

"Amazing Sketchbook Advance"

Shinji Mizuno

Aichi Institute of Technology, Japan

Kenji Funahashi

Nagoya Institute of Technology, Japan

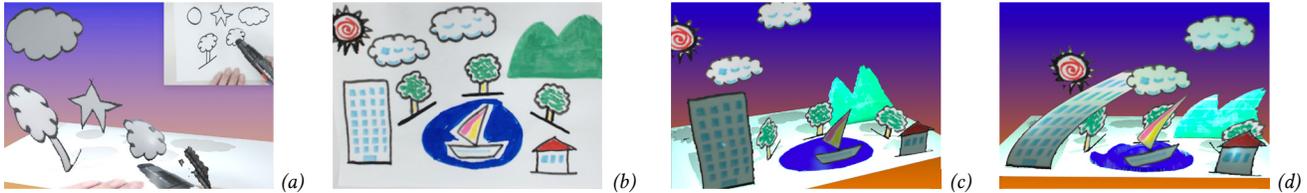


Figure 1: (a) A user just draws a picture on an ordinary sketchbook with ordinary color pens, and a video camera monitors the picture. (b) The picture in the sketchbook. A user adds black contour lines to hand-drawn objects to generate multi-color 3DCG objects from them. (c) A 3DCG scene with 3DCG objects is created in real time. (d) A user can deform and move 3DCG objects interactively by shaking the sketchbook or touching hand-drawn objects.

CCS CONCEPTS

• **Computing methodologies** → *Graphics input devices;*

KEYWORDS

3DCG, sketch based modeling, interaction

ACM Reference Format:

Shinji Mizuno and Kenji Funahashi. 2017. "Amazing Sketchbook Advance". In Proceedings of SIGGRAPH '17 Posters, Los Angeles, CA, USA, July 30 - August 03, 2017, 2 pages. <https://doi.org/10.1145/3102163.3102210>

1 INTRODUCTION

The authors have been developing an interactive CG creation system "Amazing Sketchbook" in which, a user can create a 3DCG scene composed of many 3DCG objects interactively by drawing a picture in an ordinary physical sketchbook with ordinary physical color pens, and can interact with the 3DCG scene by touching the picture or shaking the sketchbook [Kondo et al. 2013]. "Amazing Sketchbook" does not require any drafts to color in, and a user can create 3DCG scenes from arbitrary hand-drawn pictures interactively, unlike other systems with the same concepts [Clark et al. 2011][teamLab 2013].

The system monitors the sketchbook with a video camera and generates a 3DCG scene by analyzing a frame image of the video camera. The former system separated a picture in each frame image into regions based on colors, and generated 3DCG objects from each region. Therefore, each 3DCG object generated was a single color. In this research, we improve the system to generate a multi-color 3DCG object from a hand-drawn object. We also introduce a method to control a position of the 3DCG object by adding a simple line to the hand-drawn object. The user can generate and deform 3DCG objects interactively by drawing in and shaking the sketchbook like the former system.

e-mail: s_mizuno@aitech.ac.jp

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.

Copyright is held by the owner/author(s).
SIGGRAPH '17 Posters, July 30 - August 03, 2017, Los Angeles, CA, USA ACM 978-1-4503-5015-0/17/07.
10.1145/3102163.3102210

2 GENERATING 3DCG OBJECTS FROM HAND-DRAWN OBJECTS

The components and usages of the improved system are almost the same as those of the former system. A user draws a picture in a sketchbook and the system generates 3DCG objects from it.

The improved system separates the picture into regions based on not only colors but black contour lines. A user can generate a multi-color 3DCG object by drawing a black contour around an object (Fig. 1(b)). A user can also control the position of 3DCG objects in the scene by drawing a straight line under each object. We adopt the following steps experimentally to detect thin black lines and regions enclosed by black contours stably.

1. Converting the color of a frame image from to an HSVspace.
2. Applying adaptive thresholding to the V image to detect dark pixels, which might include not only black but also blue and green pixels (Fig. 2(a)).
3. Applying fixed thresholding to the S image to detect low saturation pixels, which might include not only black but also white pixels. Then applying erosion operation (Fig. 2(b)).
4. Calculating the product set of two images of Step 2 and 3, and filling regions enclosed by the pixels (Fig. 2(c)(d)).

Circumscribing rectangles of each region enclosed by a black contour are calculated (Fig. 3(a)). Then the system checks a black straight line using the Progressive Probabilistic Hough

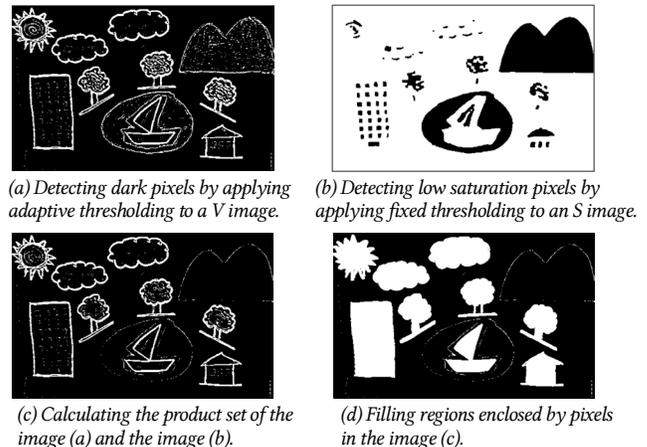


Figure 2: Detecting thin black lines and regions enclosed by black contours from a frame image.

