Design of Pop-Up Cards Based on Weighted Straight Skeletons

Kokichi Sugihara
Graduate School of Advanced Mathematical Sciences
Meiji University
JST, CREST
Nakano, Tokyo 164-8525, Japan
Email: kokichis@isc.meiji.ac.jp
Telephone: +81-3-5343-8366

***Abstract*—We investigated the use of weighted straight skeletons in the computer-aided creation of pop-up cards. A pop-up card is a sheet of paper that can be folded to a flat plane but, when opened, produces a meaningful three-dimensional structure. Weighted straight skeletons are a special type of Voronoi diagrams, and they are closely related to the foldability of a sheet of paper along specified lines. We characterize the weights that make a paper foldable and apply them to an interactive system for the design of pop-up cards.**

I. Introduction

Pop-up art is a class of paper craft in which an initially folded sheet of paper generates a three-dimensional structure when it is opened. This kind of paper craft is mainly used for pop-up cards and pop-up books [16], [21]. Traditionally, pop-up structures have been designed by professionals using empirical know-how, but systematic treatments have also been developed.

For a special class of these structures in which all the fold lines are parallel, the basic methods are known [16] and software systems have been implemented [22]. Hara and Sugihara [13] presented a general method for designing one-lower-dimensional structures, i.e., planar link structures, for any polygon. Though the result of their method may cause collisions when opening the structures, Abel et al. [1] give another method in which no collisions occur.

For three-dimensional pop-up structures, however, general systematic methods have not yet been presented; the construction of complicated pop-up structures usually relies on the knowledge of skilled designers.

This paper proposes a computational approach to the computer-aided design of three-dimensional pop-up structures. Our method is based on a class of generalized Voronoi diagrams called weighted straight skeletons.

The Voronoi diagram is one of the most fundamental structures in computational geometry [5], [12], [23]. The most basic version is defined as the assignment of the points in a plane to the nearest site, as determined by the Euclidean distance, among a given set of sites that are called the generating points. This concept has been generalized by replacing the generating points with other types of generators and/or replacing the Euclidean distance with other metrics [4], [19], [20], [23].

One of the remarkable characteristics of the basic Voronoi diagrams is that the edges are composed of straight line segments (including half-lines). Because of this property, they can be a useful tool for solving geometric problems efficiently and robustly. In most of the generalized versions, on the other hand, this property is not preserved; the boundary edges, in general, are composed of curved lines, which makes the diagrams complicated in theory and computationally unstable in practice.

However, there are some exceptional cases in which the generalized versions do have boundary edges consisting of straight line segments. These include power diagrams [3] (also called Laguerre Voronoi diagrams [15]), the L_1-distance and L_∞-distance Voronoi diagrams [19], and the offset-distance Voronoi diagrams [6], [7], [8], [25] (also called straight skeletons [2], [11], [14]). This class of generalization is important from a practical point of view because these can all be constructed relatively easily from a numerical robustness point of view.

Let us give a remark on why we can regard the straight skeleton as a generalized Voronoi diagram. An ordinary Voronoi diagram can be interpreted as the crystal structure. Suppose that at time 0 crystals start grawing at generating points simultaneously, and they graw and fill space isotropically by the same speed until forefronts of the crystals collide each other. The resulting structure is the Voronoi diagram. Similarly, suppose that some crystal-like virtual material starts grawing at all the edges of a polygon toward both sides simultaneously. This material graws in such a way that the forefronts keep straight parallel to the original edges, stretching or shrinking so that they contact neighbor forefronts, and stops grawing when the forefronts collide. The resulting structure is the partition of the plane into regions belonging to the edges of the polygon, and the straight skeleton is the boundaries of these regions.

