ICIC Express Letters
Volume **11**, Number **3**, March **2017**

ICIC International ©2017 ISSN 1881-803X
pp. 705–710

A PHYSICALLY-BASED DEFORMATION METHOD OF SCANNED PLANT MODELS

Sheng Wu[1], Boxiang Xiao[2,3,4,5,*] and Xinyu Guo[2,3,4,5]

[1]School of Information Science and Technology
Beijing Forestry University
No. 35, Qinghua East Road, Haidian District, Beijing 100097, P. R. China

[2]Beijing Research Center for Information Technology in Agriculture
[3]Beijing Academy of Agriculture and Forestry Science
[4]National Engineering Research Center for Information Technology in Agriculture
[5]Beijing Key Lab of Digital Plant
No. 11, Shuguanghuayuan Mid Road, Haidian District, Beijing 100097, P. R. China
*Corresponding author: xiaobx@nercita.org.cn

Received August 2016; accepted November 2016

Abstract. *With the rapid and universal development of 3D laser scanner, point-sampled geometry is becoming ubiquitous in graphics and geometric information processing. Deformation of scanned point cloud models of plant is an attractive research issue both in botany and computer graphics, and it is important in virtual plant modeling and simulations. This paper systematically describes a physically-based deformation method for scanned point cloud models of plants. Firstly, we present an octree-based discretization method to divide and manage the scanned point cloud models. The constructed framework models are composed of nodes with mass and links with elastic properties. Secondly, we construct a constraint model for each extracted framework model by mass-spring system and fit all values of masses and springs according to the structure, density and skeleton of scanned models. Finally, a physically-based algorithm is used to get resolutions of mass-spring system models and to simulate the deformation process of scanned plant models. The experimental results show that our method is effective in virtual plant modeling and simulations based on scanned point cloud models. Furthermore, the main contributions and limitations are discussed in document.*

Keywords: Physically-based deformation, Scanned, Point cloud, Virtual plant, Simulation

1. **Introduction.** In recent decades, the three-dimensional laser scanning technologies as well as scanning devices have been rapidly and universally developed. They have been used in many fields and applications such as landscape design, industrial engineering, architecture, medical simulation, and virtual reality [1-6]. Deformation of scanned point cloud models of plant is an attractive research issue both in botany and computer graphics, and it is important in virtual plant modeling and simulations. Many researchers have carried out various methods to achieve highly precise and highly efficient modeling and simulation based on scanned point cloud models [7-10]. Especially for plant objects, Li and colleagues [11] proposed a new approach on sampling and surface reconstruction of large-scale point cloud data. Livny et al. [12] proposed an approach to automatic reconstruction of tree skeletal structures from point clouds. Wang et al. [13] presented a data-driven method for deformation capture and modeling of general soft objects. Su et al. [14] carried out a skeleton extraction algorithm for tree models based on point cloud contraction using constrained Laplacian smoothing. For deformations and simulations, Pirk et al. [15] presented a novel method (SPH) for combining developmental tree models with turbulent wind fields.

Although relevant works are fruitful and the previous researches have promoted the advances of point cloud based modeling, robust and effective deformation algorithms for scanned point cloud data are still challenging and have not been fully resolved by the state-of-the-art graphical techniques. To tackle the aforementioned problems, in this paper we explore a new physically-based deformation method of scanned plant models, where point cloud models are divided and control by mass-spring system framework. Furthermore, a physically-based resolution and simulation approach is performed to implement deformation process. The main flowchart is shown in Figure 1.

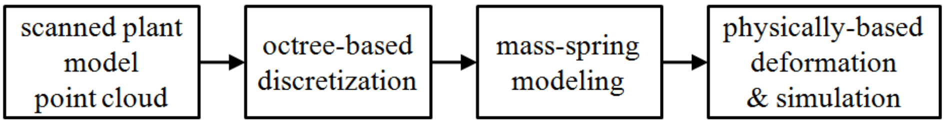

FIGURE 1. Main flowchart of this work

2. **Point Cloud Model Discretization.** In this work, we select trees as examples to introduce our approach. We acquire all our experimental data by use of a laser scanner named ‘FARO Focus3D’ in agricultural tree garden in Beijing. Data includes point cloud models of apple trees, pear trees and other fruit trees in their leaf fall period. To obtain a higher precision, each tree was scanned from four perpendicular directions, and the scanned data were integrated by common makers in scene. Figure 2 shows a typical point cloud tree model.

