Geometric Deformation by Merging a 3D-Object with a Simple Shape

Philippe Decaudin
INRIA
Rocquencourt, BP 105
78153 Le Chesnay Cedex, FRANCE
Philippe.Decaudin@inria.fr
http://www-rocq.inria.fr/syntim/research/decaudin

Published in Graphics Interface'96 proceedings, May 21-24, 1996, Toronto (Canada)

Abstract

Deformation techniques are often used to model the shape of geometric objects. This paper presents a new geometric deformation technique allowing local deformation of the shape of an object by merging the object with a simple 3D-shape (sphere, ellipsoid, ...). Two kinds of effects can be obtained: the simple shape is used to produce either a bump or a dent on the object. The object is deformed so that it includes *or* it embeds the simple shape. In order to deform the object, the 3D-space where it lies is continuously deformed so that the object is bumped and the shape of the bump corresponds to the simple 3D-shape. If the surfaces of both the original object and the simple shape are smooth (continuously differentiable), the surface of the deformed object is also smooth. Moreover, topological properties of the object are unchanged, and the volume variation of the deformed object is controlled by the volume delimited by the simple shape.

An interactive deformation tool based on this technique is presented. It comprises a convex 3D-shape and a center (a 3D point included in the shape). The user can interactively control the deformation by changing the parameters of the tool shape and the position of the tool center. Control of the parameters is particularly intuitive.

Résumé

Les techniques de déformation sont souvent utilisées pour modeler la forme d'objets géométriques. Cet article présente une nouvelle technique de déformation qui permet de déformer localement un objet en le *fusionnant* avec une forme 3D simple (sphère, ellipsoïde, ...). Deux types d'effets peuvent être obtenus: la forme simple est utilisée pour produire une déformation en bosse ou en creux. L'objet est déformé de façon à inclure *ou* à englober la forme simple. Afin de déformer l'objet, l'espace 3D dans lequel il est placé est déformé continument de telle façon que l'objet soit bosselé et que la forme de la bosse corresponde à la forme simple. Si les surfaces de l'objet initial et de la forme simple sont lisses (continument différentiables), la surface de l'objet déformé l'est aussi. De plus, les propriétés topologiques de l'objet sont inchangées et la variation de volume de l'objet déformé est contrôlé par le volume délimité par la forme simple.

L'article présente un outil de déformation interactif basé sur cette technique. Il est composé d'une forme 3D convexe et d'un centre (un point 3D inclus dans la forme). L'utilisateur peut contrôler interactivement la déformation en modifiant les paramètres caractérisant la forme de l'outil et la position du centre de l'outil. Le contrôle des paramètres est particulièrement intuitif.

1 Introduction

Many deformation techniques have been developed; they can be used to modify either globally or locally the shape of a 3D-object. They can be classified in two categories:

Physically based deformations use a set of linked particles. Particles are associated with a 3D-object. They are subject to laws of physics and their displacement generates a deformation on the object [14, 9, 15]. These deformation techniques are useful to deform an object according to a natural phenomenon [4], or to deform a surface so that it approximates the boundary of 3D data [16].

Geometric deformations directly deform the geometric model of a 3D-object in different ways:

- By modifying the parameters which define the model of an object. For instance, an object defined by spline patches can be deformed by moving its control points [5, 10]; an object implicitly defined by a potential field [17, 19, 18] can be deformed by modifying the potential parameters or by merging it with an other implicit object.

- By applying a mathematical transformation. In this case, the 3D-space where the object lies is deformed, hence communicating a deformation to the object. For instance, the Free Form Deformation technique [13, 6, 7] deforms a 3D spline lattice by moving its control points, and the part of the object enclosed in the lattice is deformed. In [3], a transformation is applied directly to the object by moving some points of its surface, the object is deformed consequently. This kind of deformation does not depend on the object representation.

Geometric deformations are often implemented in interactive modeling softwares. The user deforms an initial (simple) object to model a more complex one. He completely controls the deformation which is often computed in real time.

