H-048

Fundamental Study of Reconstructing Botanical Tree's 3D Models from Kinect's Images and Dynamical Behaviors of Non-leafy Trees

Khanh N. Do†   Jun Ohya†

1. Introduction

It is very difficult to synthesize the behavior of non-rigid objects such as smoke, water and botanical tree due to their complicated structure and behavior. Among these non-rigid objects, this paper reports our fundamental study of reconstructing botanical trees' behaviors.

Giacomo et al. [1] proposed two methods for animating a botanical tree: a procedural method and a physically-based method which allows user interaction with the trees. However, both methods can only animate the trunk motion and the leaf motion is not animated. Therefore, the botanical tree motion is not as close as natural motion. Yang [2] introduced a new approach to quickly design a 3D model of botanical trees and it enables the simulation of tree animation by introducing physically-based transformation matrix calculations for hierarchical branch patterns. Although this method can give motions to leaves, this motion is artificial so that the botanical tree behaviors do not look like natural behaviors. C. Li's paper [3] developed a method to track the dynamical behavior of a botanical tree. It is based on the Maximum Stable Extremal Regions (MSERs) method [4] to track the dynamic texture of a botanical tree's foliage. This method received huge notices from computer graphics and animation because it is the first method which allows tracking the foliage of botanical tree. Unfortunately, this method can only track the texture of the foliage; therefore, it cannot work on non-leafy botanical tree.

Our study is to develop a method to reconstruct the natural dynamical behaviors of non-leafy botanical tree in virtual space. In order to do that, the motion of the botanical tree's leaves and trunks need to be tracked and then, the recognized motions should be embedded to 3D botanical model to create the dynamical motion. In this paper, we studies particle filter based method for tracking the dynamical behaviors of non-leafy botanical tree and reconstructing botanical tree's 3D models from Kinect's images.

The rest of this paper is organized as follows. Section 2 explains how to get the 3D model of botanical tree. Section 3 explains the Particle filter based method for tracking the dynamical behaviors of a botanical tree and experimental. Section 3 concludes this paper with discussion about our plan.

2. Reconstructing Botanical Tree's 3D Model

Since our ultimate objective is to bring the naturally dynamical behaviors of a real botanical tree to its 3d model, we need to create 3D model which must look similar to the real 3D model.

Our solution to this problem is to use Kinect as a 3D scanner to reproduce the 3D model. We use Kinect Fusion algorithm [6] to do it.

The Kinect Fusion algorithm is explained as follow. First, the

† Graduate School of Global Information and
Telecomunication Studies, Waseda University

raw depth data is obtained from Kinect sensor and filtered with a bilateral filter to remove the error. Second, the raw depth map is

Figure 1. Non-leafy botanical tree (left) and its 3D model (right)

converted into vertices with corresponding normal vectors. Third, the Iterative Closest Point (ICP) algorithm is applied to the two vertex and normal maps. Fourth, the new depth data is merged
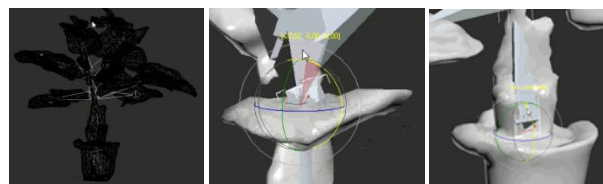

Figure 2. Left: Creating bone structure.
Middle: Animating leaf. Right: Animating trunk

with the current model using the pose we obtained from the previous step. At last, the final 3D model is generated by performing a ray-casting algorithm. The result is shown in Fig. 1. The leaves are reconstructed well but the trunks are not because they are too thin. This is due to the limitation of Kinect hardware so that at this moment, we manually added the trunk to the 3D model. We could also manually add the texture using Zbrush tool in the near future.

The 3D model can be animated, for example, in 3DS Max environment by adding the bone structure to the model as shown in Figure 2 (Left). It also needs detach each leaf and trunk from the model and attach them to the bone. The animation can be created by generate a different leaf's pose and trunk's pose at a certain frame. 3DS Max can automatically interpolate their poses between frames so that that animation is smooth.

3. Particle Filter Based Method for Tracking Dynamical Behaviors of a Botanical Tree

3.1. Initialization and Particle Definition

Figure 3. Overview the method

| 1st frame | 40th frame | 80th frame |
| 120th frame | 160th frame | 200th frame |

