MVA2005 IAPR Conference on Machine VIsion Applications, May 16-18, 2005 Tsukuba Science City, Japan
8-9

Understanding Folding Process of Origami Drill Books Based on Graphic Matching

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

Abstract

This paper introduces an approach for recognizing illustrations of origami drill books with a view to understanding folding process of origami. For this purpose, comparing and matching two kinds of graphic forms are required. One is a graph generated from an origami illustration image by applying image processing techniques to it. The other is that obtained from an internal model which is produced and updated dynamically according to the results of understanding of the folding process stepwise. In order to comparing and matching these graphics, we propose a method for extracting USGs from origami illustration images. This method is based on Hough transform and takes into account the characteristics of origami illustration images. We also propose a method for matching two kinds of graphic forms based on a genetic algorithm. Experiments show that our methods are effective in recognizing illustration images including a lot of noise.

1 Introduction

Origami, one of the Japanese traditional cultures, is perceived worldwide as the art of paper folding which has the abundant potential globally. Traditionally, people play origami based on drill books or materials on web pages in which the folding processes consist of simple folding operations are illustrated by diagrams.

However, as origami works are complicated, it is usually the case that people give up obeying the instructions of illustrations because there are hidden portions of origami in 2-D diagrams. In order to solve this problem, 3-D animation system which represents the folding process from all angles is required. As related work, a system that represents dialogical operations of origami in 3-D space has been introduced [1]. The research described in this paper tries to develop a fully automatic interface that represents the folding process automatically from the illustrations of origami drill books.

This paper first outlines this system and indicates the necessity for matching two kinds of graphic forms in Section 2. Then, Section 3 describes our method for generating one kind of graphic forms that express the state of origami from an origami illustration image. Section 4 presents the matching method based on a genetic algorithm. Finally, we show the effectiveness of our approach by experiments in Section 5 and give the conclusions in Section 6, respectively.

2 Approach

2.1 Definitions of Graphic Forms

This section describes some definitions and characteristics of graphic forms used in our recognition system.

Drill Book Image. This is a two-facing-page image scanned from an origami drill book with 256 gray scale level, which illustrates a folding process (a sequence of folding operation) in number order. It consists of some origami illustrations, explanatory texts and numbers and so on. The explanatory texts and numbers are attached to the illustrations.

Origami Illustration Image. This image is extracted from a drill book image and illustrates the state of origami and one folding operation to be performed. In concrete terms, it consists of a diagram which illustrates the state of origami after the latest folding operation, an arrow which shows the direction of the present folding operation, and a folding line which gives the information of position and the type of the folding operation. The notable feature of this image is having a part of face's overlapping to assist users in understanding the origami state. This part becomes noise in image processing. The input of the method proposed in this paper is an origami illustration image.

Unrefined Shape Graph (USG). A USG is defined as a graph generated from an origamai illustration image by applying image processing techniques. It consists of line segments corresponding to edges of origami and their intersections (called feature points).

Ideal Shape Graph (ISG). An ISG is generated by projecting the internal model, a data structure which maintains 3-D information about changes of origami states step by step, onto a 2-D plane from the same viewpoint as that in the next illustration. The system compare and match the USG with one or more possible ISGs to determine the correct folding operation at each step (see Section 2.2 for details).

2.2 System for Recognition of Folding Process

This section outlines the system for recognition of folding process. Figure 1 shows an example of a USG and an ISG.

The system firstly extracts crease information, both positions and types of the creases, from an origami illustration image [2]. Then, the feasible (physically possible) folding operations are constructed based on the crease information.

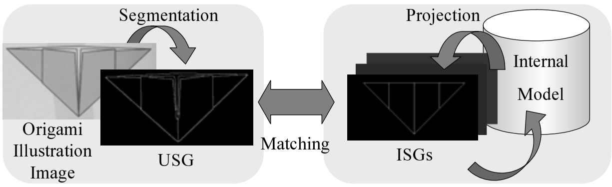

Figure 1. An example of a USG and ISGs.

