Anisotropic Scaling of Model Using Cage-Based Deformations

Huanyu Yang¹, Kuangrong Hao² and Danli Xu¹
1College of Technology, Shanghai Open University, P.R. China
2College of Information Sciences and Technology, Donghua University, Shanghai 201620, P.R. China
Email:yhuanyu77@163.com

Abstract. This paper proposes a method for anisotropically adjusting model size based on cage deformation technology. In this method, firstly the initial cage is constructed for the model according to the model features or the content definition. Then, the scale variations of each side of cage are adjusted in different directions by the minimum elastic energy method. Finally, the deformation result conforming to the model size adjustment constraints is calculated using the mean value coordinates. The result of the experiments on multiple different complexity models demonstrate that the proposed method can achieve anisotropic model size adjustment while maintaining the sensitivity features of the model.

1. Introduction

Processing of geometric model for reacquiring new geometric model is an effective geometric modeling approach. Through the scale deformation of model, i.e. "the scaling of geometric model", a new geometric model can be obtained on the basis of the original model. Since the human visual perception of various model parts is non-uniform, some zones will receive more visual attention. Therefore, geometric scaling should consider the maintenance of visual naturalness of model size adjustment, in order to protect the visually sensitive zones of the geometric model. The model size adjustment should be multidirectional scaling of model based on actual needs, i.e. anisotropic model size adjustment, rather than simple increase and shrinkage. Dimensional adjustment of 3D model is essentially a special deformation technology.

With the rapid development of 3D model processing technology, the computer processed model has become increasingly accurate, so the applicability and computational efficiency of geometric processing algorithm are also becoming growingly demanding. The use of wrapping mesh similar in shape to the original model but with fewer vertices as the proximal geometry for bridging complex model and algorithm is a trend. Cage is a polygon mesh that is capable of wrapping the original model, which has greatly reduced number of vertices and facets compared to the original model. Using cage technology, the problem of dealing with high-precision complex model can be transformed into indirect processing of simple control mesh, which allows the algorithm to be independent of the geometric representation and complexity of the model, thereby greatly reducing the complexity of the algorithm and extending its applicability. Therefore, cage is widely applied in the research fields like spatial deformation, deformation migration, collision detection and mesh subdivision. The cage-based deformation technology has become the hot topic of deformation technology research in recent years.

In this paper, a method for anisotropic model scaling based on cage deformation technology is introduced. The anisotropic deformation process of model is mapped as the deformation process of cage control mesh. Firstly, the model is divided into different zones according to the model feature variation requirements, and the initial cage control mesh is constructed for different zones, where the computation model corresponds to the mean value coordinates of the initial mesh. Then, the target
cage of model deformation is reconstructed based on the minimum elastic energy method with the size adjustment or deformation requirement as the constraint. Finally, size-adjusted model is obtained by calculation based on the newly generated target cage and the mean value coordinates. The experimental results show that the application of method presented in this paper allows anisotropic size adjustment of model while preferably maintaining its sensitivity features.

2. Related Works

2.1. Cage-Based Deformation

The cage-based deformation is a kind of free deformation technology, whose theoretical basis comes from the barycentric coordinate interpolation. In 1986, Sederberg and Parry first proposed the cage-based deformation technique [1]. Afterwards, the technology gained wide application since it directly used free control mesh to control the surface deformation, which was intuitive in form and did not require addition of other constraints. Figure 1 demonstrates the implementation of cage-based model deformation as in reference [2]. It mainly involves the research on the construction process of cage, the definition mode of cage coordinates and the mode of user interaction.

