

A MODELING METHOD OF FLUTTERING LEAVES BASED ON POINT CLOUD

Jing Tang^a, Yinghui Wang^{a,*}, Yanni Zhao^{a,b}, Wen Hao^a, Xiaojuan Ning^a, Ke Lv^c, Zhenghao Shi^a, Minghua Zhao^a

^a Institute of Science and Engineering, Xi'an University of Technology, Xi'an, 710048, China - tj890724@126.com, wyh_925@163.com, haowensxf@163.com, fly-snow2001@163.com, ylshi@xaut.edu.cn, zhaominghua@xaut.edu.cn

^b Department of Computer Science, Shaanxi Vocational & Technical College, Xi'an, 710100, China - feifei6513@163.com

^c University of Chinese Academy of Sciences, 100049, China - luk@ucas.ac.cn

Commission IV, WG IV/3

KEY WORDS: Point Cloud, Leaves, Falling, Trajectory, Geometric Transformation, OpenMP

ABSTRACT:

Leaves falling gently or fluttering are common phenomenon in nature scenes. The authenticity of leaves falling plays an important part in the dynamic modeling of natural scenes. The leaves falling model has a widely applications in the field of animation and virtual reality. We propose a novel modeling method of fluttering leaves based on point cloud in this paper. According to the shape, the weight of leaves and the wind speed, three basic trajectories of leaves falling are defined, which are the rotation falling, the roll falling and the screw roll falling. At the same time, a parallel algorithm based on OpenMP is implemented to satisfy the needs of real-time in practical applications. Experimental results demonstrate that the proposed method is amenable to the incorporation of a variety of desirable effects.

1. INTRODUCTION

Leaves falling gently or fluttering are common phenomenon in nature scenes. The complexity of the air flow and the collision produced in leaves falling makes the simulation process of leaves fluttering more difficult. Hence the accurate and realistic reconstruction of the leaves fluttering has been a great challenging topic in computer graphics for many years. The result has directly influence on the quality of tree modeling in virtual scene.

At present, most researches of leaves fluttering modeling are focused on simulation rather than actual measured. We present a novel method for leaves fluttering modeling. First, the leaves are measured in natural scene and can be variable in size, shape and other aspects. It lays a physical foundation for the real simulation of falling leaves. Second, according to the shape of the leaves, the weight of leaves and the speed of the wind, three kinds of basic trajectories of leaves falling are defined. Then we select an appropriate trajectory to accomplish falling leaves simulation in different wind field. It makes the leaves falling model more realistic.

2. RELATED WORK

After years of researches, leaves fluttering modeling have obtained fruitful results (Zhong, 2011; Auguste, 2013). The early researches simply applied the fluid dynamics to the leaves fluttering. Wejchert et al. (Wejchert, 1991) proposed a method based on aerodynamics to simulate and control the motion of objects in fluid flows, which can also be applied to an animation that involves hundreds of flexible leaves. However, the falling leaves behaviors and trajectories have not been categorized. Tanabe and Kaneko (Tanabe and Kaneko, 1994)

estimated the force and torque using Kutta-Joukowski's theorem to calculate the force acting on leaves and obtained a fluttering and tumbling motion. However, their method is restricted to two spatial dimensions and it is not able to model spiraling.

After that, Desbenoit et al. (Desbenoit, 2006) presented a system for modeling autumn leaves covering vegetation and monuments. They rendered leaves and simulated the aging process, but the behaviors of falling leaves are simulated by using certain patterns. So the leaves falling path is not built based on physical simulation. Vázquez et al. (Vázquez, 2008) put forward a graphics hardware rendering method which models the process of thousands of leaves falling in real time, and they also presented a method for path reuse in order to make longer falling trajectories by reusing the same information data. However, it emphasizes the simulation of thousands of leaves in real time, and lacks of details of falling leaves. In the paper of (Li, 2010), leaves falling motion was divided into two factors such as movement trajectory and leaves posture on each point of the trajectory. Based on it, an algorithm which can be used to produce stylized and expressive falling leaves animation is presented and implemented. Although it has achieved some effects, their leaves are still simulated and somewhat is different from the real leaves. Li et al. (Li, 2015a) proposed a rapid approach for leaf falling dynamic simulation based on the plant physiology principle of leaf abscission. Considering the leaves age and wind force, the specific falling leaves are identified. Then leaves falling motion in the air can be simulated by using synthetic path method based on the current state of falling leaves. However it does not consider the shape of the leaves falling process such as rotation, flip and so on. Quigley et al. (Quigley, 2017) proposed a three-dimensional spiralling motion by introducing the azimuthal offset angle for the torque direction. The leaves will fall from the branches when the torque on the joint exceeds the threshold. While the leaf they

