1 Introduction

Observation-based modeling automatically constructs solid models from real objects using computer vision techniques, as an alternative to building them manually. Applications of observation-based modeling include, among others, creating models for animation, reconstructing human body parts for surgical planning, recovering machine parts for virtual factory simulation, building CAD models for model-based recognition.

Usually Observation-based modeling systems work with a sequence of images of the object(s), where the sequence spans a smoothly varying change in the positions of the sensor and/or object(s). The task of observation-based modeling is essentially the problem of merging multiple views using an appropriate representation. Many previous observation-based modeling techniques involve motion estimation between successive pairs of views in a sequential manner [13][17][22] as shown in Figure 1(a). Whenever a new view is introduced, it is matched with the previous view. The transformation between these two successive views is estimated before the object model is updated. This sequential method usually does not work well in practice because of errors in local motion estimation due to noise and missing data. These errors accumulate and are propagated along the sequence, yielding erroneous object models.

In this paper, we present a new technique for observation-based modeling of free-form objects. This technique, called *integral approach,* is illustrated in Figure 1(b). Rather than sequentially integrating successive pairs of views, we propose to reconstruct a statistically optimal object model that is simultaneously most consistent with all the views. Our method makes use of significant redundancy existing among all the views, i.e., it is likely that any part of the object will be observed a number of times along the sequence of images although each single view provides only partial information. The key idea of integral object modeling is to enable a system to integrate a complete object model in terms of observable features.

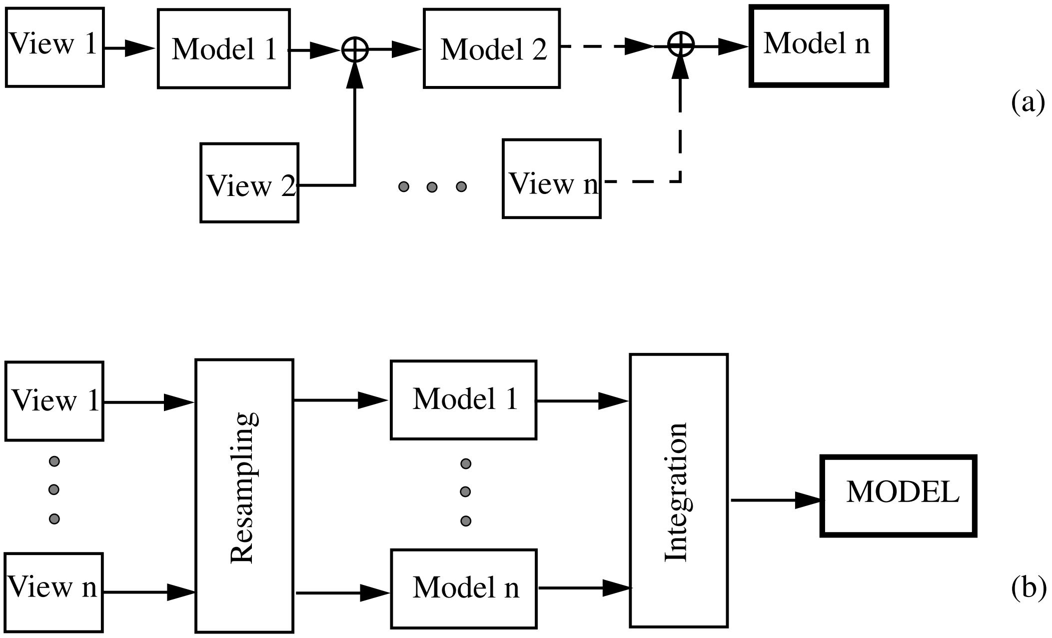


Figure 1 **Observation-based model systems: (a) sequential modeling; (b) integral modeling**

In previous work, we have applied the integral approach to polyhedral object modeling, where polyhedral object models and view transformations are recovered simultaneously by employing the principal component analysis with missing data (PCAMD) algorithm [20]. Unlike planar patches in polyhedral object modeling [20], however, correspondence of range images of curved surfaces is a difficult problem. Due to discrete sampling, data points in two different views generally do not correspond with each other. On the other hand, salient features such as high curvature points do correspond with each other but can not be extracted reliably from noisy range data.

We propose to determine correspondence by resampling each view of the free-form object. In particular, this resampling process combines top-down topological knowledge (a spherical surface representation) with bottom-up geometrical properties (approximated local curvature). For each range image, we first build a discrete mesh that approximates the object's surface, and encodes the local curvature at each node. The local curvature at each node is integrated from its neighborhood. During the process of mesh approximation, local connectivity among different nodes is always preserved, i.e., each mesh node has exactly three neighbors. Mesh matching is based on the local curvature distribution at each mesh node. Global resampling is applied on the spherical mesh coordinate system to establish one-to-one correspondence among mesh nodes from multiple views.

1.1 Organization of the Paper

The remainder of this paper is organized as follows. After reviewing previous work on modeling from multiple range images in Section 2, we introduce our integral approach to object modeling in Section 3. We show that the integral modeling can be viewed as a solution to a combination of two subproblems: *what to integrate* and *how to integrate*. In Section 4 we address the problem of "what to integrate" by presenting a novel global resampling scheme which can be used to determine correspondence among different views. Each range image is resampled using a global spherical representation. Correspondence is established by matching the local curvature distribution in different views. In Section 5 we show "how to integrate" by solving a two-step weighted least-squares problem from the measurement matrix which is formed by resampling the whole sequence of range images. We demonstrate the robustness of our proposed PCAMD method by applying it to synthetic data and real range images. Experimental results using synthetic data of a free-form object indicate that our approach is robust with respect to noise and surface mismatching. From sequences of real range images, free-form object models are accurately reconstructed using our new approach. We close with a discussion of future work along with a summary.

2 Previous Work

A significant amount of work has been done in object modeling from a sequence of range images. Bhanu [4] modeled object by rotating it through known angles. Ahuja and Veenstra [1] constructed an octree object model from orthogonal views. By finding the correspondences from intensity patterns in all eight views, Vemuri and Aggarwal [24] derived the motion and transformed all eight range images with respect to the first frame. Most work assumed that the transformation between successive views is either known or can be recovered, so that all data can be transformed with respect to a fixed coordinate system.

To accurately recover the transformation between two views, different range image registration techniques using various features have been proposed. Ferrie and Levine [12] used correspondence points which were identified by correlation over the differential properties of the surface. Wada et al. [25] employed facets of an approximated convex hull of range images. Parvin and Medioni [17] proposed the construction of boundary representation (B-rep) models using segmented surface patches. The difficulty of feature-based registration is in realizing robustness, especially in the case of free-form objects.

