Machine Vision and Applications (2005) 16: 148–156
Digital Object Identifier (DOI) 10.1007/s00138-004-0165-2

Machine Vision and Applications

A multiview 3D modeling system based on stereo vision techniques

Soon-Yong Park[1], **Murali Subbarao**[2]

[1] Computer Engineering Department, Kyungpook National University, Daegu, 702-701 Korea
[2] Electrical and Computer Engineering Department, State University of New York at Stony Brook, Stony Brook, NY 11794-2350, USA

Received: 2 August 2003 / Accepted: 20 September 2004
Published online: 25 February 2005 – © Springer-Verlag 2005

Abstract. This paper introduces a stereo vision system to automatically generate 3D models of real objects. 3D model generation is based on the merging of multiview range images obtained from a digital stereo camera. Stereo images obtained from the camera are rectified, and a correlation-based stereo matching technique reconstructs range images from them. A turntable stage is also employed to obtain multiple range images of the objects. To register range images into a common coordinate system automatically, we introduce and calibrate a turntable coordinate system with respect to the camera coordinate system. After the registration of multiview range images, a 3D model is reconstructed using a volumetric integration technique. Error analysis on turntable calibration and 3D model reconstruction shows the accuracy of our 3D modeling system.

Keywords: Stereo vision – Multiview – 3D modeling – Turntable calibration

1 Introduction

Generating a complete 3D model of an object has been a topic of much interest in recent computer vision and computer graphics research. Many computer vision techniques have been investigated to generate complete 3D models. There are two major approaches in this research. The first one is based on merging multiview range images into a complete 3D model [4,7,23]. The second one is based on processing photographic images using a volumetric reconstruction technique, such as *voxel coloring* and *shape-from-silhouettes* [5,21]. This paper presents a computer vision system to automatically generate 3D computer models by merging multiview range images of real objects. We employ a stereo vision camera and a turntable stage to develop an automatic and inexpensive vision system.

Multiview 3D modeling has been done by many active or passive ranging techniques. Laser range imaging and structured light techniques are the most common active techniques. These techniques project special light patterns onto the surface of a real object to measure the depth to the surface by a simple triangulation technique [4,7]. Some common approaches of the structured light technique employ a single line pattern [9], a multiline pattern [17], a color-coded pattern [24], and a space-time coded pattern [19]. Advantages of using the active techniques are accuracy and speed of depth acquisition [24, 19]. However, active techniques are still more expensive than passive techniques.

In contrast, however, relatively less research has been done using passive techniques, such as stereo image analysis. This is mainly due to the inherent problems (e.g., mismatching and occlusion) of stereo matching. Several stereo matching techniques have been introduced; however, only a few of them are empolyed for multiview 3D modeling [20]. Okutomi et al. [13] presented a multibaseline stereo matching technique to reduce matching ambiguity, and their approach is empolyed in many multiview 3D modeling techniques [18]. Chen and Medioni [3] used a stereo camera to obtain range images and integrated them using a volumetric method. In each viewpoint, they used a 3D voxel grid to find a disparity surface using a dynamic programming technique. Rander et al. [18] and Vedular et al. [22] also used stereo vision techniques to create 3D models of a dynamic scene for virtual reality applications. They used a considerable number of video cameras and a multiple-baseline stereo matching technique proposed by [13].

In order to generate complete 3D models, we obtain multiview range images using stereo vision techniques. We use two inexpensive digital still cameras to capture stereo images of an object. The cameras are calibrated by a projective calibration technique, and stereo images from them are rectified accordingly. A range image is then obtained from a pair of rectified stereo images. Multiview range images are obtained by changing the viewing direction to the object. Different approaches to changing viewing direction exist. Among them are a moving object on a turntable with a fixed sensor [4,5], a moving sensor with a fixed object [1,12], and other variations [7,8]. One advantage of using a turntable is the ease of calibration between different views. We also employ a turntable stage to rotate the object and to obtain multiple range images. Multiple range images are then registered and integrated into a single 3D model. In order to register range images automatically, we define and calibrate a turntable coordinate system (TCS) with respect to the camera's coordinate system (CCS). To integrate multiple range images into a single mesh model, we use a vol-

Correspondence to: S.Y. Park (e-mail: sypark@mail.knu.ac.kr)

umetric integration technique [10,14]. Error analysis on real objects shows the accuracy of our 3D model reconstruction.

Section 2 presents the calibration and rectification of stereo images and a comparison of 3D range reconstruction techniques. Section 3 presents the definition and calibration of the TCS with respect to the CCS. In Sect. 4, we present a 3D modeling technique of merging multiview range images and error analysis of our 3D modeling system. Finally, we conclude the paper in Sect. 5.

2 Range image acquisition

2.1 Stereo calibration and rectification

In this paper, we employ a projective camera model to calibrate our stereo camera. Calibration of the projective camera model can be considered as an estimation of a projective transformation matrix from the world coordinate system (WCS) to the camera's coordinate system (CCS).

Let $\mathbf{w} = [x\ y\ z]^T$ be the coordinates of a 3D point W with respect to the WCS, $\mathbf{p} = [u\ v]^T$ the coordinates of the projection of $\mathbf{w}$ to the retinal (CCD) plane of a camera, and $\mathbf{p}' = [u'\ v']^T$ the coordinates of $\mathbf{p}$ in the picture plane (pixels). The mapping from 3D coordinates to 2D coordinates is the *perspective projection*, which is represented by a linear transformation in *homogeneous coordinates*. Let $\tilde{\mathbf{p}} = [u\ v\ 1]^T$, $\tilde{\mathbf{p}}' = [u'\ v'\ 1]^T$, and $\tilde{\mathbf{w}} = [x\ y\ z\ 1]^T$ be the homogeneous coordinates of $\mathbf{p}$, $\mathbf{p}'$, and $\mathbf{w}$, respectively. Then a 3×4 perspective transformation is given by matrix $\tilde{\mathbf{M}}$:

$$\tilde{\mathbf{p}}' = \mathbf{K}\tilde{\mathbf{p}} \cong \mathbf{K}\tilde{\mathbf{M}}\tilde{\mathbf{w}}\,, \tag{1}$$

where $\cong$ means equal up to a scale factor. The camera is therefore modeled by a transformation matrix $\mathbf{K}$ and its perspective projection matrix (PPM) $\tilde{\mathbf{M}}$, which can be decomposed into the product

$$\tilde{\mathbf{M}} = \mathbf{A}[\mathbf{R}|\mathbf{t}]\,. \tag{2}$$

The matrices $\mathbf{K}$ and $\mathbf{A}$ depend on the intrinsic parameters only and have the following forms:

$$\mathbf{K} = \begin{bmatrix} k_u & 0 & 0 \\ 0 & k_v & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{A} = \begin{bmatrix} f_u & \gamma & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where, f_u, f_v are the focal lengths in the horizontal and the vertical directions, k_u, k_v are the scaling factors from the retinal plane to the picture plane, (u_0, v_0) are the coordinates of the principal point in the retinal plane, and γ is a skew factor.

