Cage-Based Tree Deformation

Chao Zhu[1,2], Weiliang Meng[1,2], Yinghui Wang[3], and Xiaopeng Zhang[1,2]

[1] Sino-French Laboratory LIAMA, CAS Institute of Automation, Beijing, China
[2] National Laboratory of Pattern Recognition, CAS Institute of Automation, Beijing, China
[3] Department of Computer Science & Engineering,
Xi'an University of Technology, Xi'an, China
zhuchao8@gmail.com, weiliang.meng@ia.ac.cn,
wyh_925@163.com, xpzhang@nlpr.ia.ac.cn

Abstract. Tree models are broadly used in multimedia applications, but it is a challenge task to deform trees with plenty of branches. We propose a novel method to deform the tree model interactively using cages based on trees' property, and we design a framework to control the hierarchy deformation of trees. The bounding box of the trunk is used as the global control cage, while the bounding boxes of branches are set as local cages. The linear combination of mean value coordinates computed from local cages and the global cage decides the vertices deformed positions of the deformed tree. This framework is straightforward and effective, which could also preserve local details of the tree. Experimental results show that our technique is suitable for deforming trees with many branches.

Keywords: deformation, mean value coordinates, tree model.

1 Introduction

Deformation of model is always the hot topic. Since the 1960s, plenty of achievements on research are obtained based on free-form surfaces. The users could deform the complicated models by adjusting the point, the line and the faces of models. This method is popular deformation technique in business software, but it is difficult to hold the geometric detail. Now the cage-based deformation methods on characters are proposed such as MVC (Mean Value Coordinates)-based method [1], [2], HC (harmonic Coordinates)-based method [3], GC (Green Coordinates)-based method [4] and so on. We mainly use the MVC to deform the model of tree. Once the MVC is computed, it is fixed during the whole deformation process. The deformed vertices positions of the tree model are the linear combinations of cage vertices positions, which make sure the computation of deformed models fast.

The plants are popular objects in nature scene, so virtual plant has more and more applications in computer games and 3D films. The number of tree models is limited, but the real trees are in wide range of species and have various shapes. Traditional deformation methods [5] usually do not work for trees. When the tree is deformed as usual, the tree may be split, or the new deformed position can't be computed at all. The deformation based on space is a better choice. However, choosing a suitable

M. Chang et al. (Eds.): Edutainment 2011, LNCS 6872, pp. 409–413, 2011.

space to deform is a delicacy problem. Vast space will weaken the agility of deformation for trees as they have more branches than other models. Here we propose a method to deform the models of tree based on cage, and this method could change the shape of models interactively and flexibly, which in turn generate diversity of the shape of tree model in one species.

This paper focus on the deformation of tree models by using cage, in which the cages are the bounding boxes of region of interest and the hierarchy is constructed by skeleton of the tree. We can choose the whole tree model or the branches of tree model to deform. Using our method and current existing tree models, we can generate more tree shapes which can be used navigation of large-scale environment simulation, virtual reality and similar applications. Here the main contribution of our work is listed as follows:

- A novel method to deform the tree model interactively based on cage, and this method could avoid manipulating the vertices of tree model directly.
- A framework to compute the vertices coordinates of deformed tree model, which could hold the detail of tree model.

2 Tree Deformation

In general tree model has two sections: branches and leaves. It is obvious the trunk and branches decide the whole shape of tree. Deforming the branches that we focus on can decide the main contour of the tree.

2.1 Local Deformation

In our interactive system we input the mesh model of tree, and skeleton model of tree which is gotten using [6]. We manipulate the mesh model and the skeleton of tree provides us with hierarchical information. The branches and trunks of trees are nearly cylindrical, and using bounding box as the cage to deform the tree is enough and straightforward. We build the bounding box as cage of the branches (Fig.1 (a)), and we only compute MVC of the vertices in cage. When we adjust the vertices of the cage, the branches in cage are deformed.