Many algorithms also exist for featureless range data point matching. Besl and Kay [3] used the iterative point matching (ICP) method to project points from one surface to another during matching. A similar approach was proposed by Chen and Medioni [6]. Zhang [28] improved Besl and Kay's ICP algorithm by using robust statistics and eliminating the requirement that one surface be a strict subset of the other. Champleboux et al. [5] used the Levenberg-Marquart nonlinear minimization algorithm to minimize the sampled distance to surface using octree-splines.

These featureless registration algorithms are locally optimal; they work well for free-form objects only if a good initial transformation is given. Higuchi, Hebert and Ikeuchi [13] proposed a registration method which eliminates problems for both feature-based and featureless methods. Their method builds discrete spherical meshes representing the surfaces observed in each range image, and computes local curvature at each mesh node. The local curvature information is extracted reliably because of the use of local regularity where each mesh node has exactly three neighbors. Registration of different images is then achieved by comparing local curvature distribution of spherical meshes.

After transforming all range images to a world coordinate system using the registration result, an object model is usually obtained by running a connectivity algorithm (such as the Delaunay triangulation [10]) at the last step. Hoppe et al. [14] used graph traversal methods. Connectivity can also be modified and determined as more views are incorporated. Parvin and Medioni [17] used an adjacency graph to represent the connectivity of each segmented view. In their adjacency graph, nodes represent surface patches with attributes, and arcs represent adjacency between surfaces. Soucy and Laurendeau [21] made use of the structured information about where the images are taken; they proposed to triangulate each view and merge multiple views via a Venn diagram when the transformation is known. The common parts of different views are then re-sampled. However, constructing such a Venn diagram is combinatorial in nature (only four-view merging is presented in their work). Turk and Levoy [22] proposed a similar approach but avoided the problem of Venn diagram construction by merging only two adjacent views at each step.

Most of the previous approaches to modeling from a sequence of views are sequential. To merge any two views, a rigid transformation has to be computed accurately. Thus, transformation errors accumulate and propagate from one matching to another, which may result in noticeably imprecise object models. To make use of the redundancy among multiple views, Shum, Ikeuchi and Reddy [20] showed that multiple view merging can be formulated as a weighted least-squares problem and applied a PCAMD algorithm to polyhedral object modeling.

3 An Integral Approach to Object modeling

There exist several problems for observation-based object modeling. Data acquisition and image registration introduce significant errors. Range images have missing data points due to occlusion and self-occlusion [2]. Several important issues related to the realization of a practical observation-based modeling system have not been resolved satisfactorily. These issues include:

- What kind of representation should be used to model the object?
- How can a sequence of images be integrated?
- Can model reconstruction be made statistically optimal?
- How can objects be sampled sufficiently and unambiguously?

To solve two of the essential problems associated with observation-based object modeling, *what to integrate* and *how to integrate*, we propose a new approach termed *integral object modeling*. The key idea of integral object modeling is to enable a system to resample a sequence of images of an object in terms of observable features, and then integrate them using principal component analysis with missing data (PCAMD). Integral object modeling works by integrating the partial observation provided by each view to yield a complete object model. Although object modeling from a sequence of images is nonlinear and possibly ill-conditioned, the integral approach makes use of redundant data so that statistically optimal reconstruction is feasible.

3.1 What to Integrate?

To integrate multiple views in a statistically optimal fashion, we first need to determine what is to be integrated from a sequence of range data. In our previous work on polyhedral object modeling [20], we can successfully make use of the redundancy in multiple views because planar surface patches can be tracked over the sequence. Ideally, for free-form object modeling, we would like to register each data point from one view to another. However, the physical correspondences between data points in two different views are usually not known a priori. As a result, we have to search for some salient features which do have correspondence among different views.

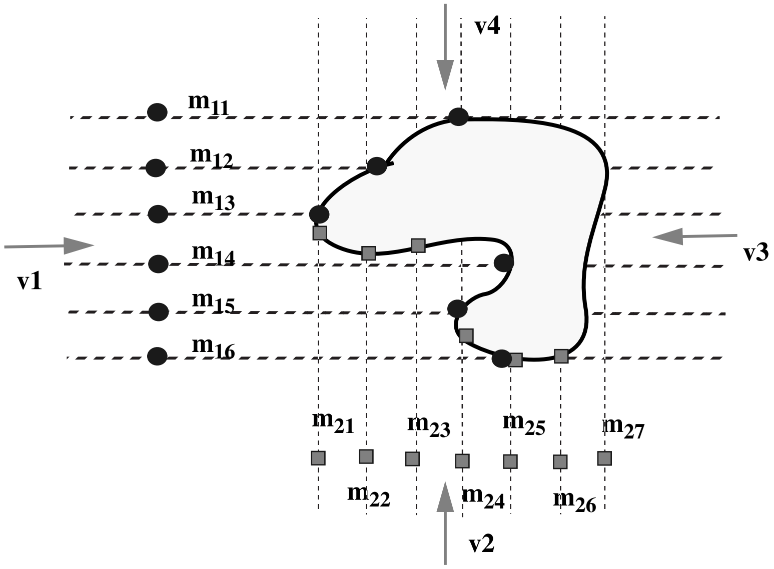


Figure 2 **The problem of resampling (black dots and grey squares are measurements in the sensor coordinates obtained from range sensor readings in different views).**

To appreciate the difficulties in free-form object modeling, we examine, in Figure 2, a simple example of a free-form curve observed from multiple views. Two major problems exist due to discrete and noisy sampling in practice: lack of physical correspondence among sampling points in different views, and lack of global sampling for all sampling points. For example, the measurements of view 1, m_{11}, ..., m_{16} do not necessarily correspond to other measurements in other views. In addition, ordering information is unknown among all sampling points because of the non-existence of global sampling. In other words, we have to somehow figure out the proper linkage between the points in different views (e.g., m_{13}, m_{21}, m_{22}, m_{23}, m_{14}, ...) in order to recover an accurate model for the curve. The situation in the three dimensional world is even more complicated because the result of transforming all range data from multiple views into a single coordinate system is a “cloud” of noisy 3D points. It is difficult to make a surface model from this cloud of noisy 3D points.

Because curvature is invariant under Euclidean group transformations, we would have physical correspondence among distinguished high curvature points in different views. In practice, however, from a particular viewing direction we usually obtain only a finite number of sampling points of a surface. These points are measured in the sensor coordinate system. The normals, or the tangent planes, as well as the curvatures, are not explicitly or directly known. Although surface normals can be approximated by fitting local planar surface patches of points, the process of computing curvature at each point using range data is known to be highly noise sensitive and may be numerically unstable.

To address the problem of what to integrate from multiple range images of known topology object, we propose a global resampling scheme in Section 4. Our global method resamples the object by combining top-down knowledge of topological information (spherical representation) and bottom-up geometrical information (range data and local curvature) at each view. The details of our global resampling method, including spherical surface representation, deformable mesh generation and one-to-one mesh node correspondence, are described in Section 4.

