

## Article

# Visualization Simulation of Branch Fractures Based on Internal Structure Reconstruction

Meng Yang<sup>1,2,\*</sup>, Yi Zhang<sup>1</sup> and Benye Xi<sup>3</sup> 

- <sup>1</sup> School of Information Science and Technology, Beijing Forestry University (BFU), Qinghua East Road 35, Haidian District, Beijing 100083, China
- <sup>2</sup> Engineering Research Center for Forestry-Oriented Intelligent Information Processing of National Forestry and Grassland Administration, Beijing 100083, China
- <sup>3</sup> The College of Forestry, Beijing Forestry University (BFU), Qinghua East Road 35, Haidian District, Beijing 100083, China
- \* Correspondence: yangmeng@bjfu.edu.cn

**Abstract:** This paper presents a visualization algorithm for wood fracture simulation based on wood science and wood internal structure reconstruction. The algorithm can simulate a reasonable and realistic wood fracture effect. First, the 3D point-cloud data of the bark structure are obtained using a laser scanner, and the cross-section of the branch is obtained by voxelization of the surface mesh model. Then, the outer contour of the cross-section is shrunk inward to reconstruct the annual rings and wood fiber bundles, and reasonable internal structures of branch 3D models are generated. The internal structure consists of a hierarchical model composed of several ring-like annual rings, and each annual ring is divided into a series of continuous fan rings. On the basis of the reconstruction results, the wood fracture surface model generated by the parameter control can be mapped to the irregularly shaped 3D branch model. In this research, the internal structure of branches and the shape of annual rings on the fracture surface of branches are analyzed to provide a reliable fracture model for different branch fractures of trees. In addition, the realistic fractured tree branch model generated by this algorithm can be widely applied in fields such as animation film special effects, game scene simulation, virtual reality scene construction, and mechanical research on broken tree branches.

**Keywords:** branch fractures; internal structure reconstruction; 3D point cloud; visualization simulation



**Citation:** Yang, M.; Zhang, Y.; Xi, B. Visualization Simulation of Branch Fractures Based on Internal Structure Reconstruction. *Forests* **2023**, *14*, 1020. <https://doi.org/10.3390/f14051020>

Academic Editors: Huaiqing Zhang, Juan Suárez-Minguez, Qi Chen, Yunsheng Wang, Safa Tharib, Hua Sun, Weipeng Jing, Huaguo Huang, Ting Yun, Meili Wang and Huaiqing Zhang

Received: 4 April 2023  
Revised: 10 May 2023  
Accepted: 14 May 2023  
Published: 16 May 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Specific fields such as film special effects, real-time gaming, and geometric modeling often require highly detailed models to create realistic scenes. Woody plants are an essential element of nature, and their 3D simulation plays a significant role in constructing natural scenes. In real-life situations, the shape of a tree can vary, and almost every tree has broken branches. Different factors, such as the tree species and environmental conditions, can affect branch breaking in different ways. Therefore, it is crucial to reconstruct branch fractures realistically and effectively to better simulate real-life scenes.

Natural disasters such as wind, rain, hail, blizzards, and tornadoes can also occur in the simulation of natural scenes. These disasters can cause damage to trees, resulting in broken branches. For instance, strong winds can blow branches, heavy snow on branches can break them, and tornadoes can uproot entire trees. The cracks in the broken branches caused by different natural disasters have varying effects. To create a more realistic 3D natural scene, it is essential to simulate the natural phenomenon of these branch fractures.

Tree branch fracture is caused by mechanical factors and involves both wood science and physics. Therefore, simulating the generation of fractured tree branches can be an effective way to study wood's resistance to external forces. Additionally, visualizing the fracture of tree branches can provide an intuitive representation of the tree's structural failure, which can aid in the development of strategies for garden protection.

Wood fracture simulation is a crucial aspect of natural landscape simulation. Wood is a complex biological organism, and the internal structure of branches varies among different tree species, making it difficult to simulate the fracture of wood and branches using traditional methods. Fractured branch models collected through 3D scanning instruments often have many voids due to the unevenness of the wood spurs at the fracture site. Using random noise to generate the fracture surface of the wood leads to a lack of realism in the simulation, as it oversimplifies the complex properties of wood. Therefore, there is a pressing need for a 3D plant technique for geometric modeling that can simulate detailed and realistic branch fracture models, allowing for a better analysis of the internal morphological structure and function of woody plants.

This paper introduces a novel approach for reconstructing the internal structure of tree rings and wood fiber bundles using the external contours of branches. The primary objective of this research is to develop a method for generating precise and evenly distributed cross-sectional slices and fragment models of tree rings, which can significantly enhance the creation of detailed and lifelike tree models. Moreover, the resulting model of fractured branches is particularly valuable for simulating the response of trees to external forces and evaluating their mechanical characteristics.

## 2. Related Work

In current graphics research, there is a focus on the dynamic simulation of trees and the development of material fracture algorithms. However, there are few simulation algorithms that are effective in capturing the unique properties of wood and simulating its fracture phenomenon. Furthermore, there are limited direct simulations of tree branch fracture. Wood is a complex biological organism that is porous, anisotropic, and viscoelastic, possessing exceptional mechanical properties. Previous research on tree fracture has mainly focused on two areas: computer graphics and wood fracture mechanics. In computer graphics, the focus has been on simulating trees in a realistic manner from a visualization perspective, including the modeling of trees and their interactions with various environmental factors such as external forces, wind, and rain.

### 2.1. Research on Tree Modeling Algorithm

In the early days of tree modeling research, procedural modeling methods were commonly used. One example is the work by Fernández et al. [1], who studied the function-structure model of *Pinus radiata* and were able to reproduce the development and growth of the species using mechanical formulas. Ancelin et al. [2] proposed a numerical model designed to simulate the biomechanical behavior of growing trees. The model is based on the transfer matrix method and is adjusted incrementally to calculate the evolution of trunk biomechanics during growth. Weber et al. [3] proposed an algorithm to create a tree model. The algorithm can adjust the morphological characteristics such as the rotation angle interval between branches, the length of the sub-branch, and the number of branches through multiple parameters. Initially, by learning a framework, the 3D shape, camera, and texture of an object are recovered from a single image [4]. Then, Hu et al. [5] proposed a self-supervised mesh reconstruction (SMR) method to enhance the 3D mesh attribute learning process. By requiring only contour mask annotations, SMR can be trained in an end-to-end fashion and can generalize to reconstruct natural objects. With the development of 3D scanning technology, many 3D point cloud-based tree modeling methods have emerged. This method generates point clouds that allow for 3D models that are more consistent with real trees. For example, Livny et al. [6] proposed a method to automatically generate realistic tree models from scanned 3D point clouds of trees. This work is based on pre-generated classified tree information and is able to handle large area scans and generate models automatically. Liu et al. [7] proposed TreePartNet, a neural network aimed at reconstructing tree geometry from point clouds obtained by scanning real trees.