Straight skeletons, in particular, can be applied to the design of three-dimensional structures. For example, Eppstein and Erickson [10] applied them to roof design, and Tomoeda and Sugihara [24] applied them to designing illusory solids. Straight skeletons have also been applied to paper-folding designs that create an arbitrary polygon by means of a single straight cut [9]. Kelly [17] pointed out that a straight skeleton can be generalized to a weighted version, and the result can also be used for roof design, where it allows more degrees of

978-0-7695-5037-4/13 $26.00 © 2013 IEEE
DOI 10.1109/ISVD.2013.9

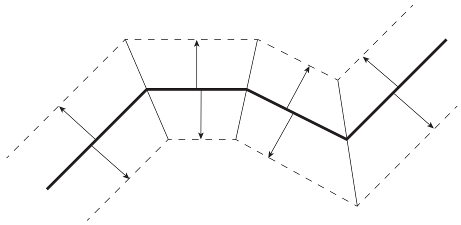

Fig. 1. Straight forefronts propagating from edges by the same speed.

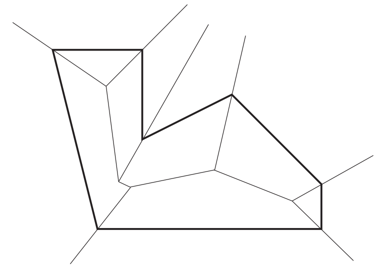

Fig. 2. Example of a straight skeleton.

freedom in the choice of roof inclination.

In this paper we show that weighted straight skeletons can be used in the design of pop-up structures. In Section 2, we review the definition and basic properties of weighted straight skeletons. In Section 3, we characterize the foldability of pop-up structures by using the weights of straight skeletons, and in Section 4, we propose a method for the interactive design of pop-up cards and offer some examples. Finally, we offer our concluding remarks in Section 5.

II. Weighted Straight Skeletons

In this section, we briefly review the straight skeleton and its weighted generalization.

A. Straight skeletons

Let P be a simple polygon, and let $(v_1, e_1, v_2, e_2, \ldots, v_n, e_n, v_1)$ be a cyclic sequence of vertices $v_1, v_2, \ldots, v_n$ and edges $e_1, e_2, \ldots, e_n$ that forms the boundary of P counterclockwise. We consider e_i a directed edge such that the left side of e_i corresponds to the inside of P. We consider a partition of the plane into regions, each belonging to an edge, in the following manner.

Suppose that, as shown in Fig. 1, waves with straight forefronts propagate from each edge inward and outward by the same speed, until they disappear at the points where their forefronts collide with other forefronts. Note that this wave is not physical wave, but imaginary one. We assume that the forefronts is straight and it changes the length in such a way that the forefront ends at the points of intersection with the two neighbor forefronts. Hence the forefronts initially form two polygons; one is inside P and the other is outside P. Each point on the forefronts stops moving when it collides with another forefront. Thus, they sweep the plane. We call the region swept by the waves starting at edge e_i the *Voronoi region* of e_i. The plane is partitioned into the Voronoi regions of the edges and their boundaries. We call this partition the *straight skeleton* of P. Fig. 2 shows an example of the straight skeleton where thick lines represent the polygon P and thin lines represent the straight skeleton.

The straight skeleton can be interpreted as a roof structure in the following way.

Let $f_i^{\mathrm{i}}(t)$ and $f_i^{\mathrm{o}}(t)$ be the forefronts at time t of the waves that starts from edge e_i at time 0 inward and outward, respectively. We consider an (x, y, z) Cartesian coordinate system and fix the polygon P to the xy plane. Suppose that the forefront line segments $f_i^{\mathrm{i}}(t)$ and $f_i^{\mathrm{o}}(t)$, that are initially (i.e., at $t = 0$) coincide with the edge e_i, goes downward so that the z coordinate is $z = -t$. Thus, $f_i^{\mathrm{i}}(t)$ and $f_i^{\mathrm{o}}(t)$ sweep slanted polygons so that they form a pair of roof planes having ridge e_i. Note that this interpretation is different from a common interpretation [10]; in the common interpretation the skeleton edges inside the polygon correspond the ridges of the roof, whereas in the present interpretation, edges of the polygon correspond to the ridges. This interpretation helps us to understand basic properties of the straight skeleton intuitively.