* Corresponding author

used is represented by a thin triangular prism and it is not matched by the actual leaf, which has different thicknesses and shapes in various angles of view.

Alternatively, there has a data-driven approach as in (Martin, 2015) or even randomly sample prescribed spline curves as in (Li, 2015b).

It is clearly to see that the research of leaves falling simulation has made great progress already, but there are also some aspects that need further improvement. Such as using the real natural leaves model, taking into account different geometric features of leaves, classifying leaves falling trajectories by considering various factors and so on. For the reasons outlined above, we propose a method of leaves fluttering based on point cloud in this paper.

The significant contributions of this paper are:

- 1) Our method accomplishes the simulation of leaves falling based on the measured leaves model, rather than the simulation. On the basis of it, our leaves can be variable in size, shape and other aspects, and therefore it is more real for growing on a tree.
- 2) The reconstructed falling leaves have three kinds of leaves falling trajectory which can better describe the dynamic characteristics of leaves. The models of leaves motion in rotation falling, roll falling and screw roll falling are all rebuild and that makes the reality sense of the leaves falling stronger.
- 3) The parallel algorithm is implemented by using OpenMP in order to provide means for real time.

In the next section, we describe the tree model used in our paper. In Section 4, we introduce the trajectories for falling leaves. The analysis of the experimental results is described in Section 5, and the conclusion is presented in Section 6.

3. LEAF AND TREE MODEL

The process of tree modeling is composed of three steps, including the modeling of trunk and branch, the modeling of leaves and the growing of leaves on the branch.

3.1 Modeling of trunk and branch

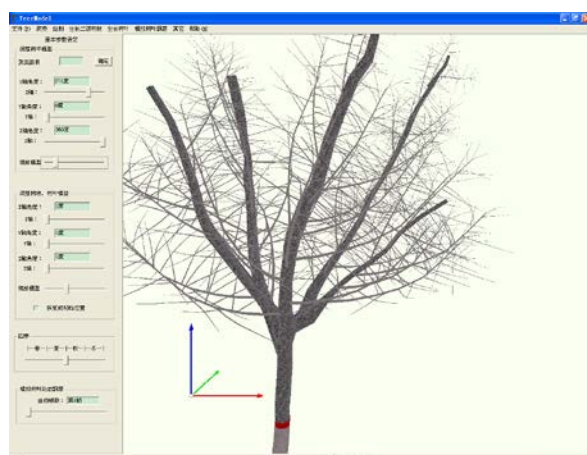


Figure 1. Final trunk model

The method of modeling trunk and branch is obtained from (Wang, 2013a). First, the optimal segmentation of the point cloud is carried out. Second, we extract the skeleton points of the tree and carry on the connection of the tree pole skeleton lines. In the end, the reconstruction of tree model is accomplished, and as shown in figure 1.

3.2 Modeling of leaves

The leaves modeling are also realized by (Wang, 2013b). The veins and margins are firstly extracted by using image processing from the denoised and smoothed leaf point clouds. At the same time, the leaf point clouds are simplified except the margins and veins. The basic structure and shape of leaf has been obtained at this time. Then the texture mapping method is implemented to the simplified point clouds on the basis of triangulation and produced a normal leaf model. Finally, the leaves with different shapes are generated by using Laplacian mesh deformation algorithm and grow to the trunk model. The experimental results are shown in figure 2, and it can be seen that every leaf has different shape with each other and has different thicknesses in various perspectives, thus increasing the visual sense of reality.