In addition to research on modeling algorithms for whole trees, there are many studies on tree growth animation. For example, Kratt et al. [8] studied the animation simulation of

wood growth according to the principle of tree cambium growth, including an algorithm for generating bark folds. Xiao and Chen [9] developed a model of the plant leaf drying phenomenon. The model is driven by the differential contraction of leaf tissue, and it also simulates the leaf vein system on plant leaves.

## 2.2. Research on the Interactive Simulation Algorithm of Trees and Environment

There are several geometric representations of plant models. Different representations have their own advantages and disadvantages in simulating the interaction of trees with environmental factors. Quigley et al. [10] proposed an interactive real-time animation method. This method represents the tree model as multiple articulated rigid bodies. By setting the stiffness of different branch node rigid bodies to avoid the animation effect of excessive softness and bending when subjected to external forces. The tree animation is processed in real time on a tree scene with tens of thousands of branches or a wooded scene composed of many trees with complex branch structures, such as wind and collision. Diener et al. [11] proposed a method to simulate in real time a complex scene of tens of thousands of trees drifting with the wind under a user-controlled wind field, where the branch nodes calculate the acceleration, velocity, and displacement of the object on the basis of the applied external forces, mass matrix, damping matrix, and stiffness matrix.

Yang et al. [12] used an extended trigonal spring model to represent the tree structure. Using the theory of nonrigid tree leaf surfaces with hydrophilic properties to simulate branch and raindrop interactions. Xie et al. [13] used a leaf model based on a mass–spring model representation to simulate the effect of hail striking and tearing leaves.

Pirk et al. [14] proposed a biologically sound approach to simulate tree combustion. Their tree combustion model can establish a link between the description of fine mechanisms (pyrolysis of wood, mass loss of logs, insulation effect of charcoal, and temperature change due to water evaporation) and the description of macroscopic effects (forest fire spread phenomenon), which can realistically simulate the effect of tree combustion.

Jernej et al. [15] decomposed the model volume mesh into several subdomains using the finite element method. A simplified deformable model is constructed for each subdomain to achieve a real-time simulation of deformed objects. Li et al. [16] proposed a simulation method based on the theory of material mechanics that can set stable anisotropic material parameters and can accurately simulate the deformation effect of plant twisting and bending. Bohan et al. [17] proposed a method based on the power law relationship between branch length and its diameter and between length and natural vibration frequency to automate the hardness setting of tree models according to plant biomechanics, which can simulate the dynamic effects of trees with richer layers.

## 2.3. Fracture Modeling and Animation in Graphics

For the simulation of material fracture, materials are usually considered as ideal brittle or isotropic materials. James et al. [18] analyzed the stress tensor calculated using a finite element model. They modeled where the cracks should start on the model volume mesh and in which direction they should propagate. Pfaff et al. [19] proposed a method for adaptive crack propagation on a thin sheet model. This method dynamically reconstructs a high-quality triangular mesh and adaptively maintains the details of the simulation. Chen et al. [20] proposed a user-friendly method for designing and controlling the effect of fracture surfaces. The method refines the low-resolution fracture surface mesh into a high-resolution and detail-rich fracture surface according to the user-set material strength field. Desbenoit et al. [21] proposed an interactive approach to model cracks by mapping editable 2D curve patterns onto a 3D model. Hädrich et al. [22] proposed a new method to simulate wood as an anisotropic material. This method uses the shape-matching method as the basis for modeling the isotropic properties of wood. A fiber model based on the Cosserat rod theory is used to generate fracture.

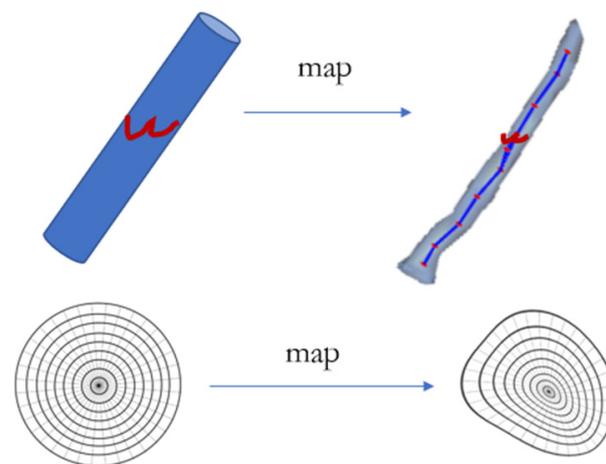
### 2.4. Image Detection of Wood Annual Rings

The annual rings of wood can reflect important information such as the age of the tree and the growth environment of the tree. Tree rings not only are a dating tool, but also provide information on environmental factors during the generation of tree rings, serving as a proxy for environmental changes. The width of tree rings is considered a function of the total number of cells and the radial size of cells, while cell growth rate is a function of other climatic factors such as temperature, soil moisture, and light conditions. It is this functional relationship that transforms the statistical correlation between tree ring growth and external climate factors into a causal relationship [23,24]. Existing work has processed wood cross-section images through image enhancement [25], image processing [26], edge detection [27], and other methods, obtaining information such as the number of annual rings by detecting annual rings.

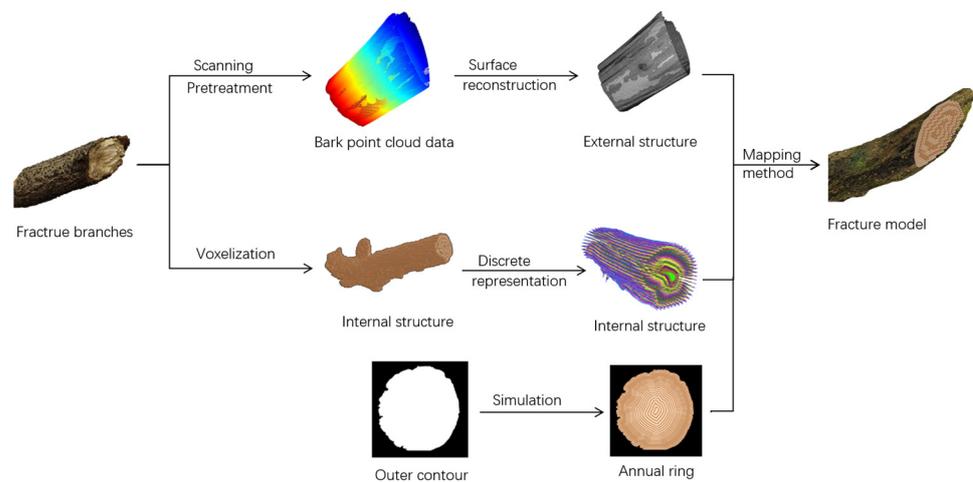
Previous methods approximated the branch fracture surface as a smooth elliptical surface, and they lacked certain rationality and realism. The effect of the branch fracture surface depends on the material difference of different species of wood and the material strength of the internal annual structure of the branch. In this paper, a better model of the branch fracture surface is obtained by reconstructing the internal structure of branches.