Stereo rectification determines a transformation of each image plane such that pairs of conjugate epipolar lines become parallel to the horizontal image axes. Using projection matrices of the left and the right cameras of the stereo vision system, we rectify stereo images by using the rectification technique investigated by Fusiello et al. [6].

A transformation matrix $\mathbf{T}_i$ that rectifies a homogeneous pixel $\tilde{\mathbf{p}}_\mathbf{o}$ in an original image plane to a new pixel position $\tilde{\mathbf{p}}_\mathbf{n}$ is estimated as

$$\tilde{\mathbf{p}}_\mathbf{n} = \mathbf{T}_\mathbf{i}\tilde{\mathbf{p}}_\mathbf{o}\,. \tag{4}$$

The picture coordinates $\mathbf{p}'_n = [u', v', 1]^T$ of the image point $\mathbf{p}_n$ are then obtained by multiplying the transformation matrix $\mathbf{K}$ to the image coordinates:

$$\tilde{\mathbf{p}}'_n = \mathbf{K}\tilde{\mathbf{p}}_n\,. \tag{5}$$

However, when we save a rectified image to a 2D array of a picture frame, we need to consider the translation of the principal point. Otherwise, we may lose some portion of the image outside of the original picture frame. This is because of an offset between the original principal point (u_{o0}, v_{o0}) and the new principal point (u_{n0}, v_{n0}), which is due to the rotation of the optical axis of the camera. In order to translate the rectified image back into the picture frame, we compute the new principal point (u_{n0}, v_{n0}) by adding the offset to the old principal point. The offset of the principal points can be computed by mapping the origin of the retinal plane onto the new retinal plane:

$$\tilde{\mathbf{o}}_n = \mathbf{T}_i \begin{bmatrix} u_{o0} \\ v_{o0} \\ 1 \end{bmatrix}, \tag{6}$$

and the new retinal coordinates are

$$\tilde{\mathbf{p}}'_\mathbf{n} = \mathbf{K}(\tilde{\mathbf{p}}_\mathbf{n} - \tilde{\mathbf{o}}_\mathbf{n})\,. \tag{7}$$

We consider the offset only in the x direction because rectifying the transformation rotates the image plane around the y axis.

2.2 Stereo system configuration

Our stereo camera consists of two identical digital still cameras, which are *Olympus C-3020 Zoom*. The two cameras are installed on a vertical stereo mount. We fix the cameras on the mount with an arbitrary toed-in angle so that the optical axes of the cameras converge to about 600 mm from the camera. Two digital cameras are connected to a personal computer, running on a 1.8-GHz Intel Pentium, through two USB ports. Figure 1 shows a picture of the stereo camera and a turntable stage.

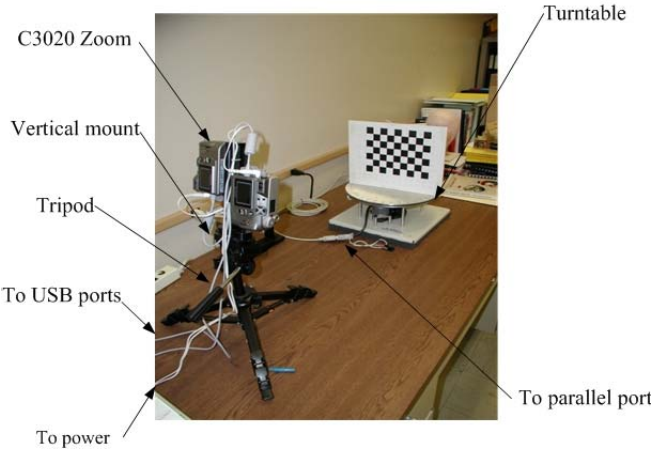


Fig. 1. Stereo camera system

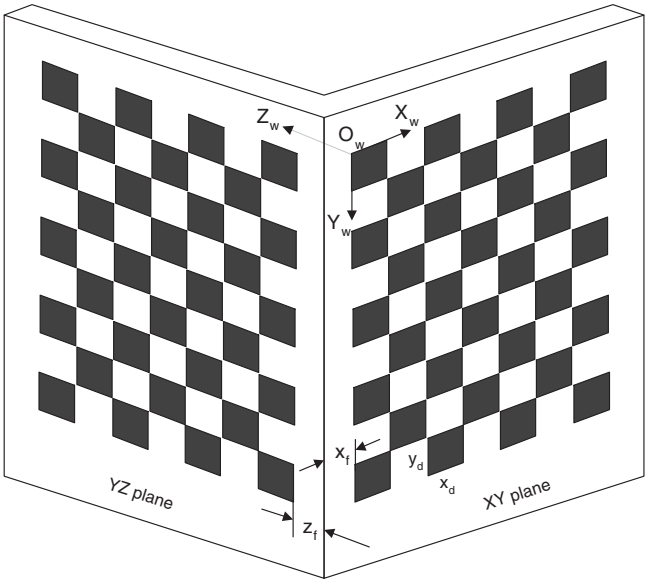


Fig. 2. Checkerboard pattern

We use a checkerboard pattern to calibrate the stereo camera. The pattern has two planes that are parallel to the xy and the yz planes of the WCS. On each plane are 48 control points that compose a set of 3D world coordinate points. Figure 2 shows a diagram of the calibration pattern. The specifications of the pattern are as follows:

- Size of a black square in x direction: $x_d = 17.2\,\mathrm{mm}$.
- Size of a black square in y direction: $y_d = 17.18\,\mathrm{mm}$.
- Offset to yz plane in x direction from origin: $x_f = 6\,\mathrm{mm}$, which means $x = -6\,\mathrm{mm}$ on the yz plane.
- Offset to edge of rightmost square on xy plane in z direction: $z_f = 11.5\,\mathrm{mm}$.