![Figure 1. Implementation of cage-based model deformation [2]](image)

Cage is a closed 2-manifold mesh, whose geometry approximates the target model but with fewer vertices. The problem of Cage generation can be formally described as follows: Given a point set \( p_j \in \mathbb{R}^3 \) and a constant \( c \) in a 3D space, a closed polygon mesh \( \Omega = \{V, F\} \) (usually triangular) is sought, where \( V \) represents the set of mesh vertices, and \( F \) represents the set of mesh surfaces. The space surrounded by the polygonal grid \( \Omega \) is \( M \), whose volume should be as small as possible while satisfying \(|F| < c \) and establishing \( p_j \in M \) for any \( j \). The volume constraint of \( M \) is for ensuring the similarity between the cage and the target model, which has great influence on the deformation control and deformation quality. There should also be some constraints on the number of facets for Cage, because excessively many facets means high complexity of cage, which increases the complexity of calculating cage coordinates. Currently, the majority of cage construction adopts manual or semi-automatic mode. Xian et al. [3] proposed a voxel-based automatic cage mesh generation method, where the triangular mesh model was voxelized using the axis-aligned bounding box to extract the faces of adjacent voxels on the outside of voxels that intersects the model surface. Then, triangulation and smoothing operation were performed to generate the cage. Deng et al. [4] employed the idea of progressive convexity-preserving simplification to construct a cage. Ju et al. [5] introduced a cage deformation with reusable skin template, where fixed topology polygon templates wrapping the model were predefined at the joints and bones, and an initial cage was formed by connecting adjacent templates. Li Lin et al. [6] used the distance field of triangular mesh model to create equidistant cage meshes for the model. Chen Xue et al. [7] inversely simulated the visual hull generation process. For images of different perspectives, corresponding polygonal contours were generated and then projected as cones, which intersected to generate the cage of model. In this paper, no strict restrictions are placed on the cage volume constraints due to the simplicity of model deformation conditions involved.

The theoretical basis for the Cage-based deformation technology is the barycentric coordinate interpolation. In 1827, Mobius proposed the concept of barycentric coordinate for the first time when...
studying functional interpolation [8]. Any point \( v \) in the plane triangle \( \Delta v_i v_{i+1} v_{i+2} \) can be represented by the linear combination of the three vertices on the triangle, and the coefficient of the linear combination is precisely the area coordinates in the finite element analysis. That is, the barycentric coordinates of point \( v \) are the area ratio \( \Delta v_i v_{i+1} v_{i+2} \) to \( \Delta v_i v_{i+3} v_{i+2} \). The generalized barycentric coordinates are widely used in digital geometry processing, mainly in the areas of computer simulation, FEM calculation, 3D model parameterization and deformation of 3D mesh model.

3. Algorithm

3.1. Overview
The main steps of the anisotropic model size adjustment algorithm based on cage deformation technology are shown in Figure 2.

![Figure 2. Algorithm](image)

The specific description is as follows:
Step 1: Model \( H \) is partitioned into \( t \)-number of different subsets \( H = \{ H_1, H_2, \cdots, H_t \} \) based on the feature definition or content definition;
Step 2: Bounding boxes \( B_1, B_2, \cdots, B_t \) are created for each subset in \( H_1, H_2, \cdots, H_t \), which are then boolean intersected following appropriate transform to obtain the initial cage control mesh \( HC_0 \) of the model.
Step 3: The mean value coordinates \( O_{MVC} \) for all points of model \( H \) are calculated based on the model’s initial cage control mesh \( HC_0 \).
Step 4: Elastic energy function is constructed for the change \( \Delta e \) at each edge \( e \) of the initial cage control mesh \( HC_0 \), and the size adjustment constraints for different zones of model are converted into the constraints of cage variation. Under the anisotropic size change constraints, the energy function for cage variation is solved to get the target cage control mesh \( HC \).
Step 5: The final size-adjusted model is obtained by calculation based on the mean value coordinates \( O_{MVC} \) of all points in the target cage control mesh \( HC \) and model \( H \).