Our deformation technique, described in section 2, is a geometric deformation, it corresponds to a 3D-space transformation. The deformation tool based upon this technique is composed of a convex shape located in space (the tool shape) and a 3D-point (the tool center) included in the tool shape. The tool can deform an object in two ways:

- if the tool center is inside the object, the tool shape is used to produce a bump on the object so that the deformed object includes the tool shape (figure 1),

- if the tool center is outside the object, the tool shape is used to produce a dent on the object (the complement of the object is merged with the tool shape ,figure 2).

This technique has some interesting features:

- The technique is interactive. Deformations are computed in real time on a standard graphic workstation (it has been implemented in C++ on an SGI Indigo workstation) and the user can interactively modify the tool attributes.

- Using the tool is intuitive. Consider that the object is made of a soft material (like rubber) and that a balloon is inflating inside or outside of it (depending on the position of the tool center), the shape of the balloon corresponds to the tool shape. This bumps the object and the shape of the bump corresponds to the tool shape. Thus, the user may easily guess what the result will be.

- The presented technique does not depend on the representation of the object. Especially, objects described by a polygonal representation of their surface, which are often used in interactive modelers, are well adapted. This case is presented in section 3. A simple method which refines the mesh to approximate more closely the deformed object's surface is proposed.

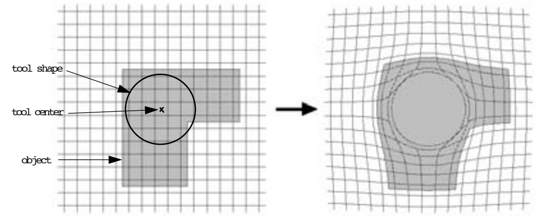

Figure 1: The tool produces a bump on the object

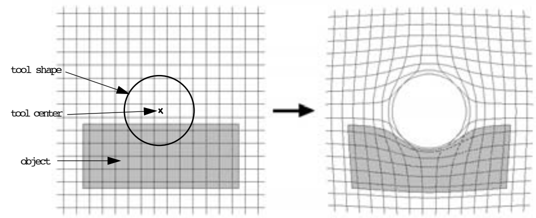

Figure 2: The tool produces a dent on the object

- This technique allows deformations that only composition techniques can produce. A significant example consists in a drop which forms on the surface of an object. Implicit modeling simulates this by adding a new potential field to the object but it requires an implicit description of the object. Our tool simulates this by merging the explicitly described object with a tool having a spherical shape. The shape of the tool is not restricted to sphere shapes, any convex shape would fit. Other techniques allowing composition such as the techniques based upon the Minkovski sum [11] or a fusion technique [8] directly compose two objects. The construction of the resulting object involves complex mesh operations and topological transformations, so it is not easy to use them in an interactive modeler as a deformation tool.

2 The deformation technique

2.1 Description

In order to deform an object, we deform the 3D-space (named $\mathcal{E}$) in which the object is located. The tool shape ($\mathcal{S}$) and the tool center (H) define the deformation. We assume that $\mathcal{S}$ is convex and H is included into $\mathcal{S}$. The deformation derives from a simple transformation which maps $\mathcal{E}-\{H\}$ onto $\mathcal{E}-\mathcal{S}$ (a hole is created in the space, see figure 3). Polar coordinates are used to define this transformation.

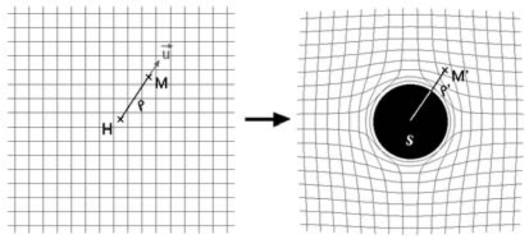

Figure 3: Space deformed by a spherical tool (in 2D)