2.3 Stereo matching

From a rectified stereo image pair we acquire a range image by employing a multiresolution stereo matching technique using a Gaussian pyramid [2]. A Gaussian pyramid for an image I is a sequence of copies of I, where each successive copy has half the resolution and sample rate. The levels of a Gaussian pyramid for a given image I are calculated as

$$g_k(i,j) = \sum_{m=-2}^{2}\sum_{n=-2}^{2} w(m,n)g_{k-1}(2i+m, 2j+n)\,, \quad (8)$$
$$g_0(i,j) = I(i,j)\,,$$

where $w(m,n)$ is a 5×5 Gaussian kernel. Because this kernel is separable, we use a 1D Gaussian kernel $w(m)$ whose length is 5. The weights of the Gaussian kernel are

$$w(0) \;= 0.4\,,$$
$$w(1) = w(-1) = 0.25\,,$$
$$w(2) = w(-2) = 0.05\,.$$

Three levels of a Gaussian pyramid are used from level 0 to level 2. The level 0 image corresponds to the original image, and the level 2 image corresponds to the smallest image. The original image size is 1280×960, and the image size at level 2 is 320×240. We use a variable size of matching block for stereo matching – (15-2k)×(15-2k) for the kth pyramid level.

An object's silhouettes in the stereo images are segmented by a *blue screen* technique. A binary morphological closing and opening operation is used to remove noise in the image segmentation. The matching algorithm finds stereo correspondence only on the object areas in the left and right images. The object's silhouettes are also used later for volume intersections in a multiview integration process.

SSD (sum of squared difference)-based stereo matching is done at each level of the Gaussian pyramid from low resolution to high resolution. At the first level of the stereo matching, the initial search range of stereo disparity SR_0 at level 0 is set to [0,sr_0]. Then, at the lowest level of the pyramid, where $k = 2$, initial stereo disparity SR_2 becomes [0,sr_0/(2k)]. At successive levels of the pyramid, the result of the stereo disparity at the lower levels decides the search range of the corresponding level. If the disparity at the lower level is D_i, then the search range of the current level SR_i is restricted to within $[2*D_i-2, 2*D_i+2]$ so that the stereo matching algorithm can correct possible mismatches in the previous level. When there is a pair of stereo images, ${g_k}^{(l)}$ and ${g_k}^{(r)}$ for left and right images, which are at level k of the pyramid, SSD(i,j) at image coordinate (i,j) is

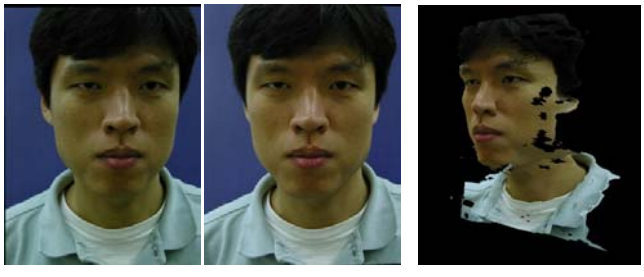
$$SSD(i,j) = \sum_{k=-m}^{m}\sum_{l=-m}^{m}\Big\{{g_k}^{(l)}(i,j) - {g_k}^{(r)}(i+k, j+l)\Big\}, \quad (9)$$

where $2m+1$ is the size of a matching block.

Figure 3a shows a pair of rectified stereo images of a human face, and Fig. 3b shows the result of 3D reconstruction. The depth to a 3D point is measured using the disparity between projected points in stereo images. The horizontal image offsets $\tilde{\mathbf{o}}_n$ of the principal points are taken into account for depth measure. In the left and right retinal planes, they are about $(0.0742, 0)$ and $(-1.299, 0)$ mm, respectively. In the next section, we compare two methods of depth computation from stereo disparity.

2.4 Depth from triangulation

After stereo rectification, we consider the new stereo configuration as a parallel stereo camera. Therefore, we use a simple



a **b**

Fig. 3. Stereo matching results. **a** Rectified left and right stereo images of a human face. **b** Texture-mapped range image

equation for depth computation. Let $\mathbf{p}_l$ and $\mathbf{p}_r$ be the projections of a 3D point $\mathbf{w}$ to the left and right retinal planes. If the disparity between two image points is d'_u in the x direction, the depth $\mathbf{w}_z$ to a 3D point from the origin of the camera coordinate system (CCS) is

$$\mathbf{w}_z = \frac{f \cdot B}{d'_u/k_u + (u_{n1} - u_{n2})}\,, \tag{10}$$

where $B = \|\mathbf{c}_1 - \mathbf{c}_2\|$ is the length of the baseline of the stereo camera; in our system it is 74.25 mm. u_{n1} and u_{n2} are x coordinates of the new principal points in the left and right images, respectively. For the focal length f of the camera we average the calibration results for both left and right focal lengths f_u, f_v, and it turns out to be 11.65 mm.

2.5 Depth of a linear equation

The depth from two conjugate image points is also reconstructed by using Eq. 1. Given two conjugate points $\tilde{\mathbf{p}}_1 = [u_1, v_1, 1]^T$ and $\tilde{\mathbf{p}}_2 = [u_2, v_2, 1]^T$ and the two projection matrices $\tilde{\mathbf{M}}_{n1}$ and $\tilde{\mathbf{M}}_{n2}$, we can write an overconstrained linear system:

$$\mathbf{A}\mathbf{w} = \mathbf{y}\,, \tag{11}$$

where

$$\mathbf{A} = \begin{bmatrix} (\mathbf{a}_1 - u_1\mathbf{a}_3)^T \\ (\mathbf{a}_2 - v_1\mathbf{a}_3)^T \\ (\mathbf{b}_1 - u_2\mathbf{b}_3)^T \\ (\mathbf{b}_2 - v_2\mathbf{b}_3)^T \end{bmatrix} \mathbf{y} = \begin{bmatrix} -a_{14} + u_1 a_{34} \\ -a_{24} + v_1 a_{34} \\ -b_{14} + u_2 b_{34} \\ -b_{24} + v_2 b_{34} \end{bmatrix}. \tag{12}$$

Then $\mathbf{w}$ gives the position of the 3D point projected to the conjugate points. Column vectors $\mathbf{a}_i$ and $\mathbf{b}_i$ are entry vectors of $\tilde{\mathbf{M}}_{n1}$ and $\tilde{\mathbf{M}}_{n2}$, respectively.

The 3D point $\mathbf{w}$ is represented with respect to the WCS. In this paper, however, we transform the world point to reference coordinates in order to represent it with respect to the CCS. Suppose we let the right camera's coordinate system (RCCS) be the reference. Then we can transform the point by simply using the external calibration parameters $[\mathbf{R}|\mathbf{t}]$ of the right cameras.