They are obtained by maintaining consistency of crease patterns under some geometrical constrains [3]. All the feasible candidates are simulated against the internal model. As a consequence, several different origami states are obtained from each candidate. Then, the ISGs are generated by projecting each states of the internal model.

In order to determine the correct folding operation, each ISG should be compared with the USG which corresponds to the state of origami in the next illustration. Furthermore, in order to get the position of the next crease, the ISG should be matched with the USG. Namely, the system must match the feature points of the USG with those of the ISG approximately. Since existing methods for comparing and matching between a USG and an ISG are not so well, this paper proposes a new method for constructing USGs and matching two kinds of graphics.

2.3 Our Approach

By an existing segmentation method [4] based on the linear approximation of origami illustration image, a number of line segments corresponding to one edge of origami (i.e. one line segment of ISG) are generated. Because of the parts for describing overlapping faces and other elements such as arrows, many line segments and feature points are likely extracted from an origami illustration image and appear in the USG. That leads to the deterioration of matching accuracy.

From this reason, we propose a new method for generating USG based on Hough transform so as to improve matching accuracy. In our method, images including lines in origami illustrations are transformed by Hough transformation. Before applying inverse Hough Transform, candidates of linear parameters are filtered and narrowed down to reduce superfluous line segments or feature points. Instead of extracting short line segments that may come from the parts of overlapping faces, our method extracts main line segments with accuracy in consideration of an original image.

The issue to match an ISG with a USG can be actually viewed as an optimization problem, i.e. how to choose feature points from the ISG so that they are best matched with the feature points of the USG. We solve the problem by using a genetic algorithm. The evaluation function is based on both the positional error of feature points and the angular error of line segments.

3 Extraction of USG

This section discusses how to generate a USG. Figure 2 shows the outline of processing for generation of a USG.

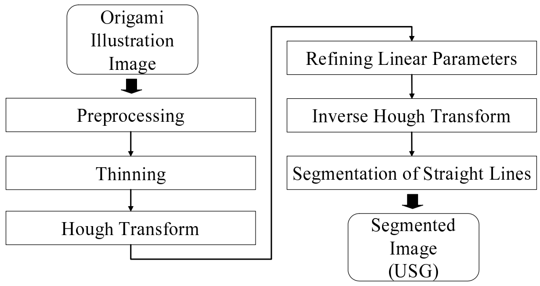

Figure 2. Outline of generating USG.

3.1 Preprocessing

Digitalization. Discriminant analysis is performed for determining the threshold for digitalization of a gray scale image whose histogram has two peaks. Based on the discriminant analysis, the threshold for digitalization of an origami illustration image whose histogram normally has three peaks is determined. The algorithm is described as follows.

1. Apply discriminant analysis to an origami illustration image. The resultant value is indicated by T.
2. Apply discriminant analysis to the parts whose pixel values are less than T and the resultant value is indicated by T.
3. Digitalize the image by threshold T'.

In this paper, the foreground of a digitalized image is represented by white pixels (max value) while the background is represented by black pixels (0 value).

Thinning. A thinning method based on 8-connectivity relations among pixels is applied on digitalized image

3.2 Refining Linear Parameters

Hough transform maps an image on x-y plane into the parameters on θ-ρ plane by the following equation. We spare someone the detail of Hough Transform in this paper.

$$\rho = x \cdot \cos\theta + y \cdot \sin\theta \quad (1)$$

We will use an accumulator array which keeps the number of votes obtained, and an accumulator image is generated to display this array. The peaks in the accumulator array, the pixels which have high values, mean that straight lines likely exist.

The First Step of Refining. In order to extract the candidates of linear parameters, we first digitalize the accumulator image, and label connected components in the digitalized image.

Subsequently, the candidates of parameters corresponding to the part of an arrow are eliminated on the basis of the fact that an arrow in the original image would lead to elongated connected component in the digitalized accumulator image. A connected component is considered as existence of an arrow when the following conditional expressions are satisfied.

$$s/d < m_1 \text{ (= const) and} \quad (2)$$
$$d_\theta > m_2 \text{ (= const)} \quad (3)$$

where s is the area (i.e. the number of pixels) of the connected component, d is the Euclidean distance between two end-points of the component, and d_θ is the θ-direction distance between them.

