvar, and F. Pighin. Making faces. In *Proc. ACM SIGGRAPH 98*, pages 55-66, 1998.

[11] M. A. Halstead, B. A. Barsky, S. A. Klein, and R. B. Mandell. Reconstructing curved surfaces from specular reflection patterns using spline surface fitting of normals. In *Proc. ACM SIGGRAPH 96*, pages 335-342, 1996.

[12] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. In *Proc. ACM SIGGRAPH 92*, pages 71–78, 1992.

[13] S. B. Kang. Quasi-Euclidean recovery from unknown but complete orbital motion. Technical Report CRL 97/10, Cambridge Research Laboratory, 1997.

[14] Y. Matsumoto, H. Terasaki, K. Sugimoto, and T. Arakawa. A portable three-dimensional digitizer. In *Proc. of Int. Conf. on Recent Advances in 3-D Digital Imaging and Modeling*, pages 197–204, 1997.

[15] D. Mehren and V. Rodehorst. Gestaltsanalyse komplexer Objekte bei kontrollierter Bewegung, Diplomarbeit, TU Berlin, Fachbereich Informatik, 1994.

[16] ModelMaker. 3D Scanners Ltd, UK. http://www.3dscanners.com.

[17] ReCaSs. Fraunhofer Institute for Computer Graphics (IGD), Germany. http://www.igd.fhg.de/projects/index.html.

[18] REPLICA. 3D Scanners Ltd, UK. http://www.3dscanners.com.

[19] H. Saito, S. Baba, M. Kimura, S. Vedula, and T. Kanade. In *Proc. 2nd Int. Conf. on 3-D Digital Imaging and Modeling*, 1999.

[20] Synchronized Laser Scanners. National Research Council, Canada. http://www.vit.iit.nrc.ca/.

[21] R. Szeliski and S. B. Kang. Recovering 3D shape and motion from image streams using non-linear least squares. *Journal of Visual Communication and Image Representation*, 5(1):10–28, 1994.

[22] S. Teller. Toward Urban Model Acquisition from Geo-Located Images. In *Proc. of Pacific Graphics'98*, pages 45–52, 1998.

[23] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *IJCV*, pages 137–154, 9(2), 1992.

[24] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 3(4):323–344, 1987.