Type	Perspecti ve 1	Perspecti ve 2	Perspecti ve 3	Perspecti ve 4
Original plane				
Deforma tion effect 1				
Deforma tion effect 2				
Deforma tion effect 3				

Figure 2. Figure placement and numbering

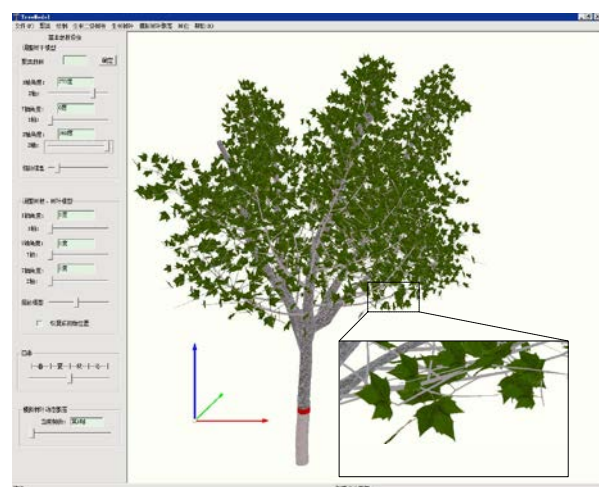


Figure 3. Final tree model

3.3 Growing of leaves on the branch

As in the (Gang, 2014), leaves will grow on the branch by translation transformation and rotation transformation. The generated leaves models are firstly translated to the specified locations, and their sizes are changed by using ratio transformation. The angles between branches and leaves can be adjusted by rotation transformation of coordinate axis to obtain the leaves with different forms. As shown in figure 3.

4. ALGORITHM OF FALLING LEAVES

4.1 The leaves falling trajectory

The leaves falling is a complex simulation process, which is influenced by various factors, such as the shape of leaves, the gravity of leaves, the wind speed and direction, and so on. It is clearly to see that the simulation of leaves falling is so complex and it is difficult to calculate the motion path for every leaf. Therefore, we simplify the calculation of leaves falling and define three trajectories to express the basic process of leaves falling without the analysis of forces. The complex and natural falling leaves process is formed by the mixing of three basic trajectories.

As shown in Figure 4, the falling trajectories are defined according to the shape and the weight of leaves and the wind speed. In the breeze, the wind force is smaller than the weight of leaves, and the gravity becomes a dominant in the process of leaves falling. Then the leaves will rotate or roll down to the ground in a straight line, as shown in figure 4(a) and 4(b). As the wind speed becomes larger, the wind force becomes a dominant force in the trajectory of leaves falling. The leaves, driven by the swirling flow of air, will screw down to the ground as shown in figure 4(c).

The three trajectories of leaves falling is modeled by the translation transformation and the rotation transformation of coordinate axis. The rotation angles of leaves around X, Y, Z axes are $angle_x$, $angle_y$, $angle_z$. The scaling factors of leaves along X, Y, Z axes are $factor_x$, $factor_y$, $factor_z$. The angular increments of rotation around the X, Y and Z axis are $angle_x_offset$, $angle_y_offset$ and $angle_z_offset$. Our algorithm assumes that the ground plane is parallel to the XOY plane. The sampling times of leaves falling from the branches to the ground are equal to pre-set value. The offset of leaves position between the current time and the last time is calculated according to the sampling times.

$$\begin{cases} posi_offset_x = 0.0 \\ posi_offset_y = 0.0 \\ posi_offset_z = ground_point.z / sampling_times \end{cases} \quad (1)$$

where $posi_offset_x$, $posi_offset_y$, $posi_offset_z$ = the offset of the leaves position between the current time and the last time

$ground_point.z$ = the z axis value of any point on the ground plane

$sampling_times$ = the given sampling times

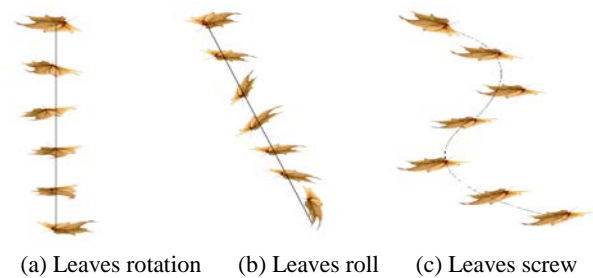


Figure 4. Leaves falling trajectory

4.1.1 The leaves rotation falling with straight line: In the breeze, the gravity of leaves is bigger than the wind force, and the trajectory of leaves falling is determined by the gravity. It makes the leaves fallen slowly in a rotation way with an approximate straight lines, as shown in Figure 4(a). The detailed steps are as follows:

- 1) The rotation angles, the scaling factors and the angular increments of rotation are first initialized. In our experiment, values are respectively set as (0, 0, 0), (0.5, 0.5, 0.5) and (0.00, 0.01, 0.50).
- 2) The translation vector of leaves produced by the rotation transformation and the ratio transformation is calculated by the rotation angles and the scaling factors.

$$\begin{cases} trans_vect.x = point'.x - posi_offset_x - point.x \\ trans_vect.y = point'.y - posi_offset_y - point.y \\ trans_vect.z = point'.z - posi_offset_z - point.z \end{cases} \quad (2)$$

where $trans_vect.x$, $trans_vect.y$, $trans_vect.z$ = the translation vector

$point.x$, $point.y$, $point.z$ = any point of leaf model

$point'.x$, $point'.y$, $point'.z$ = the translated point of leaf model

$posi_offset_x$, $posi_offset_y$, $posi_offset_z$ = the offset of the leaves position between the current time and the last time

- 3) Then the current position of leaf can be calculated by the corresponding translation vector. Update the values of the rotation angles with the angular increments. At the same time, add the values of leaves position offset to the next offset position calculated by the equation (1), and thereby obtaining the new offset of leaves position.
- 4) Return to the step 2) until the loop count is smaller than the given sampling times.

The experimental results are shown in Figure 5. Each leaf is a state in the process of a leaf falling and then combining all of them to create a leaf falling trajectory.

4.1.2 The leaves roll falling with straight line: In the breeze, as the same with the rotation falling, the gravity is bigger than the wind and hence to determine the leaves fall slowly as roll way in approximate straight line trajectory, shown in Figure 4(b). Experimental results are shown in Figure 6.