(a) Real scan scenario (b) A scanned tree model

FIGURE 2. Point cloud scanning

A main problem in meshless volumetric simulation is processing of unstructured point cloud data. Segmentation and discrete point cloud model is the first and crucial step for data processing. Octree-based method is effective for segmentation and discretization of point cloud data in three-dimensional space. In practice, we firstly calculate the largest boundary of the whole model, and construct the initial bounding cubic box which covers all points. And then, we compute divided sub layer cubic boxes for the model, and the bounding boxes are organized in an octree structure. The octree subdivides the volume of the object represented as point set surface into cubes, giving a non-overlapping discrete representation of the domain, where efficient numerical integration schemes could be used. The octants provide the basic unit to construct the patches and allow the efficient determination of patch interactions. Figure 3 shows a result of octree-based discretization of a tree’s point cloud model. Points in the model are divided into different cub boxes, and they are respectively controlled by their bonding boxes in deformation.

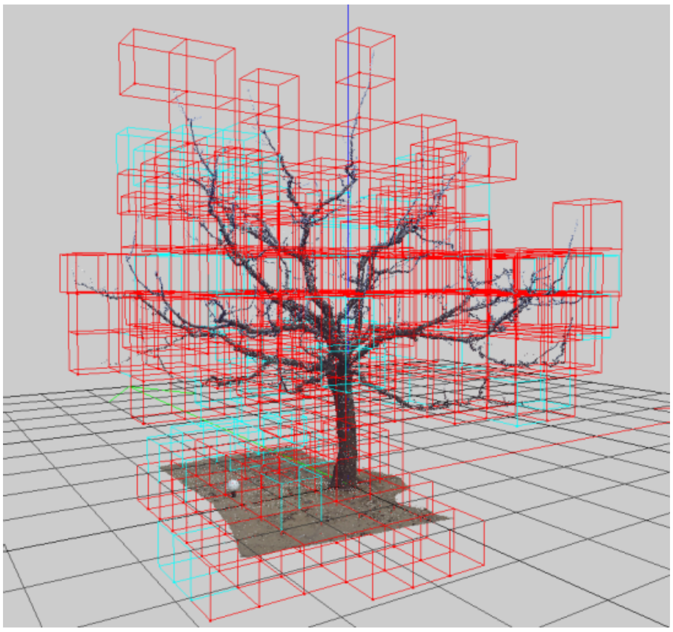

FIGURE 3. Point cloud models' segmentation and discretization

3. **Mass-Spring System Models.** Physically-based deformation is generally driven by physical models, and here we make use of the most popular mass spring system model to construct our physics dynamic system. To implement physics-based modeling, a general approach is to discretize a continuous dynamic surface into a collection of mass-points connected by a network of springs across the nearest neighbors (and/or along both diagonals) in the parametric domain. The users can also add other additional springs into the discretized surface if certain types of dynamic behavior are more desirable. The mass-spring system is widely used because of its simplicity and the critical need of real-time surface sculpting.

In motion, the velocity $\boldsymbol{v}$ and acceleration $\boldsymbol{a}$ of an N-linked mass are relevant to its position $\boldsymbol{P}$. Considering damping $\boldsymbol{k}_d$ and gravity $\boldsymbol{G}$, its acceleration $\boldsymbol{a}$ can be calculated by all linked spring parameters and external forces, including pull or push of spring $\boldsymbol{f}_{ki}$, gravity $\boldsymbol{f}_g$, damping $\boldsymbol{f}_d$, external force $\boldsymbol{f}_{ext}$ and mass value $\boldsymbol{m}$. Balance equation can be expressed as:

$$\sum_{i=1}^{N} \boldsymbol{f}_{ki} + \boldsymbol{f}_d + \boldsymbol{f}_g + \boldsymbol{f}_{ext} = \boldsymbol{ma} \tag{1}$$