Property 2.1. ([2], [11], [10] A straight skeleton consists of straight line segments (including half-lines).

Proof. The boundary edges are the orthographic projection of the intersection of roof planes, and hence are parts of straight lines. □

B. Weighted straight skeletons

In the definition of straight skeletons, we generated a roof structure using the same inclination of the roof surfaces. By changing this inclination edge by edge, we can introduce a weighted version of the straight skeleton.

For $i = 1, 2, \ldots, n$, let $w(e_i)$ be a positive real, called the *weight* of the edge e_i. We assume that the forefront line segments $f_i^{\mathrm{i}}(t)$ and $f_i^{\mathrm{o}}(t)$ propagate in the xy plane $w(e_i)$ times faster than the unit speed. Each point on the forefronts stops propergation when it collides with another forefront. The region swept by the forefronts $f_i^{\mathrm{i}}(t)$ and $f_i^{\mathrm{o}}(t)$ is called the *weighted Voronoi region* of e_i, and the partition of the plane into the weighted Voronoi regions and their boundaries is called the *weighted Voronoi diagram* or the *weighted straight skeleton* [17], [18].

We can interpret the weighted straight skeleton as a roof structure in the following way. We extend the motion of the forefronts $f_i^{\mathrm{i}}(t)$ and $f_i^{\mathrm{o}}(t)$ from two-dimensional to three-dimensional so that their z coordinates are $z = -t$ at time t. This means that the forefronts sweep slanted planes with different inclinations. We define θ_i by

$$\theta_i = \arctan\left(\frac{1}{w(e_i)}\right). \tag{1}$$

By the above sweeping procedure we obtain the arrangement of the roof structures in such a way that the roof planes of edge e_i are generated by rotating the horizontal half-planes in both sides of the edge by angle θ_i in the negative direction of the z axis. The partition of the plane obtained by the projection

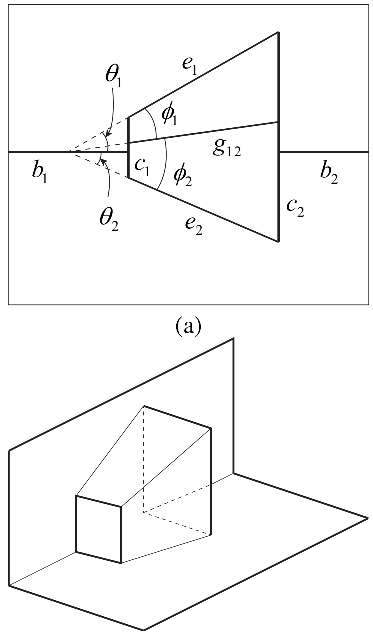

Fig. 3. Folding unit composed of two cut lines and five folding lines.

of this roof structure onto the xy plane coincides with the weighted straight skeleton.

The following property can also be obtained in a similar way as was used for straight skeletons.

Property 2.2 [17], [18]. A weighted straight skeleton consists of straight line segments (including half lines).

Note that a larger weight implies a larger region. This is because a larger weight implies a faster propagation of the forefronts.

An algorithm for constructing the weighted straight skeleton has been discussed by Eppstein and Erickson [10] and Kelly [17].

III. Foldability and the Weighted Straight Skeleton

Pop-up structures can be divided into two classes: one-piece structures and multi-piece structures [16]. The former is made by cutting and folding only a single sheet of paper, while the latter allows additional pieces of paper to be attached (such as by glue). In this paper, we concentrate on one-piece structures.