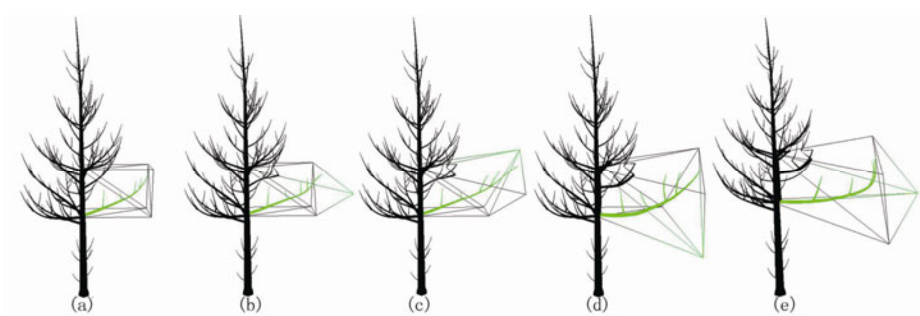

Fig. 1. (a) build the cage of the selected branches, (b)-(e) deform the branches

2.2 Global Deformation

The tree model is made up of trunk and branches, and the contour of the tree is constructed by the trunk and branches. In botany, the trunk makes influence on the whole tree growth, so the deformation of trunk will affect all branches, when we deform the cage of trunk interactively (Fig.2). We compute the MVC based on Ju et al's method [7]. When we adjust the cage, the section of the model outside of cage is also deformed as in Fig.2, which leads to more diversification.

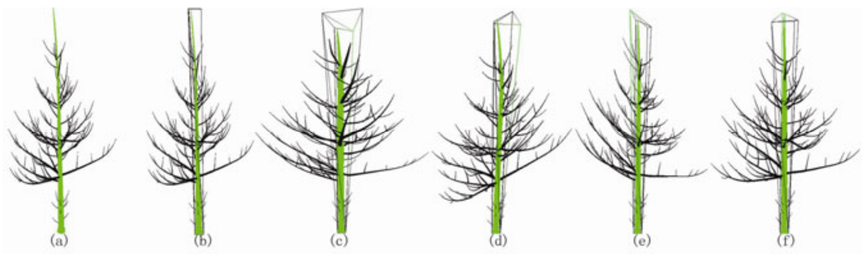

Fig. 2. (a) choose the trunk, (b) build the cage of the selected trunk, (c)-(f) four deformed trees

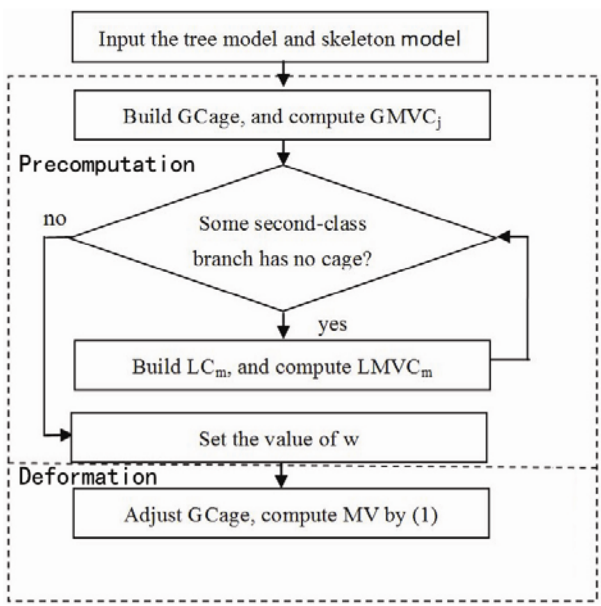

Fig. 3. The pipeline of global deformation

In order to make the branches outside of cage deform under the control, we use the bounding boxes of every second-class branch and its subbranches, which is called local cage (LC_m), and compute MVC according to local cage which is called local MVC ($LMVC_m$). We call the trunk cage global cage (*GCage*) and compute the MVC according to *GCage* which is called global MVC ($GMVC_j$). When we adjust the cage of trunk the coordinate of every vertice of model (*MV*) is computed as follows:

$$MV = w*(\Sigma\, GCage* GMVC_j)+(1-w)*(\Sigma\, LC_m * LMVC_m) \quad (1)$$

We use the parameter *w* to adjust the weight of local MVC and global MVC which belongs to the interval [0, 1]. Fig.3 shows the pipeline of global deformation. The global deformation of tree model is in two steps:

1. Precomputation. In this step the global MVC and local MVC of model are computed.
2. Deformation. We compute the vertices coordinates of the deformed model by (1).

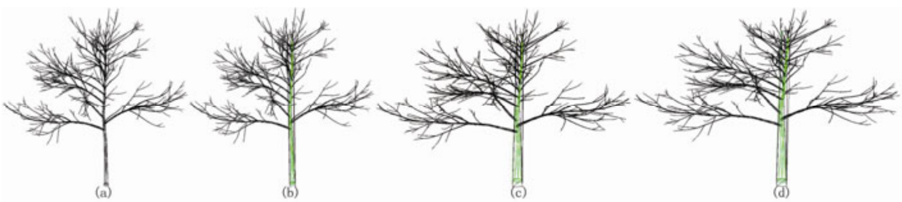

Fig. 4. Apple tree deformation. (a) original tree model, (b) build the cage of the trunk, (c)(d) two deformation, *w*=0.2.

