MVA2002 IAPR Workshop on Machine Vision Applications, Dec. 11 - 13, 2002, Nara- ken New Public Hall, Nara, Japan

3—8

Constituting Feasible Folding Operation Using Incomplete Crease Information

Hiroshi Shimanuki* Jien Kato† Toyohide Watanabe‡
Graduate School of Engineering, Nagoya University§

Abstract

This paper proposes a novel approach to constituting all the feasible ways of folding, based on *crease information* obtained from an image of illustrations of general origami drill books. Since crease information from 2D plane figures is superficial and incomplete, how to only generate all the feasible ways is a problem. This paper deals with this subject.

Origami operations can be classified into basic operations and complex operations. For the basic ones, we propose a method that can create creases corresponding to the operations on a sheet of extended paper called unfolded plan, and an algorithm, called origami-section method, that is used to test physical feasibility of basic operations. For the complex ones, we propose several algorithms to produce correct creases for each operation, which keep the consistency of the crease patterns under some geometrical constrains. Some experimental results are given to show the practical efficacy of the proposed methods.

1 Introduction

As international conferences on origami science has been held three times, researches related with origami are conducted in various fields such as mathematics, engineering and art. Most of the researches are done by mathematicians. They attempt to elucidate geometrical properties of origami by use of mathematical methods [1]. On the other hand, the study described in this paper is to generate feasible ways of folding stepwise during a virtual process of paper folding, based on a sequence of illustrations of origami drill books. It aims at constructing an interface to transform the 2D illustrations into 3D animation automatically. So, it is a completely new challenge.

We have developed a recognition system that successfully extracts edges of origami and crease information from images of the illustrations [2]. Crease information includes both of the position and direction of a folding operation. To realize our objective by use of the crease information, there are three difficulties here. First, generating a folding operation means to create all the resulting creases on faces of an origami model in 3D virtual space. Obviously, the crease information obtained from 2D plane figures (illustrations) is superficial and incomplete, and not enough for what we want to do. Second, given the incomplete crease information, many interpretations about the way of folding can be made. As the result, the creases cannot be determined uniquely. Finally, some methods which exclude infeasible ways of folding are needed. To deal with these problems, we apply some geometric constrains existing among creases to limit candidate crease patterns only to those that correspond with feasible folding operations.

*E-mail: simanuki@watanabe.nuie.nagoya-u.ac.jp
†E-mail: jien@watanabe.nuie.nagoya-u.ac.jp
‡E-mail: watanabe@nuie.nagoya-u.ac.jp
§Address: Furo-cho, Chigusa-ku, Nagoya, Aichi, 464-8603, Japan.

2 Definitions and Constrains

Generation of creases is performed through a sheet of extended paper called ***unfolded plan***. Using unfolded plans makes it possible to convert the problem of generating a folding operation against a 3D origami model into the problem of generating the creases on a 2D plane. All the algorithms we describe in the following sections are based on such an unfolded plan. Before we introduce them, we first give some necessary definitions and constrains below.

Folding operations of origami can be classified into basic operations and complex operations. Basic operations only consist of mountain and valley folding, and the resulting creases have uniform attributes (mountain or valley). On the other hand, complex operations mainly include tucking in, covering and expanding, and produce the creases with mixed attributes (mountain and valley).

When an operation is applied on a face of origami, the moving portion of the face is usually folded up (rotate 180 degrees around the crease). That leads to a restriction called flatness of the origami model. This restriction requires that during a folding process all the faces of the origami model must be in parallel. Necessary and sufficient conditions for the creases connected with an inner point of unfolded plan to keep an origami model flat are as follows.

[Theorem] Local flatness conditions [1]

i) The number of creases is even number.

ii) $|N_M - N_V| = 2$

N_M : the number of mountain creases

N_V : the number of valley creases

iii) The alternate sum of the angle among each crease makes 180 degrees.

iv) If the angle among adjacent two creases is obtuse, these creases' attribute (mountain/valley) is equal.