Let M be a point of $\mathcal{E} - \{H\}$,

let $\vec{u}$ be a normalized vector in the direction of $\overrightarrow{HM}$ (i.e. $\vec{u} = \frac{\overrightarrow{HM}}{||\overrightarrow{HM}||}$),

let ρ be the distance between H and M (i.e. $\rho = ||\overrightarrow{HM}||$) $\Longrightarrow M = H + \rho\vec{u}$,

let I be the intersection[1] between the boundary of $\mathcal{S}$ and the half straight line $(H, \vec{u})$,

let ρ_0 be the distance between H and I (i.e. $\rho_0 = ||\overrightarrow{HI}||$) $\Longrightarrow I = H + \rho_0\vec{u}$.

The point M is transformed onto M' so that

$$M' = H + \rho'\vec{u} \quad where \quad \rho' = \sqrt[3]{\rho_0{}^3 + \rho^3} \quad (1)$$

We have chosen this transformation because it has several interesting properties described below.

2.2 Properties

- Smoothness:

The formula (1) defines a continuous mapping of $\mathcal{E} - \{H\}$ onto $\mathcal{E} - \mathcal{S}$: given a $\mathcal{C}^n$ surface (continuously differentiable at order n), if the surface of the tool shape is $\mathcal{C}^n$, the deformed surface is also $\mathcal{C}^n$. This property insures that our tool produces smooth deformations.

- Locality:

The deformation is located around the tool shape. If a surface is far from $\mathcal{S}$, the deformation is negligible:

$$\rho' = \sqrt[3]{\rho_0{}^3 + \rho^3} = \rho\sqrt[3]{1 + \left(\frac{\rho_0}{\rho}\right)^3}$$

If $\rho \gg \rho_0$, then $\rho' \approx \rho$.

- Volume conservation:

An object of volume V including H, will be transformed into an object of volume V' so that $V' = V + V_0$ where V_0 is the volume of $\mathcal{S}$. An object that does not include H is transformed into an object having the same volume.

This property is implied by the following feature:

The volume of a transformed angular sector is increased by the volume of the intersection of the sector and S. This will be shown in a 2D-space (figure 4):

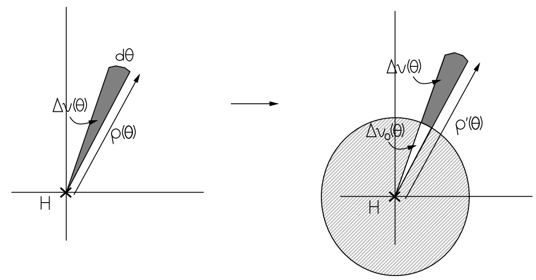

Figure 4: Volume conservation (in 2D)

Let $\rho_0(\theta)$ and $\rho(\theta)$ be the polar functions describing the boundary of $\mathcal{S}$ and the arc of an angular sector centered on H respectively. Let $\Delta v_0(\theta)$ and $\Delta v(\theta)$ be the areas of angular sectors defined by $(\rho_0(\theta), d\theta)$ and $(\rho(\theta), d\theta)$ respectively.

$$\Delta v_0(\theta) = \frac{1}{2}\rho_0(\theta)^2 d\theta$$

$$\Delta v(\theta) = \frac{1}{2}\rho(\theta)^2 d\theta$$

Let find a polar function $\rho'(\theta)$ associated to the deformed arc so that $\Delta v'(\theta)$, the area of the angular sector defined by $(\rho'(\theta), d\theta)$, satisfies:

$$\Delta v'(\theta) = \Delta v_0(\theta) + \Delta v(\theta)$$

This formula can be interpreted as the conservation of the area inside an angular sector.

$$\Delta v'(\theta) = \frac{1}{2}\rho'(\theta)^2 d\theta$$

$$\frac{1}{2}\rho'(\theta)^2 d\theta = \frac{1}{2}\rho_0(\theta)^2 d\theta + \frac{1}{2}\rho(\theta)^2 d\theta$$

By solving this equation, we determine that

$$\rho'(\theta) = \sqrt{\rho_0(\theta)^2 + \rho(\theta)^2}$$

The extension to 3D-space is immediate: by solving $\Delta v' = \Delta v_0 + \Delta v$ where $\Delta v'$, Δv_0, Δv are the volumes of three dimensional angular sectors, we determine that

$$\rho' = \sqrt[3]{\rho_0{}^3 + \rho^3}$$

[1] I is unique because $\mathcal{S}$ is convex and H is inside $\mathcal{S}$.