### 3. Materials and Methods

As shown in Figure 1, the result of parametric modeling [28] is a cylinder with a variable-height branch fracture surface. The cross-section is still a circle, uniformly divided into  $N_{ring}$  circles and  $N_{sec}$  sectors. However, the shapes of real branches and the 3D branch models used in the animation are different. According to actual observations and knowledge of wood science, the internal structure of tree branches has irregularly shaped annual rings with uneven width, and real wood fiber bundles are also irregular thin rectangles. Therefore, in order to generate a more realistic branch fracture surface that meets the animation requirements, a reasonable and effective method of mapping the fracture surface generated on the regular cylinder to the 3D branch model is needed. As shown in Figure 2, this paper obtains real fracture branches, then further generates bark point-cloud data and internal voxelized data, and finally generates a complete tree broken model through mapping.



**Figure 1.** Schematic diagram of the method of mapping the fracture surface generated on a regular cylinder to a 3D branch model.



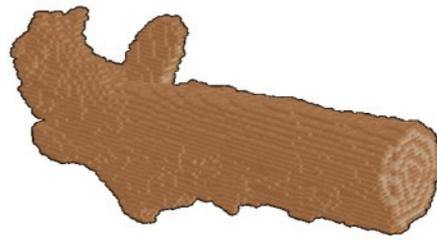
**Figure 2.** Flowchart of this method.

### 3.1. Branch Voxelization Model

According to related research in wood science and botany, the structure of the stem of a branch woody plant is composed of four parts: bark, cambium, essential part, and pith, from outside to inside [29–31]. The voxelization method converts the geometric form representation of an object (vertices and facets) into a voxel representation that is closest to the object at a given resolution  $N$ . The voxelization method is used to convert the branch surface mesh into volume data. The spatial voxels used to represent a 3D model are similar to the two-dimensional pixels used to represent an image. It can be understood as an extension from a two-dimensional pixel unit to a three-dimensional cube unit. The internal distribution and arrangement of trees are similar to the arrangement of plant cells. The voxelized branch model can obtain a branch shape composed of many voxels, which is similar to the arrangement of cells inside the branch. According to the arrangement and combination of voxels to form the various parts of the branch woody plant structure, the changes in the shape and position of different parts can be better realized. The outermost voxels can form the shape of annual rings on the branch fracture surface. The outermost voxels can be used to deform and control the shape of the fractured part, generate different fractured surfaces, and form the shape of annual rings on the fractured surface.

In order to obtain the contours of each layer of the 3D branch model. The branch model needs to be divided into multiple segments in the axial direction. The voxelization method [32,33] divides the model space into  $N \times N \times N$  grids, and sets the grid to 0 or 1 depending on whether the grid is covered by the model. The voxelization method eventually generates volume data of size  $N \times N \times N$  consisting of 0 s and 1 s, where 0 denotes that the voxel is not in the model, and 1 denotes that the voxel is in the model. Each voxel represents a particle.

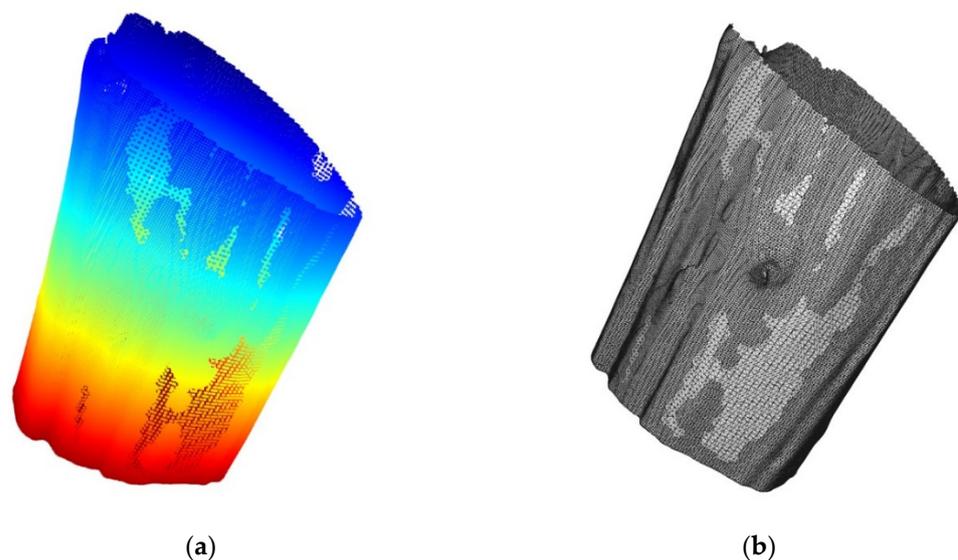
In order to obtain a clearer contour map of the branch interface, the 3D model of the branch should be voxelized using a high-resolution  $N$ . When  $N = 512$ , volume data with 134,217,728 values are generated. However, since the axial length of the branch model is much larger than its cross-sectional width and height, if the whole branch is voxelized directly, it is not possible to generate high-quality volume data even with a larger resolution of 512. As shown in Figure 3, a part of the voxelized branch data is shown. It can be seen that the cross-section of the branch consists of only a few hundred voxels, and the width and height of each cross-section are a few tens of voxels wide. Therefore, in order to obtain a clear cross-section, the long branch model should be divided into  $N_{model}$  segments of the same axial length before voxelization, with each segment being slightly longer than the cross-sectional width of the branch model.



