# EXTRACTION OF DEFORMABLE PART MODELS

**Alex Pentland**

Vision and Modeling Group, The Media Lab, Massachusetts Institute of Technology
Room E15-387, 20 Ames St., Cambridge MA 02139[1]

Many important objects consist of approximately rigid parts connected by hinges and other sorts of joints, so that the obvious way to describe these objects is in terms of the shapes of the component parts. Furthermore, if we are interested in the *behavior* of these parts, or if they are not completely rigid, then we must also account for their non-rigid shape and dynamics using a technique such as the finite element method.

Use of a 3-D dynamic model based on the finite element method was first suggested by Terzopoulos, Witkin, and Kass [1]. This approach to modeling is also known as the "thin plate" model. I will begin, therefore, by reviewing the finite element method.

## THE FINITE ELEMENT METHOD

The finite element method (FEM) is the standard technique for simulating the dynamic behavior of an object. In the FEM energy equations (or functionals) are derived in terms of nodal point unknowns and the resulting set of simultaneous equation iterated to solve for displacements as a function of impinging forces:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} \tag{1}$$

where $\mathbf{u}$ is a $3n$ x 1 vector of the $(x, y, z)$ displacements of the $n$ nodal points relative to the objects' center of mass, $\mathbf{M}$, $\mathbf{D}$ and $\mathbf{K}$ are $3n$ by $3n$ matrices describing the mass, damping, and material stiffness between each point within the body, and $\mathbf{f}$ is a $3n$ x 1 vector describing the $(x, y, z)$ components of the forces acting on the nodes. This equation can be interpreted as assigning a certain mass to each nodal point and a certain material stiffness between nodal points, with damping being accounted for by dashpots attached between the nodal points. Normally the damping matrix $\mathbf{D} = s_1\mathbf{M} + s_2\mathbf{K}$ for some scalars $s_1$, $s_2$.

One of the major drawbacks of the finite element method is its large computational expense, due to the large size of the $\mathbf{M}$, $\mathbf{D}$, and $\mathbf{K}$ matrices. For instance, an object whose geometry is defined by 100 points produces 300 x 300 matrices, corresponding to the 300 unknown coordinates of the 100 nodal points, $(x_i, y_i, z_i)$. Furthermore, for 3-D models the computation scales as $O(n^3)$ as the number of points $n$ defining the object geometry increases.

[1]This research was made possible in part by National Science Foundation, Grant No. IRI-8719920.

Another related drawback of the finite element method, at least as applied to vision, is the large numbers of unknowns that must be solved for. Because the number of unknowns is typically much larger than the number of measurements available from sensor data, external information such as axis direction and shape, and heuristics such as symmetry and smoothness must be used to achieve useful extraction of shape.

A final drawback of the finite element approach is the non-unique and unstable nature of the descriptions produced. Because the number of degrees of freedom in the model is at least as large as the number of sensor measurements available, the final position of a particular nodal point is strongly constrained in only in the direction perpendicular to the object's surface. Thus it is not in general possible to compare the shape of two finite element models directly; instead, one must sample the surface of one model, synthesize 3-D points, and then measure the distance between those points and the surface of the second model. Further, unless special care is taken in the sampling step, the comparison is not transitive or unique with respect to rotation and translation.

## MODAL DYNAMICS

A better method of describing non-rigid object behavior — at least for vision — is by use of *modal dynamics*, that is, by describing an object's behavior in terms of its natural *strain* or *vibration* modes. The modal method is equivalent to the finite element or thin-plate method in expressiveness and accuracy, but has the additional virtue that it separates non-rigid object behavior into independent modes of deformation, each of which may be separately analyzed and (often) solved in closed form. This in turn can lead to a much more efficient and stable computational scheme.

An object's strain modes may be found by simultaneously diagonalizing $\mathbf{M}$, $\mathbf{D}$, and $\mathbf{K}$. Because these matrices are normally positive definite symmetric, and $\mathbf{D}$ is a linear function of $\mathbf{M}$ and $\mathbf{K}$ , Equation 1 can be transformed into $3n$ independent differential equations by use of the *whitening transform*, which is the solution to the following eigenvalue problem:

$$\lambda\phi = \mathbf{M}^{-1}\mathbf{K}\phi \tag{2}$$

where $\lambda$ and $\phi$ are the eigenvalues and eigenvectors of $\mathbf{M}^{-1}\mathbf{K}$.

