Interactive DPM for Thin Plants with the Latency Measurement

Tomoki Sueyoshi
Kyushu University
sueyoshi.tomoki.661@s.kyushu-u.ac.jp

Yuki Morimoto
Kyushu University
morimoto@design.kyushu-u.ac.jp

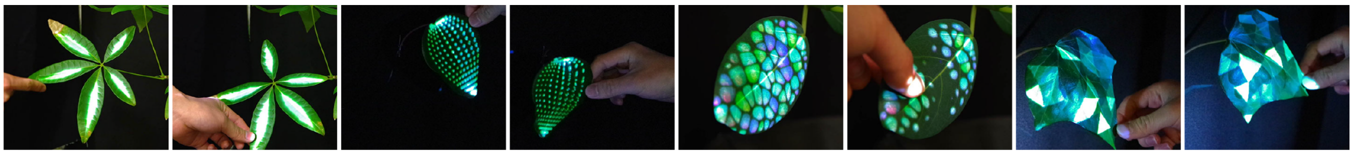

Figure 1: Our results: DPM for thin plants using four aesthetic effect animations by considering user's contact.

CCS CONCEPTS

• **Applied computing** → **Media arts**.

KEYWORDS

Interactive Projection Mapping, Image Processing, Visual Effects

ACM Reference Format:
Tomoki Sueyoshi and Yuki Morimoto. 2021. Interactive DPM for Thin Plants with the Latency Measurement. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Posters (SIGGRAPH '21 Posters), August 09-13, 2021.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3450618.3469151

1 INTRODUCTION

We propose a system to generate an interactive projection mapping onto thin plants automatically. Recently, various objects have become targets for projection mapping (PM). For example, in the artwork "Projections in the Forest" [van Schoor and Mawad 2014], plants and creatures are the projection targets. In this video artwork, the bioluminescent-like plants and creatures in the forest were created by PM and then filmed and edited. However, this work required considerable time (six weeks) to register effect animations (effects) manually for targets in the forest. It is thus very time-consuming to create PM works, particularly for natural objects. Our system therefore registers projection areas on targets semi-automatically. When plants, our projection targets, move and deformed by wind or are touched by a hand, the system can track the targets. Therefore, this system enables dynamic PM (DPM) for thin plants such as leaves and flowers. Previous works [Bermano et al. 2017; Narita et al. 2015] applied pre-input 3D models of the targets and/or markers attached to the targets in advance. A previous DPM generation method for unknown shapes without such measures [Miyashita et al. 2018] required expensive hardware.

Therefore, our system generates DPM semi-automatically using low-cost hardware with reduced prior input. The system enables DPM creation on plants by considering the user's contact with the plants. The system calculates occlusion of the projection target. This calculation is robust against interaction because the occluded areas are recalculated by applying a rigid deformation. We also measured the latency caused by our tracking method.

In addition, the projected effects can be changed interactively by using the occluded position with the contact time obtained using a capacitive sensor. The concept of our effects is to make plants themselves appear luminous, e.g., as if they were bioluminescent. The system generates effects automatically using the shape of the projection target (Fig. 1).

2 METHOD

Our system consists of a ProCam (projector and camera) system (Sec 2.1) that acquires target surfaces using the camera as candidate projection areas and projects images applied to these surfaces via the projector. In our system, points of each contour of the specified projection area in the previous frame are first tracked on every frame (Sec 2.2) to map the effects. Second, a capacitive sensor detects touch events on target objects for interactive effects generation (Sec 2.3). These effects are generated using our concept. Finally, these effects are projected onto real-world plants.

2.1 System Configuration

Our system comprises the ProCam, an IR light and a capacitive sensor. We use a registration method [Sueyoshi and Morimoto 2018] that reduces the time required to fit an input image to the entire projection area by acquiring calibrated information about the ProCam during preprocessing. The capacitive sensor comprises a computer and a resistor. The sensor detects touches on targets by controlling transmission and reception of voltage signals. Both sensor electrodes are connected via a resistor and a branched electrode is inserted into the potted plant.

2.2 Tracking Method for Plants

Our system tracks specified projection areas for projection onto dynamic and deformable targets. Initially, the projection area is specified in the first frame (t=0). After the first frame (t>0), corresponding points between the previous and current frames are calculated to map the effect via texture mapping. Each point is checked to verify if they are occluded or not and occluded points

SIGGRAPH '21 Posters, August 09-13, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8371-4/21/08.
https://doi.org/10.1145/3450618.3469151

are recalculated. In this section, for simplicity, we describe our method when a single region is to be tracked.

2.2.1 Initial Specification of Projection Area. In the first frame (t=0), the projection area is specified as a contour. The creator specifies a contour via a mouse click as the projection area for a plant. The contour vertices of the area are p_i for i=1,2,…, n, where n is the number of vertices in the area. Each projection area is divided into triangular meshes using p_i by constrained Delaunay triangulation. The interior angle of each point is also calculated. If the interior angle is $0\sim\frac{2\pi}{3}$, $\frac{4\pi}{3}\sim 2\pi$, p_i^0 is defined as feature pointq_j for j=1,2,…, m, where m is the number of feature points.

2.2.2 Detection of Occluded Points. First, the optical flow $\boldsymbol{o}_i^t$ is a motion vector of p_i^{t-1} from the input images I^{t-1}, I^t used to calculate p'^t_i (=p_i^{t-1}+ $\boldsymbol{o}_i^t$). Next, the brightness difference is calculated from the area around p_i^{t-1}, p'^t_i on I^{t-1}, I^t and the differences in the last 10 frames are averaged as the error value e_i^t for each vertex. The condition of each vertex is determined to "occluded" or "not occluded" with e_i^t by using k-means clustering.

