2D PHYSICS-BASED DEFORMABLE OBJECTS USING FAST FREE VIBRATION MODAL ANALYSIS

Stelios Krinidis and Ioannis Pitas

Aristotle University of Thessaloniki
Department of Informatics
Box 451
54124 Thessaloniki, Greece
email: {skrinidi,pitas}@zeus.csd.auth.gr

ABSTRACT

This paper presents an accurate, very fast approach for the deformations of 2D physically based shape models representing open and closed curves. The introduced models overcome the main shortcoming of other deformable models, i.e. computation time. The approach relies on the determination of explicit deformation governing equations, that involve neither eigenvalue decomposition nor any other computationally intensive numerical operation. The approach was evaluated and compared with another fast and accurate physics-based deformable shape model, both in terms of deformation accuracy and computation time. The conclusion is that the introduced model is completely accurate and is deformed very fast on current personal computers.

1. INTRODUCTION

A key problem in machine vision is how to describe features, contours, surfaces, and volumes, so that they can be segmented, recognized, matched, or any other similar underlying process. The primary difficulties can be summarized as: a) object descriptions are sensitive to noise, b) objects can be nonrigid, and c) the shape of the 2D object projection varies with the viewing geometry. These problems have motivated the use of deformable models [1]-[5], to interpolate, smooth, and warp raw data, since these models provide reliable shape reconstruction tools that are both robust and generic.

The class of deformable shape models originates with the method of active contours ("snakes") introduced by Kass *et al.* [1], that are used to locate smooth curves in 2-D images. Since then, deformable models have been used for a number of applications in 2-D and 3-D by Terzopoulos, Witkin and Kass [6].

All approaches are quite slow, requiring dozens or even up to hundreds of iterations. Hence, all the deformation-based modelling approaches are computationally intensive since each iteration turns out to be very time-consuming. Thus, although finite elements are a powerful tool in computer vision applications, they may be computationally intensive for certain applications, especially the real time ones. We address this problem by introducing a 2D physics-based deformable approach representing open and closed curves, based on modal analysis, using explicit functions. This leads to a very fast model deformation approach, involving only the calculation of an explicit function for the description of the deformations of the object under examination, as opposed to the eigenvalue decomposition that is necessary in classical shape modal analysis.

Our approach was motivated by the technique presented in [3, 5], which consists of analyzing non-rigid motion, with application to medical images. Nastar *et al.* [3] approximated the dynamic deformations of the object contour, using a physically based deformable curve/surface. In order to reduce the number of parameters describing the deformation, modal analysis was exploited that provides a spatial smoothing of the curve/surface. On the same basis, we employ modal analysis and, after simplifying the deformation governing equations, we prove that the deformations of open and closed 2D curves have explicit governing equations, involving neither eigenvalue decomposition nor any other computationally intensive operation.

The remainder of the paper is organized as follows. The 2D physics-based deformable shape models [3] are presented in Section 2. In Section 3 the proof and the properties of the new deformable physics-based governing equations are introduced. A comparison between the initial deformation models [3] and the novel ones, are presented in Section 4, and final conclusions are drawn in Section 5.

2. 2D PHYSICS-BASED DEFORMABLE SHAPE MODELLING BASED ON MODAL ANALYSIS

In this Section, physically based deformable shape models [3, 5] exploiting modal analysis are introduced. We consider both the surface and volume properties of the objects at hand. We restrict ourselves to elastic deformations, i.e. we assume that the object recovers its original configuration as soon as all applied forces causing the deformation are removed.