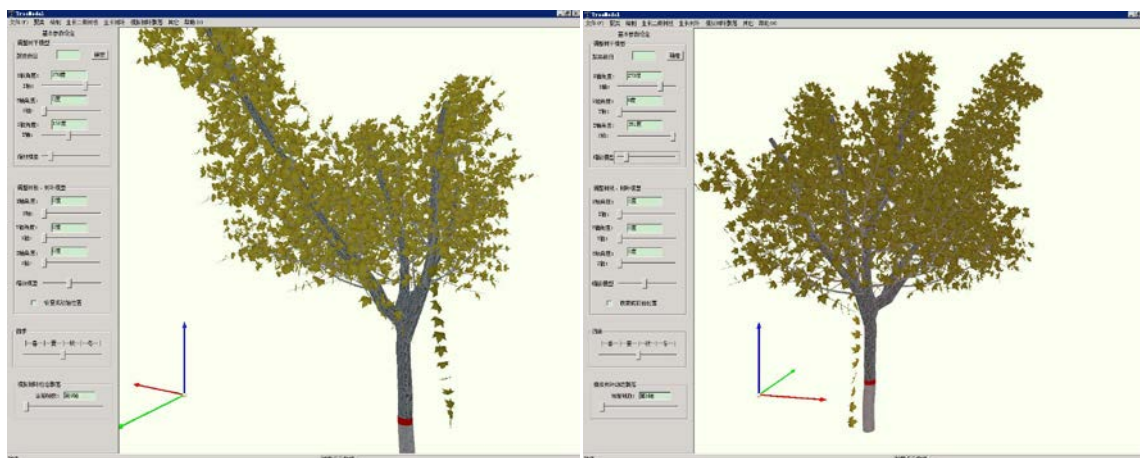


Figure 5. Two angles of the rotation falling with straight line

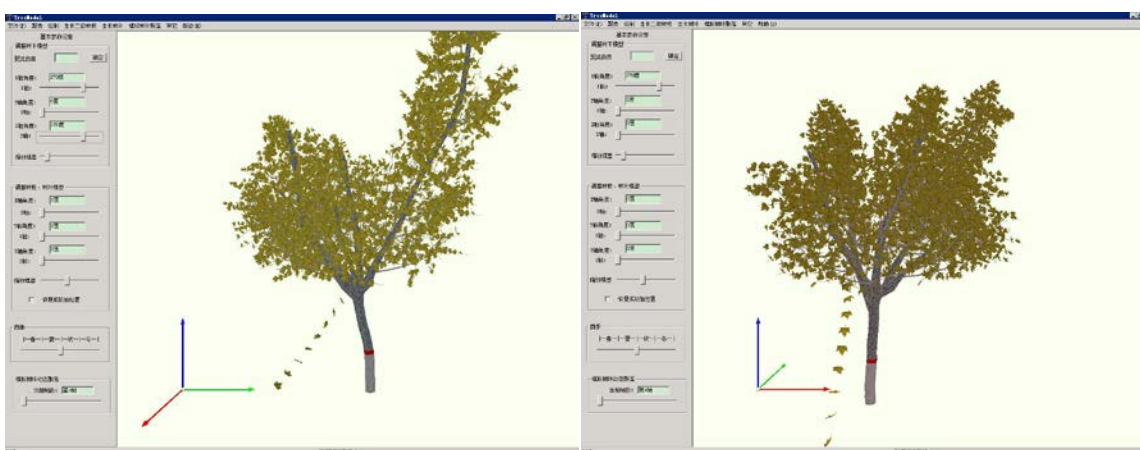


Figure 6. Two angles of roll falling with straight line

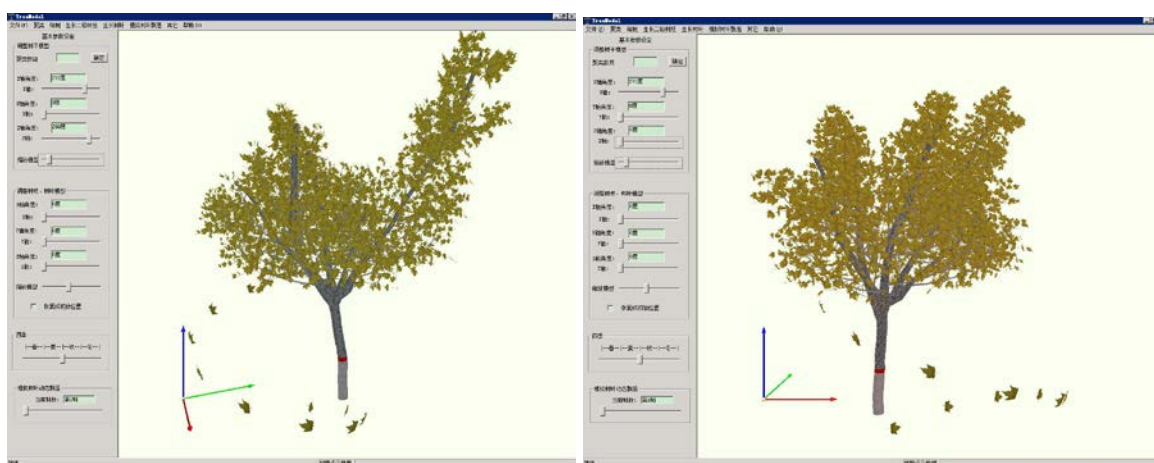


Figure 7. Two angles of screw roll falling

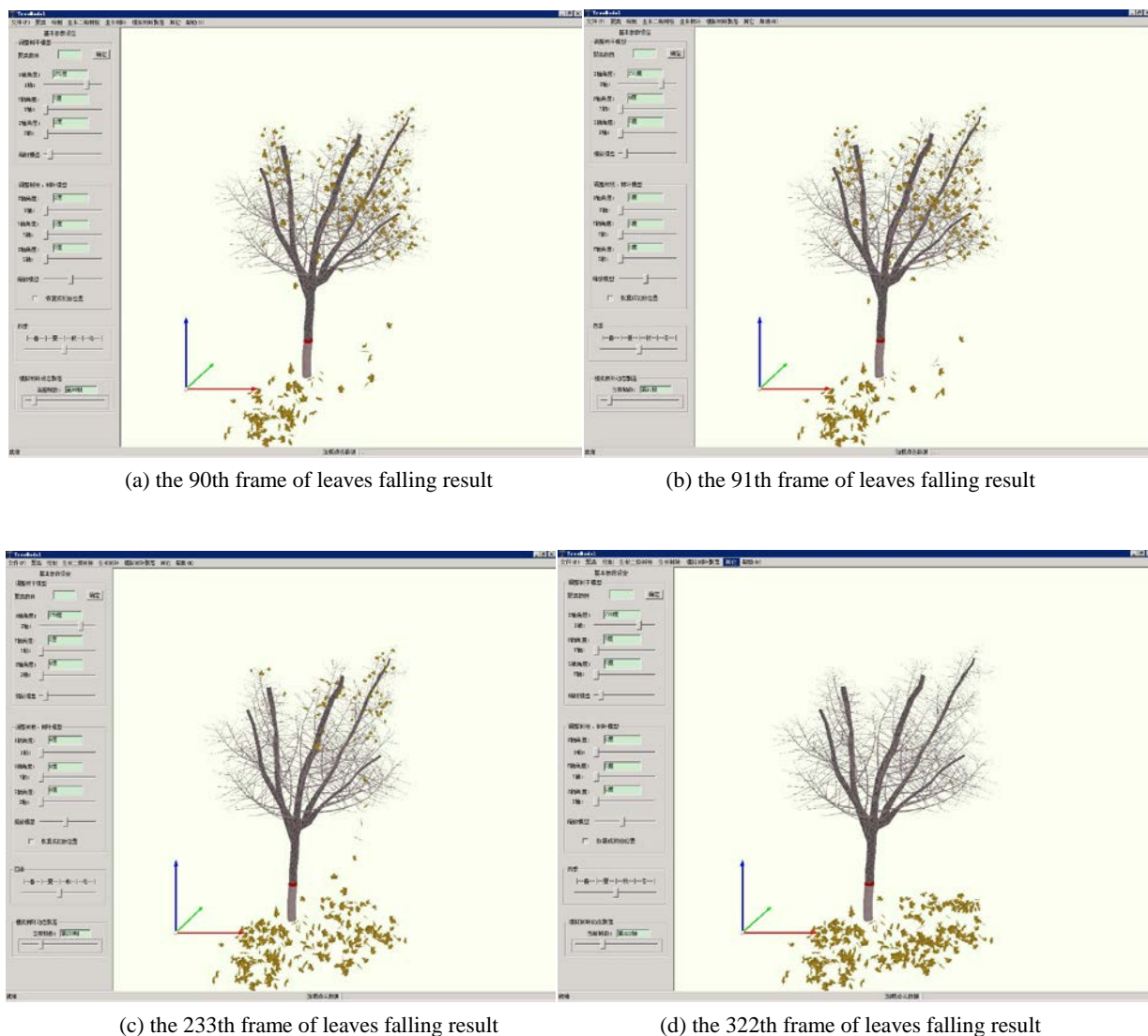


Figure 8. The leaves continued falling results

4.1.3 The leaves screw roll falling: When the wind speed becomes stronger, the wind force is bigger than the gravity of leaves and hence to dominant the trajectory of leaves falling. In that situation, the leaves will fall slowly in a screw roll way, as shown in Figure 4(c). The detailed steps are as follows:

- 1) Initialize the rotation angles, the scaling factors and the angular increments of rotation as (0, 0, 0), (0.5, 0.5, 0.5) and (0.25, 0.01, 0.50) respectively.
- 2) Calculate and store the translation vector of leaves, which is produced by the rotation transformation and the ratio transformation, combining with the equation (2).
- 3) Calculate the current position of leaf by the corresponding translation vector. Update the values of the rotation angles with the angular increments, and update the values of leaves position offset by adding to the equation (1).
- 4) Return to the step 2) until the loop count is smaller than the given sampling times.

The experimental results are shown in Figure 7.

4.2 Random process of falling leaves

After describing three trajectories of leaves falling, taking the rotation falling with straight line as an example, leaves random falling is modeled in this section.