As shown in Fig. 3(a), we consider a sheet of paper having two cut lines c_1 and c_2 and five folding lines b_1, b_2, e_1, e_2, and g_{12}, where b_1 and b_2 are collinear. We want to generate the three-dimensional structure shown in Fig. 3(b), by bending along the folding lines. Assume that the sheet of paper can be bent along the folding lines, that is, we can change the dihedral angles associated with the folding edges from π to another angle, but each part of the sheet will remain planar. We then

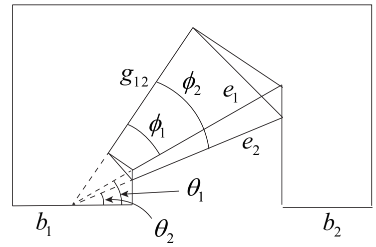

Fig. 4. Flat surface obtained by folding the structure.

bend along b_1, b_2, e_1, and e_2 to form valleys, while we bend along g_{12} to form a ridge. We call the structure composed of the two cut lines and the five folding lines a *folding unit*, and the segments b_1 and b_2 the *base line*. As shown in Fig. 3(a), let θ_1 and θ_2 be the angles between b_1 and e_1, and b_1 and e_2, respectively, and let ϕ_1 and ϕ_2 be the angles between e_1 and g_{12}, and e_2 and g_{12}, respectively. Note that

$$\theta_1 + \theta_2 = \phi_1 + \phi_2. \quad (2)$$

The following is the most basic property required for the design of a foldable structure [16].

Property 3.1. Suppose that the folding lines e_1, e_2 and g_{12} are not parallel to b_1 and b_2. Then, a folding unit can be folded so that the dihedral angle of b_1 and b_2 becomes 0, if and only if the following two conditions are satisfied:

(i) when extended, b_1, e_1, e_2, and g_{12} have a common point of intersection;

(ii)

$$\phi_1 = \theta_2 \qquad \text{and} \qquad \phi_2 = \theta_1. \quad (3)$$

Proof. First assume that (i) and (ii) are satisfied.

Suppose that the folding unit can be folded to a flat plane. Then the final situation of the folding unit is as shown in Fig. 4. Therefore we have

$$\theta_1 + \phi_1 = \theta_2 + \phi_2. \quad (4)$$

From eqns. (2) and (4), we have (3). Therefore if (i) and (ii) are satisfied, the folding unit is foldable to a flat surface.

Suppose that (i) is not satisfied. Then, the four lines b_1, e_1, e_2, and g_{12} are skew, and the structure cannot be folded flat.

Finally, suppose that (i) is satisfied but (ii) is not satisfied. Then, eq. (4) is contradicted.

Thus, (i) and (ii) are necessary and sufficient for it to be possible to fold a structure to a flat surface. □

We next consider the relation between foldability and a weighted straight skeleton. So far, we have considered both the straight skeleton and its weighted version for the case where a polygon is used as a generator. Now let us consider the case where two edges e_1 and e_2 are used as generators.

Suppose that e_1 and e_2 are two edges in the plane and that the line containing e_1 does not intersect e_2 and the line containing e_2 does not intersect e_1. This situation is satisfied by the e_1 and e_2 of the folding unit shown in Fig. 3(a). The common boundary of the weighted Voronoi regions of e_1 and e_2 is called the *bisector* of e_1 and e_2. We denote the bisector of e_1 and e_2 by $B(e_1, e_2)$. We get the next property.

Property 3.2. The bisector $B(e_1, e_2)$ passes through the point of intersection of e_1 and e_2 for any pair of weights $w(e_1)$ and $w(e_2)$.

Proof. The straight skeleton for any pair of weights is generated as the intersection of the roof surfaces. Therefore, the straight skeleton passes through the intersections of e_1 and e_2. □

Suppose that we are given six line segments c_1, c_2, b_1, b_2, e_1, and e_2 such that b_1 and b_2 are collinear and b_1, e_1, and e_2 are concurrent, and that we want to generate a folding unit by locating the other line segment g_{12}. For this purpose, we assign the weights as

$$w(e_1) = \sin\theta_2, \tag{5}$$

$$w(e_2) = \sin\theta_1. \tag{6}$$

Then the bisector of e_1 and e_2 gives the line on which g_{12} lies, and we have the following property.