**Figure 3.** Voxelization result of a tree model.

### 3.2. Branch Outer Bark Structure Simulation

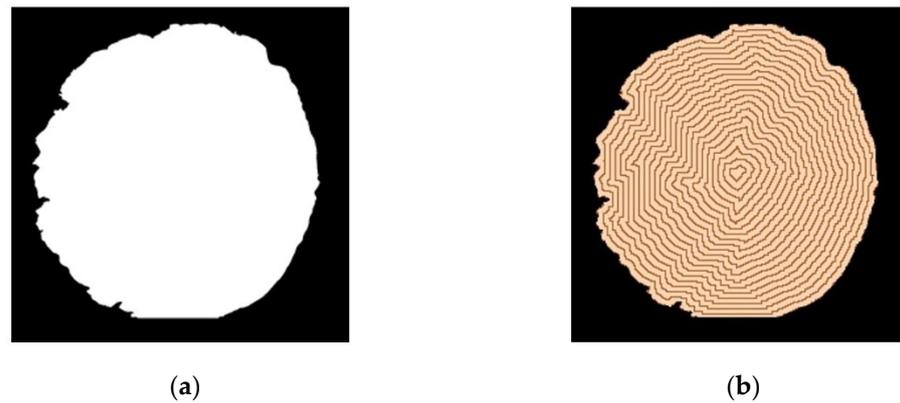
Bark is an important part of woody plants. The bark is located in the outermost part of woody plants and is in direct contact with the real environment. Influenced by different environmental factors, the structure and shape of the bark are very different from the internal structure of the branch. By using real tree branch fracture models to obtain point-cloud data, a detailed and complete outer skin structure of the fractured tree branch can be obtained, resulting in a more realistic model of the fractured tree branch. Using the PRINCE 775 two-color laser handheld 3D scanner of SCANTECH, the fractured tree branches were placed on the platform, and the handheld scanner scanned the fractured tree branches 360° to obtain the point-cloud data of the external contour of the branches. The point-cloud data were used to simulate the bark structure, which was more consistent with the real bark shape. The obtained external point-cloud data were preprocessed, noisy points and free points were removed, and point-cloud data were simplified to generate more accurate data, as shown in Figure 4a. The average distance density of this bark point-cloud data was calculated as 0.1367 points per cubic meter. The point-cloud data at the fracture surface were removed, only the bark data were retained, and the mesh structure was reconstructed using the point-cloud data, as shown in Figure 4b. The bark point-cloud data are denoted as  $p$ .



**Figure 4.** Bark structure point-cloud data: (a) pretreatment result (the different colors of the point-cloud data are used to facilitate the display effect); (b) reconstruction of the generated mesh structure.

### 3.3. Branch Cross-Sectional Annual Ring Simulation

Taking the resolution  $N = 512$ , after voxelizing the branch model,  $512 \times N_{model}$  cross-sections of the branch model were obtained. The length and width values of each cross-sectional map were 512. Figure 5a shows the extracted outer contour drawing. The black color indicates the outer part of the branch, and the white color indicates the inner part of the branch.



**Figure 5.** Broken branch annual rings simulation results: (a) outer contour drawing; (b) simulation-generated annual ring cross-section.

To get the outline of a circle such as a growth ring, a method similar to the erosion algorithm in image processing was used [34,35]. Erosion is performed on the white part of the black-and-white section; each time the white part is corroded, the circle shrinks. By recording this circle of corrupted white pixels, the shape of a chronicle is obtained. Iterative etching is performed until all white pixels in the cross-section are set to black, and the annual wheel is obtained from the outside to the inside.

The erosion method uses a filter kernel of size  $K$  to traverse all pixels in the cross-sectional view, and the filter kernel can cover  $K \times K$  pixels. When the pixel in the center of the filter kernel is white and there are  $M_K$  black pixels currently covered by the filter kernel, the white pixel value in the center is marked and stored. After traversing all pixels, the stored white pixels are modified to the specified color. For example, when the size of the filter core  $K = 3$  and the user set parameter  $M_K = 1$ , for white pixels, if one of the eight adjacent pixels located at top, bottom, left, right, top left, bottom left, top right, and bottom right is a black pixel, it is stored and set to a uniform color in the new erosion result graph.

According to the knowledge of wood science, it is known that annual rings can be divided into earlywood and latewood, with earlywood having a light color and latewood having a dark color. The pixel color of the odd numbered circles was set to dark brown, and the pixel color of the even numbered circles was set to light brown; the dark circles were considered latewood, and the light circles on the inner side were considered earlywood. The dark outer circle and the adjacent light inner circle form an annual ring, and the cross-sectional distribution of annual rings based on the outer contour of the branch can be obtained as shown in Figure 5b. Firstly, a filtering kernel with  $K = 1$  is used to corrode the external contour map of the tree cross-section, and the stored white pixel values are set to dark colors with RGB values of (30, 92, 186). Then, a filter kernel with  $K = 3$  for corrosion traversal is used, the stored white pixel values are set to light colors with RGB values of (184, 215, 248), and filters of sizes 1 and 3 are alternatively used to check the external contour map of the tree cross-section for corrosion until there are no white pixel points in the external contour map of the tree cross-section.

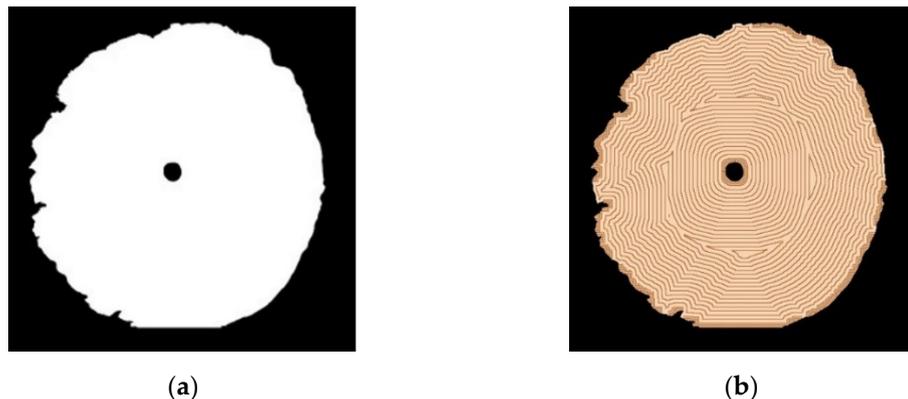
The width of the true annual rings on wood reflects the environmental conditions of the year. When the environmental conditions such as temperature, moisture, and sunlight are suitable for the growth of trees in a year, the annual rings produced in that year will be thicker, and vice versa. Therefore, the width of the annual rings on the cross-section of wood can be wide or narrow, but it is not narrower from the inside to the outside, and there is no obvious pattern. As shown in Figure 6, a cross-sectional map of size  $256 \times 256$  was processed to obtain a more realistic simulation of the annual rings in the cross-sectional area of the branch by reasonably designing the size of the filter kernel used in the corrosion process. First, we alternated the filter kernels of size 3 and 5 to generate a few thin annual rings in the outermost circle. Then, only the filter kernel of size 3 was used to generate a few annual rings with the narrowest width. Next, the kernels of size 3 and 5 were alternated

to generate a few wider annual rings. Then, the kernels of size 3 and 7 were alternated to generate a few wider annual rings, resulting in a gradual change in the width of the annual rings. Lastly, the inner annual rings with the widest width at the center of the branch cross-section were generated by alternating the filter kernels of size 3 and 9 and the filter kernels of size 3 and 11. A total of 24 annual rings were generated.