Using the transformation $\mathbf{u} = \phi\bar{\mathbf{u}}$ we can then re-write Equation 1 as follows:

$$\phi^T\mathbf{M}\phi\ddot{\bar{\mathbf{u}}} + \phi^T\mathbf{D}\phi\dot{\bar{\mathbf{u}}} + \phi^T\mathbf{K}\phi\bar{\mathbf{u}} = \phi^T\mathbf{f} \quad . \tag{3}$$

In this equation $\phi^T\mathbf{M}\phi$, $\phi^T\mathbf{D}\phi$, and $\phi^T\mathbf{K}\phi$ are diagonal matrices, so that if we let $\bar{\mathbf{M}} = \phi^T\mathbf{M}\phi$, $\bar{\mathbf{D}} = \phi^T\mathbf{D}\phi$, $\bar{\mathbf{K}} = \phi^T\mathbf{K}\phi$, and $\bar{\mathbf{f}} = \phi^T\mathbf{f}$ then we can write Equation 3 as $3n$ independent equations:

$$\bar{M}_i\ddot{\bar{u}}_i + \bar{D}_i\dot{\bar{u}}_i + \bar{K}_i\bar{u}_i = \bar{f}_i \quad , \tag{4}$$

where $\bar{M}_i$ is the $i^{th}$ diagonal element of $\bar{\mathbf{M}}$, and so forth.

What Equation 4 describes is the time course of one of the object's *strain* or *vibration modes*. The constant $\bar{M}_i$ is the generalized mass of mode $i$, that is, the inertia of the $i^{th}$ vibration mode. Similarly, $\bar{D}_i$, and $\bar{K}_i$ describe the damping and spring stiffness associated with mode $i$, and $\bar{f}_i$ is the amount of force coupled with this vibration mode. The $i^{th}$ row of $\phi$ describes the *deformation* the object experiences as a consequence of the force $\bar{f}_i$,

and the eigenvalue $\lambda_i$ is proportional to the natural resonance frequency of that vibration mode.

To obtain an accurate simulation of the dynamics of an object one simply uses linear superposition of these modes to determine how the object responds to a given force. Because Equation 4 can be solved in closed form, we have the result that for objects composed of linearly-deforming materials *the non-rigid behavior of the object in response to a simple force can be solved in closed form for any time t.* In complex environments, however, numerical solution is preferred.

**Number Of Modes Required.** The modal representation decouples the degrees of freedom within the non-rigid dynamical system of Equation 1, but it does not by itself reduce the total number of degrees of freedom. However modes associated with high resonance frequencies (large eigenvalues) normally have little effect on object shape. This is because (on average) the displacement amplitude for each mode is *inversely* proportional to the *square* of the mode's resonance frequency, and because damping is proportional to a mode's frequency. The combination of these effects is that high-frequency modes generally have very little amplitude. Experimentally, I have found that most commonplace multi-body interactions can be adequately modeled by use of only[2] rigid-body, linear, and quadratic strain modes [2].

Although discarding high-frequency modes has little effect on accuracy, it can have a profound effect on computational efficiency. Not only does it result in having to solve fewer equations in fewer unknowns, but (because of Nyquist considerations) we can also employ a much larger time step. For a problem involving 100 nodal points, for instance, the modal method (using 30 modes) will be roughly *two orders of magnitude* more efficient than the standard finite element approach, and yet will typically have roughly the same accuracy.

Another equally important benefit of using only low-order modes to describe object deformations is that they change very slowly as a function of object shape. Consequently the same modes $\phi$ may be used for a *range* of different — but similar — undeformed shapes without incurring substantial error. This allows the modes to be precomputed, avoiding the expense of solving for $\phi$ at run time.

**Advantages of the Modal Representation.** The modal representation has several advantages over a representation based on the standard finite element method. First, of course, it is much more efficient — typically one or two orders of magnitude more efficient — and scales as $O(n)$ rather than the $O(n^3)$ scaling of the standard finite element method.

Perhaps more importantly, however, it provides a natural hierarchy of scale, so that we can smoothly vary the level of detail by adding in or discarding high-frequency modes. That is, the modal representation provides a natural multi-scale representation for three-dimensional object shape in much the same manner that the Fourier transform provides a multi-scale representation for images. By matching the level of detail (the number of modes) to the number of sensor measurements available the shape recovery problem can be kept overconstrained without resorting to heuristics or external knowledge.

Further, by keeping the number of modes less than the number of sensor measure-

[2]Note, however, higher-order modes are required to accurately model the objects whose dimensions differ by more than an order of magnitude.

ments, we can calculate a shape description that is *unique* with respect to sampling and viewpoint (assuming, of course, that a sufficient distribution of surface measurements is available). In the finite element/thin-plate approach a recovered description is not unique because changes in sampling or viewpoint cause the nodal points to move about on the object's surface; this is a direct consequence of having more degrees of freedom than surface measurements. Polynomial and spline representations also suffer from these same problems. In the modal representation, however, the high-frequency modes that allow nodal points to move relative to one another have been discarded, and as a consequence the representation is insensitive to sampling and viewpoint.

Because of this uniqueness property, the modal representation is well-suited for object recognition and other spatial database tasks. To compare two objects described using a modal representation one simply compares the vector of mode values $\bar{\mathbf{u}}$; if the dot product of the $\bar{\mathbf{u}}$ for each object is small, then the objects are similar (excepting some degenerate conditions).

## USE OF VOLUMETRIC MODELING PRIMITIVES

A disadvantage of any vertex or knot based representation is the expense of calculating the distance between 3-D points (e.g., sensor data) and the modeled surface; this calculation has a computational complexity of $O(Nn)$ where $N$ is the number of data points and $n$ is the number of points defining the object's geometry. Consequently this distance calculation is often a large fraction of the total computational cost.