When creating the initial cage, model partitioning must be performed first. Depending on the complexity of model, different module partitioning approaches can be selected. If the model features are simple, the simple K-mean clustering can be used. If the model features are more complicated or have attitude changes, other model partitioning methods can be chosen. On the basis of model partitioning, initial cage of model is created. The construction procedure of the initial cage will be described in detail in Section 3.2. After creating the initial cage mesh, the mean value coordinates \( O_{MVC} \) of the model are obtained by calculating the 3D mean value coordinates. The construction of target cage will be described in detail in Section 3.3. Finally, in Section 3.4, the computational process for obtaining the target model will be elaborated.

3.2. Construction of Initial Cage
Depending on the complexity of model, different methods are adopted for cage construction. The adjustment of model size is not demanding on the details of model deformation. The deformation of
model is mainly the translation of model points, which basically does not involve the rotation of model. Creation of complex cage that is tightly integrated with model requires a long computational time, so we will adopt a relatively simple method for automatically building cage control meshes. To enable the anisotropic size adjustment of model according to the definition of content or model features, we need to consider the association between the model features and the control mesh when constructing the cage. During the construction of cage control mesh, triangular meshing of the generated mesh framework is needed. For the algorithm in this paper, NetGen software is used, which is a mesh (surfaces and solids) generation software programmed by an Austrian scientist Joachim Schoeberl. As an extremely advanced and sophisticated meshing technology, it has gained widespread use in the field of 3D mesh generation. The construction steps of initial cage in the algorithm presented are shown in Figure 3. The details are as follows:

Step 1: The model is partitioned using the model partitioning algorithm. Figure 3(a) displays the point cloud partitioning result of the block model.

Step 2: Bounding boxes \( B = \{ B_1, B_2, \ldots, B_t \} \) are created separately for the model blocks. If \( be_i \) is the diagonal of \( B_i \), \( i = 1,2,\ldots,t \), a relaxation coefficient \( \xi \) is set up, and the diagonal of \( B_i' \) is assumed to be \( be_i' = be_i + \xi \). The purpose of setting up the relaxation coefficient \( \xi \) is to avoid the overlapping of boundaries between control mesh and model. \( B' = \{ B_1', B_2', \ldots, B_t' \} \) is obtained. Figure 3(b) presents the bounding box set of the block model.

Step 3: \( B' = \{ B_1', B_2', \ldots, B_t' \} \) is boolean intersected with the initial framework of cage, then the mesh framework is subjected to triangular meshing on the NetGen software. Figure 3(b) displays the initial cage meshing result of the block model.

3.3. Target Cage Construction Based on Energy Optimization Method

The construction of target cage has been the focus and challenge of research on the cage-based deformation technology. In general, the target cage is constructed manually via user interaction. Cage is actually a 3D model of a triangular mesh. In this paper, we will transfer the cage deformation process of model to the deformation process of the model. Therefore, the energy optimization-based deformation method is applied to the construction of target cage first. Unlike the deformation process of original 3D model, cage requires less points, surfaces, edges and simpler constraints.

We created a discrete point energy equation for curve with each point on the curve as the computing unit, thereby achieving the energy optimization-based curve deformation with the arc length as constraint. The specific equation is shown in Eq. (1).

\[
E(V) = \frac{1}{2\Delta L} \sum_{i=1}^{n-1} \left( 2 - 2 \cos(\Phi(i)) \cos(\Phi(i-1)) - 2 \sin(\Phi(i)) \sin(\Phi(i-1)) \right)
\]

In this paper, we construct an elastic energy equation for the cage mesh by introducing elastic energy for each edge \( e \) of the cage triangular mesh, that is, by treating each edge as a spring. According to the elastic energy function of physics: \( F = k \Delta x \). If each edge \( e_i \) of mesh model and the elastic modulus \( k_i \) changing along the size adjustment direction are determined, it is possible to calculate
the elastic force generated from the shrinkage of each edge along the size adjustment direction, thereby allowing the construction of elastic energy system that changes with the length of cage mesh. The original cage mesh is precisely in an elastic energy balance state.