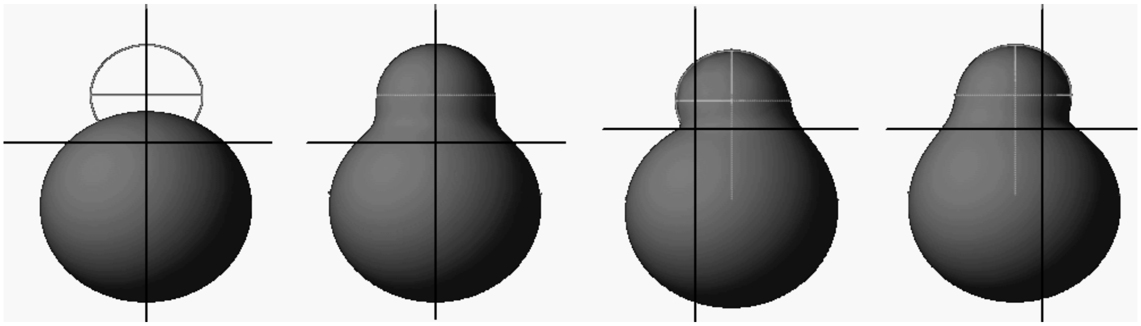

Figure 5: Influence of the position of the tool center

Hence, using this formula, we built a transformation that preserves the volume along an angular sector. This implies that the volume of a transformed object is unchanged or increased by the volume of $\mathcal{S}$.

- Control of the deformation:
The deformation is easily controllable by changing the parameters of the tool shape. Therefore, it can be scaled, rotated and moved interactively by the user. Figures 7 and 8 illustrate these manipulations. By applying several simple deformations to a simple 3D-object, complex objects can be created; on figure 9 a simple 3D-object (an ellipsoid) is deformed thanks to several tools to model a cat.

Relative position of the tool center inside the tool shape also controls the deformation. For instance, in figure 5, a sphere is deformed by a spherical tool. If the tool center is inside the sphere, the sphere is deformed so that it includes the tool. The volume of the deformed object is increased by the volume of the tool shape and the relative position of the tool center inside the tool shape controls the direction to which the volume is added.

3 Case of an object described by a mesh

In computer graphics, a 3D-object is often represented by a vertex/edge/facet network which approximates its surface. In order to deform such an object, vertices are moved using the transformation (1) defined in section 2.1. But, the resulting mesh has also to be refined (i.e. facets are split, which creates new vertices) to approximate more closely the deformed surface. For instance, let's consider a cube, it could be represented by 6 facets, 12 edges and 8 vertices. If it is deformed by a spherical tool, more than 6 facets are needed to approximate its surface, especially near the tool shape. See figure 6.

We propose a simple method which refines the mesh of a deformed surface. This method is adapted to the deformation technique presented in this article.

It is based upon the locality of the deformation. If $\rho \gg \rho_0$, the deformation is negligible (see section 2). An influence zone of the tool is defined by the criterion $\frac{\rho}{\rho_0} < \alpha$ where α is specified by the user ($\alpha > 1$). If a facet is included into or intersects the influence zone, it is marked and it will be split to fit the deformation.

The splitting algorithm is iterative. We know the shape of the tool and how to refine a mesh so that it fits well the surface of this simple tool shape: in order to control the refinement, a useful criterion is the length of the facets edges. For instance, if the tool shape is a sphere of radius r, we consider that a mesh that fits well the surface of the tool shape should have edges so that their length is less than βr where β is also specified by the user ($0 < \beta < 1$). In order to refine the mesh of the deformed object, marked facets are split if at least one of its edges does not satisfy the length criterion. Hence, new facets and vertices are created; vertices are moved onto the deformed surface using the transformation (1). This operation is iterated until every marked facets or newly created ones satisfy the criterion.