Modelling an elastic 2D boundary can be achieved by an open or closed chain topology of N virtual masses on the contour. Each model node has a mass m and is connected to its two neighbors with identical springs of stiffness k. The ratio $a = \frac{k}{m}$ constitutes the so-called *characteristic value* of the model, which is a constant value that describes its physical characteristics and determines its physical behavior. The node coordinates of the model under examination are stacked in vector:

$$\mathbf{X}_t = \left[x_1^t, y_1^t, \ldots, x_N^t, y_N^t\right]^T \quad (1)$$

This work has been supported by the European Union research project IST-2000-28702 Worlds Studio "Advanced Tools and Methods for Virtual Environment Design and Production".

where N is the number of vertices (masses) of the chain and t denotes the t-th deformation time instance. The model under study, is a physics-based system governed by the fundamental equation of dynamics:

$$\boldsymbol{f}_e(x_i^t) + \boldsymbol{f}_d(x_i^t) + \boldsymbol{f}_{ext}(x_i^t) = m_i \ddot{x}_i^t \tag{2}$$

where m_i is the mass of the point under study and $\ddot{x}_i^t$ its acceleration under total load of forces. $\boldsymbol{f}_d(\cdot)$ is a damping force, $\boldsymbol{f}_{ext}(\cdot)$ the external load on node under study, and $\boldsymbol{f}_e(\cdot)$ is the elastic force due to node neighbors. The above governing equation is expressed for all model nodes, and setting the natural length l_0 of the springs equal to zero, we could assume that the model can be considered within the framework of linear elasticity. As a consequence, our solution lies in a set of linear differential equations with node displacements decoupled in each coordinate, regardless of the magnitude of the displacements.

The governing equation can now be written in a matrix form [7]:

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{C}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}_t \tag{3}$$

where $\mathbf{U}$ is the nodal displacements vector. $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ [8] are the mass, damping, and stiffness matrices of the model, respectively, and $\mathbf{F_t}$ is the external force vector, usually resulting from the attraction of the model by the object contour (sometimes based on the Euclidean distance between the object contour and the node coordinates). Equation (3) is a finite element formulation of the deformation process.

Instead of solving directly the equilibrium equation (3), one can transform it by a change of basis:

$$\mathbf{U} = \mathbf{\Psi}\tilde{\mathbf{U}} \tag{4}$$

where $\mathbf{\Psi}$ is the square nonsingular transformation matrix of order N to be determined, and $\tilde{\mathbf{U}}$ is referred to as the *generalized displacement* vector. One effective way of choosing $\mathbf{\Psi}$ is setting it equal to $\mathbf{\Phi}$, a matrix whose entries are the eigenvectors of the generalized eigenproblem:

$$\mathbf{K}\boldsymbol{\phi_i} = \omega_i^2 \mathbf{M}\boldsymbol{\phi_i} \tag{5}$$

$$\mathbf{U} = \mathbf{\Phi}\tilde{\mathbf{U}} = \sum_{i=1}^{N} \tilde{u}_i \boldsymbol{\phi_i} \tag{6}$$

Equation (6) is referred to as the modal superposition equation. The i-th eigenvector, i.e. the i-th column of $\mathbf{\Phi}$, denoted by ϕ_i, is also called the i-th *vibration mode*, $\tilde{u}_i$ (the i-th scalar component of $\tilde{\mathbf{U}}$) is its amplitude, and ω_i is the corresponding eigenvalue (also called *frequency*). If the matrix $\tilde{\boldsymbol{C}} = \mathbf{\Phi}^T \boldsymbol{C} \mathbf{\Phi}$ is diagonal (standard Rayleigh hypothesis [3]), then in the modal space the governing matrix-form equations decoupled into N scalar equations:

$$\ddot{\tilde{u}}_i^t + \tilde{c}_i \dot{\tilde{u}}_i^t + \omega_i^2 \tilde{u}_i^t = \tilde{f}_i^t, \qquad i = 1, \ldots, N. \tag{7}$$

Solving these equations at time t leads to $\left(\tilde{u}_i^t\right)_{i=1,\ldots,N}$, and the displacement $\boldsymbol{U}$ of the model nodes is obtained by the modal superposition equation (6).