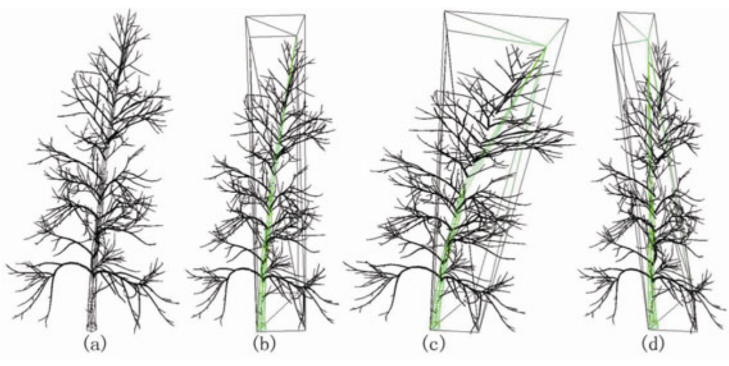

Fig. 5. Birch deformation. (a) original tree model, (b) build the cage of the trunk, (c)(d) two deformations, *w*=0.3.

3 Experiments

We use our framework to deform other tree models and adjust the vertices of the cage almost horizontally. The test is on PC with Intel Core(TM)2 Quad CPU at 2.4GHz and 4G RAM, and our method is written with C language with the support of operation system Windows7. Fig.4 and Fig.5 show the deformation of two trees with no leaf. All the deformation is finished in real time after the precomputation. From the results, we can see that local details of the deformed trees are preserved well.

The parameter w is set to 0.2, 0.3 individually. Compared with other methods of tree deformation such as [8], our method provides less control points to deform the tree model, which is more convenient.

4 Conclusion and Future Work

Tree deformation is a challenge topic because of their abundant branches, which make traditional deformation methods out of action.

In this paper, we propose a cage-based method to deform the tree model, and design a framework which avoids manipulating the vertices of tree model directly and could preserve the local details of tree model perfectly. Experimental results show that our method is straightforward and effective.

Currently the deformation is interactively by controlling the cage manually. In the future we could make the cage move according to the botanic principle of trees to simulating the growing process. We also consider the physical methods to set the cage movement in order to get the animation of windy trees. The collision detection to avoid intersection of branches during the deformation should also be considered and dealt.

Acknowledgments. The authors would like to show thanks to Dr. Hongjun Li for providing tree models. This work is supported by National Natural Science Foundation of China with projects No. 60970093, 61072151, 60872120, and No. 60902078; and by the Strategic Cooperation between Project Guangdong Province and Chinese Academy of Sciences No. 2009B091300044 and No. 2010B090300030.

References

1. Floater, M.S.: Mean Value Coordinates. J. Computer Aided Geometric Design 20, 19–27 (2003)
2. Floater, M.S., Kos, G., Reimers, M.: Mean Value Coordinates in 3d. J. Computer Aided Geometric Design 22, 623–631 (2006)
3. Derose, T., Meyer, M.: Harmonic Coordinate. Tech. rep., Pixar Animation Studio (2006)
4. Lipman, Y., Levin, D., Cohen-Or, D.: Green Coordinates. In: SIGGRAPH 2008: ACM SIGGRAPH 2008 papers, pp. 1–10. ACM, New York (2008)
5. Sumner, R., Popvovic, J.: Deformation transfer for triangle meshes. In: Marks, J. (ed.) ACM SIGGRAPH 2004 Papers (SIGGRAPH 2004), pp. 399–405. ACM, New York (2004)
6. Li, H., Zhang, X., Zhang, Y.: Modeling trees with crown shape constraints. In: ACM SIGGRAPH ASIA 2010 Poster (SA 2010), article 38. ACM, New York (2010)
7. Ju, T., Schaefer, S., Warren, J.: Mean Value Coordinates for Closed Triangular Meshes. J. ACM Trans. Graph. 24(3), 561–566 (2005)
8. Habel, R., Kusternig, A., Wimmer, M.: Physically guided animation of tree. Computer Graphics Forum 28, 523–532 (2009)