After the elimination, the linear parameters corresponding to pixels whose values are maximal in each labeled component are extracted as the first candidates.

The Second Step of Refining. There may be several similar parameters corresponding to one edge in an origami image because different labels are assigned to the same connected component in labeling process. In these parameters, the best one which corresponds to the most similar line to the edge in the origami image should be selected.

For this purpose, in this step, we generate an thinned image of input, and let the foreground pixels of this thinned image have the ID numbers as same as those of their nearest neighbors (see Figure 3). The resulting image is called "expanded thinned image" and used for further refining of parameters as follows.

When there are some linear candidates whose parameters are similar mutually, the candidate whose ID number appears most frequently of them is selected, and the rest are eliminated. The selected linear parameters by refining processes are transformed into the image space (by inverse Hough Transform), and the straight lines are obtained.

3.3 Segmentation of Straight Lines

Selecting Requisite Intersection Points. To generate segmentation graph (i.e. USG), the requisite points, i.e. the feature points of USG, are selected from all intersection points of straight lines based on the information included in the expanded thinned image (see Figure 3). The algorithm is as follows.

1. Calculate the coordinates of all intersection points and store ID numbers of two straight lines that pass through each intersection point.

2. On the neighborhood of an intersection point of straight line i and j, check whether the pixels which have i or j value in the expanded thinned image exist.

3. Consider the intersection point as requisite point only if the answer to step 2 is yes.

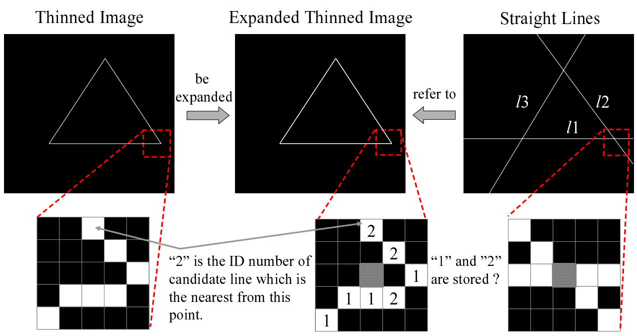

Figure 3. Generating and using the expanded thinned image for segmentation.

Generating a Segmentaion Graph. Finally, the existence of segment lines between each pair of feature points is determined. More specifically, only when two requisite points exist contiguously, the line segment between them can be generated.

4 Matching between USG and ISG

This section presents the method for matching a USG with an ISG based on a genetic algorithm (GA) [5]. We omit the idea and processing flow of GA. Definitions of the chromosome and the evaluation function in this method are described in Section 4.1 and 4.2 respectively.

4.1 Chromosome

Let U be a set of feature points of a USG, and let I be those of an ISG. The number of genes in a chromosome is supposed same with $|U|$, the number of feature points in the USG. Information of ith gene $g_i = (u_i, i_j)$ means that u_i, the feature point with ID i in the USG, corresponds to i_j, the feature point with ID j in the ISG in the gene.

4.2 Evaluation function

The Positional Error of Feature Points. The relative distance between $g_i = (u_i, i_j)$ and $g_m = (u_m, i_n)$ is defined as

$$d_{i,m} = \left| \left| u_i - u_m \right| - \sqrt{s_r} \left| i_j - i_n \right| \right| \tag{4}$$

where $|a - b|$ is the Euclidean distance between point a and point b, and s_r means the ratio of the area of the USG to the area of the ISG. Using this, the summation of relative distances between the points of one standard pair, one $g_i = (u_i, i_j)$, and those of every pair is expressed as

$$Ed_i = \sum_{\substack{m=1 \\ m \neq i}}^{|U|} d_{i,m} \tag{5}$$

Since Ed_i varies dramatically according to which pair is standard, every pair should be standard pair and each Ed_i should be calculated and summed. Consequently, the positional error of feature points is calculated as