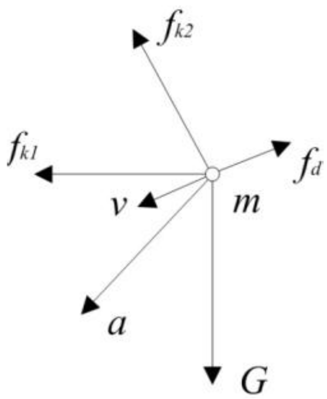

FIGURE 4. Balance of mass point under forces

For a certain mass in a physical continuous motion, both the velocity and the acceleration of $\boldsymbol{P}$ can be discretized along the time axis analogously:

$$\ddot{\boldsymbol{P}} \approx \left(\boldsymbol{P}^{t+\Delta t} + \boldsymbol{P}^{t-\Delta t} - 2\boldsymbol{P}^{t}\right)/\Delta t^2 \tag{2}$$

$$\dot{\boldsymbol{P}} \approx \left(\boldsymbol{P}^{t+\Delta t} - \boldsymbol{P}^{t-\Delta t}\right)/2\Delta t \tag{3}$$

where t is the deformable process time coordinate in time domain, and Δt is the discretized time pace. It is also equivalent to motion capture data.

Here the mass-spring model is created by the framework model of discretization. The framework model is constructed by divided bounding boxes, where the vertices of boxes are nodes of framework model which are considered as mass points of mass-spring system, and edges of bounding boxes are links of framework model which are considered as springs of mass-spring system. All values of mass are determined by density of point cloud data as well as the topological structure of objects which could be considered as the distribution of real mass. Values of springs are determined in an empirical and experimental way to obtain visual-correct simulation.

4. **Physically-Based Deformation.** Physically-based methods are widely developed and used in dynamic simulations for many applications, and they are able to simulate the natural process under suitable parameters control and constraints [7]. Most of plant objects blend rigid and flexible parts with both two kinds of deformations. In an elastic deformable process, the model in elastic deformable process can be expressed by the position, velocity and acceleration which are generally determined by the material properties such as mass, damping, and stiffness distributions. By Lagrangian mechanics, a general physics-based model can be described as:

$$\boldsymbol{M}\ddot{\boldsymbol{P}} + \boldsymbol{D}\dot{\boldsymbol{P}} + \boldsymbol{K}\boldsymbol{P} = \boldsymbol{f} \tag{4}$$

where $\boldsymbol{P}$ is the set of vertexes, $\dot{\boldsymbol{P}} = \dfrac{\partial \boldsymbol{P}}{\partial t}$ expresses the velocity, $\ddot{\boldsymbol{P}} = \dfrac{\partial \boldsymbol{P}}{\partial t^2}$ is the acceleration, $\boldsymbol{M}$ is mass matrix, $\boldsymbol{D}$ is the damping, $\boldsymbol{K}$ is the stiffness of model, and $\boldsymbol{f}$ is the external forces in total.

In a general physics-based model, the values of material and mechanical properties are defined over the model surface as functions respectively, which oftentimes can be simply considered to be constant at certain time.

By means of the constructed mass-spring models, we perform the deformation process by physically based functions. The frameworks models are deformed firstly, and then all points of scanned models recalculate the coordinates values according to the values of vertices of their bonding box. Here we adopt the volume coordinate algorithm to compute the new positions of points in scanned models [16], and the corresponding position of a point in a hexahedron after deformation is illustrated in Figure 5. The basic volume coordinate definition can be defined as:

$$L_i = \frac{V_i}{V}, \quad i = 1, 2, 3, 4 \tag{5}$$

$$\sum_{i=1}^{4} L_i = 1 \tag{6}$$

where V_i is the volume of the subdivided tetrahedron composed of given point and relevant 3 points on the ith face of subdivided tetrahedron, and V is the volume of the whole tetrahedron. The change of nodes of mass-spring mode is conducted by physical constraints, and then, by use of deformed cubic bounding boxes we recalculate all positions of points in hexahedrons according to their relevant bounding boxes.