Since the creases generated on unfolded plan have to meet above theorem, these conditions are used in our proposed methods.

Finally, we give the definition about origami-symmetry, a concept that will be frequently used in our algorithms, below.

[Definition] Origami-symmetry

In creases patterns in unfolded plans, if two creases C_1 , C_2 which make the mutually equal angle to certain crease C_0 exist, C_1, C_2 are *origami-symmetry* about C_0. Then, the attribute or length of creases are not considered.

This definition actually is based on local flatness conditions.

3 Calculation Methods of Folding Operations

3.1 Basic Folding Operations

Some geometric rules exist among creases. Therefore, it is possible to calculate using the following characteristcs.

i) Each terminal point of generated creases exists on the outside edges or shares a terminal point with the other generated crease.

ii) Two generated creases which share a terminal point are symmetrical to a certain existing crease and have different attributes (Mountain/Valley).

3.1.1 The Algorithm of Generating Creases

The position of a crease is determined on a face specified by crease information. However, this crease is imperfect. Because some creases must be generated when two or more faces are folded. Then, the algorithm which generates suitable creases is shown below.

Creases generation algorithm

$C = \{c_i\}$: the generated creases

$P = \{p_m\}$, $Q = \{q_n\}$: sets of terminal points of creases

(STEP 1) $i \leftarrow 0, m \leftarrow 0, n \leftarrow 0, C = \{\}$.
A crease c_0 is generated to the face specified in the input. Two terminal points of c_0 are p_0 and q_0.

(STEP 2) If both p_m and q_n are on the outside edges, C is outputed, $\Rightarrow$ Stopped.

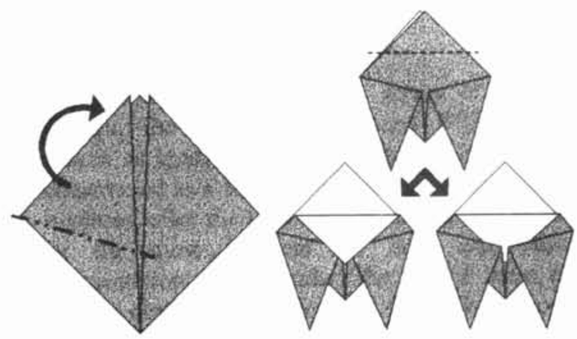
(STEP 3) If p_m is on the crease e_i, the crease c_{i+1} is generated symmetrically to e_i. Then, the attribute (Mountain/Valley) of c_{i+1} is antithetical to c_i. Moreover, the terminal point other than p_m of C_{i+1} is p_{m+1}. $i \leftarrow i+1$, $m \leftarrow m+1$, $C \leftarrow C \cup \{c_i\}$.

(STEP 4) If q_n is on the crease e_i, the crease c_{i+1} is generated symmetrically to e_i. Then, the attribute (Mountain/Valley) of c_{i+1} is antithetical to c_i. Moreover, the terminal point other than q_n of C_{i+1} is q_{n+1}. $i \leftarrow i+1, n \leftarrow n+1, C \leftarrow C \cup \{c_i\}$.
$\Rightarrow$ (STEP 2).

By using this algorithm, consistency of creases can be supported.

3.1.2 Origami-section Method

Some problems are mentioned in case a basic folding operation is calculated. Fig.1(a) is shown the way of folding that is impossible physically, because the moved face run into other faces. Moreover, since crease information is command only to fold by the crease specified in the input, there are many considered ways of folding. In Fig.1(b), many ways of folding occur because of the defference of the interfolded position of a face. In order to solve these problems, all the feasible ways of folding must be consituted without constituting the impossible ways of folding. Therefore, we propose the calculation method using sections of origami. The section of origami is the section which cut the origami model perpendicularly to the generated crease. A section is expressed as a set of line segments.