3.2 How to Integrate?

Our task is to model a free-form object from a sequence of range images. Suppose that the model consisting of *20* nodes is observed from *4* views (Figure 3). If the correspondence among *20* mesh nodes of the object over *4* views is obtained using global resampling, we may form a 16×20 measurement matrix

$$W = \begin{bmatrix} \mathbf{p}_1^{(1)} & \mathbf{p}_2^{(1)} & \mathbf{p}_3^{(1)} & \mathbf{p}_4^{(1)} & \dots & \mathbf{p}_{14}^{(1)} & \mathbf{p}_{15}^{(1)} & * & \dots & * \\ \mathbf{p}_1^{(2)} & \mathbf{p}_2^{(2)} & \mathbf{p}_3^{(2)} & \mathbf{p}_4^{(2)} & \dots & * & \mathbf{p}_{15}^{(2)} & \mathbf{p}_{16}^{(2)} & \dots & * \\ \mathbf{p}_1^{(3)} & \mathbf{p}_2^{(3)} & * & * & \dots & \mathbf{p}_{14}^{(3)} & * & \mathbf{p}_{16}^{(3)} & \dots & * \\ * & * & * & * & \dots & * & \mathbf{p}_{15}^{(4)} & \mathbf{p}_{16}^{(4)} & \dots & \mathbf{p}_{20}^{(4)} \end{bmatrix} \qquad \text{(EQ 1)}$$

where $\mathbf{p}_p^{(f)} = \left(x_p^{(f)}, y_p^{(f)}, z_p^{(f)}, 1\right)^T$, $f=1,...,4$, $p=1,...,20$ represents homogeneous coordinates of vertex p at frame f, and every * indicates an unobservable point. The task now is to recover the positions of all *20* points in a fixed coordinate system.

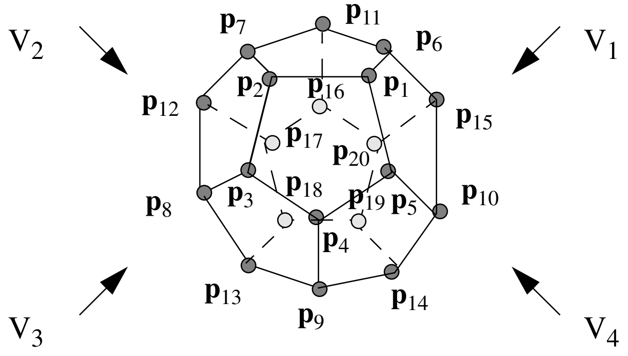


Figure 3 **Distinct views of a free-form object with 20 mesh nodes and known connectivity**

If the measurement matrix were complete, our task would be to average all those *20* points over *4* views, assuming that the input data is noisy. The standard way to solve this problem is to apply the singular value decomposition SVD to the matrix W, whose rank is at most *4*. The measurement matrix can subsequently be factorized, with proper normalization, into a left matrix Q of transformation parameters and a right matrix P of plane coordinates $W = QP$, where

$$P = \begin{bmatrix} \mathbf{p}_1 \ \mathbf{p}_2 \ \cdots \ \mathbf{p}_{20} \end{bmatrix}, Q = \begin{bmatrix} Q^{(1)} \\ Q^{(2)} \\ Q^{(3)} \\ Q^{(4)} \end{bmatrix}, \qquad \text{(EQ 2)}$$

$Q^{(f)}$ is the transformation of *f*th view with respect to the fixed world coordinate system, and $\mathbf{p}_p$ is the *p*th vertex in the same world coordinate system.

Unfortunately, in practice the measurement matrix is often incomplete; it is not unusual for a large portion of the matrix to be unobservable. When the percentage of missing data is very small, it is possible to replace the missing elements by the mean or by an extreme value; this is a useful strategy in multivariate statistics [9]. However, such an approach is no longer valid when a significant portion of the measurement matrix is unknown.

In the sequential modeling from multiple range images, we have to first recover the transformation between view *1* and view *2* if there are at least three matched points that are non-planar [11]. Then using the recovered transformation, we obtain the invisible points (mesh nodes) in view *1* from its corresponding points in view *2* which are visible. This process is repeated for the whole sequence. The major problem with the sequential method is that once an error in the estimated transformation occurs at any step, the resulting erroneous results will accumulate and propagate all the way along. To obviate this problem, Shum, Ikeuchi and Reddy [20] proposed the PCAMD approach making use of more rigorous mathematical tools developed in computational statistics. These tools cater for missing data without resorting to error sensitive extrapolation. A closed related work was proposed by Poelman and Kanade [18] in shape from motion where they assigned a confidence factor to each measurement to handle occlusion. In Section 5 we will review the PCAMD approach and apply it to free-form object modeling from multiple view merging.

4 What to Integrate: Global Resampling

In this section, we describe the concept of global resampling of a free-form object. For each range image, we first build a discrete mesh that approximates the object's surface, and encodes the local curvature. Mesh matching is based on the local curvature distribution at each mesh node. Using the spherical mesh coordinate system, global resampling is used to establish one-to-one correspondence among mesh nodes from multiple views.

The concept of global resampling is based on previous work on spherical surface representation and matching for object recognition [8]. We describe only the basic approach of the spherical surface representation and the main result of curvature-based matching, and refer the reader to [8] for a detailed description of the algorithms.

4.1 Surface Representation Using Deformable Mesh

Many deformable models, such as irregular meshes [23], finite element models [16] and balloon models [7] have been proposed for 3D shape reconstruction. Our basic surface representation is a discrete connected mesh that is homeomorphic to a sphere. The free-form objects are restricted to a topology of genus zero (i.e., no holes) in this paper. A distinctive feature of our surface representation is its global structure, i.e., the connectivity of the mesh is such that each node has exactly three neighbors. The total number of mesh nodes depends on the resolution of the mesh.

Different attributes may be associated with each node of the mesh. The most prominent attribute for the purpose of building models is the curvature. More precisely, an approximation of curvature, called the simplex angle [8], is computed at each node from its position and the positions of its three neighbors. We denote by $g(P)$ the simplex angle at node P.

Given a set of data points from a range image, the mesh representation is constructed by first placing a spherical or ellipsoidal mesh at the approximate center of the object, then iteratively deforming the shape of the mesh in response to “forces” generated by the data points and the image features, as well as internal smoothness constraints.

An example of reconstruction of a surface model from a partial view is shown in Figure 4. The mesh is always a closed surface. However, since only part of the object's surface is visible from a given viewpoint, some of the mesh nodes do not correspond to visible surface data. These nodes, corresponding to occluded part of the object, are flagged as interpolated nodes so that they can be treated differently when used. Figure 5 shows the same deformable surface with and without interpolated portion.

