Constituting Origami Models from Sketches

Hiroshi Shimanuki Jien Kato Toyohide Watanabe
Department of Systems and Social Informatics,
Graduate School of Information Science, Nagoya University, Japan
simanuki@watanabe.ss.is.nagoya-u.ac.jp {jien, watanabe}@is.nagoya-u.ac.jp

Abstract

This paper proposes an approach to constituting craese patterns, the unfolded origami models, by using skeletons obtained from 2-D images such as handwriting sketches. Firstly, we describe a method for constructing a data structure which represents all the parts of an origami model and their relationships based on an extracted skeleton, and then give an algorithm for constituting a crease pattern using this data structure. To show the validity of proposed method and algorithm, we finally demonstrate how a crease pattern is generated and how a origami model is actually realized using some real illustrations taken from a origami drill book.

1. Introduction

Origami is Japanese traditional paper play which forms various handiworks from square papers. The pleasure and beauty of origami are understood by people in the world now. Additionally, very many origami works are announced by the origami creator in origami drill books, web pages, and so on. Illustrations of origami drill books are drawn to be easy to understand folding operations for readers. When readers want to fold a certain work, they must look for the illustration from the origami drill books which they own, or must search for the illustration from Web. However, there is sometimes no origami work of the kind and the shape which the reader aims at. In such a case, because the reader must purchases origami drill books which the illustration appears or must look it up on the Internet, great labor and a burden are assigned to the reader.

Therefore, we propose a method for constituting origami models from 2-D images such handwritten sketches. When there is no illustration which users want to fold, origami models are constituted from images which users input or draw. It becomes possible to constitute origami models which users intend at any time.

As the international meeting of origami science has been held three times until now, researches related with origami are conducted in various field. In the field of the design of origami, the methods which constitute complete crease patterns from tree graphs which represent characteristics of folding objects are proposed [1, 3]. However, the most is calculated by using compasses and straight edges on the desk and are dependent on human senses. In such researches, [3] proposes an approach to cpnstituting crease patterns on computers, formalizing origami design gives a great influence in the field. However, it is the application for the designers who know the origami design well by inputting a tree graph and it burdens with users.

This paper describes a method for automatically constituting complete crease patterns from handwriting sketches.

2. Approach

The outline of the process is shown in Fig. 1. The first, a data structure, called a skeleton graph, of an object is extracted from a sketch given as input. The Second, the skeleton graph is arranged onto a square, and polygons are constituted in the square. The Third, the polygons are divided into origami molecules. Finally, creases are created based on the origami molecules and the complete crease pattern are constituted.

Skeleton graphs are constructed by segmented the skeleton images of sketches. Figure 2 shows an example of extracting a skeleton image from a sketch and constructing a skeleton graph. the skeleton graph is defined as following.

$$\begin{aligned} \mathcal{SG} &= (\mathcal{V}, \mathcal{E}) \\ \mathcal{V} &= \{v_i \mid 1 \le i \le N\} \\ \mathcal{E} &= \{e_{ij} = (v_i, v_j) \mid v_i, v_j \in \mathcal{V}\} \\ \mathcal{X} &= \{x_i\}_{v_i \in \mathcal{V}} \end{aligned}$$

$\mathcal{SG}$ represents skeleton graph, which is a pair that consists of vertices and edges. Moreover, the vertices $\mathcal{V}$ have their coordinates $\mathcal{X}$ of N points extracted from the skeleton image. $\mathcal{SG}$ constituted from the image thinned the silhouette image must be a tree graph, which is a connect graph without cycles.

0-7695-2128-2/04 $20.00 (C) 2004 IEEE

The skeleton graph of Fig. 2 has symmetry. In the field of origami desin, symmetrical origami models are generally desined, e.g. a living thing. Therefore, this skeleton graph is detected the axis of symmetry whose nodes are expressed with black dots "•", and the other nodes are expressed with white circles "o".