(a) Impossible way of folding physically

(b) Many considered ways of folding

Figure 1: Uncertainty of a basic folding operation

The acquisition algorithm of origami section

$C = \{c_i\}$: a set of creases which cross line segments of a section

$S = \{s_i\}$: a set of line segments of a section

$P = \{p_i\}$: a set of terminal points of line segments

O : the existing crease which the symmetrical axis to the genarated creases

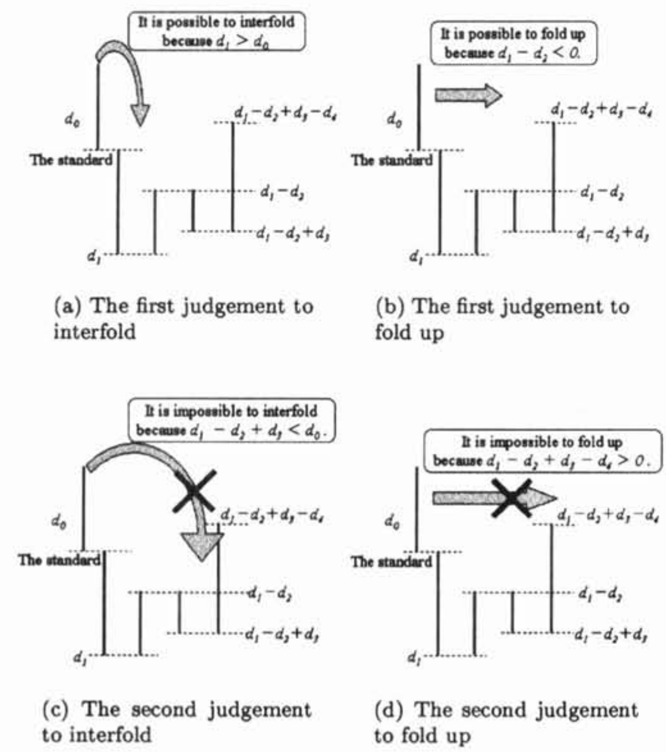
(STEP 1) Two segments perpendicular to the generated crease c_0 are generated. The segment which has terminal points on the outside edges or O is s_0 and another is s_1. The terminal point of s_1 on c_0 is p_0 and the terminal point of s_1 on c_1 is p_1. $S \leftarrow \{s_0, s_1\}$, $P \leftarrow \{p_0, p_1\}$, $i \leftarrow 1$.

(STEP 2) If p_i is on the outside edges or on O, $\Rightarrow$ Stopped.

(STEP 3) If p_i is on the crease c_i, s_{i+1} that is symmetrical with s_i to c_i is generated. Then, the terminal point which is not p_i of s_{i+1} is p_{i+1}. $S \leftarrow S \cup \{s_{i+1}\}$, $P \leftarrow P \cup \{p_{i+1}\}$, $i \leftarrow i+1$
$\Rightarrow$ (STEP 2).

Thereby, the section of origami is obtained as a set of line segments S. Several kinds of such sections are acquired at equal intervals to all generated creases. The physical folding possibility is judged based on these sections. The judgment algorithm is explained using

Fig.2. First, it investigates whether the moved face can be interfolded into the hithermost "trench" (a valley between a face and a face) in Fig.2(a). In this case, since the length of the section of the moved face is shorter than the depth of the trench ($d_1 > d_0$), it is possible to interfold. Next, it investigates whether the moved face can be folded up to the following "wall" (a mountain between a face and a face) in Fig.2(b). In this case, since the height of the wall is below a standard based on the generated crease ($d_1 - d_2 < 0$), it is possible to fold up. In the same way, when it investigates the second trench, the length of the section of the moved face is longer than the depth of the trench ($d_1 - d_2 + d_3 < d_0$) in Fig.2(c). Therefore, the moved face can not be interfolded into this trench. Moreover, when it investigates the second wall, the height of the wall is above a standard ($d_1 - d_2 + d_3 - d_4 > 0$) in Fig.2(d). Therefore, it is impossible to fold up to the face after this wall, so this algorithm is stopped.