4.2 Mesh Matching

In order to compare two meshes, we need to find the correspondence between the two sets of mesh nodes. In general, such a correspondence may not exist because the distributions of nodes on the two surfaces may be completely different. To obviate this problem, we introduce a regularity constraint on the distribution of the nodes on the surface. This constraint can be evaluated locally at each node which has exactly three neighbors and is incorporated in an iterative fitting algorithm as an additional force. When this constraint is enforced, meshes representing the same surface viewed from different viewpoints have the property that their nodes do correspond. Higher resolutions of semi-tessellated sphere [8] can be used to ensure the existence of mesh correspondence.

Given meshes from two different views, the matching proceeds by comparing the values of the curvature at the nodes of the two meshes. Specifically, if $\{P_i\}$ and $\{Q_j\}$ are the nodes of the two meshes defined in the sensor coordinates, the matching problem can be defined as finding the set of correspondences $C = \{(P_i, Q_j)\}$ such that the value of the curvature at any node P_i is as close as possible to the value of the corresponding node Q_j in the other mesh. Specifically, C is the correspondence that minimizes the “distances” between two meshes:

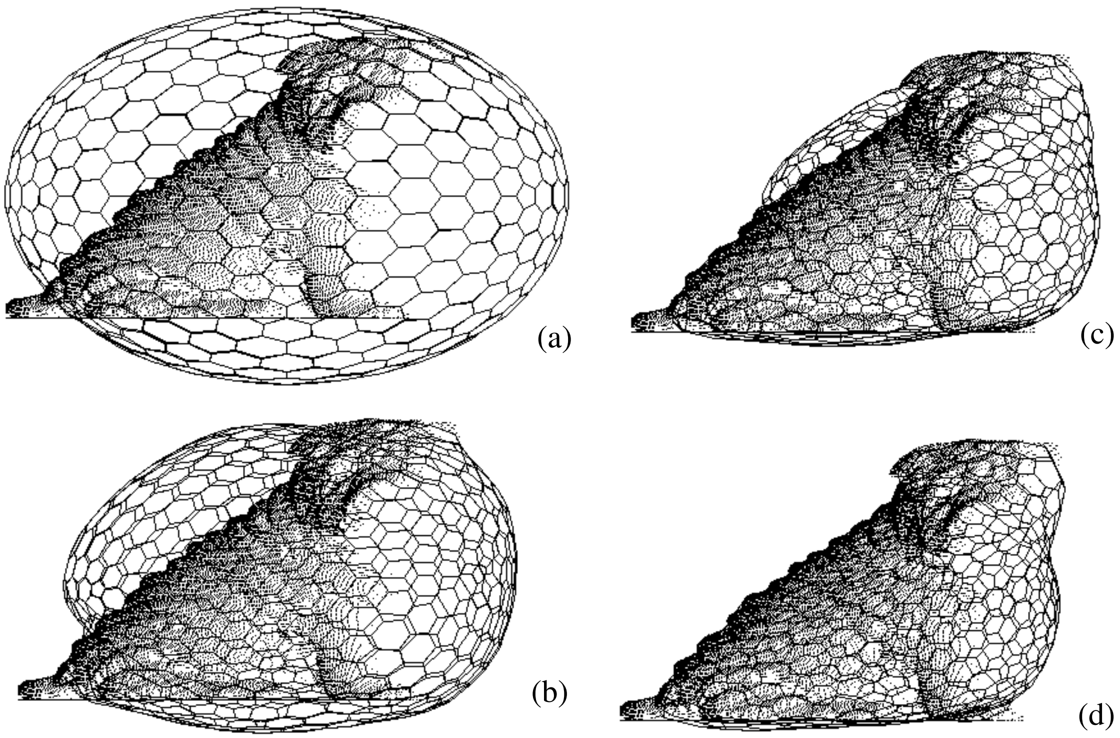


Figure 4 **Deformable surface reconstruction at different iteration steps (dots are range data, solid lines are mesh models) (a) n=0 (start of deformation); (b) n=20; (c) n=50; (d) n=100 (end of deformation).**

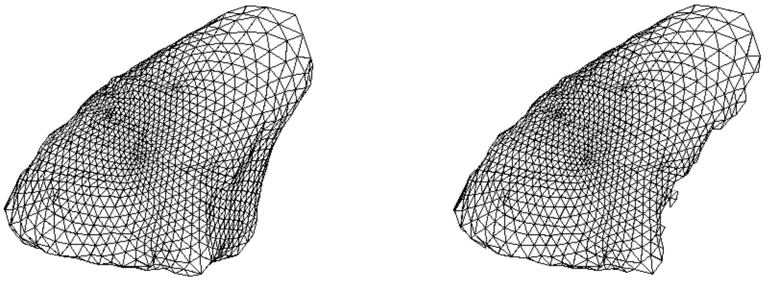


Figure 5 **Deformable surface. (a) with interpolated part; (b) without interpolated part**

$$D(S, S') = \sum_{C} (g(P_i) - g(Q_j))^2 \qquad \text{(EQ 3)}$$

where *g(P)* is the simplex angle at a mesh node *P*. Since the local curvature is independent of the translation, we can search for the best match *C* only in the rotation space. The connectivity among all mesh nodes can further simplify the mesh matching [13].

Once the best set of correspondences *C* is obtained, we compute an estimate of the transformation *(R, T)* between the two views by minimizing the distance:

$$\sum_{C} \left\| P_i - (RQ_j + T) \right\|^2 . \qquad \text{(EQ 4)}$$

The resulting transformation *(R, T)* is only an initial estimate because the nodes P_i and Q_j do not correspond exactly due to the resolution of the mesh. In object recognition, mesh discreteness is not a problem because object recognition is based only on the goodness of fit between the model and the scene data. In object modelling, however, the estimate is accurate enough to be used as a starting point for the ICP algorithms [3][28], for example. We will show in Section 5 how the estimate of *(R, T)* and the node correspondence set *C* are used as input to the PCAMD algorithm to integrate multiple views.

4.3 One-to-one Correspondence

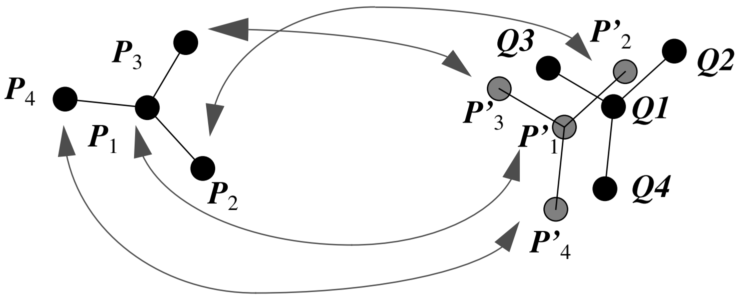
In order to apply the PCAMD to take advantage of redundancy of multiple views, we need one-to-one mapping among different views. The matching procedure above, however, does not guarantee one-to-one correspondence between two sets of mesh nodes generated from two views because of the discrete sampling and the requirement of local regularity.