**Figure 6.** Simulation results of annual rings in cross-section of branches with different annual ring widths.

In the internal reconstruction of some particularly irregular outer contours, in order to conform to the real situation of wood, it should be ensured that the innermost annual rings that are very close to a perfect circle can be obtained. Therefore, the outer contour and the innermost annual ring contour are extracted at the same time, and the annual rings are generated from the outside to the inside and from the inside to the outside respectively. As shown in Figure 7, since the tree growth process is initially less affected by external factors, early growth rings that are approximately perfect circles can be generated by using the outer contour of the innermost ring. When the tree grows to a certain year, the shape of the annual ring will change due to the influence of specific external factors. The outer contour can be used to generate the late growth ring that matches the later growth.



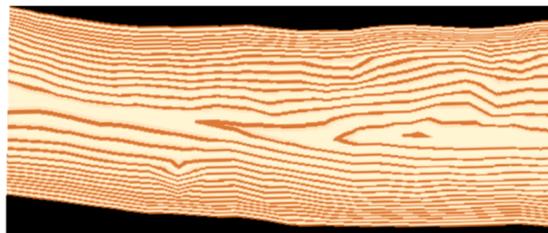
**Figure 7.** Simulation results of annual rings in branch cross-sections generated with different contours: (a) outer contour drawing; (b) simulation-generated annual ring cross-section.

During tree growth, the formation of annual rings in the same circle is easily affected by sunlight intensity and temperature. It is also related to the influence of the local environment and climate. When trees grow in different directions in different environments, it is easy to form eccentric growth rings, as shown in Figure 8. During the simulation process, the range covered by each corrosion ring is varied in the  $x$ - and  $y$ -directions, in order to achieve different widths of the same annual ring in different directions and generate eccentric annual ring images.



**Figure 8.** Simulation results of tree rings in branch cross-sections with different widths of the same annual ring.

The tree cross-section annual ring simulation was carried out on 512 sections of the volume data of a branch model, and the calculated results were stored in a volume data structure with a color. The visualization effect is shown in Figure 9. The longitudinal sections of the wood interior were visually very similar to real wood.



**Figure 9.** Longitudinal section of reconstruction of timber interior structure.

### 3.4. Discrete Representation of Branch Model

The tree branch model is discretized into particles with information about the index value of the annual rings in which they are located, the sector location, and the number of layers in the axial direction. In this way, the information about the internal structure of wood can be stored with a small amount of data. Therefore, a method is proposed to discretize the branch model into particles on the basis of the reconstructed internal structure of wood.

The internal structure of a branch reconstruction is stored in colored body data sections. Each section  $i$  has  $N_{ring(i)}$  annual rings, and each annual ring can be discretized into  $N_{sec}$  particles. Thus, the number of particles to be discretized from the wood internal reconstruction data with resolution  $N_{model}$  can be expressed as follows:

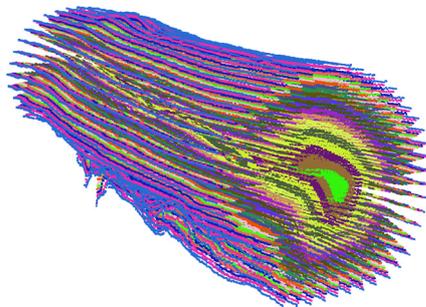
$$Sum_{discrete\_particles} = \sum_i^{N_{model}} N_{sec} N_{ring(i)}. \quad (1)$$

The method of discretizing an annual cycle into  $N_{sec}$  particles is as follows: before each profile erosion operation, the center position of the current branch cross-section is calculated. The center position is equal to the average position of all white dots in the current section. Then, every  $(360/N_{sec})$  degrees, a straight line is emitted from the center outward, and the position of the first black point in the path of each straight line is found along the direction from the center to the outside, which is recorded as the position of a discrete particle. The result of discretizing a cross-section into particles on the basis of annual structure of the wood cross-section is shown in Figure 10, with the discrete particles in red. Figure 11 shows a section of a branch represented by discrete particles, with different colors of particles at different annual rings. Figure 12 shows the surface mesh of the branch reconstructed from the discrete particles. The surface mesh of the “bark” was reconstructed

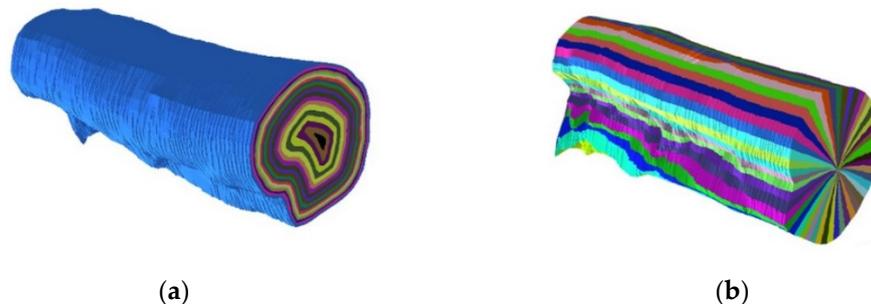
using the position information of the particles located in the outermost annual rings of all sections. The surface meshes of the two ends of the branch were reconstructed using the position information of all particles in the first and last sections.



**Figure 10.** Discrete particles represent a cross-section of the wood.



**Figure 11.** Discrete particles represent the wood; the color of the particle indicates the annual ring it is in.



**Figure 12.** Branch model for discrete particle reconstruction, where the color of the particle indicates the annual ring it is in: (a) different annual rings; (b) different sectors.

#### 4. Mapping Method

By mapping the vertex information of the fracture surface generated on the regular cylinder to the internal wood particles reconstructed according to the external contour of the branch, the fracture surface that fits the contour of the branch and conforms to the internal structural rules of the wood could be obtained. The method in [25] was used to generate a vertex  $V_{ij}$  at the  $i$  annual ring and the  $j$  sector on the regular cylinder, and its height was  $H_{V_{ij}}$ , where  $i$  and  $j$  are greater than 0 but less than 40, and  $H_{V_{ij}} = 16.8$  cm.  $\alpha_{map}$  denotes the ratio of the axial length of the dendritic mesh model to the total number of sections  $N_{model}$  occupied by the dendritic voxel data in the axial direction, where  $N_{model} = 512$ . The algorithm in this paper was used to get a discrete particle  $P_{ijk}$  located on the  $k$  section, where  $k$  is greater than 0 but less than  $N_{model}$ . The vertex of the new fracture surface obtained by mapping is represented by  $P'_{ijk}$ .  $V'_{ij}$  represents the vertices at the  $i$  annual ring and

the  $j$  sector of the new fracture surface obtained through mapping.  $V'_{ij}$  is calculated using Equations (2) and (3):

$$k' = \left[ \alpha_{map} H_{V_{ij}} \right], \quad (2)$$

$$V'_{ij} = P_{ijk'}. \quad (3)$$

The mesh model of the new fracture surface  $k'$  was constructed according to all  $V'_{ij}$  obtained by calculation. The bark point-cloud data sections were divided according to the total number of sections. The vertex  $V'_{ij}$  on the outermost ring of the fracture plane was connected with the bark point-cloud data point  $p$  on the same section layer to form a complete branch fracture model.