2.2.3 Tracking of Non-Occluded Points. Non-occluded points are calculated by fitting points to the contour extracted from the edge image E^t. Initially, the feature points q_j^t that can be detected most easily are calculated. Around the point q'^t_j (=p_j^{t-1}+ $\boldsymbol{o}_i^t$), the point which has the closest internal angle to that of q'^t_j on E^t is calculated as q_j^t. Next, the nonfeature points are calculated. The closest point in the edge pixels on E^t from p'^t_i is calculated as p_i^t. Second, the Laplacian $\Delta\boldsymbol{p}_i^0$ of each point is considered to maintain the initial contour shape. The point that has the closest Laplacian to $\Delta\boldsymbol{p}_i^0$ is calculated from the edge pixels via iterative calculations.

2.2.4 Tracking of Occluded Points. The tracking results for the occluded points are calculated by rigid deformation [Schaefer et al. 2006] with the non-occluded points as control points.

2.2.5 Kalman Filter Application. For display purposes, a Kalman filter is applied to p_i^t to perform denoising using the predicted value.

2.3 Interaction and Effect Animation

The touched position p^t_{touch} and touch duration t_{touch} (≥0) on plants are calculated using a combination of occlusion and touch detection. The touch event is detected by the capacitive sensor. The occluded point p_i^t with the highest e_i^t is p^t_{touch} and is used as the source of interactive effects.

The concept of our effect is to make plants themselves appear luminous. Therefore, the effects are generated automatically from the target shapes using the motifs of bioluminescence and light or shine from the natural world. The effects can also be changed using the touch duration t_{touch} and the touched position p^t_{touch}.

3 RESULTS AND CONCLUSION

We show projection results for the effect animations in Fig. 1. These results show that our system can track thin plants automatically and project onto targets in the real world. Here we describe the times required for tracking (Fig. 2 left). The system can track a single target with 10 points in 3.6 ms. Although the tracking time increases with increasing numbers of targets (tracked points), the system can track 9 targets with 94 points in 16.6 ms (60 fps).

Latency Measurement. We also measured the projected area outside the projection target as the latency with and without tracking for comparison. For this measurement, we moved the leaf horizontally at a constant speed. With our tracking method, we projected the area of the tracked target onto white. Without our tracking method, an image taken at each frame is binarized, and its foreground is projected in white. We obtained the image with a camera (1920×1080) and measured the projected white area outside the target leaf (Fig. 2 right). The area of the leaf on the image was 34235px. Without tracking, the projected area outside of the target was 1152px/frame (3.4%). This area is considered to be the system latency caused by hardware. With our tracking, the projected area outside the target was 1261px/frame (3.7%). Therefore, the latency caused by our tracking method was 109px/frame (0.3%), and it is small enough compared to the system latency.

To generate interactive DPM onto thin plants, we have proposed a tracking method that is robust against interactions with occlusions based on an interactive system with a touch sensor. In future work, we would like to improve our system to increase its scale and the number and variety of plants for mapping.

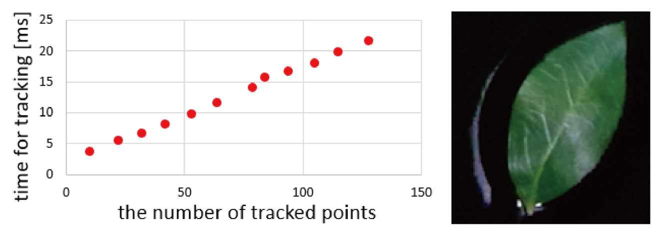

Figure 2: Time for tracking and the number of tracked points (left), the projected leaf to measure latency (right).

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number 18K11956. This work was also supported by HAYAO NAKAYAMA Foundation for Science & Technology and Culture.

REFERENCES

Amit H. Bermano, Markus Billeter, Daisuke Iwai, and Anselm Grundhöfer. 2017. Makeup Lamps: Live Augmentation of Human Faces via Projection. *Comput. Graph. Forum* 36, 2 (May 2017), 311–323.

Leo Miyashita, Yoshihiro Watanabe, and Masatoshi Ishikawa. 2018. MIDAS Projection: Markerless and Modelless Dynamic Projection Mapping for Material Representation. *ACM Trans. Graph.* 37, 6, Article 196 (Dec. 2018), 12 pages. https://doi.org/10.1145/3272127.3275045

Gaku Narita, Yoshihiro Watanabe, and Masatoshi Ishikawa. 2015. Dynamic Projection Mapping onto a Deformable Object with Occlusion Based on High-Speed Tracking of Dot Marker Array. In *Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology* (Beijing, China) *(VRST '15)*. Association for Computing Machinery, New York, NY, USA, 149–152. https://doi.org/10.1145/2821592.2821618

Scott Schaefer, Travis McPhail, and Joe Warren. 2006. Image Deformation Using Moving Least Squares. *ACM Trans. Graph.* 25, 3 (July 2006), 533–540. https://doi.org/10.1145/1141911.1141920

Tomoki Sueyoshi and Yuki Morimoto. 2018. Automatic Generation of Interactive Projection Mapping for Leaves. In *SIGGRAPH Asia 2018 Posters* (Tokyo, Japan) *(SA '18)*. Association for Computing Machinery, New York, NY, USA, Article 2, 2 pages. https://doi.org/10.1145/3283289.3283310

Friedrich van Schoor and Tarek Mawad. 2014. Projections in the Forest.