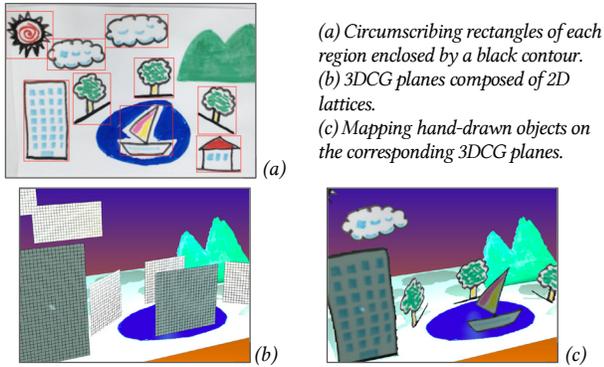


Figure 3: Generating multi-color 3DCG objects from hand-drawn objects.

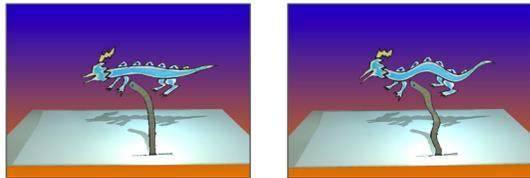


Figure 4: Interactive deformation of long narrow 3DCG objects.

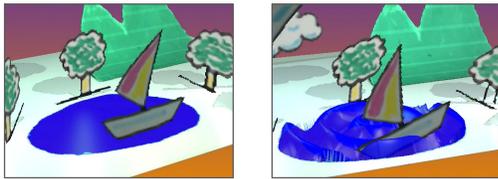


Figure 5: A 3DCG ship pitched by the waves of a 3DCG pond.

Transform in the lower part of the rectangles.

The system generates 3DCG planes composed of 2D lattices based on each circumscribing rectangle (Fig. 3(b)). The system trims hand-drawn objects from a frame image using the rectangles. Then each trimmed hand-drawn object is mapped on corresponding 3DCG plane while using the region enclosed by a black contour as a mask (Fig. 3(c)). As the result, multi-color 3DCG objects like stage settings are generated.

The position of each multi-color 3DCG object in the 3DCG space is decided automatically based on the position of the corresponding hand-drawn object in the sketchbook. A 3DCG object drawn at the upper part of the sketchbook is usually put above the ground. However, when a black straight additional line is detected under the hand-drawn object, the system regards the line as the ground, and the corresponding 3DCG object would be put on the ground in the 3DCG space. The direction of the black straight line is used to decide the positioning angle of the 3DCG object. In Fig. 3, each 3DCG tree is put on the ground with the angles of the line drawn under each hand-drawn tree.

After generating multi-color 3DCG objects from hand-drawn objects enclosed by black contour, the system generates single color objects in the same way as the former system. In Fig. 1, a green mountain and a blue pond are single-color 3DCG objects.

3 INTERACTION WITH A 3DCG SCENE

The system detects motion of a sketchbook using optical flow in the video camera. A user can interact with a 3DCG scene by shaking the sketchbook.

A multi-color 3DCG object would be deformed and move with the motion of the sketchbook. Each 3DCG object has a 2D lattice structure, and the deformation is realized by moving points of the lattice structure. The way of deformation and motion would change according to the position and the shape of an object. A 3DCG object put on the ground will swing at the same point. A 3DCG object put above the ground will fly around. A long narrow 3DCG object will move like a wave (Fig. 4).

A single color 3DCG object would be also deformed with the motion of the sketchbook in the same way as the former system, and it would affect the motion of neighboring multi-color 3DCG objects. For example, when a user draws a ship with a black contour inside a blue circle, a 3DCG ship would be put on a 3DCG pond. The motion of the sketchbook would make waves on the surface of the 3DCG pond, and the 3DCG ship would be pitched by the waves (Fig. 5).

4 EXPERIMENTS AND CONCLUSIONS

We implemented a prototype "Amazing Sketchbook Advance" and carried out a preliminary experiment. About 30 people between 5 to 25 years old experienced 3DCG creation and interaction with the system. Most of them were beginners of 3DCG, but they could create various 3DCG scenes easily and enjoyed our system very much. Fig. 6 shows examples of 3DCG scenes created by the system.

We improved "Amazing Sketchbook" to realize more various 3DCG scenes creation and interaction. More various motions of 3DCG objects based on detailed shape analysis is a future work.

REFERENCES

- KONDO, N., GOTO, S., MIZUNO, S., 2013, Amazing Sketchbook: extended drawing on a sketchbook using 3DCG, *SIGGRAPH 2013 Posters*, Article No. 17, <https://doi.org/10.1145/2503385.2503404>.
 CLARK, A., DUNSER, A., GRASSET, R., 2011, An Interactive Augmented Reality Coloring Book, *SIGGRAPH Asia 2011 Emerging Technology*, Article No. 25, <https://doi.org/10.1145/2073370.2073394>.
 teamLab, 2013, *Sketch Aquarium*, <https://www.teamlab.art/w/aquarium/>.



(a) A hand-drawn picture and the created 3DCG scene.



(b) A hand-drawn picture and the created 3DCG scene.

Figure 6: Examples of 3DCG creation from hand-drawn pictures.