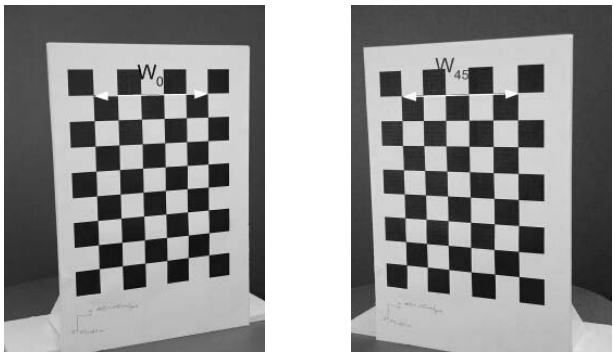
However, two transformations can be considered. One is to the old RCCS before rectification, and the other is to the new RCCS after rectification. By taking into account a multiview registration, which will be presented in the next section, we transform the point to the old RCCS by

$$\mathbf{p}_r = [\mathbf{R}_{o2}|\mathbf{t}]\,\mathbf{w}\,, \tag{13}$$

where $[\mathbf{R}_{o2}|\mathbf{t}]$ is the old external calibration parameters of the RCCS. Because the camera needs to calibrate the turntable for registration of multiview range images, we represent all range images with respect to the old RCCS.

2.6 Comparison of reconstruction methods

We compare the accuracy of the two 3D reconstruction methods. To compare the results with the ground truth, we use another checkerboard pattern to compute the 3D positions of all control points. The pattern is placed on the table and rotated by 0° and by 45°. We take a pair of stereo images at each angle, detect all corners, and compute the depth of the corners using the disparity between their conjugates.



a **b**

Fig. 4. A test checkerboard pattern at **a** 0° and **b** 45°

Table 1. Results of the width of the test pattern (ground truth is 100 mm)

Method	w_0	w_{45}
Linear eq. (mm)	99.6	99.2
Triangulation (mm)	104.1	99.7

As shown in Fig. 4, we measure the horizontal length between two control points at the upper-left and upper-right corners on the pattern. In order to minimize noise effects in the measurement, we also average all eight horizontal lengths between the leftmost and rightmost corners. The triangulation method using Eq. 10 shows a small reconstruction error. As shown in Table 1, there is also a difference between the two lengths measured at 0° and at 45°. This difference can cause serious problems if some multiview models are registered and integrated into a single 3D model. In fact, when we use this method to integrate multiple range images, a geometric distortion occurs on the 3D model. In contrast, using Eq. 11, we can reconstruct more accurately the 3D model. Table 1 shows that the linear equation method is more accurate than the triangulation method.

3 Turntable calibration

3.1 Rotation axis calibration

As presented in an earlier section, we employ a turntable to change the stereo camera's viewing direction to an object. To merge multiple range images, we need to know the rigid transformation of each image with respect to a common coordinate system. Because we calibrate the stereo camera only once before taking multiple range images, each range image obtained at different angles has an independent coordinate system. To register all multiview range images, we have to know the rigid motions between all viewpoints.

Suppose there are N viewing directions from $\mathcal{V}_0$ to $\mathcal{V}_{N-1}$ and the $\mathcal{V}_0$ is the reference viewpoint. When there is a 3D point $\mathbf{p}_i^i$ that is obtained and represented by the ith view point,

we can register it to a new point $\mathbf{p}_i^0$ in the reference view as follows:

$$\mathbf{p}_i^0 = \mathbf{T}_{cs}\mathbf{R}_i{\mathbf{T}_{cs}}^{-1}\mathbf{p}_i^i\,, \tag{14}$$

where $\mathbf{R}_i$ is the rotational transformation from $\mathcal{V}_i$ to $\mathcal{V}_0$, and $\mathbf{T}_{cs}$ is the transformation from the TCS to the CCS as shown in Fig. 5, which is represented by

$$\mathbf{T}_{cs} = [\mathbf{R}_{cs}|\mathbf{t}_{cs}]\,. \tag{15}$$

Let us define two independent coordinate systems in 3D space, the WCS and the TCS, whose origins are O_w and O_s, respectively. Suppose we know the transformation $\mathbf{T}_{cw}$, which is from the origin of the WCS O_w to that of the CCS O_c. If we know another transformation $\mathbf{T}_{ws}$ that is from the origin of the TCS O_s to that of the WCS O_w, then we can derive the transformation

$$\mathbf{T}_{cs} = \mathbf{T}_{cw}\mathbf{T}_{ws} \tag{16}$$

$$= [\mathbf{R}_{cw}|\mathbf{t}_{cw}]\,[\mathbf{R}_{ws}|\mathbf{t}_{ws}]\;. \tag{17}$$

Suppose there is the WCS in 3D space with its origin at O_w as shown in Fig. 6. In the figure, $\mathbf{p}_0$ is the origin of the WCS (but it is not necessary) and $\mathbf{p}_0'$ is the same point after being rotated by angle θ along the $\mathbf{Y}_s$ axis of the turntable. Given two 3D points and the rotation axis $\mathbf{Y}_s$, we can define a plane Π as shown in the figure. Then we know the vector product

$$(\mathbf{p}_0' - \mathbf{p}_0)\cdot\mathbf{Y}_s = 0\,. \tag{18}$$

In other words,

$$\begin{bmatrix} p'_{0x}-p_{0x} \\ p'_{0y}-p_{0y} \\ p'_{0z}-p_{0z} \end{bmatrix}^T \begin{bmatrix} Y_{sx} \\ Y_{sy} \\ Y_{sz} \end{bmatrix} = 0\,. \tag{19}$$

When we have at least three points in world coordinates, we can solve an overdetermined linear equation

$$\mathbf{AY} = \begin{bmatrix} p'_{0x}-p_{0x} & p'_{0y}-p_{0y} & p'_{0z}-p_{0z} \\ p'_{1x}-p_{1x} & p'_{1y}-p_{1y} & p'_{1z}-p_{1z} \\ & \cdots & \\ p'_{Nx}-p_{Nx} & p'_{Ny}-p_{Ny} & p'_{Nz}-p_{Nz} \end{bmatrix} \begin{bmatrix} Y_{sx} \\ Y_{sy} \\ Y_{sz} \end{bmatrix} = 0 \tag{20}$$