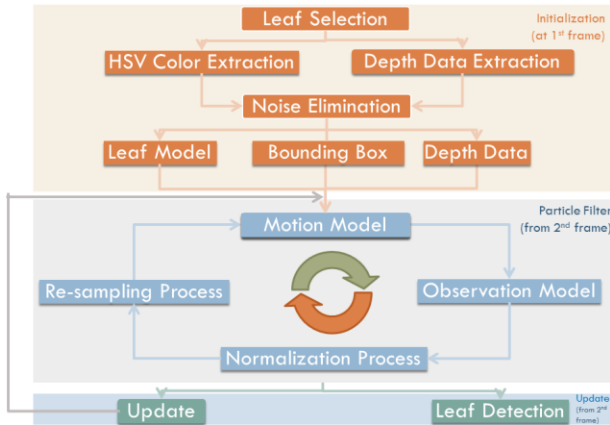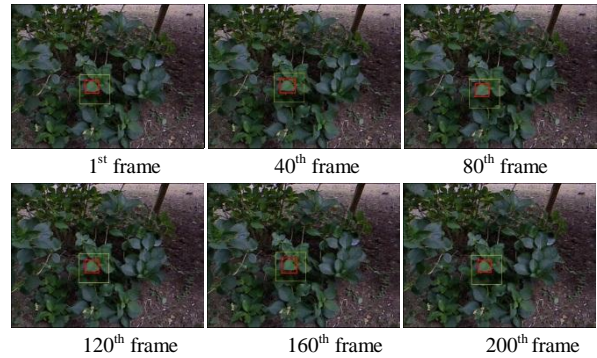Figure 4. Tracking a leaf using proposed method. The tracker is able to follow object which has similar color to background during the whole sequence.

In this paper, a particle, which has two factors, is defined as $s_t^n = (x_t^n, w_t^n)$, where t denoted time t. The first factor represents the target leaf's state estimation and the second factor represents the particle's weight. Specifically, x_t^n is defined as $[a_t, b_t, r_t, h_t, mind_t, maxd_t, \alpha_t]^T$, where a_t, b_t are the coordinates of top-right points of the target leaf' bounding box in image plane. r_t, h_t are the width and height of the target leaf's bounding box, respectively. $mind_t$ and $maxd_t$ are the minimum and maximum distances between the camera and a point on the target leaf, respectively, α is the rotation angle around z axis between two consecutive frames.

3.2. Motion Model

The motion model p($x_t \mid x_{t-1}$) is applied to all N particles independently. That is $x_t = F_t x_{t-1} + noise_t$ $\qquad$ (1)

where F_t is the linear motion model. $noise_t$ is the noise. In Eq. (1), $F_t x_{t-1}$ predicts a new location for each particle. Each particle has different state estimate in previous frame. Therefore, the particle predictions are be different. $noise_t$ in Eq. (1) is the noise.

3.2. Observation Model

In the observation model, template matching is used to compare the template with the target leaf.

First, in the RGB image, all the pixels whose depth values are out of the range (mind+Δ, maxd+Δ) are deleted.

Second, the sum of squared differences (SSD) algorithm is used to compute the sums of the squared error of absolute difference between pixels in the template and the current frame based on Eq. (3).

$$d(T, I) = \sum_{i=1, j=1}^{m,n} \left| T_{i,j} - I_{k,l} \right| \qquad (3)$$

where d(T,I) is the sum of squared differences between pixel values of the template image T and corresponding pixel value of the current image I. The size of T is m x n pixels.

3.3. ROI Update

The leaf's motion vector between the first frame and second frame is used as the motion vector of the ROI between the second and the third frame. Therefore, the motion vector of the ROI between $(t-1)^{th}$ frame and t^{th} frame is defined by the motion vector of the leaf between the $(t-2)^{th}$ frame and $(t-1)^{th}$ frame, where t is the current frame by Eq. (4), where $ROI_t(x,y)$ is the position of ROI at frame t^{th}, $L_t(x,y)$ is the position of the target leaf at frame t^{th}.

$$ROI_t(x, y) = L_{t-1}(x, y) - L_{t-2}(x, y) + ROI_{t-1}(x, y)$$

4. Conclusion

The experiments show that the model of tree can be reproduced by using Kinect as a 3D scanner and the Particle filter method can used to track the dynamical behavior of the tree.

Some problems that we need to solve are (1) the trunks are not reconstructed well due to its size, (2) the texture is not added automatically, (3) how to add the tracking data to add to the 3D model.

Our future work is to find a way to add the tracking data to the 3D model and improve the 3D model with texture.

References

[1] Thomas Di Giacomo, Stéphane Capo, and François Faure. 2001. An interactive forest. In *Proceedings of the Eurographic workshop on Computer animation and simulation*, W. Hansmann, W. Purgathofer, and F. Sillion (Eds.). Springer-Verlag New York, Inc., New York, NY, USA, 65-74.

[2] Meng Yang; Bin Sheng; Enhua Wu; Hanqiu Sun; "Multi-level tree branch modeling and animation," *Computer-Aided Design and Computer Graphics, 2009. CAD/Graphics '09. 11th IEEE International Conference*, pp.96-99, 19-21 Aug, 2009.

[3] C. Li and P. Hall, *Learning a Stable Structure to Describe Dynamic Texture*, British Machine Vision Conference. 23-32, 2008.

[4] J. Matas, O. Chum, M. Urba, and T. Pajdla. "*Robust wide baseline stereo from maximally stable extremal regions*." Proc. of British Machine Vision Conference, pages 384-396, 2002.

[5] Michael Isard and Andrew Blake. "*CONDENSATION -- conditional density propagation for visual tracking*". Int. J. Computer Vision, 29, 1, 5--28, (1998).

R. N. Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Steve Hodges, Pushmeet Kohli, Andrew Davison, and Andrew Fitzgibbon, *"KinectFusion: Real-Time Dynamic 3D Surface Reconstruction and Interaction,"* in SIGGRAPH Talks, ed: ACM SIGGRAPH 2011.