

ThinkingSketch: A Tool for Artwork Creation

Yoshiaki MIMA, Ken-ichi KIMURA, Hidekatsu YANAGI
School of Systems Information Science
Future University - Hakodate
116-2 Kamedanakano
Hakodate, Hokkaido 041-8655 JAPAN
+81-138-34-6331
{mima, kimura, yanagi}@fun.ac.jp

ABSTRACT

ThinkingSketch is a drawing software, which enables user to create series of artworks with same design taste. The basic keyword for art work creation on this tool is "trace and copy." On this tool, user can generate graphic elements by tracing scanned artworks as seed objects for artworks. After making the elements then combine, transform, and copy them to canvas. This tool will help both beginners and professionals of visual art creation.

KEYWORDS: Art, Design, Interaction, Reflection

INTRODUCTION

ThinkingSketch is an object based two dimensional graphic software. On top of the basic graphic drawing engine, user can define object generation rules and a set of graphic objects as a seeds for graphic generation. Colors are also assigned by mapping from color palettes. Several positioning rules are provided as built-in tools and user tunes its parameters.

User's role is to find a good rules and seeds for generating new artworks. Once user finds the set of layouting rules, graphical data as seed object, and color mapping parameters, this software will generate pictures with same taste using randomly generated parameters (Figure 1).

This system is now available on internet[2].

APPLICATION

There are at least two possible types of users for ThinkingSketch. (1) Beginners of artworks, who are going to learn about art style. (2) Professional visual creators, who need supports on drawing creative works.

Application as an Education Tool

It is not easy to establish ones own artistic style from scratch. For the beginners, it is not easy only to be aware the existence

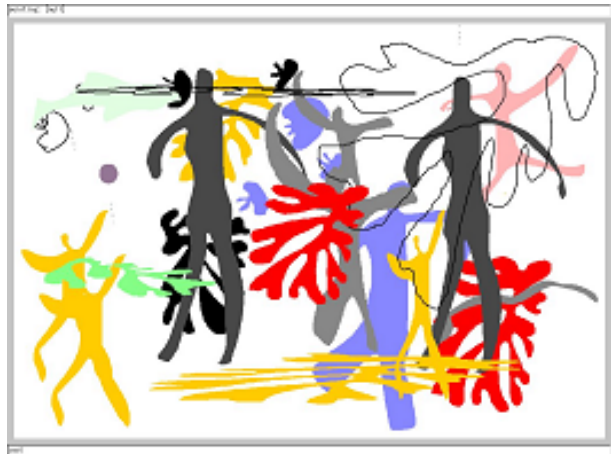


Figure 1: Example of Generated Picture

of styles. Our first development motivation for ThinkingSketch was to provide a quicker way to establish one's own style on visual expression.

The operation of defining rules, generating pictures, and reflection is a good experience of understanding what artistic styles or tastes are. Using this system, user can repeatedly create many pictures from the same settings and find they have their own style.

Application as a Creation Tool

Once user defined any style, ThinkingSketch can be used as an artwork generation tool. Even if the pictures requires complexed process for generation, this tool will automatically execute specified process.

For example, let's think about a case that a designer has an order of a series of book covers. This system can be helpfull for the designer. Once the designer could successfully define a generation rule as a combination of basic object set and a generation rule set, it will automatically generates many artworks as candidates. The designer's work is just pick up from the output and modify them in case the modification is effective.

BASIC OPERATION FOR GRAPHIC GENERATION

Drawing with ThinkingSketch is performed by generating objects using randomized parameters and applying them to generation rules. Thinking Sketch automatically generates such series of drawings. At the early stage of the tuning, user might not be satisfied with generated pictures. After several steps of parameter tuning, user will get to have preferable drawings. The stepwise instruction of making picture on ThinkingSketch is as follows.

Step 1: Creator specifies a set of primitive pictures. In current system, we can read scanned image of existing artworks (it can be a photograph) and then trace the shape on the picture. Then we will keep the traced or drawn data in the archives of this system. (Generation of primitive is not necessary because user can choose basic geometrical objects like lines, rectangles or circles and use them.)

Step 2: Creator defines a set of colors as color palettes. This color palette is used for object generation and used to specify coloring strategy.

Step 3: creator defines how to place objects on canvas. An object will be chosen from the object set that was generated in the first step. Then, following the automatic primitive placement procedure, the position of the object is specified with random numbers. Creator specifies this generation process to get the final process.

User can specify the procedure of generation as textaul command description. We call this as macro program specification.

EXAMPLES

Following is an example of picture generation macro 'igo.mac' (Figure 2). On this example, we do not specify originally defined figure as seed but use simple circle by adding grid feature as design constraint. What user specifies here is 'both height and width to be 50', 'specify grid as 50', and 'draw 25 times filled circles and draw 25 times circle outline.'

```
// sample macro 'igo.mac'
// specify object size
wmax 50
wmin 50
hmax 50
hmin 50
// grid
gridw 50
gridh 50
// number of objects to gen.
repeat 25
// object generation
fill fill
*bubble
fill outline
*bubble
```

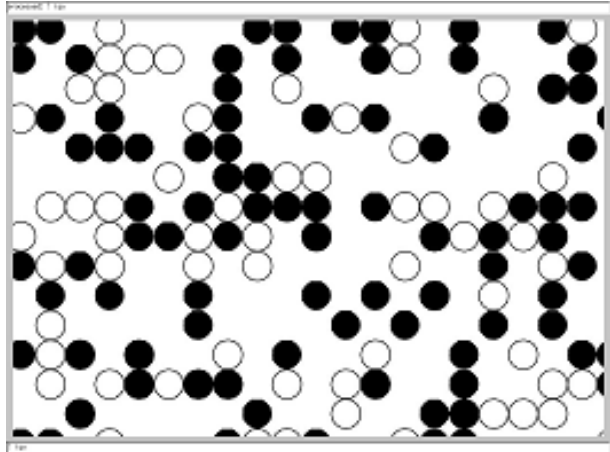


Figure 2: A Pattern Igo

RELATED WORKS

Harold Cohen's AARON[1] is a famous autonomous drawing system, which generates original artworks. Basically AARON is a rule based system, which has knowledge of objects such as human, plant, stone, etc. AARON draws objects one by one by checking the status of the pictures drawn on canvas image. It generates pictures swiftly and draws them.

Difference between AARON and ThinkingSketch is as follows. AARON is a program that is tuned up well and realizing a style of a high level artist. "ThinkingSketch" is designed as a tool for finding new style/taste of artwork generation with frequent interaction between human and computer by checking artworks as its output.

ACKNOWLEDGMENTS

For the development of "ThinkingSketch" project had been supported from Exploratory Software Project of IPA. Especially, Prof. Ikuo Takeuchi gave us useful comments and important suggestions on our research direction. We also thanks to Prof. Harold Cohen, though who have disagreements about many of the underlying concepts in this paper from him, for explaining his system AARON and giving us important suggestions for designing ThinkingSketch at the very early stage.

REFERENCES

1. Pamela, McCorduck, *Aaron's Code*, W. H. Freeman and Company, New York, 1991.
2. ThinkingSketch Unit, <http://www.sketch.jp/>