If the mesh is resized to a certain scale \( l \), the elastic energy produced by change of each mesh edge by \( l \) is calculated with least square method. The \( \Delta e_i \) value satisfying the energy function minimization is solved, thereby obtaining the new energy balance of the whole mesh. The energy function is as follows:

\[
E = \sum_{i \in E} \left| k_i (\Delta e_i) \right|^2 = \sum_{i \in E} \left| k_i \tilde{s}_i - k_i l s_i \right|^2
\]

(2)

Where \( s_i \) represents the original projection of edge \( e_i \) in the scaling direction \( u \); \( \tilde{s}_i \) represents the projection of edge \( e_i \) in the scaling direction after adjustment; \( l \) denotes the scaling multiple of adjusted target cage; \( k_i \) is the elastic coefficient of edge, and \( k_i \in [0,1] \). By setting the direction of model size variation \( u \); the edge \( e_i \) of cage needing adjustment; and the scaling multiple \( l \) of target under adjustment, the target cage mesh can be obtained. The problem of cage mesh deformation that minimizes the energy can be written as follows:

\[
\begin{align*}
\min & \ E(K) \\
\text{s.t.} & \ l = s
\end{align*}
\]

(3)

In the problem of solving the mesh elastic energy optimization shown in this paper, the objective function is a nonlinear function, while the constraint condition is a linear function. We used the sequential quadratic programming (SQP) algorithm to solve the quadratic programming problem with the linear constraint of this paper.

Figure 4 illustrates the cage mesh deformation process of the block model. Figure 4(a) shows the block model and its initial cage mesh. Figure 4(b) presents the deformation result of target cage mesh that is magnified by double at both ends of the block model. Clearly, the target cage mesh changed the length of edges at the two ends of mesh. Nonetheless, the middle part of the cage mesh remains unchanged.

Figure 4. Cage-based mesh deformation process of the block model (a) Initial cage mesh (b) Deformation result of target cage mesh (c) Cage-based mesh deformation result

3.4. Implementation of Model Deformation

Let all the points in model \( H \) be \( P = \{ p_i \} \), \( i = 1,2 \ldots n \); all the points in the initial cage control mesh \( HC_0 \) be \( C = \{ c_j \} \), \( j = 1,2 \ldots m \); and all the points in the target cage control mesh \( HC \) be \( C' = \{ c'_j \} \), \( j = 1,2 \ldots m \). If the mean value coordinates of point \( p_i \) are \( \{ \lambda p_1, \lambda p_2, \ldots \lambda p_m \} \), and \( p_i \) changes to \( \hat{p}_i \) after the implementation of model deformation, then:

\[
\hat{p}_i = \sum_{j=1}^{m} \lambda c'_j
\]

(4)
According to the method in Section 3.2, the initial cage mesh is constructed for the block model as shown in Figure 4(a). If the requirement of model size adjustment is to elongate the cylinders on two sides without changing the middle block, the target cage mesh constructed by method in Section 3.3 will be used as shown in Figure 4(b). The mean value coordinates for each vertex of the model are obtained by the initial cage mesh calculation. After deformation of cage mesh, the final model is calculated by using the Figure3 as shown in Figure 4(c).

4. Results and Analysis

The algorithm in this paper is implemented on a computer with the Intel® Core™ i5 CPU 1.60 GHz and 4.0 GB memory. The model partitioning algorithm, the construction of initial cage mesh, the construction of target cage mesh and the generation of target model are implemented using Matlab software, while the computation of mean value coordinates is realized with C++ language in Visual Studio 2010 programming environment.

4.1. Comparison of Algorithm

Kraevoy et al. [10] proposed the setting up of a corresponding bounding box for the model under scaling. Then, the model points were embedded into the hexahedral protective mesh cells, and the vulnerability of each mesh cell is evaluated with Slippage analysis and normal curvature. The model size adjustment is driven by the scaling of hexahedral mesh. We compared the present algorithm with that in literature [10].