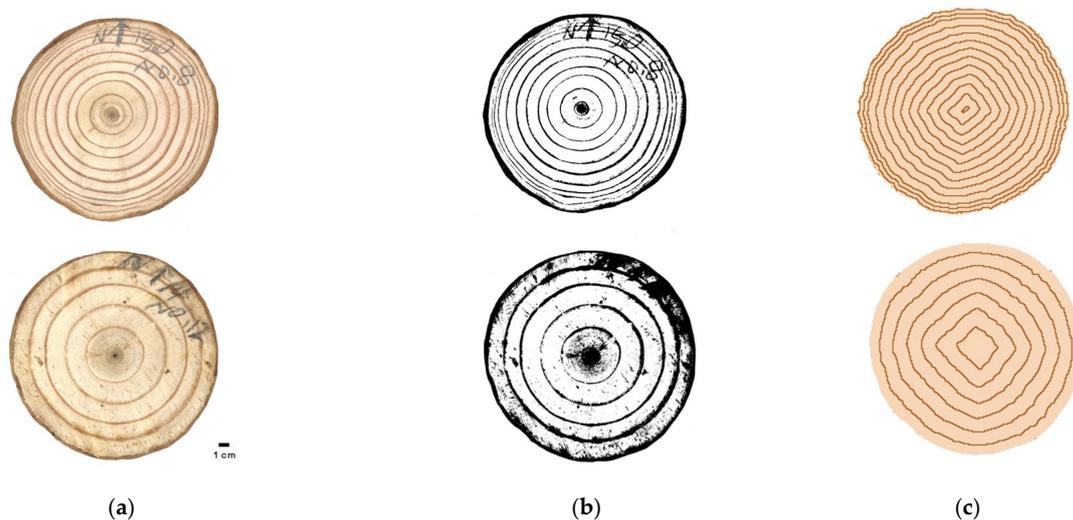
## 5. Results

### 5.1. Experimental Environment

The algorithm in this paper was run in the following hardware and software experimental environment: 16 GB memory, Windows 10 64 bit operating system, CPU AMD Ryzen 7 5800H, GPU NVIDIA GeForce RTX 3060, Visual Studio 2017, Python 3.6, and OpenGL 3.3.

### 5.2. Experimental Results

Cross-sectional tree ring images of tree branches were generated using methods similar to corrosion algorithms. Different filtering kernels could be used to generate rings with different widths. Adding colors with the same RGB values as the real image to each ring could generate a more realistic cross-sectional ring image of fractured branches. Figure 13 shows a comparison of the tree ring contour extraction method based on the image processing method [25] and the tree ring reconstruction results in this paper. The first line is the tree ring reconstruction of *China fir* with a thoracic diameter of 36 cm, and the second line is the tree ring reconstruction of *Cryptomeria fortune* with a thoracic diameter of 24 cm.



**Figure 13.** The comparison of the method of extracting annual rings based on image processing method and the reconstruction results in this paper: (a) real tree rings; (b) image processing method [25]; (c) the method of this paper.

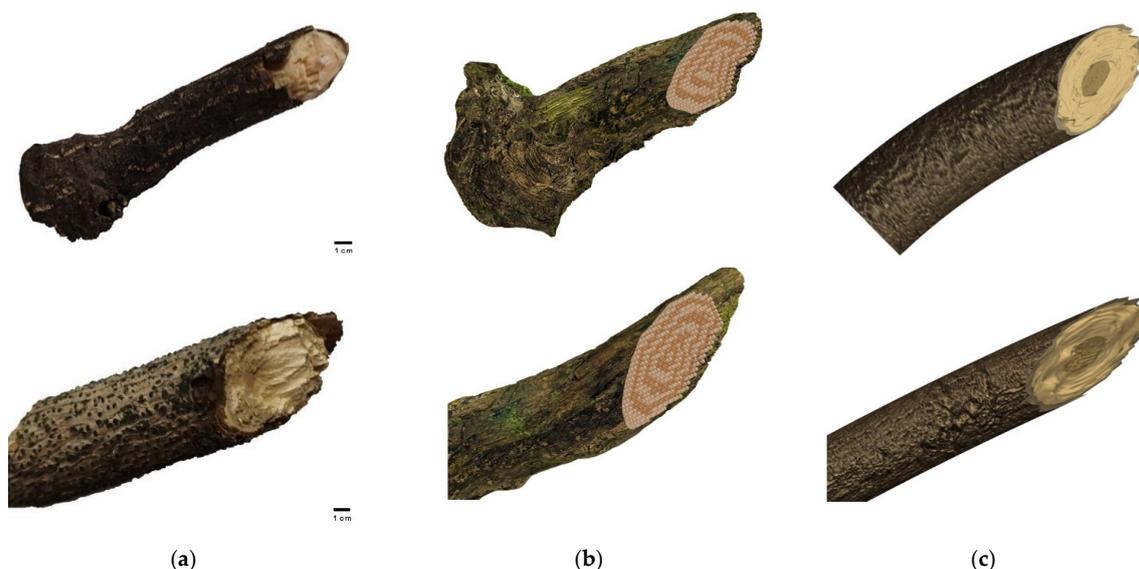
The internal contours were constructed using  $3 \times 3$  filter kernels for different sizes of cross-sectional maps, as shown in Table 1. A  $128 \times 128$  cross-sectional map can generate 41 circles of contours, a  $256 \times 256$  cross-sectional map can generate 80 circles of contours, and a  $512 \times 512$  cross-sectional map can generate 156 circles of contours. A larger resolution takes a longer time to build the internal contour.