Property 3.3. Let $B(e_1, e_2)$ be the bisector of e_1 and e_2 for the weights given by eqns. (5) and (6). Then, the folding unit can be folded to a flat surface if and only if g_{12} is on the bisector $B(e_1, e_2)$.

Proof. As shown in Fig. 5, let O be the point of intersection of e_1 and e_2. Recall that because of our assumption, b_1 and b_2 pass through O. From Property 3.2, $B(e_1, e_2)$ also passes through O. Suppose that we place the line segment g_{12} on $B(e_1, e_2)$. Let Q be a point on the line g_{12}. Let $d(P, Q)$ be the Euclidean distance between two points P and Q, and let $d(P, e)$ be the Euclidean distance from point P to line e. Then, we have

$$\sin\phi_1 = \frac{d(Q, e_1)}{d(O, Q)} = w(e_1), \tag{7}$$

$$\sin\phi_2 = \frac{d(Q, e_2)}{d(O, Q)} = w(e_2). \tag{8}$$

From eqns. (5), (6), (7), and (8), we get $\phi_1 = \theta_2$ and $\phi_2 = \theta_1$. Therefore the folding unit is foldable to the flat plane.

If g_{12} is placed outside $B(e_1, e_2)$, eq. (3) is not satisfied, and hence the folding unit is not foldable. □

The folding unit is thus foldable to the dihedral angle 0 if weights are assigned to e_1 and e_2 using eqns. (5) and (6). Using this property, we can construct an interactive system for designing pop-up structures, as shown in the next section.

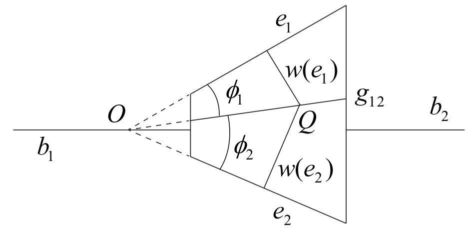

Fig. 5. Weights that place the bisector on g_{12}.

IV. Interactive Design of Pop-Up Structures

In this section, we provide an example to show how to interactively design pop-up cards. Suppose that we want to construct a pop-up structure that produces the shape shown in Fig. 6(a). Let us call this a *goal shape*. The design procedure of this pop-up structure is as follows.

First, a user extracts a polygon that represents the global structure of the goal shape. Suppose that the polygon represented by thick lines in Fig. 6(a) is chosen. Let us call this polygon the *fundamental polygon* of the goal shape.

Next, we construct the (unweighted) straight skeleton for the fundamental polygon, as shown in Fig. 6(b). The user chooses longer skeletal edges that share two regions corresponding to nonadjacent edges. In the example in Fig. 6(b), we choose five skeletal edges g_i that are on the bisectors $B(e_i, e_i{}')$, for $i = 1, 2, \ldots, 5$. For each pair e_i and $e_i{}'$, we compute the point of intersection and name it O_i, $i = 1, 2, \ldots, 5$.

Third, we deform the fundamental polygon so that the five points $O_1, O_2, \ldots, O_5$ become collinear, as shown in Fig. 6(c). We treat this common line b as the common base line, and construct five folding units using e_i and $e_i{}'$ together with the common base line. For that purpose we assign the weights to the edges in the same manner as was done in eqns. (5) and (6). That is, for each $i = 1, 2, \ldots, 5$, let θ_i and $\theta_i{}'$ be the angles between b and e_i and between b and $e_i{}'$, and define the weights as

$$w(e_i) = \sin\theta_i{}', \tag{9}$$

$$w(e_i{}') = \sin\theta_i. \tag{10}$$

Fourth, we construct the weighted straight skeleton, as shown in Fig. 6(d).