(a) The first judgement to interfold

(b) The first judgement to fold up

(c) The second judgement to interfold

(d) The second judgement to fold up

Figure 2: Examples of finding the possible ways of folding

The specific algorithm is shown below.

The judgment algorithm of the folding possibility

$S = \{s_i\}$: a set of line segments of a section

d_i : the length of s_i

$C = \{c_i\}$: a set of the creases which cross line segments of a section

a_i : the digitized attribute of c_i

$$a_i = \begin{cases} 1, & \text{when the attribute of } c_i \text{ is Mountain.} \\ -1, & \text{when the attribute of } c_i \text{ is Valley.} \end{cases}$$

(STEP 1) $i \leftarrow 1$.

(STEP 2) Judgement in the following two cases.

i) If $\sum_{k=0}^{i} a_k = 2a_0$ and $\sum_{k=1}^{i} a_0 a_k d_k < d_0$, it is **impossible** to fold to the face containing s_i.

ii) If $\sum_{k=0}^{i} a_k = a_0$ and $\sum_{k=1}^{i} a_0 a_k d_k > 0$, it is **impossible** to fold to the face after the face containing s_i, ⇒ stopped.

(STEP 3) If all line segments are scaned,⇒ stopped.

(STEP 4) $i \leftarrow i + 1$.
⇒ (STEP 2).

Some ways of folding with the faces judged "foldable" are constituted and outputted. a series of processes is named *origami-section method.*

3.2 Complex Folding Operations

3.2.1 Tucking in and Covering

In tucking in and covering, there are common characteristics and antithetical characteristics. They are summarized to below.

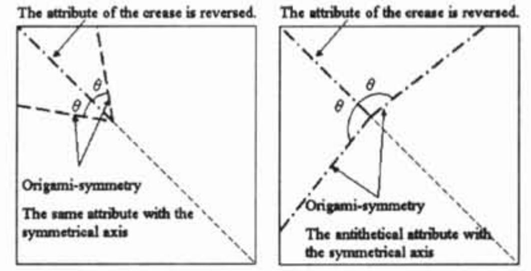
Common characteristics

- Generated creases are symmetrical.
- Some crease's attributes of the origami symmetrical axis are reversed.

Differences

Tucking in	Covering
Creases which have **same** attribute to the symmetry axis are generated	Creases which have **antithetical** attribute to the symmetry axis are generated
Acute angle is made to the reversed crease	**Obtuse angle** is made to the reversed crease

The calculation method is explained using Fig.3. The first, the symmetrical crease is generated like a basic folding operation. The next, with an acute angle in case of tucking in or with an obtuse angle in case of covering, the crease is generated, and the attribute of a part of the symmetrical axis between generated creases is reversed. Crease patterns are called the basic crease pattern of tucking in or covering, respectively.



(a) Basic crease pattern of tucking in

(b) Basic crease pattern of covering

Figure 3: Basic crease pattern of complex folding operations

3.2.2 Expanding

The calculation method of expanding is more complicated than tucking in and covering. Because the crease pattern of expanding is not expressed with only one clearly defined pattern. The basic crease pattern is shown in Fig.4. However, a origami model cannot be flat folded by using only the basic crease pattern consisting of three creases. Then, the algorithm of detecting another crease using local flatness conditions is proposed.

The algorithm of detecting a consistent crease
A crease is detected on the intersection of three creases of the basic crease pattern so that a origami model becomes flat.

Attr : the crease's attribute

$Attr = Mountain\ or\ Valley$

(STEP 1) N_M and N_V are the number of mountain and valley creases, respectively. In the crease pattern of expanding,

i) If $N_M > N_V$, $Attr = Mountain$.

ii) If $N_M < N_V$, $Attr = Valley$.

(STEP 2) It investigates whether the crease whose attribute is *Attr* and is not the crease generated by expanding is eliminable. If it is eliminable, *Attr* of the crease updates into "opening", ⇒ stopped.