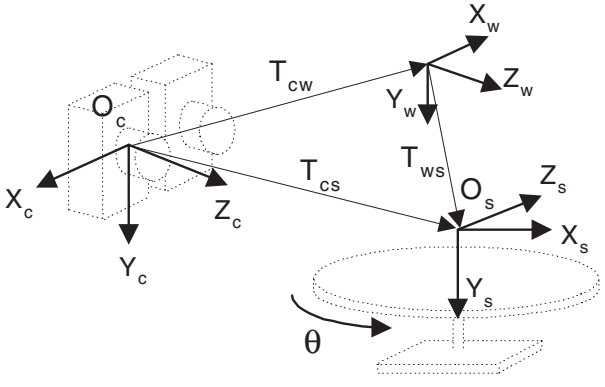


Fig. 5. Geometry of the vision system

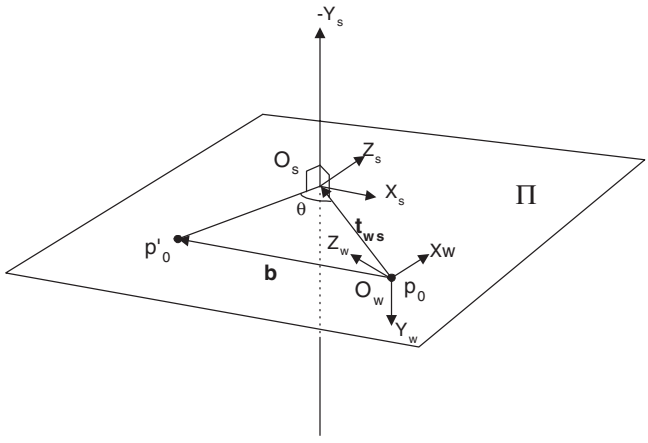


Fig. 6. Rotation axis calibration with respect to the WCS

using the SVD technique. When matrix $\mathbf{A}$ is decomposed such that $\mathbf{A} = (UDV^T)$, the solution of the equation is a column vector of $\mathbf{V}$ that corresponds to the column of the least eigenvalue in the $\mathbf{D}$ matrix. We then normalize vector $\mathbf{Y}_s$ to $\hat{\mathbf{Y}}_s$. If the computed Y_{sy} is negative, then we change the direction of the axis so that the axis is in the same direction as the $\mathbf{Y}_w$ axis of the WCS.

To compute the $\mathbf{X}_s$ and $\mathbf{Z}_s$ axes of the TCS, we apply the following computations. Let us initialize the $\mathbf{X}_s$ axis to $(1.0, X_{sy}, 1.0)$. Then

$$\begin{aligned} &\mathbf{X}_s\cdot\mathbf{Y}_s = 0\,, \\ &X_{sy} = (-X_{sx}Y_{sx} - X_{sz}Y_{sz})/Y_{sy}\,, \\ &\hat{\mathbf{X}}_s = \mathbf{X}_s/\|\mathbf{X}_s\|\,, \\ &\text{and}\;\; \hat{\mathbf{Z}}_s = \hat{\mathbf{X}}_s\times\hat{\mathbf{Y}}_s\,. \end{aligned}$$

Finally, the rotation matrix from the turntable to the WCS is defined as

$$\mathbf{R}_{ws} = \begin{bmatrix} (\hat{\mathbf{X}}_s)^T \\ (\hat{\mathbf{Y}}_s)^T \\ (\hat{\mathbf{Z}}_s)^T \end{bmatrix}^T\;. \tag{21}$$

Let us now consider a translation from the origin of the TCS to the origin of the WCS. The origin of the TCS is defined as the intersection of the axis $\mathbf{Y}_s$ and the Π plane. If we transform two 3D points $\mathbf{p}_0$ and $\mathbf{p}_0'$ using the rotation in Eq. 21, the transformed points are on the xz plane of the TCS.

Suppose the two points $\mathbf{p}_0$ and $\mathbf{p}_0'$ are transformed, with respect to the TCS, to new points $\mathbf{p}_{s0}$ and $\mathbf{p}'_{s0}$, respectively. Then the three points $\mathbf{O}_s$, $\mathbf{p}_{s0}$, and $\mathbf{p}'_{s0}$ are on the Π plane and form an isosceles triangle. Therefore, using a vector

$$\begin{aligned} &\mathbf{b} = \mathbf{p}'_{s0} - \mathbf{p}_{s0}\,, \\ &\text{where}\;\; (\mathbf{p}_{s0}, \mathbf{p}'_{s0}) = \mathbf{R}_{ws}^T(\mathbf{p}_0, \mathbf{p}_0') \end{aligned}$$

and the rotation angle θ, we can compute a translation vector $\mathbf{t}_{ws}$ from $\mathbf{p}_{s0}$ to the origin O_s.

Let us consider the Π plane on which the origin is moved to $\mathbf{p}_{s0}$ and the y component is zero. Then the center of rotation intersects with Π a 3D point $\mathbf{t}_i = [x, 0.0, z]^T$. Because the isosceles triangle is also on the plane, the origin $\mathbf{t}_i$ is one of the intersection points of the two circles c_1 and c_2, as shown in Fig. 7. On the Π plane, the center of c_1 is

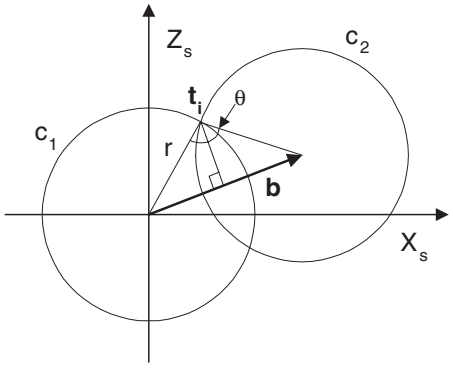
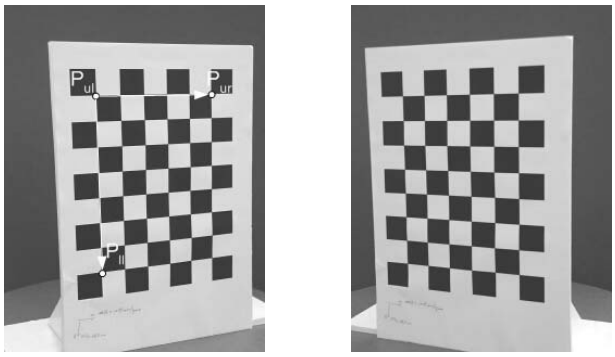


Fig. 7. The rotation center is one of the intersections of two circles c_1 and c_2



a **b**

Fig. 8. Checkerboard patterns for turntable calibration (**a**) 0°; (**b**) 45°

at $[0.0, 0.0, 0.0]^T$ and its diameter is $\|\mathbf{t}_i\|$. Similarly, the center of c_2 is at $[b_x, 0.0, b_z]^T$ and its diameter is also $\|\mathbf{t}_i\|$. Let $r = \|\mathbf{t}_i\|$ and $b = \|\mathbf{b}\|$; then we derive two circles' equations

$$x^2 + z^2 = r^2\,, \tag{22}$$
$$(x - b_x)^2 + (z - b_z)^2 = r^2\,.$$

Using the equations we get

$$z = \frac{b_x^2 + b_z^2 - 2b_x x}{2b_z}\,. \tag{23}$$

From Eq. 22 we also get

$$r^2 = x^2 + \frac{(b^2 - 2b_x x)^2}{4b_z^2}\,,$$
$$0 = 4b^2x^2 - 4b_x b^2 x + (b^4 - 4r^2 b_z^2)\,, \tag{24}$$

where $r = \dfrac{b/2}{sin(\theta/2)}$.