We establish the one-to-one correspondence by resampling each deformable surface using the global spherical coordinate. Once a set of spherical mesh nodes (along with its local curvature attributes) is obtained, it is possible to interpolate any point on the spherical coordinate from this set. For example, in Figure 6, although there exist many-to-one mappings between P and Q (both P_1, and P_2 are matched to Q_1), mapping between P' and P is one-to-one because P' results from the rotation of P. The new mesh node at P'_1 and its SAI value can be interpolated from its nearest point Q_1 on set Q and three neighbors of Q_1. Let $g(P')$ and $g(Q_1)$ be the values of the simplex angles at node P' and its nearest node Q_1, respectively. We have the following local interpolation:

$$g(P') = \sum_{i=1}^{4} w_i Q_i \qquad \text{(EQ 5)}$$

where Q_2, Q_3, and Q_4 are three neighbors of Q_1, and w_i are the weights depending on the distance between P' and Q_i $(i=1,2,3,4)$. The coordinates of the mesh node at P' can be interpolated in the same way. Because of the one-to-one mapping between any two views, one-to-one mapping among multiple views can be established. When the measurement matrix is ready, we can apply the PCAMD algorithm to compute the complete set of mesh nodes and transformations among different views. The object model is then obtained based on the known connectivity among all mesh nodes.

(a)



(b)

node number →

1	2	3	4	---	K
P_1	P_2	P_3	P_4	---	P_K
P'_1	P'_2	P'_3	P'_4	---	P'_K
Q_1	Q_1	Q_3	Q_4	---	Q_K

Figure 6 **: One-to-one matching: (a) Valid correspondence between nodes; (b) Table of correspondences**

5 How to Integrate: Principal Component Analysis with Missing Data

The problem of object modeling from a sequence of views shown in Section 3 can be formulated as a problem of principal component analysis with missing data (PCAMD), which has been extensively studied in computational statistics [19][27]. Shum, Ikeuchi and Reddy [20] modified Wiberg's formulation of principal component analysis with missing data, and generalized the problem as a weighted least square (WLS) problem. We briefly describe the PCAMD algorithm proposed in [20] before applying it to our modeling problem.

Suppose that an $F \times P$ measurement matrix W consists of P individuals from an F-variate normal distribution with mean $\bar{\mu}$ and covariance Σ. Let the rank of W be r. If the data is complete and the measurement matrix filled, the problem of principal component analysis is to determine $\tilde{U}$, $\tilde{S}$, and $\tilde{V}$ such that $\|W - \mathbf{e}\mu^T - \tilde{U}\tilde{S}\tilde{V}^T\|$ is minimized, where $\tilde{U}$, and $\tilde{V}$ are $F \times r$ and $P \times r$ matrices with orthogonal columns, $\tilde{S} = diag(\sigma_i)$ is an $r \times r$ diagonal matrix, μ is the maximum likelihood approximation of the mean vector and $\mathbf{e}^T = (1, \ldots, 1)$ is an F-tuple vector with all ones. The solution to this problem is essentially the SVD of the centered (or registered) data matrix $W - e\mu^T$.

If the data is incomplete, we have the following minimization problem:

$$min \quad \phi = \frac{1}{2}\sum_I (W_{f,p} - \mu_p - \mathbf{u}_{f.}^T \mathbf{v}_{p.})^2 \qquad \text{(EQ 6)}$$

$$I = \{ (f,p) : \ W_{f,p} \ is\ observed \}$$

where $\mathbf{u}_{f.}$ and $\mathbf{v}_{p.}$ are column vector notations defined by

$$\begin{bmatrix} \mathbf{u}_{1.}^T \\ \ldots \\ \mathbf{u}_{F.}^T \end{bmatrix} = \tilde{U}\tilde{S}^{\frac{1}{2}}, \qquad \begin{bmatrix} \mathbf{v}_{1.}^T \\ \ldots \\ \mathbf{v}_{P.}^T \end{bmatrix} = \tilde{V}\tilde{S}^{\frac{1}{2}}. \qquad \text{(EQ 7)}$$

To sufficiently determine the problem (EQ 6) more constraints are needed to normalize either the left matrix $\tilde{U}$ or the right matrix $\tilde{V}$.

By rearranging the measurement matrix W as an m-dimensional vector, where m is the number of observable elements in the measurement matrix, Ruhe [19] proposed a minimization method to analyze a one-component model when observations are missing. The one-component model decomposes an $F \times P$ measurement matrix into an $F \times 1$ left matrix and a $1 \times P$ right matrix. Ruhe observed that the nonlinear minimization problem can be simplified as a bilinear problem and proposed an NIPALS algorithm to solve it [19]. Wiberg [27] extended Ruhe's method to the more general case of arbitrary component model. The modified Wiberg's formulation has been proposed to achieve more efficient computation [20].

So far we have assumed that all weights are either one when data is observable or zero when unobservable. However, in many cases we may prefer to assign weights other than ones or zeros to individual measurement. Different sensor models can be applied to obtain a weighting matrix if necessary. The minimization problem (EQ 6) can be generalized as a WLS problem

$$min \quad \phi = \frac{1}{2}\sum_{f,p} (\gamma_{f,p}(W_{f,p} - \mu_p - \mathbf{u}_{f.}^T \mathbf{v}_{p.}))^2 , \qquad \text{(EQ 8)}$$

where $\gamma_{f,p}$ is a weighting factor for each measurement $W_{f,p}$.

Based on the modified Wiberg's formulation, principal component analysis with missing data has been formulated as a WLS problem by introducing two $FP \times FP$ diagonal weight matrices. An efficient iterative algorithm has been devised in [20].

5.1 Two-step WLS Problem

We now apply the PCAMD algorithm to solve the following modeling problem. Suppose that we have tracked P mesh nodes over F frames. In case of global resampling, the number of mesh nodes is known and fixed. We then have trajectories of point coordinates $\mathbf{v}_p^{(f)}$

for $f=1,...,F$, and $p=1,...,P$, where $\mathbf{v}_p^{(f)}$ is the *p*th point in the *f*th frame. Instead of forming a $4F \times P$ measurement matrix as in Section 3, we assemble *registered measurement matrices* $W^{(v)}$ and $W^{(d)}$ from the original measurement $3F \times P$ matrix W formed from trajectories of $\mathbf{v}_{fp}$ in the same way as in (EQ 1). After removing the translation component, we get

$$W^{(v)} = W - tM \qquad \text{(EQ 9)}$$

where M is a row vector of all ones, and t is a column vector consisting of translation vector of each view with respect to the world coordinate system, i.e.,

$$M = \begin{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} & \cdots & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \end{bmatrix}, \qquad t = \begin{bmatrix} \begin{bmatrix} t_{1x} & t_{1y} & t_{1z} \end{bmatrix}^T \\ \cdots \\ \begin{bmatrix} t_{px} & t_{py} & t_{pz} \end{bmatrix}^T \end{bmatrix}. \qquad \text{(EQ 10)}$$

After removing the rotation component, we have

$$W^{(d)} = W - RV \qquad \text{(EQ 11)}$$

where R is the rotation matrix of each view with respect to the world coordinate system, and V is the point matrix in the world coordinate system, i.e.,

$$V = \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_p \end{bmatrix}, \qquad R = \begin{bmatrix} R^{(1)} \\ \cdots \\ R^{(F)} \end{bmatrix}. \qquad \text{(EQ 12)}$$

Initially R and t can be obtained from SAI matching of deformable mesh nodes.

