Dynamic Projection Mapping for Silkworms

Mayu Arano
Kyushu University
arano.mayu.529@s.kyushu-u.ac.jp

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

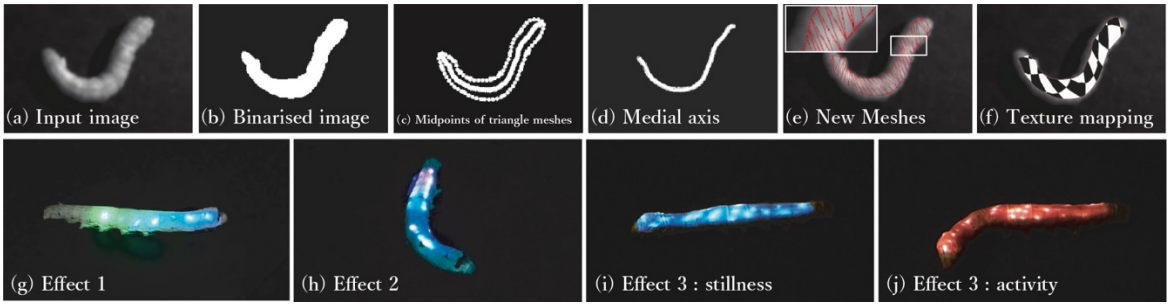

Figure 1: Overview of the system (top row) and the resulting image projected on real silkworms (bottom row). The motif is based on bioluminescent sea creatures.

ACM Reference Format:
Mayu Arano and Yuki Morimoto. 2021. Dynamic Projection Mapping for Silkworms. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Posters (SIGGRAPH '21 Posters), August 09–13, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3450618.3469153

1 INTRIDUCTION

Recently, there has been an increasing number of examples of dynamic projection mapping (DPM) applied to plants and animals. However, objects such as animals and plants are deformable, and it is difficult to obtain their shape or attach markers in advance; thus, these objects present a problem for DPM. Here, we propose an automatic generation method of DPM specially designed for silkworms. In addition, the system automatically generates visual-effect animations based on the shape of the projected object. Using this system, we conducted an experiment of projection on a real silkworm.

Projections in the Forest [Schoor and Mawad 2014] is one example of a work in which PM is applied to plants and animals. In this previous work, static projection mapping was performed for plants and animals of the forest. Image registration for the mapping was performed manually by the artists, which required a substantial amount of time. For plant leaves with individual motion, a method of DPM that considers the occlusion of the projection area by the viewer has been proposed [Sueyoshi and Morimoto 2019]. This method automatically generates effects that match the shape of the plant. However, leaves exhibit small deformations, and it is difficult to obtain corresponding large deformations using this method. A method for applying DPM to human faces has also been proposed [Bermano et al. 2017]. This method can track the movement of a face without the need for markers. However, this method requires 3D face data with various expressions to be input in advance. OpenPose [Cao et al. 2017], which uses deep learning to detect skeletal structures from an input image of a human obtained via a monocular camera, has been proposed. This approach is marker- less, requiring only the input image. However, the detection target is human-based, and non-human skeleton detection is not supported.

2 OVERVIEW OF METHOD

The system configuration of our method follows existing work [Sueyoshi and Morimoto 2018]. A camera and projector are located as close to each other as possible. They are placed directly above the projection target, and the infrared illumination is positioned so that the light reaches the projection target. Then, the projected area and captured area are matched using a homography transformation.

In each infrared image containing the silkworm (Fig. 1a), the region of the silkworm is obtained by binarization (Fig. 1b). The tracking area is then manually specified by clicking the mouse on the binarized image. A contour line is obtained from this region, and triangle meshes are generated from its vertices. The midpoints of these meshes (Fig. 1c) are obtained as the axis points (Fig. 1d). Based on the medial axis, the contour is resampled to create meshes again (Fig. 1e). Next, the effect is generated, and texture mapping is performed (Fig. 1f). From the next frame, DPM is automatically performed by tracking the contour points.

3 TRACKING METHOD

Using our method, we tracked a silkworm that flexibly deforms by extracting its axis. After the axis was extracted, we detected four points on the contour, and the remaining vertex coordinates were determined by equal sampling among the four points.

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

ACM ISBN 978-1-4503-8371-4/21/08.
https://doi.org/10.1145/3450618.3469153

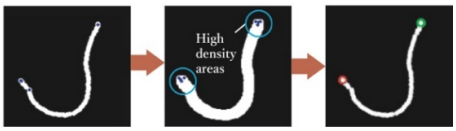

Figure 2: Detection of end points if more than two end points are detected.

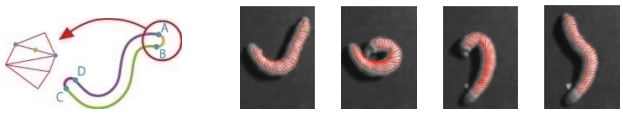

Figure 3: Division of the contour lines (left). Tracking results over time (right).