The main idea is to make the leaves falling in a certain order. That is to say, at the time of t_0 , only the leaf which is numbered one begin to fall and reach to the position of p_1^1 (the superscripts presents the position number of leaf falling, and the subscripts presents the leaf which is numbered one); at the time of t_1 , leaves numbered one and two are in falling state, and the leaf numbered one is falling to the position of p_1^2 , the leaf numbered two is falling to the position of p_2^1 ; at the time of t_2 , leaves numbered one, two and three are all in falling state, and the leaf numbered one is falling to the position of p_1^3 , the leaf numbered two is falling to the position of p_2^2 , the leaf numbered three is falling to the position of p_3^1 ; and so on until all the leaves falling down to the ground. The detailed steps of the method are as follows:

- 1) Initialization. The number of leaves in the current tree model is N , and the number of leaves falling at the same time is a pre-set value and set as $limit_falling_num$. The sampling times of each leaf falling from the branches to the ground is a pre-set value. The value of min_index , which is minimum of leaves falling in the interval at present moment, is set as zero. The maximum of leaves falling max_index is set as $min_index + limit_falling_num - 1$.
- 2) If the value of max_index is smaller than N , which is the number of leaves, it means that there has some leaves that don't fall down to the ground, and then go to step 3); otherwise stop this algorithm.
- 3) Calculate the position of leaves falling to by the algorithm in Section 5.1.1.
- 4) Return to the step 2) until all leaves have fallen to the ground.

In our experiment, we select $N = 68$, $limit_falling_num = 12$, and $sampling_times = 12$ in the testing process. The results of leaves continued falling are shown as figure 8, (a) is the result of the 90th frame, (b) is the result of the 91th frame, (c) is the result of the 233th frame and (d) is the result of the 322th frame. The experimental results show that the algorithm in this paper can simulate the falling leaves more naturally and accurately in the breeze.

From the experiment we can see that the algorithm of steps 4) in section 5.2 has the large amount of calculation. In addition, because multi-core processor has become main stream today, so in this paper, the OpenMP is used in steps 4) of section 5.2 to accomplish the parallel computing, and tests were performed in dual core, and quad core CPU computers with 3.7GHZ and 4G memory. As show in table 9, the running time decreases with the increase of the processor. Experimental results show that the algorithm can meet the needs of practical application in time and provide means for real time of leaves falling.

Number of processors	Time consumed (s)
1	2.031
2	1.094
4	0.468

Table 9. Amount of time spent

5. THE ANALYSIS OF EXPERIMENTAL RESULTS

We simulate leaves fluttering from raw 3D point clouds. Presently, there is no unified standard for evaluating the authenticity of leaves fluttering simulation, and the complexity of leaves structures further increases the difficulty of direct evaluation.

Our method of leaves fluttering based on point cloud is based on the actual measured leaves. Then the leaves used in our paper have different thicknesses and shapes with the various angles of view. While others methods more focus on the simulation and their leaves usually represented as simply geometry structures, such as in (Quigley, 2017). In that paper, the leaf is approximated to a thin triangular prism leading to the leaf has a same thickness in various angles. Figure 10 shows the result of leaves falling in (Quigley, 2017). In contrast to our method, as shown in Figure 3, the leaves have more folds and

complex thickness. It makes the leaves more reality.



Figure 10. The leaves falling results of (Quigley, 2017)

In addition, the trajectories of leaves fluttering defined in our method are basic processes, but the mixture model of them can adequately express the complex process of leaves falling in natural scene. The leaves falling trajectories of others methods, such as in (Li, 2015b), can be split into our three basic trajectories. As shown is Figure 11, Li et al. (Li, 2015b) divided leaf falling motions into 6 classes, including steady descent, periodic tumbling, transitional chaotic, periodic fluttering, transitional helical, and periodic spiral. While these motions are all composed of rotation, roll and screw roll, as shown in Figure 4. Thus, our trajectories can not only describe the dynamic characteristics of leaves, but also provide the simulation reference for the further development of falling leaves.