Fifth, we ignore the skeletal edge other than $g_1, g_2, \ldots, g_5$, and give cut lines c_1, c_2, c_3, and c_6, appropriately, as shown in Fig. 6(e).

Sixth, we add detailed substructures as shown in Fig. 6(f). In this particular example, we generate two small folding units: one with the base line e_2 to form a dorsal fin, and another with the base line $e_2{}'$ to form a belly fin. In this way we can continue to add smaller folding units, if desirable. Moreover, we can add cut lines to make small flat structures, such as a side fin in Fig. 6(f). This is the final diagram for a pop-up card, where thick lines represent cut lines and thin lines represent folding lines. The skeletal edges are to be folded convexly to form ridges of roofs, whereas the other thin lines are to be folded concavely to form valleys.

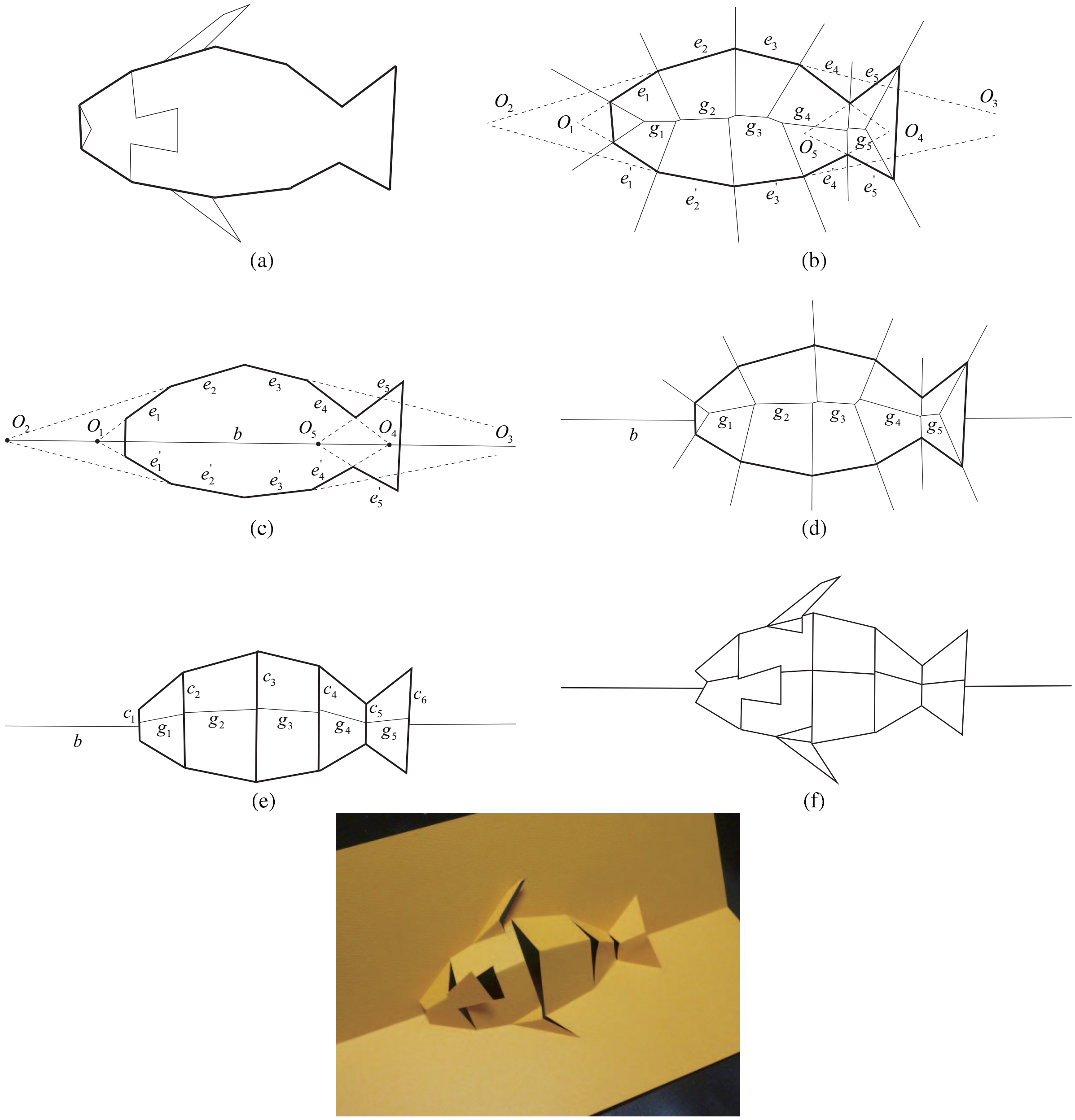

Fig. 6. Procedure for creating a pop-up card: (a) goal shape; (b) straight skeleton; (c) deformation of the goal shape; (d) weighted straight skeleton; (e) five folding units obtained by cut lines; (f) addition of detailed structures; (g) resulting pop-up card.

The resulting pop-up card is shown in Fig. 6(g).

Other examples are shown in Figs. 7 and 8. Fig. 7 shows a pop-up structure representing a stag beetle. This is also based on a single base line. On the other hand, Fig. 8 shows a pop-up structure with four base lines. As shown in this example, we can use more than one base line as long as the structure is foldable to a flat plane; our construction method can be applied to each base line independently.

V. Concluding Remarks

We proposed a computational approach for interactively creating pop-up cards from a target shape using a weighted straight skeleton. In our method, we first generate the (unweighted) straight skeleton, which gives us hints about which edges are to be paired to make the folding units. Next, for each pair of edges contributing to the main part of the skeleton, we assign weights so that the pair produces a folding unit. By applying this method to each section and subsection of the desired structure, we have a step-by-step method for designing the details of a pop-up structure. The performance of our method