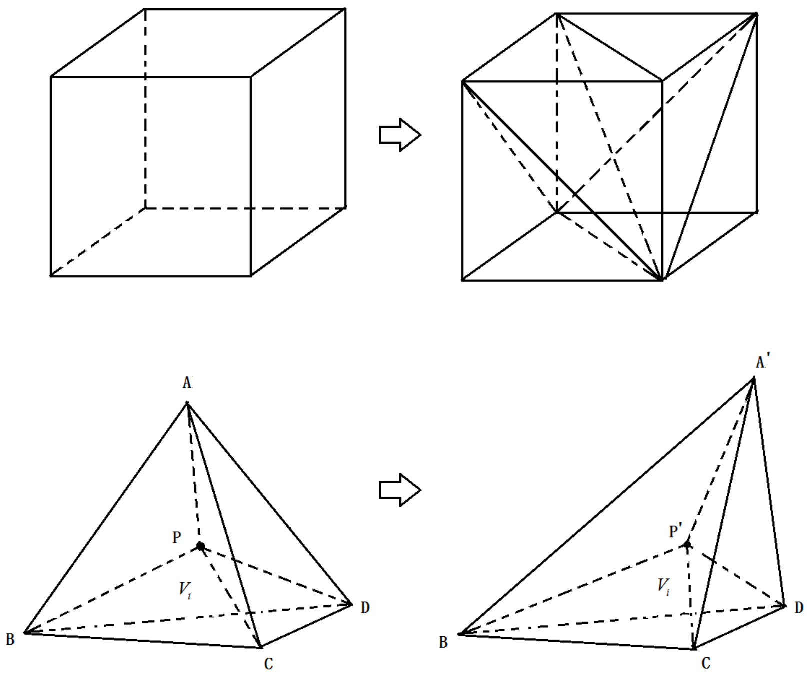

Figure 5. Volume coordinate definition

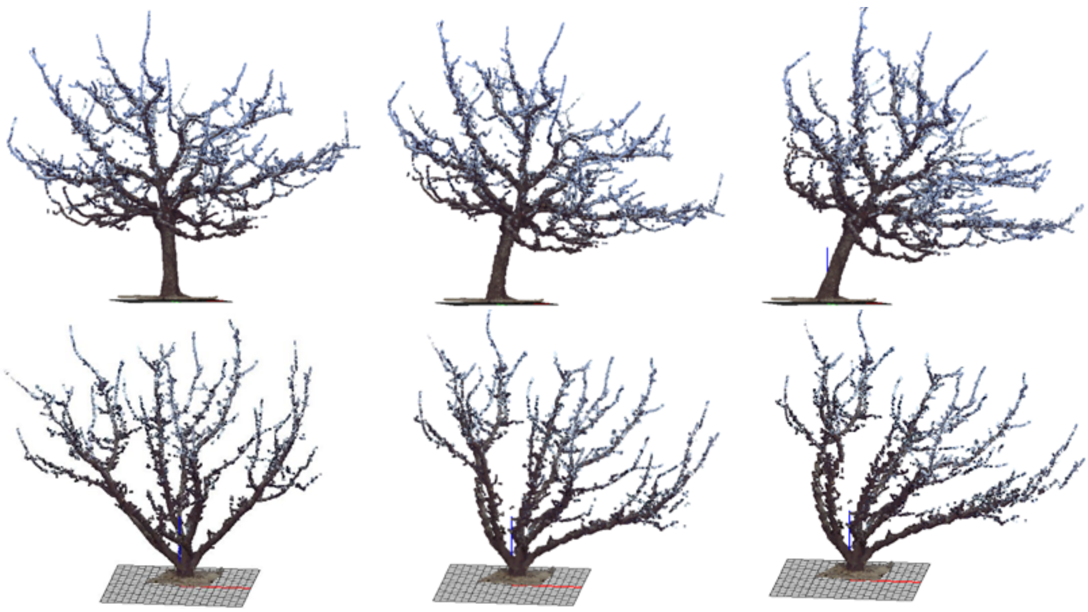

Figure 6. Result of deformation of apple tree (up) and pear tree model (down)

5. **Results.** To implement the proposed deformation and simulation method, we developed a prototype system by C++ program language and OpenGL graphics library [17]. The number of points of scanned tree models is respectively about 500000. All experiments were performed on a personal computer with 2.8GHz CPU and 16G Memory. Part of deformation results are shown in Figure 6. The experimental results showed that our physically based deformation method was effective and provided a feasible way to scanned point cloud plant models in a simply implementation.