It can be easily shown that $W^{(v)}$ has at most rank 3 and $W^{(d)}$ has rank 1 when noise-free; therefore, $W^{(v)}$ and $W^{(d)}$ are highly rank-deficient. We decompose $W^{(v)}$ into

$$W^{(v)} = R\ V. \qquad \text{(EQ 13)}$$

Similarly, we can decompose $W^{(d)}$ into

$$W^{(d)} = t\,M. \tag{EQ 14}$$

When all elements in the two measurement matrices are known, we need to solve two least-squares problems. However, since only part of the object is visible in each view, we end up with two WLS problems instead. The first least squares problem, labeled as WLS-R, is

$$min \sum_{f=1,\ldots,F,p=1,\ldots,P} (\gamma_{f,p}(W^{(v)}_{f,p} - [RV]_{f,p}))^2 \tag{EQ 15}$$

and the second one, denoted as WLS-t, is

$$min \sum_{f=1,\ldots,F,p=1,\ldots,P} (\gamma_{f,p}(W^{(d)}_{f,p} - [tM]_{f,p}))^2 \tag{EQ 16}$$

where $\gamma_{f,p} = 0$ if the point p is invisible in frame f, and $\gamma_{f,p} = 1$ otherwise. All weights can be between zero and one, depending on the confidence of each measurement.

5.2 Iterative Algorithm

We have devised a two-step algorithm which solves the WLS-R and subsequently the WLS-t by applying the PCAMD algorithm to both problems. The WLS-R problem has been decomposed into F minimization problems and solved by employing the quaternion representation of rotation in the same way as in [20]. The two-step algorithm is as follows:

Algorithm two-step WLS's

```
Step 0   Initialization
    (0.1) read in measurement matrices W
    (0.2) read in weight matrices $\gamma$
Step 1   WLS-R
    (1.1)  register $W^{(v)}$ from W and t
    (1.2)  apply PCAMD to update R and V
    (1.3)  go to (1.2) if not converged, otherwise advance to Step 2
```

Step 2 WLS-t

(2.1) register $W^{(d)}$ from W, R and V

(2.2) apply PCAMD to update the translation t

(2.3) stop if converged, otherwise go to step 1.

We have not explicitly discussed the normalization problem in our WLS approach. The normalization problem occurs because the measurement matrix is rank-deficient, and as such, there are infinite solutions to the minimization problem (EQ 6) unless an additional constraint is imposed. We have used the first coordinate as the world coordinate system in our implementation, i.e., the first rotation matrix is identity and the first translation is zero.

6 Experiments

In this section, we present results of applying our algorithm to synthetic data and to real range image sequence of objects. We demonstrate the robustness of our approach using synthetic data, and present the reconstructed models from real range images.

6.1 Synthetic Data

Our synthetic data set consists of a set of 20 points whose connectivity is assumed to be known as shown in Figure 3 in Section 3. The object size is approximately the same as a unit sphere. We study the effectiveness of our approach when the input data is corrupted by noise and mismatching occurs. Correspondence is assumed to be known. The minimization of weighted squares distance between reconstructed and given measurement matrices leads to the recovery of vertex coordinates and transformations.

To study the error sensitivity on reconstruction using our algorithm, four nonsingular views of the object are taken where each measurement is corrupted by a Gaussian noise of zero-mean and variable standard deviation. Figure 7 shows that our algorithm converges in a few steps. Two separate steps in Figure 7 are counted as one iteration in our algorithm. The cases with standard deviations σ of 0.0, 0.1, and 0.2 are studied.

If a point appears only once in the whole sequence, then its reconstruction depends on the amount of noise. When this point appears in more views, its reconstruction using our integral method is averaged over all the views. Figure 8 gives the reconstructed errors of a point which appeared 12 times in 16 views. When only two views are matched, the reconstructed point is deviated from its original position by 0.58 and 0.69 when standard deviation σ is 0.1 and 0.2, respectively. When 12 views are matched, the error decreases to 0.19 and 0.27 respectively.

When the observed points are corrupted by noise, the sequential method results in erroneous recovered shape and transformation. The errors propagate as new views are introduced and vary with different matching orders. However, our integral approach gives appreciably smaller reconstruction error by distributing the errors in all views, regardless the order of matching. In Figure 9, we use a different starting view in a sequence of 12 images for different matching order.

6.2 Real Range Image Sequence

We have applied our integral approach to real range images. All range images used in our experiments were taken using a light-stripe range finder which has a resolution of 0.2mm. The objects were placed on a rotary table about 1 meter in front of the range finder. To obtain the ground truth of the transformation, rotation axis and rotation center of the rotary table are calibrated using a known geometry calibration cube.

Figure 10 shows nine views of a sequence of a free-form object, a peach. Figure 11 shows the result of our system, two wireframe and two shaded views of a recovered object model. The peach model does not depend on a priori motion estimation, nor on the ordering of matching. The same object model peach has also been reconstructed in [13] where only 3 views are combined after transformation is recovered. Figure 12 shows the comparison between the reconstructed models both with and without applying the PCAMD. The reconstructed models are shown at a particular viewing direction where the last view of the sequence appears. The model with the PCAMD gets better averaging and has a smaller margin of error. To demonstrate the improvement resulting from the application of the PCAMD, cross-section segments of reconstructed models and range data (merged from multiple views using known transformation) are shown in Figure 13.