In practice, we wish to approximate nodal displacements $\mathbf{U}$ by $\hat{\mathbf{U}}$, which is the truncated sum of the N' low-frequency vibration modes, where $N' \ll N$:

$$\mathbf{U} \approx \hat{\mathbf{U}} = \sum_{i=1}^{N'} \tilde{u}_i \boldsymbol{\phi_i} \tag{8}$$

Eigenvectors $(\phi_i)_{i=1,\ldots,N'}$ form the *reduced modal basis* of the system. This is the major advantage of modal analysis: it is solved in a subspace corresponding to the N' truncated low-frequency vibration modes of the deformable structure [3, 4, 5]. The number of vibration modes retained in the object description, is chosen so as to obtain a compact but adequately accurate representation. A typical *a priori* value for N', covering many types of standard deformations is equal to one quarter of the number of the vibration modes.

An important advantage of the formulations described so far, in the full as well as the truncated modal space, is that the vibration modes ϕ_i and the frequencies ω_i of an open or closed chain topology have an explicit expression [3] and they do not have to be computed using eigen-decomposition techniques. The frequencies and the vibration modes of the closed chain are given by:

$$\omega_i^2 = 4a \sin^2\left(\frac{\pi i}{N}\right) \tag{9}$$

$$\boldsymbol{\phi}_i = \left[\ldots, \cos\frac{2\pi i j}{N}, \ldots\right]^T \tag{10}$$

where $i \in \{1, 2, \ldots, N\}$ and $j \in \mathcal{B}(N)$. $\mathcal{B}(N)$ is the first Brillouin zone [3] and is equal to $\left\{-\frac{N}{2}+1, \ldots, \frac{N}{2}\right\}$ for N even, and $\left\{-\frac{N-1}{2}, \ldots, \frac{N-1}{2}\right\}$ for N odd. Furthermore, the case of an open chain topology is very similar, where the frequencies and the corresponding vibration modes are obtained by:

$$\omega_i^2 = 4a \sin^2\left(\frac{\pi i}{2N}\right) \tag{11}$$

$$\phi_i = \left[\ldots, \cos\frac{2\pi(2i-1)j}{2N}, \ldots\right]^T \tag{12}$$

where $i \in \{1, 2, \ldots, N\}$ and $j \in [0, \ldots, N-1]$.

Thus, the deformations of the described *Deformable Model based on Modal Analysis* (abbreviated here as DMMA), for a closed chain as well as an open one, can be given by:

$$u_i^t = \sum_{j}^{N} \frac{\sum_{n=1}^{N} [f_n \phi_n(j)]}{(1+\omega_j^2)\sum_{n=1}^{N} \phi_n^2(j)} \phi_i(j) \tag{13}$$

whereas the deformations of the *Truncated Deformable Model based on Modal Analysis* (abbreviated here as TDMMA), for both open and closed chains, can be described by:

$$\hat{u}_i^t = \sum_{j}^{N'} \frac{\sum_{n=1}^{N} [f_n \phi_n(j)]}{(1+\omega_j^2)\sum_{n=1}^{N} \phi_n^2(j)} \phi_i(j) \tag{14}$$

3. A CLOSED-FORM REPRESENTATION OF THE 2D PHYSICS-BASED DEFORMABLE MODELS

It is obvious that the deformations described in the previous Section are still computationally intensive, since they require the calculation of a large number of summations in (13) and (14). In this Section, our goal is to simplify the deformation process (Section 2) and to introduce a new way of calculating the deformations, achieving very fast deformation computation, without loss of accuracy.

As already mentioned, the deformations $\boldsymbol{U}$ of the DMMA applied to a closed chain are given by equation (13). (13) is applied

to the x and y coordinates independently. f_i denotes the x and y components of the force acting on node i:

$$\boldsymbol{F} = (f_{1_x}, f_{1_y}, f_{2_x}, f_{2_y}, \ldots, f_{N_x}, f_{N_y})^T \tag{15}$$

and finally u_i denotes the x and y components of the displacement of the node i:

$$\boldsymbol{U} = (u_{1_x}, u_{1_y}, u_{2_x}, u_{2_y}, \ldots, u_{N_x}, u_{N_y})^T \tag{16}$$