6. **Conclusions.** To sum up, we propose a new physically-based deformation method of scanned plant models in this paper. We employ the octree-based segmentation algorithm to discretize point cloud models of plant, and then a mass-spring-based physical modeling

method is involved to construct the basic physical constraints in deformation process. To get a solution of dynamic model, we make use of a physically based dynamics algorithm for implementation of all deformation processes and simulations. The results show that our physically-based deformation method is effective and provide a feasible way to scanned point cloud plant models in a simply implementation. The main contribution of our method is that we integrate mass-spring system and physically-based solving algorithm to the scanned point cloud models of plant objects which make the deformation and reuse of scanned data in a meshless pattern based on physical constraints available. On the other hand, the main limitation of our method is that the segmentation and deformation process is lack of support of the semantic information of different part models, which could enhance reasonable and visible results of deformation in a more natural level. These problems are our further foci in future works.

Acknowledgment. This work is partially supported by the National Natural Science Foundation of China (Grant No. 61300079), by Beijing Municipal Natural Science Foundation (Grant No. 4162028). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] Y. Wang, C. Fan, D. Jiang et al., Comparison on crown characteristics and fruit quality of different tree canopy shapes, *Acta Agriculturae Boreali-Occidentalis Sinica*, vol.20, no.12, pp.93-97, 2011.

[2] T. Dornbusch, S. Lorrain, D. Kuznetsov and C. Fankhauser, Measuring the diurnal pattern of leaf hyponasty and growth in arabidopsis – A novel phenotyping approach using laser scanning, *Functional Plant Biology*, vol.39, no.11, pp.860-869, 2012.

[3] H. Xu, N. Gossett and B. Chen, Knowledge and heuristic-based modeling of laser-scanned trees, *ACM Trans. Graphics*, vol.26, no.4, pp.377-388, 2007.

[4] Q. Liu, M. Deng, Y. Shi and J. Wang, A density-based spatial clustering algorithm considering both spatial proximity and attribute similarity, *Computers & Geosciences*, vol.46, no.3, pp.296-309, 2012.

[5] Y. Akagi and K. Kitajima, Computer animation of swaying trees based on physical simulation, *Computers & Graphics*, vol.30, no.4, pp.529-539, 2006.

[6] R. Habel, A. Kusternig and M. Wimmer, Physically guided animation of trees, *Computer Graphics Forum*, vol.28, no.2, pp.523-532, 2009.

[7] H. Du and H. Qin, Dynamic PDE-based surface design using geometric and physical constraints, *Graphical Models*, vol.67, no.1, pp.43-71, 2005.

[8] M. Berger, A. Tagliasacchi, L. Seversky, P. Alliez, J. Levine et al., State of the art in surface reconstruction from point clouds, *Proc. of Eurographics 2014*, Strasbourg, France, vol.1, no.1, pp.161-185, 2014.

[9] T. Sugiura, A. Torii and M. Okutomi, 3D surface reconstruction from point-and-line cloud, *Proc. of the 2015 IEEE International Conference on 3D Vision*, pp.264-272, 2015.

[10] X. Ning, Y. Wang and X. Zhang, Hierarchical model generation for architecture reconstruction using laser-scanned point clouds, *Optical Engineering*, vol.53, no.6, pp.1-15, 2014.

[11] E. Li, X. Zhang and Y. Chen, Sampling and surface reconstruction of large scale point cloud, *Proc. of the 13th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, pp.35-41, 2014.

[12] Y. Livny, F. Yan, M. Olson, B. Chen, H. Zhang and J. El-Sana, Automatic reconstruction of tree skeletal structures from point clouds, *ACM Trans. Graphics*, vol.29, no.6, p.151, 2010.

[13] B. Wang, L. Wu, K. Yin, U. Ascher, L. Liu and H. Huang, Deformation capture and modeling of soft objects, *ACM Trans. Graphics*, vol.34, no.4, pp.1-12, 2015.

[14] Z. Su, Y. Zhao, C. Zhao et al., Skeleton extraction for tree models, *Mathematical and Computer Modelling*, vol.54, pp.1115-1120, 2011.

[15] S. Pirk, T. Niese, T. Hadrich, B. Benes and O. Deussen, Windy trees: Computing stress response for developmental tree models, *ACM Trans. Graphics*, vol.33, no.6, p.204, 2014.

[16] H. Li, S. Cen, Y. Long and Z. Cen, Volume coordinate method for hexahedral elements, *Engineering Mechanics*, vol.25, no.10, pp.12-18, 2008.

[17] D. Shreiner, M. Woo, J. Neider and T. Davis, *OpenGL Programming Guide: The Official Guide to Learning OpenGL*, Addison-Wesly, 2004.