**Table 1.** The comparison results of annual rings generated from different size cross-sectional maps.

Resolution	Contour (#)	Time (s)
32 × 32	11	0.1628
64 × 64	21	0.5469
128 × 128	41	3.9118
256 × 256	80	29.1836
512 × 512	159	240.4738

The outer contour was used to simulate the annual ring. After each erosion algorithm, the pixel value of the generated contour and the value of the remaining white pixels were calculated, and then the next erosion was performed. At the same time, according to the number of corrosions, a specific color was given to the corresponding contour to form the final effect diagram of the annual ring section. To process contour images of different sizes, the pixel values that needed to be calculated were different. A larger image resolution required larger pixel values to be calculated, thus taking longer.

The length and width of the collected tree branch were 16.8 cm and 4.5 cm, respectively. The scanned point-cloud data contained 101,422 points with a point density of 0.1367 points per cubic meter. Scanning to obtain point-cloud data of the bark structure of broken branches could preserve complete detail information, generate realistic bark surface structure, voxel the fractured branches, and add ring information to the fracture surface to better simulate the fractured branches; however, the details on the fracture surface were missing. Figure 14b shows the rendered branch fracture model, and Figure 14a shows the real branch fracture picture. It can be seen that the overall simulation effect was good, but there was still a certain gap between the simulation and the real picture in the details. Figure 14c shows the fractured branch model generated by parametric modeling [26], depicting a branch fracture surface with uneven height on a cylinder, but its cross-section was still a circle. Compared with the fractured branch model generated in this paper, the outer surface of the branch was uneven. The bark fracture was similar to the real fracture branch after adding texture, appearing realistic. The real annual rings could be seen at the fracture surface, conforming to the real branch shape. The model generated by parametric modeling was still a cylinder. However, at the fracture surface, the protrusions simulated in this paper were not obvious enough. Using parametric modeling and the fracture surface generated by filtering, it can be seen that the protrusions were more obvious, but the generated positions were still relatively random.

**Figure 14.** Rendered production of broken tree branch model compared with the real picture: (a) real branch fracture picture; (b) rendered branch fracture model; (c) other work [28].

## 6. Discussion

In this paper, we proposed a 3D plant technique to reconstruct the internal annual rings and wood fiber bundle structure on the basis of the external profile of tree trunk or branch. The reconstructed internal structure can be used to map the fracture surface and to represent the inhomogeneous material strength inside the branch. This method is an early computer graphics method to simulate wood fracture according to wood structure and wood science theory. The method incorporates basic features of wood fracture, such as the internal hierarchical structure of wood, including the annual rings and wood fiber bundles, into the parametric modeling approach. The reconstructed model plays an important role in the study and analysis of the internal morphological structure of woody plants. At the same time, the simulation of branch fracture also plays a certain role in the corresponding research of forestry. The branch fracture model is the embodiment of the details of the tree model, and the function of woody plants is accurately calculated and analyzed using the branch fracture structure model. Therefore, the simulation algorithm can be used not only in the fields of entertainment and animation games, but also to promote the research and application of plant morphology and structure in smart agriculture.

## 7. Conclusions

Although the algorithm in this paper achieved the expected results of wood fracture simulation, there were some limitations in its design and experimental process. The design of the method for extracting internal points when performing discretization is still inadequate. The circumference of the annual rings increases from inside to outside according to its position in the cross-section. The circumference of the outermost annual wheel is the longest, and the circumference of the innermost annual wheel is the shortest. The circumference of the outer annual rings is often tens of times longer than that of the inner annual rings. In the discretization result, the points of the outer annual rings are sparsely distributed, and the points of the inner annual rings are densely distributed. According to the viewpoint proposed in this paper, each annual ring is regarded as composed of  $N_{sec}$  wood fiber bundles. The cross-sectional area of wood fiber bundles in the outer annual rings is much larger than that of wood fiber bundles in the inner annual rings. However, according to actual observations of wood and theories in wood science and botany, the cross-sectional area of wood fiber bundles within the same branch cross-section should be essentially the same. For example, the number of segments of each annual rings should be set according to the circumference of the rings, with a longer circumference necessitating the divisions of more segments.

Future work can build on the algorithm in this paper to obtain more realistic and accurate wood fracture simulation results, and future research can be conducted in the following directions:

1. Bark fracture can be simulated. The fracture effect of the bark and the wood fracture effect of the xylem are very different. Using the bark gullies and grain, the bark fracture effect can be further perfected. When the bark breaks, cracks tend to spread along these gullies and form other fracture structures.
2. The effect of fracture brought by structures such as knots on wood and forks on branches can be considered. For example, knots in branches and trunks can add significant complexity to reconstructing the internal structure of wood. It is necessary to design an algorithm that can detect the position of nodes or branches and reprocess the internal structure of the wood there.
3. The real internal structure of the wood obtained from the CT scan can be compared and evaluated with the results of this paper to improve the algorithm.
4. The obtained internal structure can be used to set the strength of the wood internal material. The wood fracture animation can be generated in combination with the material point method to handle anisotropic materials.

**Author Contributions:** Conceptualization, Y.Z., M.Y. and B.X.; methodology, Y.Z., M.Y. and B.X.; validation, Y.Z., M.Y. and B.X.; formal analysis, Y.Z., M.Y. and B.X.; investigation, Y.Z., M.Y. and B.X.; resources, Y.Z., M.Y. and B.X.; writing—original draft preparation, Y.Z.; writing—review and editing, Y.Z., M.Y. and B.X.; visualization, Y.Z.; supervision, Y.Z., M.Y. and B.X.; project administration, M.Y. and B.X.; funding acquisition, M.Y. and B.X. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Key Research and Development Program of China (No. 2021YFD2201203).