3.1 EXTRACTION OF THE MEDIAL AXIS

The medial axis is extracted as the concatenation of the midpoints of the edges of the mesh, except on the contour lines. First, the extracted contours are resampled at equal intervals. To perform sampling at a sufficiently narrow interval with respect to the process described in Section 3.2, sampling was conducted for every 4 pixel in this paper. Next, a mesh is created based on those points by Delaunay triangulation. The target area (Fig. 1b), which has been reduced, is used as a mask to extract the midpoints of the created meshes as the axis points (Fig. 1d).

3.2 END POINT DETECTION

The axis points extracted in Section 3.1 are drawn as filled circles whose radii are equal to half of the resampling interval so that the drawing areas are connected. In this paper, the radius is 2 pixel. The AKAZE detector [Alcantarilla et al. 2013] is used to detect the blob feature points. Points whose feature value is greater than the threshold value (0.02) are detected. If only two points are detected, these two points are defined as the end points. If more than two feature points are detected (Fig. 2a), the axis image is subjected to a dilation process. Subsequently, the feature detection process is performed again. We then determine the regions in which the density of the feature points is high (Fig. 2b circles), and the feature points extracted for the first time (Fig. 2a) within that region are defined as end points (Fig. 2c). If there are fewer than two feature points, the point on the axis closest to the end point of the previous frame is defined as the end point.

3.3 DETERMINATION OF CONTOUR POINTS

The end points e^1 and e^2 calculated in Section 3.2 are the midpoints of the contour points A, B and C, D (Fig. 3 left). Using A, B, C, and D, the contour is divided into four parts (Fig.3 left). Each of the divided contour lines is sampled at equal intervals. In this paper, the long contour lines (green and purple lines in Fig. 3 left) are divided into 30 parts, and the short contour lines (red and yellow lines in Fig. 3 left) are divided into 2 parts.

Using the above method, the locations of all contour points can be determined by tracking the two end points. The end points are tracked by comparing the distance between $e^1_{t-1} - e^1_t$ and $e^1_{t-1} - e^2_t$ and selecting the closer point as e^1_t and the other as e^2_t. Finally, a Kalman filter is applied to all of the detected contour points to smooth the motion. In addition, the target area is slightly reduced to prevent the end point from becoming undetectable when the head and tail of the silkworm overlap. This step reduces the final projected area; however, the effects of this step are usually negligible.

4 GENERATION OF EFFECTS

We produced three types of animation effects based on bioluminescent sea creatures. The first effect represents energy propagating through the body of the silkworm along its ridgeline based on the motif of *Cavernularia obesa* (Fig. 1g). In the initial frame, the projection area is specified, and a path along the ridge is indicated by dragging the mouse. Thereafter, particles are gradually displayed in sequence along the vertices of the input path to represent light propagation.

The second effect is an effect of light running along the contour line of a silkworm, based on the motif of *Beroe abyssicola* (Fig. 1h). To represent the rainbow-colored glow of fine cilia, we created a flickering light that continuously changes color.

The third effect displays lighting points on the entire body, which change according to the magnitude of movement of the silkworm, based on the motif of the *Sepioteuthis lessoniana*.The Sepioteuthis lessoniana changes the color of its entire body all at once when hunting or excited. To create this effect, a blue particle is displayed when there is little movement, presenting an image of stillness, and a red particle is displayed when there is movement, presenting an image of activity (Fig. 1i, j).

5 RESULTS AND CONCLUSION

We tested the tracking method in an experimental video of approximately 2 minutes. Fig. 3 shows the results obtained for every 30 seconds. These results show that the system can track the large deformations of the silkworm without losing its shape. As shown in Fig. 1, we projected three types of effects onto real silkworms. Overall, the tracking was good. In rare cases, there were gaps between the real and tracked contours and flickering in the image.

In this paper, we proposed an automatic tracking method for DPM for silkworms and applied this method to real silkworms. By extracting the axis, we created corresponding local deformations and reduced the number of points to be tracked. However, the frame rate of this method was approximately 13 fps, and the behavior of the silkworms was generally fine. In the future, it may be possible to apply DPM to multiple objects by increasing the speed.

REFERENCES

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

Tomoki Sueyoshi and Yuki Morimoto. 2019. Tangible Projection Mapping onto Deformable Moving Thin Plants via Markerless Tracking. In The Adjunct Publication of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19).

Amit H. Bermano, Markus Billeter, Daisuke Iwai. 2017. Anselm Grundhöfer, Makeup Lamps: Live Augmentation of Human Faces via Projection, 2017. Eurographics.

Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh. 2017. Realtime Multi- person 2D Pose Estimation Using Part Affinity Fields. IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17).

Tomoki Sueyoshi, Yuki Morimoto. 2018. Automatic Generation of Interactive Projection Mapping for Leaves, SIGGRAPH Asia 2018 Posters (SA '18).

Pablo F Alcantarilla, Jesús Nuevo, and Adrien Bartoli. 2013. Fast explicit diffusion for accelerated features in nonlinear scale spaces. In Proceedings British Machine Vision Conference 2013. Pages 13.1–13.11