Figure 5 displays the anisotropic scaling results of the block model. The cylinders at both ends of the block model are magnified by a factor of 2. Figure 5(a) shows the original model, while Figure 5(b) presents the result of uniform magnification. The results obtained by the method in reference [10] are shown in Figure 5(c), while the model size change results achieved by the method in this paper are shown in Figure 5(d). It can be seen that after using uniform magnification, the middle block and circles of the model all underwent large distortion. Using the method in [185], although the circles in the block are kept well, the block is still stretched. In contrast, using the method in this paper, the blocks and circles are all basically kept intact.

Figure 6 presents the anisotropic scaling results of sofa model. The backrest of the sofa is raised, while the handles remain unchanged. Figure 6(a) illustrates the original model; Figure 6(b) displays the result of uniform magnification; Figure 6(c) shows the result obtained by the method in reference [10]; and Figure 6(d) presents the model size change result achieved by the method in this paper. It can be seen that after using uniform magnification, the backrest and handles of the model are simultaneously elevated. Using the method in [10], although the backrest of the sofa model is raised, the sofa handles failed to be elevated simultaneously. But from Figure 6(c), it can be seen that the surface connecting the backrest and handles is not smooth enough, with appearance of sharp corners. In contrast, using the method in this paper, the backrest of sofa model is raised while the handles did not rise simultaneously. Moreover, the connection between backrest and handles is preferably smooth.

![Figure 5. Anisotropic scaling results of the block model. (a)Original model (b) Result of uniform magnification(c) Deformation result using the method in [10] (d) Deformation result by our method](image-url)
Table 1 lists the number of original points, faces; the number of points, faces of cage control mesh; and the time required for computational steps in multiple experimental models. Where the initialization time $t_{init}$ includes the model partitioning time and the initial cage construction time, while the meshing time of NetGen software is excluded. The model size adjusting time $t_{obj}$ includes the generation and optimization of target cage and the final computing time of target model. From the table, we can see that the computation of model's mean value coordinates is most time-consuming among the computational steps. The mean value coordinate computing time $t_{MVC}$ is largely correlated with the number of points of the model and initial cage.

4.2. Algorithm Efficiency Analysis

| Model name | Number of points/faces of original point set | Number of points/faces of initial cage | Computational time (s) | Model size adjusting time $t_{obj}$ | Total time $t_{sum}$ |
|------------|---------------------------------------------|---------------------------------------|------------------------|------------------------------------|---------------------|
| elk        | 2500/5000                                   | 197/390                               | Initialization time $t_{init}$ 0.012 | Mean value coordinate computing time $t_{MVC}$ 10.6 | 0.54 | 11.152 |
| sofa       | 2502/5000                                   | 129/254                               | 0.013                  | 6.9                                | 0.38 | 7.306 |
| homer      | 5103/10202                                  | 30/56                                 | 0.022                  | 3.3                                | 0.25 | 3.77 |
| polygirl   | 9912/19809                                  | 193/382                               | 0.048                  | 41                                 | 0.49 | 41.538 |
| block      | 18083/36174                                 | 24/44                                 | 0.061                  | 9.1                                | 0.21 | 9.371 |

(a) Original model  (b) The height of the model part increases

Figure 7. Anisotropic scaling results of the elk model
Figure 7 presents the height increase result of the elk model's wheel portion. As shown in Figure 7(b), the height of the wheel portion of elk model is significantly increased, but with almost no change in the head. Figure 8 illustrates the leg size increase effect of polygirl model. As shown in Figure 8(b), the length of polygirl model's legs is significantly increased, but without any change in the upper part and the head. Figure 9 demonstrates the anisotropic size adjustment process for the homer model. In Figure 9(b), the legs of homer model are markedly increased. Figure 9(c) presents the length increase result of the homer model's abdominal zone. The height increase results of homer model's abdominal bulge zone are shown in Figs. 9(f) and (h). The deformation results of homer model conform to the requirements of anisotropic size adjustment. After analyzing the experimental results of multiple models, it can be seen that the algorithm presented in this paper can achieve the anisotropic adjustment of model size while preferably maintaining the sensitivity features of the model.