As it can be proved, the deformations of the DMMA are equal, for closed curves, to:

$$u_i = \frac{1}{\sqrt{1+4a}} \sum_{j=1}^{N} f_j a^d \frac{\left(\frac{\sqrt{1+4a}+1}{2}\right)^{N-2d} + \left(\frac{\sqrt{1+4a}-1}{2}\right)^{N-2d}}{\left(\frac{\sqrt{1+4a}+1}{2}\right)^{N} - \left(\frac{\sqrt{1+4a}-1}{2}\right)^{N}} \tag{17}$$

We call formulation (17) *Explicit Formulation of Modal Analysis* (EFMA) model. The deformation governing equation for open chain physics-based deformable models is proved to be:

$$u_i = \frac{1}{\sqrt{1+4a}} \sum_{j=1}^{N} f_j \frac{S_{i,j} + P_{i,j}}{\mu^{2N} - \lambda^{2N}} \tag{18}$$

$$S_{i,j} = a^{|i-j|}\left[\mu^{2N-2|i-j|} + \lambda^{2N-2|i-j|}\right]$$

$$P_{i,j} = a^{i+j-1}\left[\mu^{2N-2(i+j-1)} + \lambda^{2N-2(i+j-1)}\right]$$

where μ and λ are two constant values equal to $\frac{\sqrt{1+4a}+1}{2}$ and $\frac{\sqrt{1+4a}-1}{2}$ respectively, a is the model characteristic value, N is the total number of model nodes, and d is the distance (in number of nodes) between the node under examination and the node, where force f_i is applied. Hence, in practice, d is equal to $min(|i-j|, |N-|i-j||)$ for the closed case models.

Due to the fact that the model exhibits linear elasticity and that neighboring nodes cause the forces $\boldsymbol{f}_e(\cdot)$, we are able to approximate nodal displacements $\mathbf{U}$ (17) without any substantial loss of accuracy, by the truncated contribution of the $2N''$ adjacent nodes, where $N'' \ll N$:

$$u_i \approx \frac{1}{\sqrt{1+4a}} \sum_{j=i-N''}^{i+N''} f_j a^d \frac{\mu^{N-2d} + \lambda^{N-2d}}{\mu^N - \lambda^N} \tag{21}$$

The nodes i-N'' to i+N'', for each node i, form the *reduced nodal basis* of the system. This is the major advantage of the new form of the deformation governing equation: it is solved in a *nodal subspace* corresponding to the $2N''$ adjacent model nodes of the deformable structure [4, 3, 5]. From now on, the deformation calculation in a reduced nodal space (21) will be called *Truncated Explicit Formulation of Modal Analysis* (TEFMA). The number of nodes retained in (21) is chosen so as to produce a compact but adequately accurate shape representation. After performing a sequence of experiments, it has been observed that a typical *a priori* value for N'', covering many types of standard deformations, is equal to $\frac{\sqrt{1+4a}+1}{2} lnN$. Thus, only a very small number of adjacent nodes is taken into account in the calculations. This formulation is proved to be very fast, capable of use of real time applications.

4. PERFORMANCE ANALYSIS OF THE PROPOSED METHOD

In this Section, a comparison between the DMMA and TDMMA described in Section 2 and the EFMA and TEFMA introduced in Section 3 is presented. The comparison is performed in terms of the displacement estimation error and the required computation time. The relative displacement error is defined as:

$$\left\{ \begin{array}{c} E_{TDMMA} \\ E_{TEFMA} \end{array} \right\} = \frac{1}{N} \frac{\sum_i \|U_i' - U_i\|}{\max_i \|U_i' - U_i\|} \tag{22}$$

where U_i and U_i' are the displacements calculated using DMMA (13) and the deformable method under comparison (TDMMA and TEFMA) respectively, i.e., DMMA is the reference deformable method. The displacement error E_{EFMA} (22) of the EFMA and the DMMA is zero, as proven in the Appendix. All experiments are performed on a Pentium III (700 MHz) workstation under Windows 2000 Professional without any particular code optimization.