Therefore, the x coordinate of the two intersection points is the solution to the second-order binomial equation as in Eq. 24. And the z coordinate is computed by Eq. 23. Given two intersection points, only one of them is the real intersection point. If the intersection point is computed as $\mathbf{t}_i = [x, 0.0, z]^T$ on the Π plane, it should have a property such that

$$\mathbf{a} = \mathbf{b} \times \mathbf{t}_i,$$
$$\text{and } a_y > 0\,,$$

because we rotate point $\mathbf{p}_0$ by a positive angle θ along the Y_s axis of the turntable coordinates.

Let us now derive the transformation matrix from the TCS to the CCS. Because we shift the origin O_s to point $\mathbf{p}_{s0}$ to find point $\mathbf{t}_i$, the translation from O_w to O_s becomes $-(\mathbf{t}_i + \mathbf{p}_{s0})$ with respect to the TCS and $-\mathbf{R}_{ws}(\mathbf{t}_i + \mathbf{p}_{s0})$ with respect to the WCS. Finally, the transformation from the turntable to the CCS is computed as

$$\mathbf{T}_{cs} = \mathbf{T}_{ws}\mathbf{T}_{cw}$$
$$\text{where } \mathbf{T}_{ws} = [\mathbf{R}_{ws}|\mathbf{t}_{ws}] = [\mathbf{R}_{ws}| - \mathbf{R}_{ws}(\mathbf{t}_i + \mathbf{p}_{s0})]\,. \tag{25}$$

To reduce the noise effect on computing $\mathbf{t}_i$, we average the results of the vectors for some world points.

3.2 Turntable calibration experiments

To estimate the TCS, we use a checkerboard calibration pattern as shown in Fig. 8. We place the pattern on the turntable in such a way that the xy plane of the pattern faces the camera, leaving the rotation axis behind. Using the stereo camera, we take two pairs of stereo pictures at 0° and at θ°. Then we detect all 48 corner points in each picture. Using conjugate points in a pair of stereo pictures, we compute the 3D position of the corner points with respect to the RCCS.

Computing the transformation from the WCS to the CCS is done as follows. As shown in Fig. 8, the translation $\mathbf{t}_{cw}$ is the vector from the camera to the upper-left corner point $\mathbf{p}_{ul}$. The three axes of the WCS with respect to the camera system are computed as

$$\hat{\mathbf{r}}_{wx} = \mathbf{p}_{ur} - \mathbf{p}_{ul}/\|\mathbf{p}_{ur} - \mathbf{p}_{ul}\|\,, \tag{26}$$
$$\hat{\mathbf{r}}_{wz} = \mathbf{p}_{ll} - \mathbf{p}_{ul}/\|\mathbf{p}_{ll} - \mathbf{p}_{ul}\|\,, \tag{27}$$
$$\hat{\mathbf{r}}_{wz} = \hat{\mathbf{r}}_{wz} \times \hat{\mathbf{r}}_{wz}\,, \tag{28}$$
$$\text{and } \mathbf{R}_{cw} = \begin{bmatrix} \hat{\mathbf{r}}_{wx}^T \\ \hat{\mathbf{r}}_{wy}^T \\ \hat{\mathbf{r}}_{wz}^T \end{bmatrix}^T\,. \tag{29}$$

An example of the transformation matrix from the TCS to the CCS is computed as

$$\mathbf{T}_{cs} = \begin{bmatrix} 0.504629 & 0.013110 & -0.863265 & -11.003747 \\ -0.123068 & 0.990635 & -0.058155 & -72.315765 \\ 0.854513 & 0.135868 & 0.501416 & 487.191485 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{bmatrix} \tag{30}$$

We test our calibration algorithm at several positions of the stereo camera. Table 2 shows the registration error between two 3D control point sets, at 0° and at 45°, on the checkerboard pattern. The translation vector $\mathbf{t}_{cs}$ shows the distance from the CCS to the TCS.

4 3D model generation and error analysis

4.1 Multiview registration and integration

Using the range image acquisition and calibration techniques presented in earlier sections, we reconstruct 3D models of several real objects. We obtain multiple range images of an object from eight views of the object. After obtaining range images, we bring all of them to a common coordinate system

Table 2. Registration error of turntable calibration in mm

$\mathbf{t}_{cs}$			Mean error	Max. error
x	y	z		
−41.9	−64.6	393.7	0.21	0.53
−41.0	−69.3	420.6	0.21	0.53
−18.2	−80.8	488.9	0.15	0.39
−15.4	−93.3	569.4	0.11	0.20
4.93	−105.3	638.1	0.11	0.25
14.2	−120.4	727.1	0.18	0.35
21.9	−140.1	825.5	0.27	0.50

using the turntable calibration parameters. Registered range images are then refined again by using the point-to-plane registration technique we introduced in [16]. After registration, range images are integrated to obtain a 3D mesh model using a volumetric modeling technique [10,14,15]. From multiview range images of an object we find the implicit surface of the object by computing the signed distance of a voxel to the surface of the object. The implicit surface is then converted to a 3D mesh model by the Marching Cubes algorithm [10]. More details of our multiview modeling techniques can be found in [14,15].