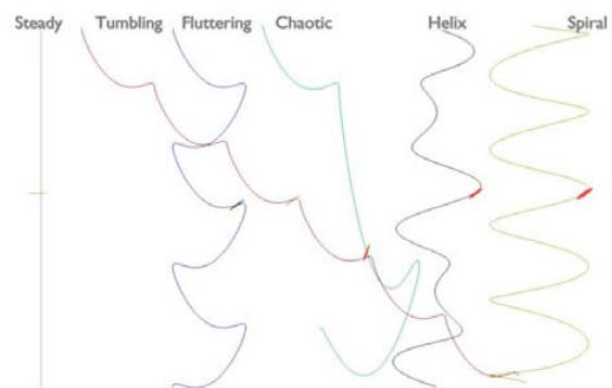


Figure 11. The Primitive motions of falling leaves of (Li, 2015b)

6. CONCLUSIONS

A method of leaves fluttering based on point cloud is proposed in this paper. Firstly, three kinds of trajectories of leaves falling, which are rotation falling, roll falling and screw roll falling, are defined according to the shape of leaves, the weight of leaves and the speed of wind. And then we model the leaves falling process by selecting different trajectories according to different wind speed. At the same time, the parallel algorithm is implemented by using OpenMP in order to provide means for real time. Experimental results demonstrate that the proposed method can model the leaves falling quickly and more natural.

However, our method just provides the basic process of falling leaves, and lacks of the physical details analysis in leaves falling, such as the vibration caused by the wind, the complex physical analysis for every leaf. In the future, we will consider more details and then producing more natural simulation effects.

ACKNOWLEDGEMENTS

This study is supported by the Nature Science Foundation of China Grant No.61472319, 61602373. and in part by the Nature Science Foundation of Shaanxi Province under Grant No.2015JZ015.

REFERENCES

- Auguste F, Magnaudet J, Fabre D, 2013. Falling styles of disk. *Journal of Fluid Mechanics*, 719(1), pp. 388-405.
- Desbenoit B, Galin E, Akkouché S, et al, 2006. Modeling Autumn Sceneries. *Eurographics*, pp. 107-110.
- Gang Wang, 2014. Photorealistic rendering leaves based on point cloud. Xi'an University of technology.
- Li H, Sun Q, Zhang H, et al, 2010. Example-based motion generation of falling leaf. *International Conference on Computer Design and Applications, IEEE*, 5(5), pp. 217-221.
- Li Z, Lin D, Song H, et al, 2015a. **A fast leaf simulation method faithful to plant physiology.** *microcomputer and application*, (13), pp. 77-79.
- Li C, Qian J, Tong R, et al, 2015b. GPU based real-time simulation of massive falling leaves. *Computational Visual Media*, 1(4), pp. 1-8.
- Martin T, Umetani N, Bickel B, 2015. Omniad: Data-driven omni-directional aerodynamics. *ACM Transactions on Graphics*, 34(4), pp. 113:1–113:12.
- Quigley E, Yu Y, Huang J, et al, 2017. Real-time Interactive Tree Animation. *IEEE Transactions on Visualization and Computer Graphics*, PP(99), pp. 1-1.
- Tanabe Y, Kaneko K, 1994. Behavior of a falling paper. *Physical Review Letters*, 73(10), pp. 1372-1375.
- Vázquez Alcocer P P, Balsa M, 2008. Rendering falling leaves on graphics hardware. *Journal of virtual reality and broadcasting*, 5(2), pp. 1-14.
- Wang Y, Chang X, Ning X, et al, 2013a. Tree Modeling Methods from 3D Point Cloud: A Survey. *Information Japan*, 16(6), pp. 3675-3694.
- Wang Y, Hao W, Wang G, et al, 2013b. A method of realistic leaves modeling based on point cloud. *ACM SIGGRAPH International Conference on Virtual-Reality Continuum and ITS Applications in Industry*, pp. 123-130.
- Wejchert J, Haumann D, 1991. Animation aerodynamics. *ACM SIGGRAPH Computer Graphics*, 25(4), pp. 19-22.
- Zhong H, Chen S, Lee C, 2011. Experimental study of freely falling thin disks: Transition from planar zigzag to spiral. *Physics of Fluids*, 23(1), pp. 011702.