Due to the non-negative constraint of the mean value coordinates, some models are unable to be calculated by the mean value coordinate method for obtaining the deformation results, such as the drillhole model in Figure 10. Figure 10(a) shows the original model, while Figure 10(b) presents the model and its cage. Due to the hole in the model, the mean value coordinates of model are unable to be calculated after creation of cage conforming to the model features. Therefore, for the drillhole model, its size cannot be adjusted using the algorithm described in this paper.

Figure 8. Anisotropic scaling results of the polygirl model

![Polygirl model](image)

(a) Original model (b) Model leg length increases

Figure 9. Anisotropic scaling results of the homer model. (a)(e)(g) Original model; (b) The legs of homer model are increased; (c) The length increase result of the abdominal; (f) The length increase result of the abdominal in front view; (h) The length increase result of the abdominal in side view
5. Conclusion
This paper introduces a method for anisotropic scaling deformation of model based on cage deformation technology. As a special case of model deformation technology, the cage-based deformation requires size adjustment of partial model zones while ensuring that the features of other zones of the model remain unchanged. Cage-based deformation technology is a kind of free deformation technology, which mainly deals with the creation of cage and the selection of barycentric coordinates.

According to the requirements of model deformation, we construct the initial cage of model deformation by using the cage construction technique similar to the model voxel bounding box method. Each edge of cage is subjected to changes at different scales by the elastic energy minimization according to the anisotropic size adjustment requirements of model. Finally, the results of anisotropic model size adjustment are obtained by calculating the mean value coordinates of the model. Through calculation of multiple models with different features and complexities, the adjustment results of anisotropic model size based on model feature or content definition are obtained. Besides, the remaining sensitivity features of model are also preferably maintained.

However, there also exist some problems with the algorithm proposed in this paper. For example, (1) Due to the non-negativity of mean value coordinates, size adjustment by the present algorithm is impossible for some models. (2) The creation of cage is not only related to the features of model, but also to the content of model. Therefore, the cage creation process of partial models needs to be achieved through user interaction.

6. Acknowledgements
This work was supported in part by the Research Project of Shanghai Open University (No. KX1713).

7. References
[1] Sederberg T W, Parry S R 1986. Free-form deformation of solid geometric models [J]. ACM Siggraph Computer Graphics, 20(4) pp 151-160.
[2] Ju T, Schaefer S, Warren J 2005. Mean value coordinates for closed triangular meshes [J]. ACM Transactions on Graphics. 24(3) pp 561-566.
[3] Xian C, Lin H, Gao S SMI 2009, Beijing, China. Automatic generation of coarse bounding cages from dense meshes. [C]. IEEE International Conference on Shape Modeling and Applications. 26-28 June. 2009 pp 21-27.
[4] Deng Z J, Luo X N, Miao X P 2011. Automatic cage building with quadric error metrics [J]. Journal of Computer Science & Technology. 26(3) pp 538-547.
[5] Ju T, Zhou Q Y, Panne M V D, et al 2008. Reusable skinning templates using cage-based deformations[J]. ACM Transactions on Graphics. 27(5) pp 32-39.
[6] Li Lin, Li Guiqing and Huang Liwei 2011. Generation of Equidistance Cages [J]. Journal of Computer-Aided Design & Computer Graphics. 23(6) pp 956-963.
[7] CHEN Xue, LIU Tao, FENG Jie-Qing. Cage Generation Based on Visual Hull [J]. Journal of Software, 2013, 24(10) pp 2379-2390.
[8] Mobius, A. F. Der barycentrische Calcul[M]. 1827.
[9] NetGen, URL: http://www.hpfem.jku.at/netgen/.
[10] Kraevoy V, Sheffer A, Shamir A, et al 2008. Non-homogeneous resizing of complex models. [J]. ACM Transactions on Graphics. 27(5) p 111.