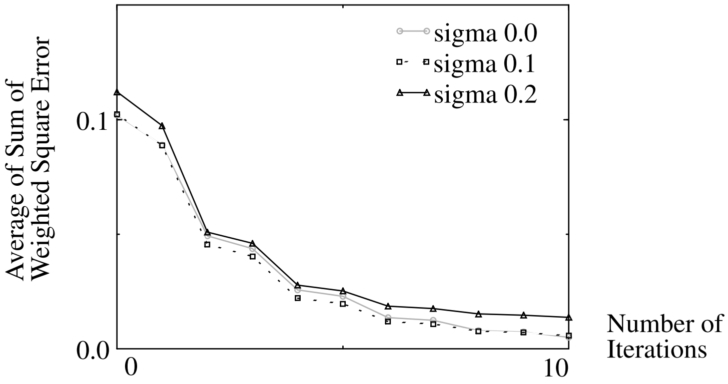


Figure 7 **Effect of noise on the convergence of the PCAMD**

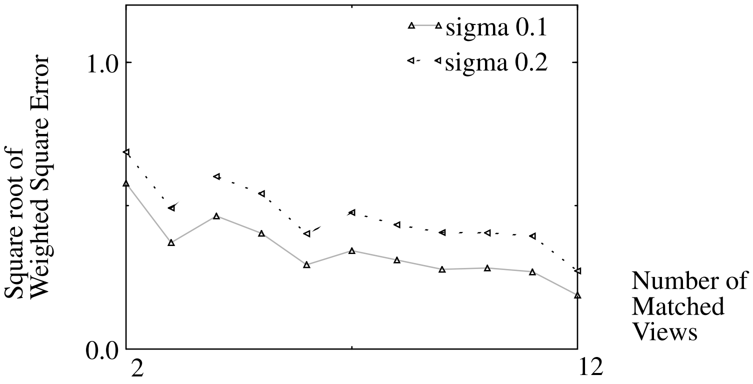


Figure 8 **Reconstructed error vs. number of matched views for a point**

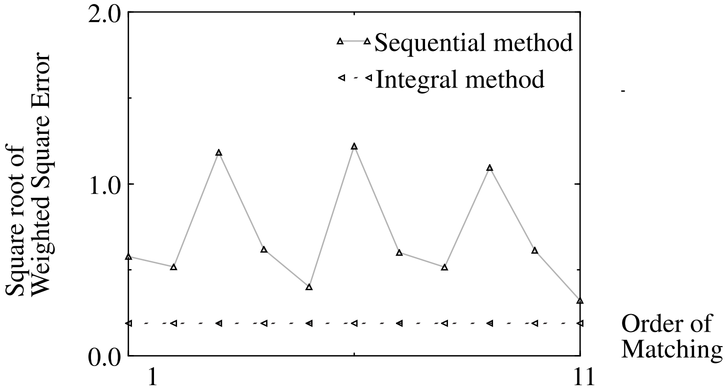


Figure 9 **Comparison between sequential method and integral method with different matching orders**

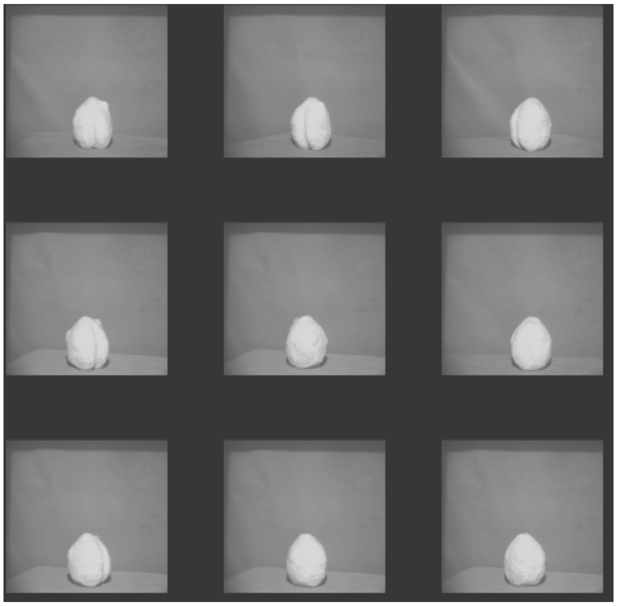


Figure 10 **A sequence of images of a free-form object (peach)**

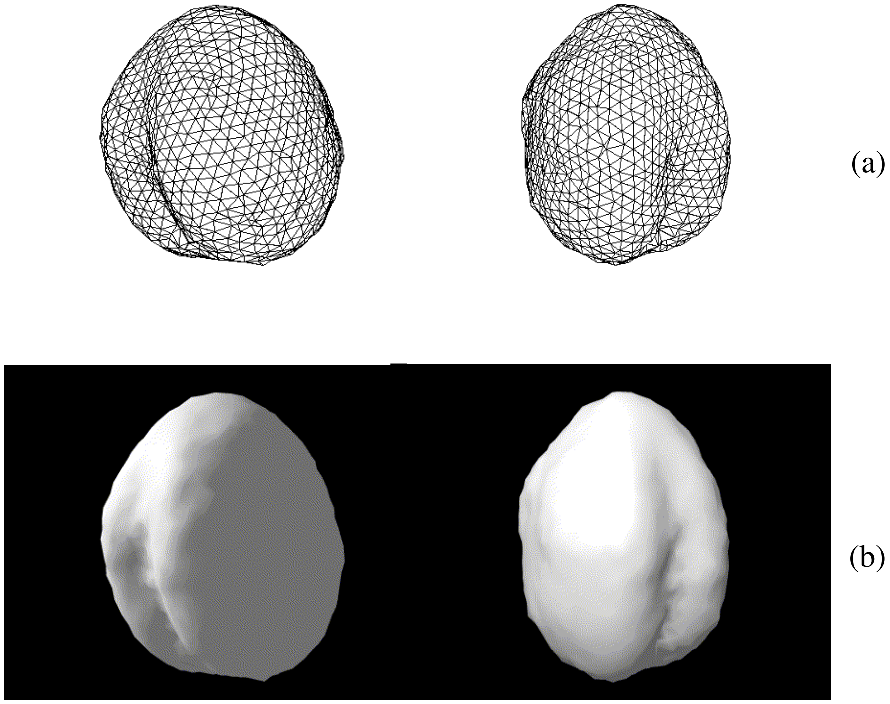


Figure 11 **Two views of a reconstructed peach model: (a) wireframe display; (b) shaded display**

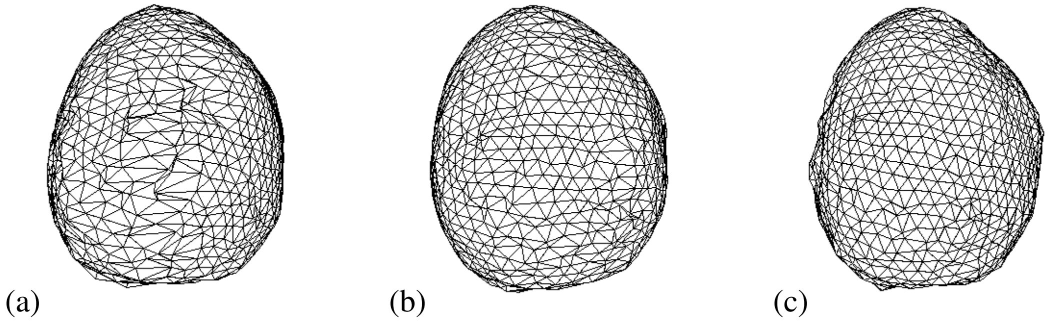


Figure 12 **Comparison between sequential reconstruction and WLS method: (a) sequential method using 8 views; (b) sequential method using 10 views; (c) WLS method using 10 views**