Fig. 7. Another example of a pop-up card with a single base line.

Fig. 8. Pop-up card with four base lines.

was evaluated by examples.

We are still at the early stages of designing a general pop-up structure, and in this paper we have concentrated only on one-piece pop-up structures. If we use multiple pieces that can be attached to the basic part of the structure, we will be able to create more complicated structures. The extension of our method to multi-piece structures is the most important aspect of future research in this area.

Another direction of future work is to decrease the requirement for human input. The present interactive method greatly depends on the users' heuristic procedures, and we want to make the method as automatic as possible.

Acknowledgment

The work is supported by Grants-in-Aid for Scientific Research of the Japanese Ministry of Education, Science, Sports, and Culture No. 24360039 and No. 24650015.

References

[1] Z. Abel, E. D. Demaine, M. L. Demaine, S. Eisenstat, A. Lubiw, A. Schulz, D. L. Souvaine, G. Viglietta and A. Winslow: Algorithms for designing pop-up cards. *Proceedings of the 30th International Symposium on Theoretical Aspects of Computer Science*, Kiel, Germany, February 27-March 2, 2013, pp. 269–280.

[2] O. Aichholzer and F. Aurenhammer: Straight skeletons for general polygonal figures. *Lecture Notes in Computer Science*, no. 1090, 1996, pp. 117–126.

[3] F. Aurenhammer: Power diagrams: properties, algorithms and applications. *SIAM Journal on Computing*, vol. 16 (1987), pp. 78–96.

[4] F. Aurenhammer: Voronoi diagrams —–A survey of a fundamental geometric data structure. *ACM Computing Surveys*, vol. 23 (1991), pp. 345–405.

[5] F. Aurenhammer and R. Klein: Voronoi diagrams. In J.-R. Sack and J. Urrutia (eds.): *Handbook of Computational Geometry*, Elsevier, Amsterdam, 2000, pp. 201–290.