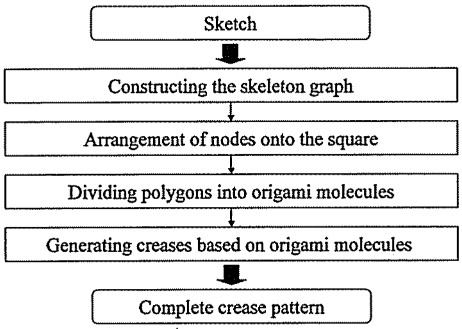


Figure 1. The process of origami design.

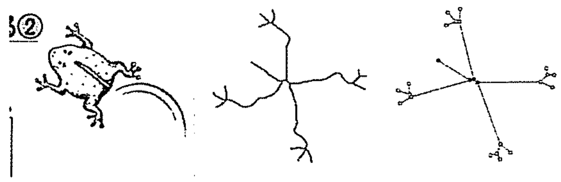


Figure 2. An example of extracting a silhouette, a skeleton and a skeleton graph.

2.1. Origami design

Origami models are designed based on *origami molecules*. origami molecules are defined as minimum units which have a certain meaning and a set of faces in origami. Although there are many types of molecules, the most general one is shown in Fig. 3. The left figure represents a triangular molecule which has creases of three bisectors of each corner and perpendiculars from the centre of gravity of the triangle to each edge. The right figure is the origami model that is folded the triangular molecule in practice. A crease pattern is constituted so that faces in the square are origami molecules.

A *flap* is one of the most important concept in origami degin. We explain this using Fig. 3. The flap means the part of origami model which is represented as "circle", and the centre of flap represents the apex of the part (A, B and C). The circles mean regions that are necessary to constitute parts of origami model. Moreover, the tree graph is 2-D projection of the origami model and terminal nodes of the tree graph are corresponding to centre points of flaps. In this paper, the crease pattern is constituted from the skeleton graph as the tree graph.

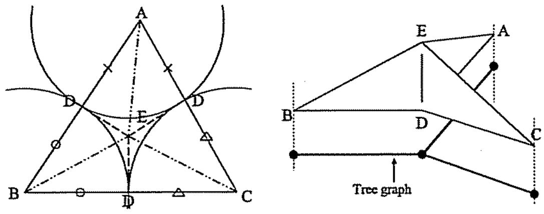


Figure 3. An example of an triangular origami molecule called "rabbit ear".

3. Arrangement of nodes onto a square

When terminal nodes of a skeleton graph are arranged onto a square, and each flap, i.e. the circle, must not be intersected each other on the square. Because flaps are necessary to constitute parts of the origami model. This condition can be formalized as following.

If there are two terminal nodes v_i, v_j in the skeleton graph, the path length between v_i and v_j is defined as l_{ij} and vertices on the square of nodes v_i, v_j are represent as u_i, u_j, then

$$|u_i - u_j| \geq l_{ij}.$$

Figure 4 shows an exmaple of arranging nodes of a skeleton graph. The skeleton graph in Fig. 4(a) consists of edges whose length is 1. Nodes of the graph are arranged as Fig. 4(b). The path length between the nodes AB becomes 2, and a distance between the vertices in the square is 2, too. Moreover, the path length between the nodes CF becomes 3, but a distance between the vertices in the square is larger than it. When equal signes of above inequality stands up, the line, called *active path*, is drawn in square.

The node arrangement optimization method proposed by Lang [3] is used. The algorithm is explained in the following.

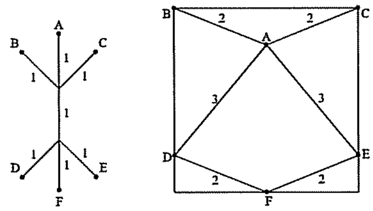
A square whose each edge length is 1 is allocated from 0 to 1 in xy coordinate system. In the example in Fig. 4, the size of the square is elastic corresponding to the skeleton graph. However, in this algorithm, square size is fixed and the scale of the skeleton graph makes m. Then an optimally efficient square is found by maximizing m over all u_i subject to the constraints:

1. $\sqrt{(u_{i,x} - u_{j,x})^2 + (u_{i,y} - u_{j,y})^2} \geq m\, l_{ij}$ for all i, j
2. $0 \leq x_i \leq 1, \quad 0 \leq y_i \leq 1$ for all i.