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors thank the peer reviewers for their valuable contributions to this research. Thank you to Yang (Liuming Yang) for providing us with assistance in the topic. Thank you to Liu's team (Yuwan Liu) for providing us with point cloud data. Thank you to Wang (Pei Wang) for providing us with professional knowledge and answers.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Fernández, M.; Norero, A.; Vera, J.; Perez, E.C. A functional–structural model for radiata pine (*Pinus radiata*) focusing on tree architecture and wood quality. *Ann. Bot.* **2011**, *108*, 1155–1178. [[CrossRef](#)]
2. Ancelin, P.; Fourcaud, T.; Lac, P. Modelling the biomechanical behaviour of growing trees at the forest stand scale. Part I: Development of an incremental transfer matrix method and application to simplified tree structures. *Ann. For. Sci.* **2004**, *61*, 263–275. [[CrossRef](#)]
3. Weber, J.; Penn, J. Creation and rendering of realistic trees. In Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA, 15 September 1995; pp. 119–128.
4. Kanazawa, A.; Tulsiani, S.; Efros, A.; Malik, J. Learning category-specific mesh reconstruction from image collections. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 371–386.
5. Hu, T.; Wang, L.; Xu, X.; Liu, S.; Jia, J. Self-supervised 3D mesh reconstruction from single images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 6002–6011.
6. Livny, Y.; Pirk, S.; Cheng, Z.; Yan, F.; Deussen, O.; Cohen-Or, D.; Chen, B. Texture-lobes for tree modelling. *ACM Trans. Graph. (TOG)* **2011**, *30*, 1–10. [[CrossRef](#)]
7. Liu, Y.; Guo, J.; Benes, B.; Deussen, O.; Zhang, X.; Huang, H. TreePartNet: Neural decomposition of point clouds for 3D tree reconstruction. *ACM Trans. Graph.* **2021**, *40*, 232:1–232:16. [[CrossRef](#)]
8. Kratt, J.; Spicker, M.; Guayaquil, A.; Fiser, M.; Pirk, S.; Deussen, O.; Hart, J.C.; Benes, B. Woodification: User-Controlled Cambial Growth Modeling. *Comput. Graph. Forum* **2015**, *34*, 361–372. [[CrossRef](#)]
9. Xiao, H.; Chen, X. Modeling and simulation of curled dry leaves. *Soft Matter* **2011**, *7*, 10794–10802. [[CrossRef](#)]
10. Quigley, E.; Yu, Y.; Huang, J.; Lin, W.; Fedkiw, R. Real-time interactive tree animation. *IEEE Trans. Vis. Comput. Graph.* **2017**, *24*, 1717–1727. [[CrossRef](#)]
11. Diener, J.; Rodriguez, M.; Baboud, L.; Revéret, L. Wind projection basis for real-time animation of trees. *Comput. Graph. Forum* **2009**, *28*, 533–540. [[CrossRef](#)]
12. Yang, M.; Wu, E.H. Approach for physically-based animation of tree branches impacting by raindrops. *J. Softw.* **2011**, *22*, 1934–1947. [[CrossRef](#)]
13. Xie, J.; Yang, M.; Liang, W.; Cui, Y.; Qi, X.; Yang, G. Simulation of hail impact on leaves based on mass-spring model. *J. Graph.* **2019**, *40*, 32.
14. Pirk, S.; Jarzabek, M.; Hädrich, T.; Michels, D.L.; Palubicki, W. Interactive wood combustion for botanical tree models. *ACM Trans. Graph. (TOG)* **2017**, *36*, 1–12. [[CrossRef](#)]
15. Barbič, J.; Zhao, Y. Real-time large-deformation substructuring. *ACM Trans. Graph. (TOG)* **2011**, *30*, 1–8. [[CrossRef](#)]
16. Li, Y.; Barbič, J. Stable anisotropic materials. *IEEE Trans. Vis. Comput. Graph.* **2015**, *21*, 1129–1137. [[CrossRef](#)]
17. Wang, B.; Zhao, Y.; Barbič, J. Botanical materials based on biomechanics. *ACM Trans. Graph. (TOG)* **2017**, *36*, 1–13. [[CrossRef](#)]
18. O'Brien, J.F.; Bargteil, A.W.; Hodgins, J.K. Graphical modeling and animation of ductile fracture. In Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, San Antonio, TX, USA, 23–26 July 2002; pp. 291–294.
19. Pfaff, T.; Narain, R.; De Joya, J.M.; O'Brien, J.F. Adaptive tearing and cracking of thin sheets. *ACM Trans. Graph. (TOG)* **2014**, *33*, 1–9. [[CrossRef](#)]
20. Chen, Z.; Yao, M.; Feng, R.; Wang, H. Physics-inspired adaptive fracture refinement. *ACM Trans. Graph. (TOG)* **2014**, *33*, 1–7. [[CrossRef](#)]
21. Desbenoit, B.; Galin, E.; Akkouché, S. Modeling cracks and fractures. *Vis. Comput.* **2005**, *21*, 717–726. [[CrossRef](#)]
22. Hädrich, T.; Scheffczyk, J.; Palubicki, W.; Pirk, S.; Michels, D.L. Interactive Wood Fracture. *SCA (Posters)* **2020**, 5–7.
23. Fritts, H.C. *Tree Rings and Climate*; Academic Press: London, UK, 1976; pp. 1–567.
24. Shao, X. Several Advances in Tree Ring Chronology. *Quat. Res.* **1997**, *3*, 265–271.

25. Wang, Y.F.; Feng, H.L.; Fang, Y.M. Improving the tree-age measurement by enhancing the tree-ring image with bilateral filter. *J. For. Eng.* **2017**, *2*, 109–114.
26. Xue, Q.; Yang, Y. The Application Advances of Image Processing and Pattern Recognition Technology in Dendrochronology. *Chin. Agric. Sci. Bull.* **2015**, *31*, 24–29.
27. Han, Q.Y.; He, J.D. Segmentation Algorithm of Tree Ring Image Based on Region Growing. *J. Agric. Mech. Res.* **2006**, *4*, 204–206.
28. Yang, L.; Yang, M.; Yang, G. Modeling fractures and cracks on tree branches. *Comput. Graph.* **2019**, *80*, 63–72. [[CrossRef](#)]
29. Sapala, A.; Runions, A.; Routier-Kierzkowska, A.L.; das Gupta, M.; Hong, L.; Hofhuis, H.; Verger, S.; Mosca, G.; Li, C.; Hay, A.; et al. Why plants make puzzle cells, and how their shape emerges. *Elife* **2018**, *7*, e32794. [[CrossRef](#)]
30. Jackson, M.; Xu, H.; Duran-Nebreda, S.; Stamm, P.; Bassel, G.W. Topological analysis of multicellular complexity in the plant hypocotyl. *Elife* **2017**, *6*, e26023. [[CrossRef](#)]
31. Montenegro-Johnson, T.D.; Stamm, P.; Strauss, S.; Topham, A.T.; Tsagris, M.; Wood, A.T.; Smith, R.S.; Bassel, G.W. Digital single-cell analysis of plant organ development using 3DCellAtlas. *Plant Cell* **2015**, *27*, 1018–1033. [[CrossRef](#)] [[PubMed](#)]
32. Min, P. binvox[CP/OL]. 2004. Available online: <https://www.patrickmin.com/binvox/> (accessed on 11 May 2020).
33. Nooruddin, F.S.; Turk, G. Simplification and repair of polygonal models using volumetric techniques. *IEEE Trans. Vis. Comput. Graph.* **2003**, *9*, 191–205. [[CrossRef](#)]
34. Dougherty, E. *Mathematical Morphology in Image Processing*; CRC Press: Boca Raton, FL, USA, 2018.
35. Shih, F.Y. *Image Processing and Mathematical Morphology: Fundamentals and Applications*; CRC Press: Boca Raton, FL, USA, 2009.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.