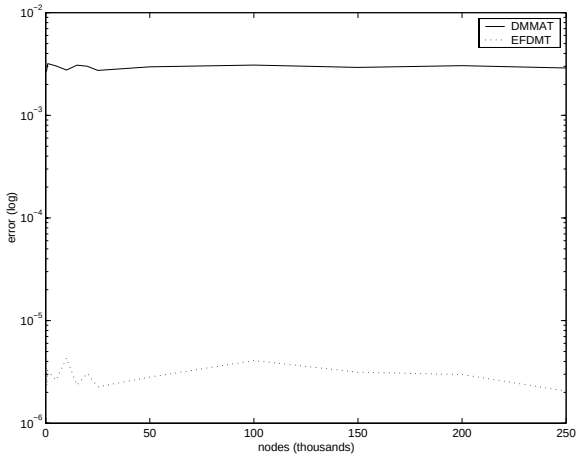

Fig. 1. ***Error comparison.*** *Errors regarding the modal and nodal reduction case respectively.*

The displacement errors E_{TDMMA} and E_{TEFMA} (22) comparison versus node number N are illustrated in Figure 1. All errors in the Figure, are in a percentage form, due to the fact that the absolute error is not that representative, since it depends on the input forces applied to the model. Note that error axis Y is a logarithmic one. In the case of TDMMA, where only 25% of the eigenvalues are taken into account, the relative average error is only 8.04721%, which is not so important. Thus, as Nastar *et al.* [3] claim, one can use the TDMMA without any particular loss of accuracy. In the case of TEFMA only $\frac{\sqrt{1+4a}+1}{2} lnN$ adjacent (neighboring) nodes are considered. The relative average error is only 0.01553%, which is negligible with respect to the global deformation of the model, and, consequently, deformations can be computed using TEFMA without any particular loss of accuracy. Besides, it is clearly depicted in Figure 1 that deformations extracted from the TEFMA have, in average, a relative error 500 times less than the corresponding deformation computed by TDMMA. It must be noted that the errors do not change with the number of nodes N.

Additionally, we have performed a computational complexity analysis of the methods under comparison. We assume that

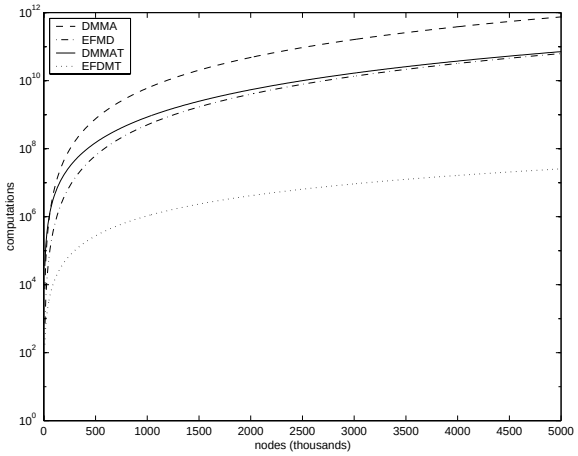

Fig. 2. ***Manual Computations.*** *Computations of the DMMA and DMMAT (25% of eigenvalues) have been manually estimated. The same task is performed for the EFDM and the EFDMT as well.*

the four elementary mathematical operations – additions, abstractions, multiplications and divisions – have the same computation time. Furthermore, sine, cosine and square root calculations are assumed to be computed by a Taylor series expansion. Then it can be proven that DMMA (13) requires approximately $6N^3+10N^2+\frac{189}{2}N$ elementary computations, where N is the number of model nodes. EFMA (17) requires only $\frac{1}{2}N^3+N^2+85$ computations, while TDMMA and TEFMA require $\frac{3}{2}N^3+\frac{11}{4}N^2+\frac{477}{8}N$ and $N^2-NlnN(1+lnN)+85$ computations respectively. These curves are plotted in Figure 2. It is worth noting that TEFMA has computational complexity of order $O(N^2)$, whereas even TDMMA has computational complexity $O(N^3)$.