$$Ed = \sum_{i=1}^{|U|} Ed_i \tag{6}$$

The Angular Error of Line Segments. The positional errors calculated from two ISGs whose feature points locate in exactly the same positions and whose line segments pass through different feature points may be equal and the correct ISG may be unable to be selected. So, the angular error of line segments has to be taken into account. If a pair of feature points $g_i = (u_i, i_j)$ is noted, the points are called noticed points and the angular error of line segment from the points is expressed as

$$Ea_i = \min_{\substack{\alpha \in A \\ \beta \in B}} \left| \alpha - \beta \right| \tag{7}$$

where A is the set of all angles which are composed of line segments connecting noticed point i_j, while B is the set of two angles composed of two lines constructing noticed point u_i. The summation of Ea_i for all i is the angular error of line segments as follows.

$$Ea = \sum_{i=1}^{|U|} Ea_i \tag{8}$$

The Evaluation Function. The evaluation function of GA is defined as

$$F = \frac{1}{\omega_d Ed' + \omega_a Ea'} \tag{9}$$

where ω_d and ω_a are the weight values, besides Ed' and Ea' are the results of the linear scaling for Ed and Ea respectively.

5 Experimental Results

This section shows the experimental results and evaluates the proposed methods. Firstly, a USG are generated by the method proposed in Section 3. Subsequently, the matching between the USG and the ISGs, including both the correct ISG and the incorrect ISG, is performed by proposed matching method.

Figure 4 show the examples of the extracted USGs and matching results. In the case of two examples, both drawn parts of face's overlapping and the arrow are not segmented. In these examples, the overlapping faces and the arrows are not extracted by our method, and only the main lines of origami appear in the USGs. That makes it possible to select the correct ISGs and to have the precise correspondence of feature points.

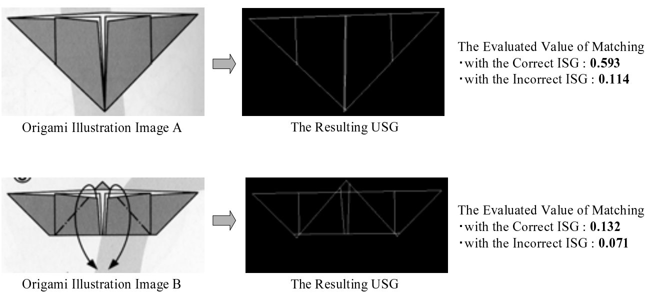

Figure 4. Examples of extraction of USGs and matching results.

6 Conclusions

This paper proposed a method for extraction of USGs from an origami illustration image and a method for matching the USGs with the origami model in 3-D virtual space. By proposed methods, the results of the matching were obtained with a high accuracy even for an image including a lot of noise. We are convinced that our methods can be applied to the recognition of 2-D freehand graphics.

As our future work, a countermeasure for wrong extraction of USGs should be proposed. Furthermore, folding lines, arrows, and explanatory texts have to be used for understanding the folding process since they have more valuable information in some cases.

References

[1] S. Miyazaki, T. Yasuda, S. Yokoi, and J. Toriwaki. "An Origami Playing Simulator in the Virtual Space". *J. Visualization and Computer Animation*, 7(6):25–42, 1996.

[2] T. Suzuki, J. Kato, and T. Watanabe. "Extraction of Contextual Information Existing among Composite Elements of Origami Books". In *Proc. of 4th IAPR International Workshop on Graphics Recognition*, pages 195–207, 2001.

[3] H. Shimanuki, J. Kato, and T. Watanabe. "Recognition of Folding Process from Origami Drill Books". In *Proc. of 7th International Conference on Document Analysis and Recognition*, pages 550–554, 2003.

[4] J. Kato, T. Watanabe, H. Hase, and T. Nakayama. "Understanding Illustrations of Origami Drill Books". *J. IPS Japan*, 41(6):1857–1873, 2000.

[5] Sadiq M. Sait and Habib Youssef. *"Iterative Computer Algorithms with Applications in Engineering"*. IEEE Computer Society, 1999.