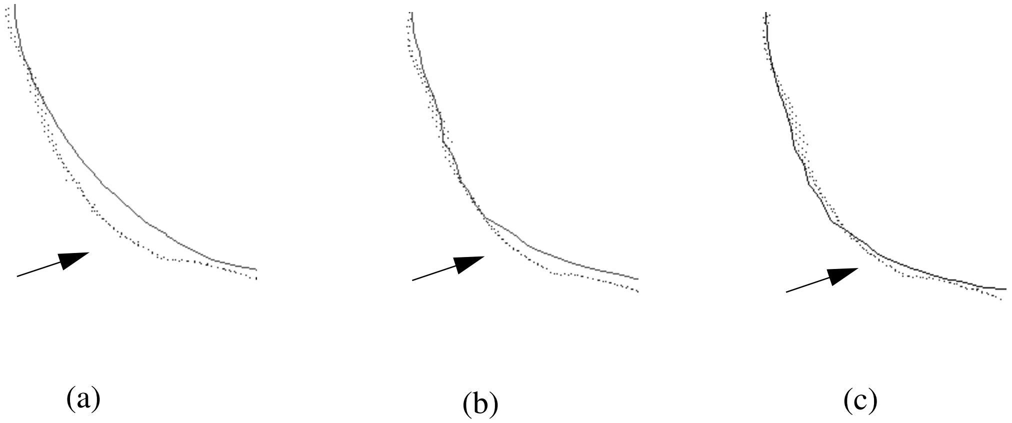


Figure 13 **Comparison using cross-section display of model (solid line) and range data (dots) (a) sequential method; (b) PCAMD after 5 steps; (c) PCAMD after 10 steps (arrows show the places where the improvement is significant)**

Another set of experiments is conducted to reconstruct a more complex object, a toy sharpei dog. Figure 14 shows the sequence of images. Figure 15 shows deformable surfaces of four different views. The occluded part of object is interpolated in the deformable surface mesh of each view. The reconstructed object models in different resolutions are shown in Figure 16. To compare the result of the PCAMD with that of the sequential method, we show the cross-section contours of reconstructed models in Figure 17, and error at each mesh node in Figure 18. We compare reconstructed models with the merged range data (using known transformation) in Figure 17. It can be observed that the integral method (Figure 17b) provides much better results than the conventional sequential method (Figure 17a), yet not as perfect as those with known transformation (Figure 17c) due to the least-squares nature of the PCAMD algorithm. In Figure 18, the mean error and maximum error for integral method are 2.7 and 9.2 as opposed to 3.7 and 16.8 for sequential method.

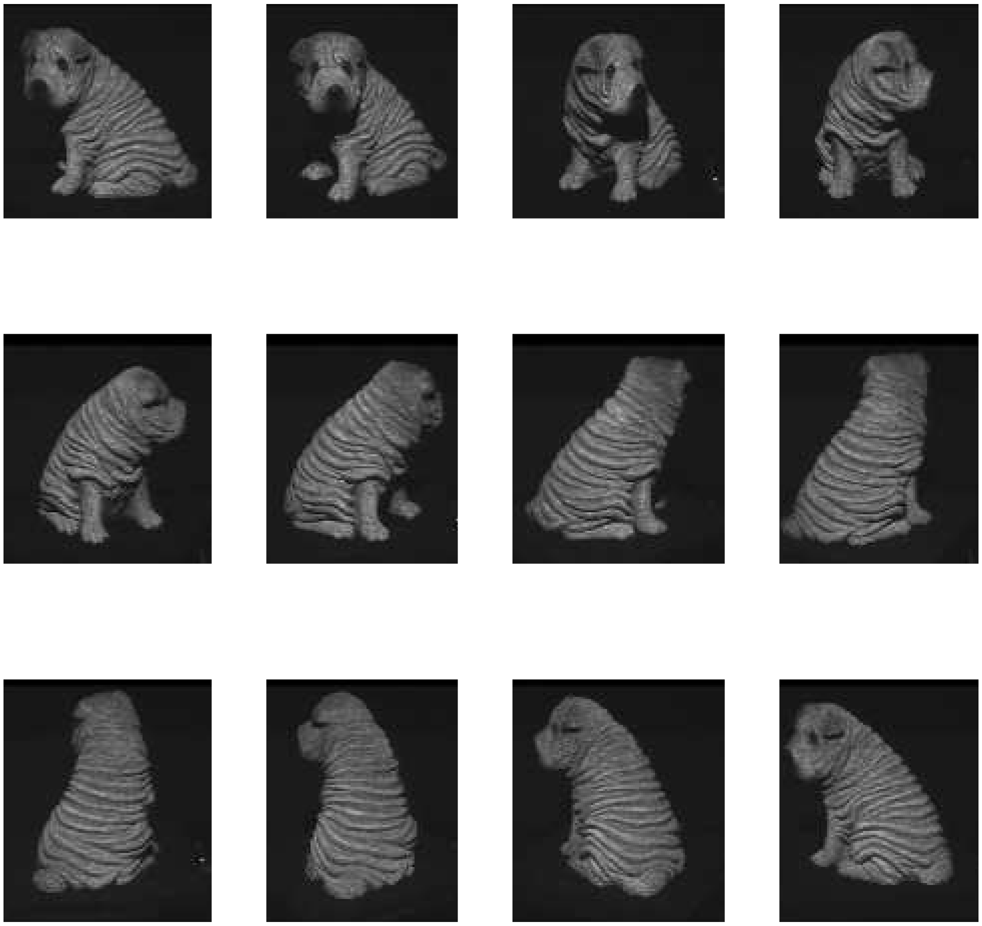


Figure 14 **A sequence of images of a free-form object (Sharpei)**

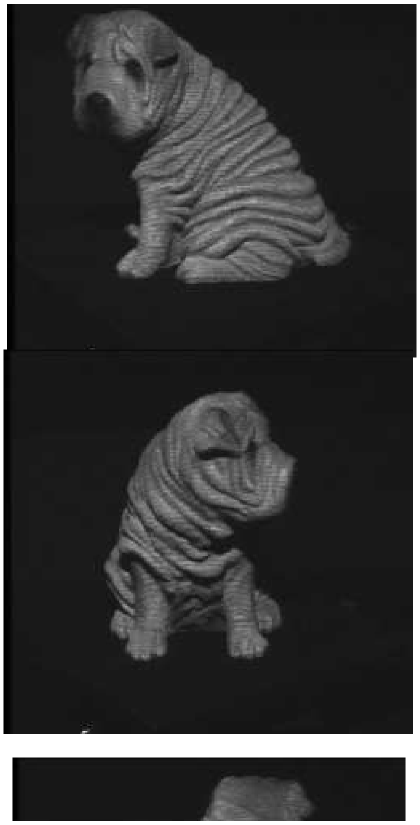


Figure 15 **Examples of deformable models from different views**