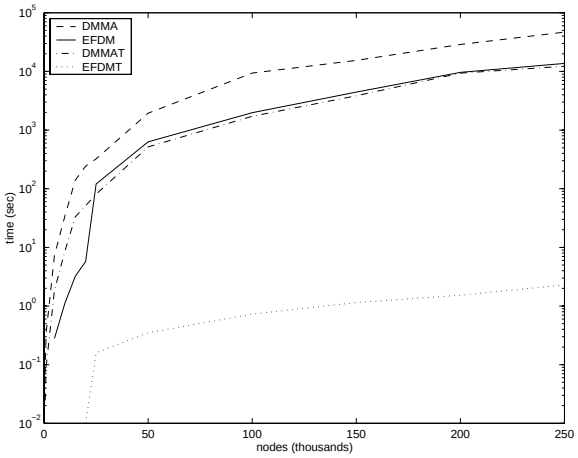

Fig. 3. ***Time comparison.*** *The EFDM compared with the DMMA with respect to computational time. The same task is performed for the DMMAT and the EFDMT. Times are measured in seconds.*

Furthermore, we have performed computation time benchmarking experimentally. Figure 3 shows the computation time for all methods under comparison. It must be mentioned that, in Figure 3, the computation times are plotted in a logarithmic axis for better visualization. EFMA and TDMMA require approximately the same computation time. Both are 3 times faster than DMMA. The computation times of these two cases are quite satisfactory, but they are still far slower than TEFMA. TEFMA computes the deformations very fast. The computation time of TEFMA is more than 1 second only when the model has more than 150.000 nodes. Therefore, TEFMA can be deformed 100 times per second for contours having 20.000 nodes and $6,25$ times per second for contours of 25.000 nodes. Hence, it can be safely used in many real time applications. TEFMA can achieve a speedup of 4-5 orders of magnitude versus DMMA for almost any contour length. It must be noted that the calculated execution speeds shown in Figure 3 are well in line with the comparing computational results shown in Figure 2.

5. CONCLUSION

In this paper, we have presented a closed-form solution for 2D physics-based shape model deformation along with its properties. The deformation equations [3] were simplified and analyzed. As a result, a closed-form solution can be reached for the objects deformation. The presented 2D physics-based deformable model drastically reduces the computation time needed to perform the deformation. This solution is very useful in analyzing the deformation behavior of the contour at hand. The extremely low computational time and the low deformation error with respect to all the other available techniques in the literature, makes the introduced physics-based deformable model a very promising tool for various image analysis and computer vision applications.

6. REFERENCES

[1] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models.," *International Journal of Computer Vision*, pp. 321–331, 1987.

[2] D. Terzopoulos, A. Witkin, and M. Kass, "Constrains on deformable models: Recovering 3D shape and mon-rigid motion," *Artificial Intelligence*, vol. 36, pp. 91–123, 1988.

[3] C. Nastar and N. Ayache, "Frequency-based nonrigid motion analysis: Application to four dimensional medical images.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 11, pp. 1069–1079, 1996.

[4] A. Pentland and S. Sclaroff, "Closed-form solutions for physically-based shape modeling and recognition.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 730–742, 1991.

[5] C. Nikou, G. Bueno, F. Heitz, and J. P. Armspach, "A joint physics-based statistical deformable model for multimodal brain image analysis.," *IEEE Transactions on Medical Imaging*, vol. 20, no. 10, pp. 1026–1037, 2001.

[6] D. Terzopoulos, A. Witkin, and M. Kass, "Symmetry-seeking models for a 3-D object reconstruction," *International Journal of Computer Vision*, vol. 1, no. 3, pp. 211–221, Oct. 1987.

[7] K. J. Bathe, *Finite Element Procedure*, Prentice Hall, Englewood Cliffs, New Jersey, 1996.

[8] C. Nastar, *Models physiques deformables et modes vibratoires pour l'analyse du mouvement non-rigide dans les images multidimensionnelles*, Ph.D. Thesis, Ecole Nationale des Ponts et Chaussees, 1994.