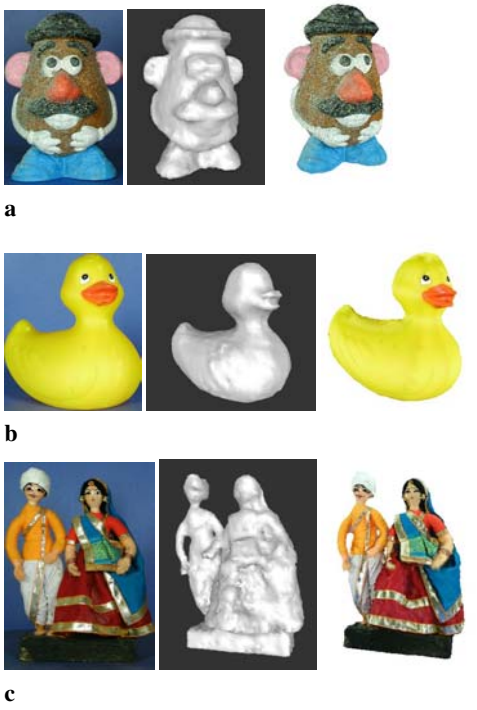
4.2 3D model results

Figure 9 shows 3D models of three real objects. The first column shows pictures of the objects, the second column shows surface representations of the reconstructed models, and the third column shows texture-mapped 3D models. If an object has little contrast on its surface, we use a slide projector to introduce a random dot pattern to enhance the performance of stereo matching. The object in Fig. 9c is very complex and difficult to reconstruct. It has non-Lambert surfaces and some concavities. We merge 16 multiview range images in this case. Texture-mapped 3D models show photorealistic reconstruction of the objects.

Table 3 shows the processing time to generate 3D models of the objects. The modeling time actually depends on the resolution of the 3D grid of voxels in the Marching Cubes algorithm is set. This table shows only some examples, where the voxel size of the objects is 3 or 4 mm. Total processing time is about 5 to 8 min depending on the complexity and size of the objects and on the number of views. In the "duck" object, we use the slide projector to take stereo pictures with a random dot pattern. After taking multiview stereo pictures with normal illumination, we take another set of stereo pictures again with the random dot pattern. Image acquisition for this object takes twice that of the "Mr. Potatohead" object.

4.3 Reconstruction error analysis

To analyze the accuracy of our modeling system, we reconstruct 3D models of two ground truth objects. Two test objects are shown in Figs. 10 and 11. One is a rectangular parallelepiped, and the other is a cylinder. We reconstruct 3D models of them and measure dimensions of the models to compare



a

b

c

Fig. 9. Reconstruction results of real objects. *Left* to *right*: Picture of objects, surface models, texture-mapped models. **a** Mr. Potatohead. **b** Duck. **c** Indian couple

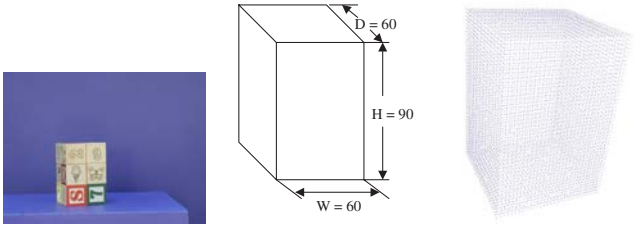
Table 3. 3D model generation time (s)

Object	Mr. Potatohead	Duck	Indian couple
Voxel size (mm)	4	4	3
No. of triangles	8284	6760	18660
No. of views	8	8	16
Image acquisition	90	180	180
Rectification	32	45	45
Stereo matching	150	130	160
Registration	12	10	6
Integration	15	6	23
Total	299	369	414

with those of the ground truth models. We choose these two objects because their dimensions are easily measured.

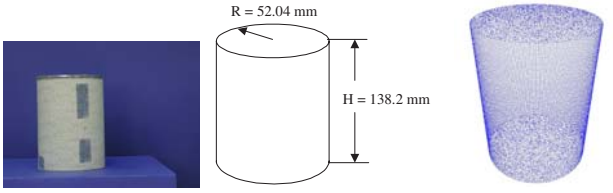
We use an ICP-based registration technique to first register point clouds of the reconstructed 3D model to that of the ground truth model. Then dimensional errors are measured between all points on the model and their closest conjugates on the ground truth. We iteratively register the reconstructed 3D model to its ground truth until the registration error between the 3D model and the ground truth converge to a constant value. As an error metric, we measure the average distance of the closest points between the two models. Figure 12 shows the results of registering two objects. We use 392 control points in the "cubes" object and 104 points in "cylinder."

After the two 3D models – the reconstructed model and the ground truth model – are registered, we measure dimensional errors on the reconstructed model with respect to the ground truth model. For the "cubes" object, the RMS errors



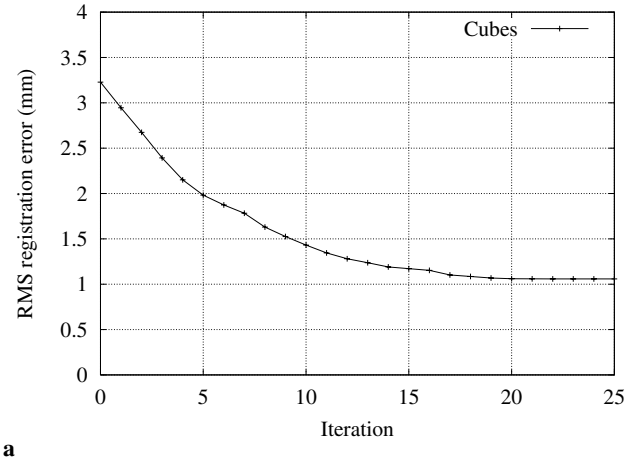
a **b** **c**

Fig. 10. "Cubes" object for error analysis. (**a**) Picture. (**b**) Dimension (mm). **c** Point clouds model

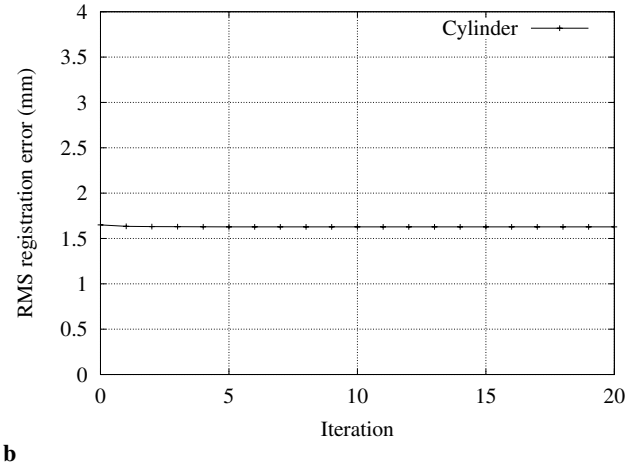


a **b** **c**

Fig. 11. "Cylinder" object for error analysis. **a** Picture. **b** Dimension (mm). **c** Point clouds model



a



b

Fig. 12. RMS registration error between the ground truth and the reconstructed 3D models. **a** Cubes. **b** Cylinder

Table 4. RMS and maximum errors of "cubes"