Results of this algorithm are shown on figures 6 and 9. The user can adjust the mesh fineness by modifying α and β. The mesh refinement is also computed in real time, which allows interactive adjustments.

4 Conclusion

By merging a simple shape with a 3D-object, deformation effects are easily obtained. A complex object can be interactively and quickly designed by deforming it several times with our deformation

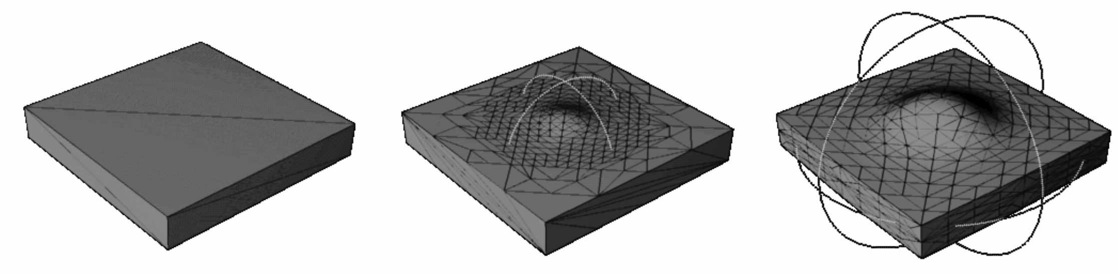

Figure 6: Mesh of a box deformed by merging a spherical tool

tool. The control of the deformation is intuitive for the user because it directly depends on the tool shape and the tool center that the user interactively modifies. The examples we have presented use spherical or ellipsoidal tools; other kinds of convex tool shapes, like cubes or cones, produce interesting effects because they create sharp edges and conic points on the deformed object (a discontinuity appears in the deformed 3D-space).

To produce animation effects, the shape of the object can be continuously deformed through time by animating the tool shape (its parameters are considered as time-dependent functions) and re-computing the deformation. An animation discontinuity will appear if the tool center moves through the surface of the object. The deformation tool may be extended to more complex tool shapes and to other kinds of tool centers like curves. But the more complex the tool is, the less intuitive it is.

References

[1] A.H. Barr. Global and local deformations of solid primitives. In *Computer Graphics (SIGGRAPH'84 proceedings)*, volume 18(3), pages 21–30, July 1984.

[2] J. Bill and S. Lodha. Sculpting polygonal models using virtual tools. In *Proc. Graphics Interface'95*, pages 272–279, May 1995.

[3] P. Borrel and D. Bechmann. Deformation of n-dimensional objects. In *Symposium on Solid Modeling Foundations and CAD/CAM Applications*, pages 351–369, June 1991.

[4] D. Breen, D. House, and M. Wozny. Predicting the drap of woven cloth using interacting particles. In *Computer Graphics (SIGGRAPH'94 proceedings)*, volume 25(4), pages 365–372, July 1994.

[5] B.S. Cobb. *Design of Sculptured Surfaces Using the B-Spline Representation.* PhD thesis, University of Utah, June 1984.

[6] S. Coquillart. EFFD: a sculpturing tool for 3D geometric modeling. In *Computer Graphics (SIGGRAPH'90 proceedings)*, volume 24(4), pages 187–196, August 1990.

[7] S. Coquillart and P. Jancène. Animated Free-Form Deformation: an interactive animation technique. In *Computer Graphics (SIGGRAPH'91 proceedings)*, volume 25(4), pages 23–26, July 1991.

[8] Ph. Decaudin and A. Gagalowicz. Fusion of 3D shapes. In *Fifth Eurographics Workshop on Animation and simulation. Oslo, Norway*, September 1994.