(STEP 3) For all angles other than the angle among creases generated by expanding, It investigates whether the crease whose attribute is $\overline{Attr}$ can be inserted within each angle. If it is possible to insert, the crease whose attribute is $\overline{Attr}$ is inserted, ⇒ stopped.

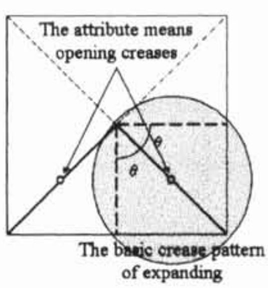


Figure 4: The basic crease pattern of expanding

4 Experimental results

With the help of an unfolded plan, we simulate all the ways of folding specified by incomplete crease information, and only constitute all the feasible folding operations using the methods described above. In this section, we present some results when arbitrary crease information is provided for the input.

Figure 5 shows the present state of the origami model that consists of two triangle-faces and a new crease on them. The creases automatically generated on unfolded plans are shown in Fig.6 and the resulting origami models in 3D virtual space in Fig.7. Two kinds of basic folding "mountain folding" and "valley folding" and two kinds of complex folding "tucking in" and "covering" are simulated. It can be observed that all and only the feasible ways of folding are generated as expected. For mountain folding and valley folding, two ways are constituted while the different portions of the faces are used as moved faces. Although the two operations are logically equivalent, they should be regarded as different operations in origami. Using our methods gives the correct results.

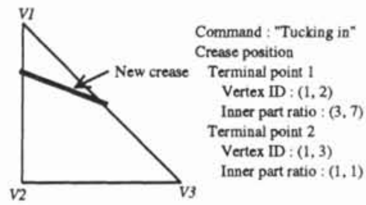


Figure 5: Experimental input

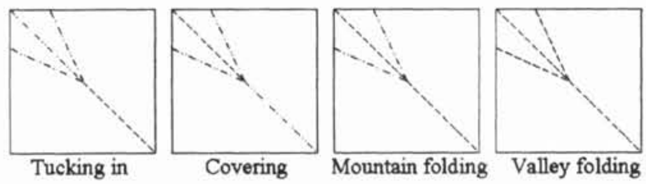


Figure 6: The calculation result on unfolded plans

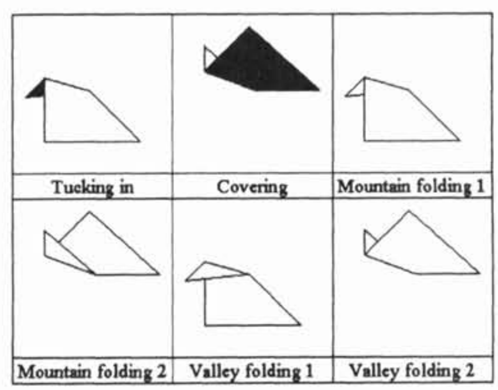


Figure 7: The resulting origami models in 3-D virtual space

5 Conclusions

Our present work has demonstrated that it is possible to generate feasible folding operations only based on incomplete crease information using proposed methods. As one of the future subjects, it is necessary to verify the validness of the algorithms under more complex states of the origami model.

References

[1] Kawasaki, T.: On the Relation Between Mountain-creases and Valley-creases of a Flat Origami, In *Proceedings of the First International Meeting of Origami Science and Technology*, Huzita, H. ed., pp.229-237 (1991).

[2] Kato, J., Watanabe, T., Hase, H. and Nakayama, T.: Understanding Illustrations of Origami Drill Books, *J. IPS Japan*, Vol.41, No.6, pp.1857-1873 (2000).

[3] Miyazaki, S, Yasuda, T, Yokoi, S, and Toriwaki, J: INTERACTIVE MANIPULATION OF ORIGAMI IN 3D VIRTUAL SPACE, *J. IPS Japan*, Vol.34, No.9, pp.1994-2001 (1993).