Dimension	W (mm)	D (mm)	H (mm)	V (mm^3)
Size	60	60	90	324000
RMS error	1.06	0.90	0.85	330542
MAX error	2.99	3.03	2.17	
(%) error (RMS)	1.76	1.50	0.94	2.04

Table 5. RMS and maximum errors of "cylinder"

Dimension	H (mm)	R (mm)	V (mm^3)
Size	138.2	52.04	1175797.4
RMS error	1.54	1.20	1179466.7
MAX error	4.56	4.42	
(%) error (RMS)	1.11	2.29	0.31

in the W, H, and D dimensions are measured for all vertices on corresponding planes – for example the top and the bottom planes for the H dimension – with respect to the closest vertices on the ground truth. Similarly for the "cylinder" object, errors in the R and H dimensions are measured using points on side surfaces, and top and bottom surfaces, respectively. Table 4 shows RMS and maximum errors in all dimensions of the "cubes" object. We also measure the volume V of the reconstructed model using a volume-measuring technique described in a reference paper [11]. Table 5 shows the results of the "Cylinder" object.

5 Conclusions

We have introduced a stereo vision system to automatically generate 3D computer models of real objects. The system consists of an inexpensive stereo camera, a turntable, and a personal computer. Calibration of the stereo camera and the turntable stage is presented. We rectify stereo images and obtain range images of an object from multiple viewpoints. Those range images are then automatically registered to a common coordinate system and integrated into a 3D mesh model. We have introduced a new turntable coo rdinate system and a simple and accurate calibration technique. Reconstruction error analysis shows the accuracy of our 3D reconstruction.

References

1. Allen P, Yang R (1998) Registering, integrating, and building CAD models from range data. In: IEEE international conference on robotics and automation, pp 3115–3120
2. Burt P (1983) The Laplacian pyramid as a compact image code. IEEE Trans Commun 31(4):532–540
3. Chen Q, Medioni G (1999) A volumetric stereo matching method: applications to image-based modeling. In: Proceedings of the conference on computer vision and pattern recognition
4. Curless B, Levoy M (1996) A volumetric method for building complex models from range images. In: Proceedings of SIGGRAPH, pp 303–312
5. Fitzgibbon A, Cross G, Zisserman A (1998) Automatic 3D model construction for turn-table sequences. In: European workshop SMILE'98. Lecture notes in computer science, vol 1506. Springer, Berlin Heidelberg New York, pp 155–170

6. Fussiello A, Trucco E, Verri A (2000) A compact algorithm for rectification of stereo pairs. Mach Vis Appl 12:16–22
7. Huber D (2001) Automatic 3D modeling using range images obtained from unknown viewpoints. In: Proceedings of the 3rd international conference on 3D digital imaging and modeling. IEEE Press, New York pp 153–160
8. Lander P (1998) A multi-camera method for 3D digitization of dynamic, real-world events. PhD dissertation, Carnegie Mellon University, Pittsburgh, PA
9. Levoy M et al (2000) The digital Michelangelo project: 3D scanning of large statues. In: Proceedings of SIGGRAPH, pp 131–144
10. Lorensen W, Cline H (1987) Marching Cubes: a high resolution 3D surface construction algorithm. Comput Graph 21(4):163–169
11. Mirtich B (1996) Fast and accurate computation of polyhedral mass properties. J Graph Tools 1(2):31–50
12. Niem W (1999) Automatic reconstruction of 3D objects using a mobile camera. Image Vis Comput 17:125–134
13. Okutomi M, Kanade T (1993) A multiple-baseline stereo. IEEE Trans Pattern Anal Mach Intell 15(4):353–363
14. Park S, Subbarao M (2002) Automatic 3D model reconstruction using voxel coding and pose integration. In: Proceedings of the international conference on image processing, pp 533–536
15. Park S, Subbarao M (2003) Automatic 3D reconstruction based on novel pose estimation and integration techniques. Image Vis Comput 22(8):623–635
16. Park S, Subbarao M (2003) An accurate and fast point-to-plane registration technique. Pattern Recog Lett 24 (16):2967–2976
17. Pulli K, Abi-Rached H, Duchamp T, Shapiro L, Stuetzle W (1998) Acquisition and visualization of colored 3D objects. In: IEEE international conference on pattern recognition, pp 11–15
18. Rander P, Narayanan P, Kanade T (1996) Recovery of dynamic scene structure from multiple image sequences. In: Proceedings of the international conference on multisensor fusion and integration for intelligence systems, pp 305–312
19. Rusinkiewicz S, Hall-Holt O, Levoy M (2002) Real-time 3D model acquisition. In: Proceedings of SIGGRAPH2002
20. Scharstein D, Szeliski R (2002) A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int J Comput Vis 47(1–3):7–42
21. Seitz S, Dyer CR (1997) Photorealistic scene reconstruction by voxel coloring. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1067–1073
22. Vedula S, Rander P, Saito H, Kanade T (1998) Modeling, combining, and rendering dynamic real-world events from image sequences. In: Proceedings of the 4th conference virtual systems and multimedia, 1:326–332
23. Wheeler M, Sato Y, Ikenuchi K (1998) Consensus surfaces for modeling 3D objects from multiple range images. In: IEEE conference on computer vision, pp 917–924
24. Zhang L, Curless B, Seitz S (2002) Rapid shape acquisition using color structured light and multi-pass dynamic programming. In: 1st international symposium on 3D data processing, visualization, and transmission, pp 24–36

Soon-Yong Park obtained his B.S. and M.S. in Electronics Engineering from Kyungpook National University, Taegu, Korea, in 1991 and 1993, respectively. He obtained his Ph.D in Electrical and Computer Engineering from State University of New York (SUNY) at Stony Brook in 2003. From 1993 to 1999, he was a senior research staff in the Advanced Robotics Lab. at Korea Atomic Energy Research Institute. He is currently a Postdoctoral Research Associate in the Department of Electrical and Computer Engineering at SUNY at Stony Brook. His research interests include 3D Sensing and Modeling, 3D Pose Estimation, and Computer Graphics.

Murali Subbarao obtained B. Tech. in Electrical Engineering from the Indian Institute of Technology, Madras, and M.S. and Ph.D. in Computer Science from the University of Maryland, College Park. He joined the faculty of the Department of Electrical and Computer Engineering, SUNY at Stony Brook, soon after his Ph.D. He is the Founder and Director of the Computer Vision Laboratory in the department. His research and teaching areas include Computer Vision, Digital Image Processing, Software Engineering, Digital Systems Design, and Web and Internet Technology.