[9] H. Delingette, Y. Watanabe, and Y. Suenaga. Simplex based animation. In *Models and Techniques in Computer Animation*, pages 13–28, 1993.

[10] D.R. Forsey and R.H. Bartels. Hierarchical B-spline refinement. In *Computer Graphics (SIGGRAPH'88 proceedings)*, volume 22(4), pages 205–212, August 1988.

[11] A. Kaul and J. Rossignac. Solid-interpolating deformations: Construction and animation of PIPs. In *Proc. Eurographics'91*, pages 493–505, September 1991.

[12] J.R. Kent, W.E. Carlson, and R.E. Parent. Shape transformation for polyhedral objects. In *Computer Graphics (SIGGRAPH'92 proceedings)*, volume 26(2), pages 47–54, July 1992.

[13] T.W. Sederberg and S.R. Parry. Free-Form Deformation of solid geometric models. In *Computer Graphics (SIGGRAPH'86 proceedings)*, volume 20(4), pages 151–160, August 1986.

[14] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. In *Computer Graphics (SIGGRAPH'92 proceedings)*, volume 26(2), pages 185–194, July 1992.

[15] D. Terzopoulos and K. Fleisher. Deformable models. In *The Visual Computer*, volume 4(6), pages 306–331, 1988.

[16] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3D shape and nonrigid motion. In *Artificial Intelligence*, volume 36(1), pages 91–123, 1988.

[17] B. Wyvill, J. Bloomenthal, T. Beier, J. Blinn, A. Rockwood, and G. Wyvill. Modeling and animating with iplicit srfaces. In *Siggraph Course Notes*, volume 23, 1990.

[18] B. Wyvill, C. McPheeters, and G. Wyvill. Animating soft objects. In *The Visual Computer*, volume 2, pages 235–242, 1986.

[19] G. Wyvill, C. McPheeters, and B. Wyvill. Data structure for soft objects. In *The Visual Computer*, volume 2, pages 227–234, 1986.