This nonlinear constrainted optimization is solved using the well-known Augmented Lagrangian Multiplier algorithm (ALM).

4. Dividing polygons into origami molecules

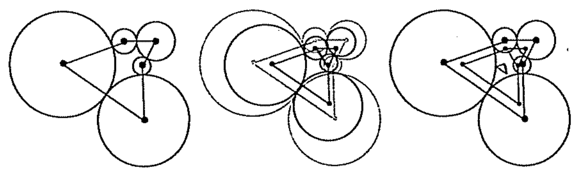
A square arranged nodes of a skeleton graph consists of polygons divided by active paths. In this section, we de-



(a) A skeleton graph.

(b) Arrangement of the skeleton graph onto a square of origami.

Figure 4. An exmaple of arranging a skeleton graph.



(a) A circle diagram of a polygon.

(b) Reducing the polygon while contacting with other polygons.

(c) Devided polygons.

Figure 5. A general idea of the universal molecule algorithm.

scribe a method for dividing these polygons into origami molecules using the universal molecule algorithm proposed by Lang [3].

The universal molecule algorithm is explained conceptually by using Fig. 5. Figure 5(a) shows a certain polygon divided by active paths and flaps on vertices of the polygon, which are expessed as circles. First, the polygon is shrinked until two circles come in contact with each other. Next, when circles come in contact with each other, center points of the circles are connected with a line. The polygon is divided into a triangle and a quadrangle in Fig. 5(b). The result of similar processing for this quadrangle is shown in Fig. 5(c). Finally, this quadrangle is divided into two triangles, three triangles are constituted. These triangles are origami molecules.

Therefore, bisectors on corners of the polygon are constituted as Fig. 6, and the polygon is shrinked in the distance h. The path length l_{ij} is also reduced as the polygon is shrinked. If the reduced path length is represented as l'_{ij},

$$l'_{ij} = l_{ij} - h(\cot\alpha_i + \cot\alpha_j).$$

Shrinkage for all polygons in the square is repeated until one or both of the following conditions will hold.

1. For two adjacent corners, the reduced path length l'_{ij} has fallen to zero and the two inset corners are degenerate; or

2. For two nonadjacent corners, $l'_{ij} = |u'_i - u'_j|$.

When the condition of 2. is satisfied, the edge between u'_i and u'_j is constituted and the polygon is divided. Moreover, in case of a triangle, shrinkage is unnecessary because bisectors of three corners cross at one point, the point of intersection of bisectors is only memorized.

Figure 7(a) shows the result of applying to crease pattern in Fig. 4(b). In this figure, divided polygons are represented as thick lines and the points of intersection of bisectors on corners of triangles are represented as black dots.

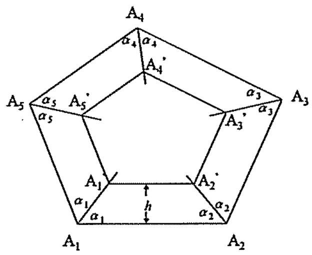


Figure 6. An example of reducing a polygon.

5. Creating creases based on origami molecules

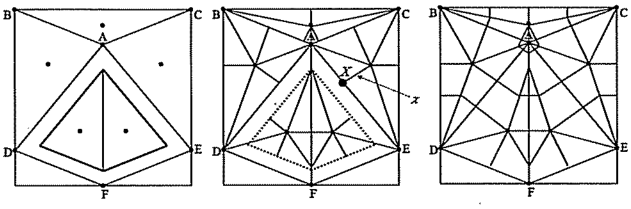
creases are created in the crease pattern based on the above-mentioned divided polygons. The created creases have relationship of creases of origami molecules described in section 2.1. Figure 7(b) shows an example of creating creases. However, this crease pattern in incomplete. Because creases are not created for some faces of this crease pattern, this crease pattern is impossible to fold.

There are various researches mentioned about *foldability* [2]. These researches judge the possibility of folding a given crease pattern. Moreover, we have already proposed a method for constituting complete crease patterns from incomplete ones which can not fold at local vertices in crease patterns [4]. This method creates the crease of symmetry about a exsiting crease. Figure 7(c) shows the result of creating using this method.