One method of reducing the cost of computing the data error term is to use a representation that has an inside-outside distance function $f(x, y, z) = d$. For such implicit function representations the distance $d$ between a point $(x, y, z)$ and the surface can be found by simply substituting the point into the distance function $f(x, y, z)$. The computational complexity of this computation is $O(N)$, a significant improvement.

The modal representation of shape can be combined with analytic shape primitives by first describing each mode by an appropriate polynomial function, and then using global deformation techniques to warp the shape primitive into the appropriate overall form. The polynomial deformation mappings that correspond to each of the modes are determined by a linear regression of a polynomial with $m$ terms in appropriate powers of $x$, $y$, and $z$, against the $n$ triples of $x$, $y$ and $z$ that compose $\phi_i$, a $3n$ x 1 vector containing the elements of the $i^{th}$ row of $\phi$:

$$\alpha = (\beta^T \beta)^{-1} \beta^T \phi_i \quad , \tag{5}$$

where $\alpha$ is an $m$ x 1 matrix of the coefficients of the desired deformation polynomial, $\beta$ is an $3n$ x $m$ matrix whose first column contains the elements of $\mathbf{u} = (x_1, y_1, z_1, x_2, y_2, z_2, ...)$, and whose remaining columns consist of the modified versions of $\mathbf{u}$ where the $x$, $y$, and/or $z$ components have been raised to the various powers. See reference [2] for more details.

By linearly superimposing the various deformation mappings one can obtain an accurate accounting of the object's non-rigid deformation. In the Thingworld modeling system [2] the set of polynomial deformations is combined into a 3 by 3 matrix of polynomials that is refered to as the *modal deformation matrix*. Because low-order modes change slowly as a function of object shape, the matrix can be used for a *range* of similar shapes, and thus may be precomputed.

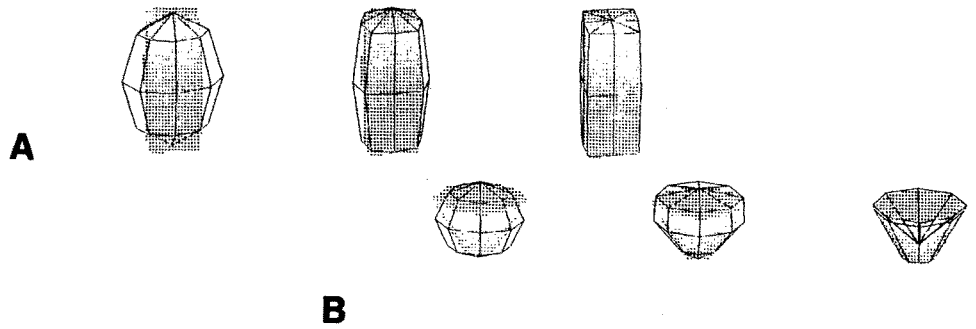


Figure 1: (a) Time-lapse sequence of a sphere being deformed to fit a vertical box, and (b) a sphere being deformed to fit a hollow vase.

## FITTING 3-D MODELS

Given a segmentation into parts (such as produced by the algorithm described in reference [3]), the next step is to fit a deformable model to each part using available data. For simple objects this can be accomplished directly from the object's axes. This is because the planes of symmetry are the singularities (zero points) of the various low-order modal deformations, so that along axes and planes of symmetry the various modes are effectively decoupled. Consequently, the unknown polynomial coefficients in the modal deformation matrix can be solved for by measuring axis length, direction, and bending. From these coefficients the modal amplitudes $\bar{\mathbf{u}}$ can be solved for directly. Such model recovery from symmetry analysis may be useful for understanding human vision, or for constructing efficient "first-pass" machine vision systems.

More accuracy can be obtained by fitting the model to range data. This can be accomplished by assigning a gravity-like potential field to each data point, so that the deformable part's surface is attracted to the data points by the resulting forces. Figure 1 shows two examples of this fitting process. In Figure 1(a) the range data is of a vertical box, and in Figure 1(b) the range data is of a hollow vase. In both cases the original spherical shape is progressively deformed by the attractive forces exerted by the range measurements, until the surface exactly fits the data. Typical fitting times on a Sun 4 using this formulation is a few seconds per part, depending upon the number of data points. These examples also illustrate that a wide variety of shapes can be generated by applying first and second order deformations to a basic spherical shape.

## REFERENCES

[1] Terzopolis, D., Witkin, A., and Kass, M., (1987) Symmetry-Seeking Models for 3-D Object Reconstruction, *Proc. First International Conf. on Computer Vision*, pp. 269-276, London, England.

[2] Pentland, A., and Williams, J. (1989), Good Vibrations: Modal Dynamics for Graphics and Animation, *Computer Graphics* ,. Vol. 23, No. 3, pp. 215-222.

[3] Pentland, A., (1989) Part Segmentation for Object Recognition, *Neural Computation*, Vol. 1, No. 1, pp. 82-91.