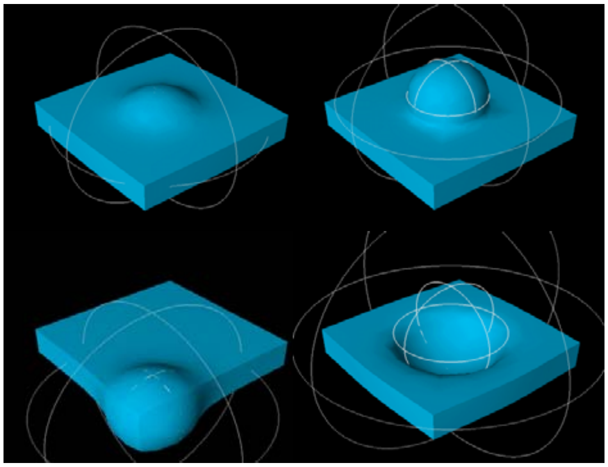

Figure 7 : a box deformed by a spherical tool

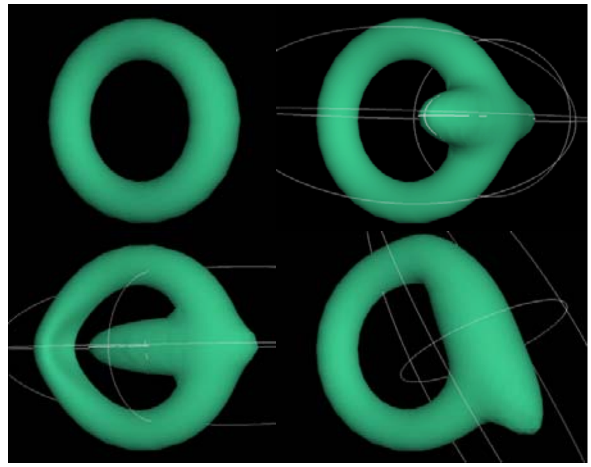

Figure 8 : a torus deformed by an ellipsoidal tool

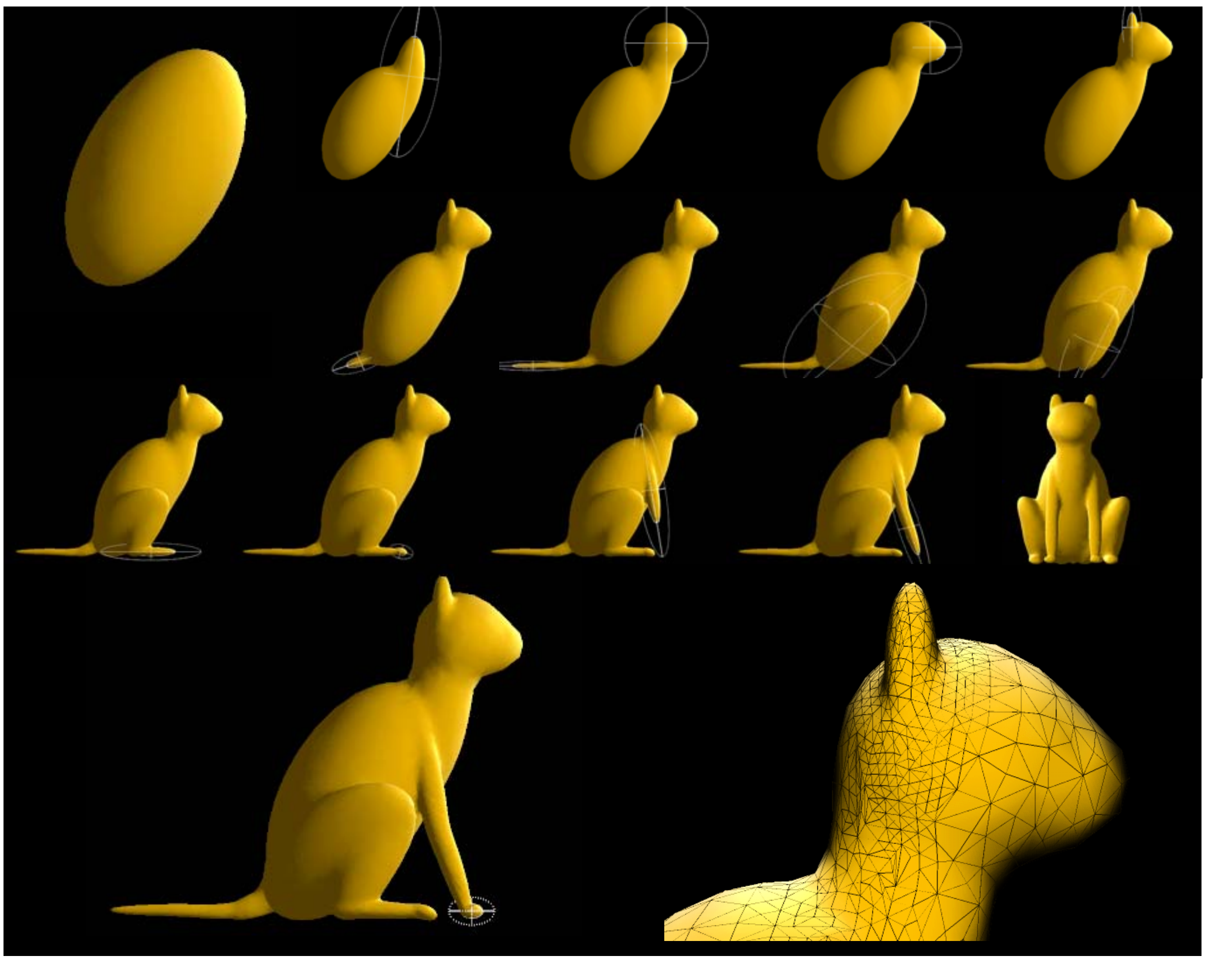

Figure 9 : an ellipsoid is deformed to model a cat