6. Experimental results

A sketch is chosen arbitrarily and skeleton graph is constituted. Moreover, the crease pattern is designed using the proposed origami design method from the extracted skeleton. Finally, based on the crease pattern, the origami model is constituted manually.

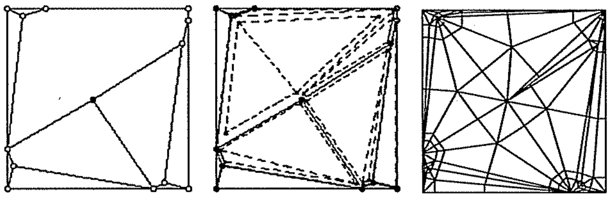
Figure 8 shows the result of constituting the origami model from the skeleton graph in Fig. 2. The nodes are



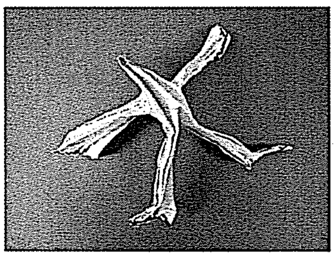
(a) (b) (c)

Figure 7. An example of generating creases.

arranged in the optimal position onto a square (Fig. 8(a)), polygons are divided into origami molecules (Fig. 8(b)) and creases are created (Fig. 8(c)). Finally, origami is actually constituted based on the crease pattern (Fig. 8(d)).



(a) Arranging onto a square of origami. (b) Dividing into origami molecules. (c) A complete crease pattern.



(d) A constructed origami model.

Figure 8. An result of constructing origami models.

7. Discussion

Recently, various origami works by designing based on origami molecules are announced, we often experience the origami like a puzzle such as "Try folding a crease pattern!". However, the mastered technique is necessary to actually fold origami from such crease patterns.

Origami models which are realized from crease patterns constituted by the proposed method have fundamental 3-D forms. However, these crease patterns are confined to constituting each part of objects, and it is infeasible to realize the details. In the field of origami design, final design is depend on a designer's sense. Figure 9 shows the result of modifing the origami model in Fig. 8(d) by authors' subjectivity.

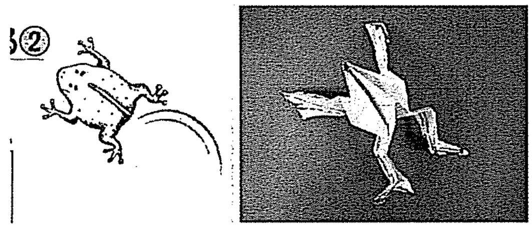


Figure 9. An example of modifing the origami model in Fig. 8(d).

8. Conclusion

This paper proposes the method for constituting craese patterns by using skeletons obtained from 2-D images such as handwriting sketches. The framework which realizes origami models from the skeleton graph is explained, which is possible to constitute crerase pattens automatically from sketches.

Although a sequence of processing is performed on the assumption that skeleton graphs are symmetry, this method can not be applied to asymmetrical objects. Moreover, we devise a method for creating folding operations from the constituted crease patterns.

References

[1] F. Kawahata. "THE TECHNIQUE TO FOLD FREE FLAPS OF FORMATIVE ART "ORIGAMI". In K. Miura, editor, *Origami Science and Art: Proc. of the Second International Meeting of Origami Science and Scientific Origami*, pages 63–72, 1994.

[2] T. Kawasaki. "On the Relation Between Mountain-creases and Valley-creases of a Flat Origami". In H. Huzita, editor, *Proc. of the First International Conference on Origami in Education and Therapy (COET91)*, pages 229–237. British Origami Society, 1991.

[3] R. J. Lang. "A Computational Algorithm for Origami Design". In *Proc. of the 12th Annual ACM Symposium on Computational Geometry*, pages 98–105, 1996.

[4] H. Shimanuki, J. Kato, and T. Watanabe. "Constituting Feasible Folding Operations Using Incomplete Crease Information". In *Proc. of IAPR Workshop on Machine Vision Applications*, pages 68–71, 2002.