[6] G. Barequet, A. J. Briggs, M. T. Dickerson and M. T. Goodrich: Offset-polygon annulus placement problems. *Computational Geometry: Theory and Applications*, vol. 11 (1998), pp. 125–141.

[7] G. Barequet, M. T. Dickerson and M. T. Goodrich: Voronoi-diagrams for polygon-offset distance functions. *Discrete and Computational Geometry*, vol. 25 (2001), pp. 271–291.

[8] L. P. Chew and R. L. Drysdale, III: Voronoi diagrams based on convex functions. *Proceedings of the 1st Annual Symposium on Computational Geometry*, 1985, pp. 235–244.

[9] E. D. Demaine and J. O'Rourke: *Geometric Folding Algorithms*. Cambridge University Press, Cambridge, 2007.

[10] D. Eppstein and J. Erickson: Raising roofs, crashing cycles, and playing pool—Application of a data structure for finding pairwise interactions. *Discrete and Computational Geometry*, vol. 22 (1999), pp. 569–592.

[11] P. Felkel and Š. Obdržálek: Straight skeleton implementation. L. Szirmay-Kalos (ed.), *Proceedings of Spring Conference on Computer Graphics*, (Budmerice, Slovakia, 1998), pp. 210–218.

[12] S. Fortune: Voronoi diagrams and Delaunay triangulations. In D. Z. Du and F. Hwang (eds.): *Computing in Euclidean Geometry*, 2nd edition, World Scientific, Singapore, 1995, pp. 225–265.

[13] T. Hara and K. Sugihara: Computer-aided design of pop-up books with two-dimensional V-fold structures. *Abstracts of the 7th Japan Conference on Computational Geometry and Graphs* (Kanazawa, November 11-13, 2009), pp. 109–110.

[14] S. Huber and M. Held: A fast straight-skeleton algorithm based on generalized motorcycle graphs. *International Journal of Computational Geometry and Applications*, vol. 22 (2012), pp. 471–498.

[15] H. Imai, M. Iri and K. Murota: Voronoi diagram in the Laguerre geometry and its applications. *SIAM Journal on Computing*, vol. 14 (1985), pp. 93–105.

[16] P. Jackson: *The Pop-Up Book*. Henry Holt and Company, New York, 1993.

[17] T. Kelly: Degeneracy in the weighted straight skeleton. http://twak.blogspot.jp/2011/01/degeneracy-in-weighted-straight-skeleton.

[18] T. Kelly and P. Wonka: Interactive architectural modeling with procedural extrusions. *ACM Transactions on Graphics*, vol. 30 (2011), no. 2, article 14.

[19] R. Klei: Abstract Voronoi diagrams and their application. *Lecture Notes in Computer Science*, no. 333 (International Workshop on Computational Geometry, Wurzburg, March, 1988), Springer-Verlag, Berlin, 1988, pp. 148–157.

[20] D. T. Lee and R. L. Drysdale: Generalization of Voronoi diagrams in the plane. *SIAM Journal of Computing*, vol. 10 (1981), pp. 73–87.

[21] C. W. McCarthy: *M. C. Escher Pop-Ups*. Thames and Hudson Inc., New York, 2011.

[22] J. Mitani and H. Suzuki: Computer aided design for origami architecture models with polygonal representation (in Japanese). *Transactions of the Information Processing Society*, vol. 45 (2004), pp. 969–976.

[23] A. Okabe, B. Boots, K. Sugihara and S.-N. Chiu: *Spatial Tessellations —Concepts and Applications of Voronoi Diagrams*, Second Edition. John Wiley and Sons, Chichester, 2000.

[24] A. Tomoeda and K. Sugihara: Computational creation of a new illusionary solid sign. *Proceedings of the 9th International Symposium on Voronoi Diagrams in Science and Engineering*, 2012, pp. 144–147.

[25] P. Widmayer, Y. F. Wu and C. K. Wong: On some distance problems in fixed orientations. *SIAM Journal on Computing*, vol. 16 (1987), pp. 728–746.