

# Scissor-Based 3D Deployable Contours

Paper

the date of receipt and acceptance should be inserted later

**Abstract** Because of its property of saving space, scissor structure, which can transform from a compact state to a expanded state, is widely applied in various fields, from furniture design, architecture to outer space application. In this paper, we solve a challenging problem: designing scissor structures that can expand from a 3D contour to another 3D contour specified by users. Given two different 3D shapes, a non-uniform concentration is required, which makes the problem non-trivial. We propose a three-step algorithm to construct a 3D scissor structure. Firstly we generate *scissor segments* that are composed of a sequence of planar scissor units based on the shape correspondence. Secondly, we compute the scissor unit parameters of each segment in a suggestive manner. Finally, we introduce ball-shaped joints with parameterized guide slit to realize the deployment of a 3D scissor structure. The results demonstrate that our algorithm is able to generate scissor structures for a wide range of 3D contours.

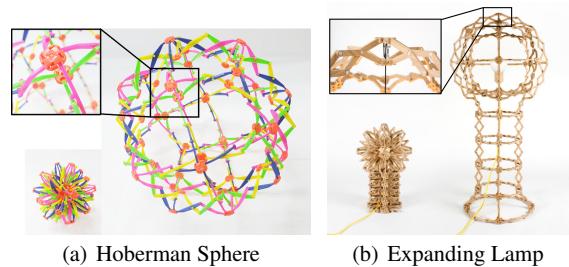
**Keywords** scissor structure, deployable, contour, non-uniform

## 1 Introduction

Deployable structures are widely used in many fields (e.g. architecture, space industry) for their space-saving capability and aesthetic appearance. When they are in expanded forms, they can be functional. On the other hand, they can be stored or serve another function in compact forms. One of the most important deployable structures is scissor structure, which can be found in daily necessities (e.g. toy, umbrella, lift, door and roof). Figure 1 shows two well-known scissor structures. One is Hoberman sphere [9] that can be compressed to a smaller sphere from a big sphere. The other one

Address(es) of author(s) should be given

is an expanding lamp that can be compressed to a smaller but similar shape [13].



**Fig. 1** Well-designed scissor structures. (a) Hoberman Sphere. (b) A expanding lamp. The junction used to deploy 3D shapes are shown on the top left.

However, all existing methods have the same weakness: **shape limitation**. Most of these 3D scissor structure designing methods are either shape preserving or with double curvature [2, 5, 9, 17]. Designing a complex deployable scissor structure requires massive human efforts, even for professional mechanisms and artists. Moreover, only one shape can be specified using these methods. However, not only the shape in expanded state is important, the shape in its compact form also plays an important role in many applications. For 2D planar cases, previous work has provided a feasible solution [15] to design scissor structure, which approximates a specific compact shape and a different expanded shape in two states. However, their method remains confined to planar curves. Given two arbitrary 3D curves, constructing a deployable scissor structure remains a significantly challenging problem.

Since scissor itself is a planar structure, only by introducing joint component, 3D structure can be constructed, as Figure 1 shows. However, this traditional joint limits the flexibility of 3D scissor structures. Only uniform collapses

are supported. Change in the angle between adjacent segments is not allowed using traditional joints. As a result, previous work is limited to some special cases.

Different from existing scissor structures of uniform collapses, we solve a more challenging problem: automatically constructing a deployable scissor structure given **two** 3D contours. While the two contours have different shapes, **non-uniform** deformations are required. It is non-trivial to construct a deployable scissor structure to approximate the given 3D shapes at different states, because of the planarity of scissor elements and unknown topologies. We propose an optimization framework to balance the shape approximation and structure stability of a scissor structure. Furthermore, we introduce a **ball-shaped joint** to produce the non-uniform deployment between different 3D contours. We show a series of simulated results as well as fabricated real models that transform between pairs of different 3D contours.

## 2 Related Work

The term, **deployable structure**, is not well defined and varies from systems to systems [7]. However, it always implies a transition from a compact state to an expanded state. In order to reveal the principle of governing the structure, classification system is proposed. The two major morphological classes are lattice structures and stressed-skin structures. The first class includes the scissor-like structure [15] and hinge structures. The second class includes the folded structure and origami structures [11]. Usually they are formed by an assemble of small planes.

**Scissor structure** is a typical deployable structure, especially in architecture area. Besides, some other practical structures including aesthetic design and toys contain scissor structure [8]. An overall description of scissor-like elements (SLE) and scissor structures can be found in [2, 5]. Structure analysis and case studies were also described in these two articles. The article [10] analyzed the kinematic properties of scissor structures, while others concentrated on mechanism analysis [16].

However, most research just considers specific cases, such as surfaces with one or two curvature [1]. Few research has been done in designing universal scissor structures. Designing a planar scissor structure, which transforms between two specific 2D curves, has been solved [15]. However, their method can not be directly extended to 3D cases. For 3D shapes, a method is proposed to generate deployable scissor structure with uniform contraction for shape-preserving in [17]. In comparison, we focus on a more challenging task to construct a scissor structure that can be transformed between two different 3D contours.

Joint structures are widely used in changing the relative position of different parts of 3D objects [3, 4]. We intro-

duced a series of **ball-shaped joints** to provide the possibility of a wide range of 3D transformation. Hubs, one type of joints used in most existing scissor structures, do not provide enough degree of freedom. As a result, the scissor structures connected by hubs can only realize uniform contraction for regular shapes. Our proposed joints support more flexible relative motions to allow non-uniform deployment between to different shapes.

## 3 Problem Statement

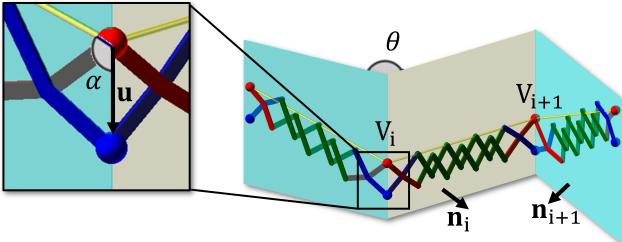
Given a 3D contour  $\mathcal{V}_0$  of the desired compact form ( $t = 0$ ) and another 3D contour  $\mathcal{V}_1$  for the expanded form ( $t = 1$ ), our goal is to construct a deployable scissor structure that expands from the given compact shape to the desired expanded shape. One-to-one vertex correspondence between two curves can be found using existing curve matching methods [6, 12] or uniform sampling. The two corresponding vertex sequences are denoted as  $\mathcal{V}_0$  and  $\mathcal{V}_1$ . Following the definition in [15], we will introduce our algorithm to generate scissor structure given two 3D shapes.

### 3.1 3D Curve Scissor Structure

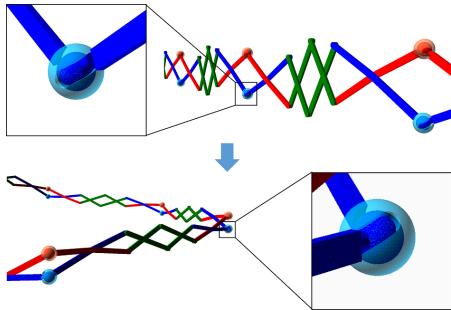
Each scissor unit  $X$  can be represented by six parameters  $\{b_1^l, b_1^r, \phi_1, b_2^l, b_2^r, \phi_2\}$ . A scissor structure  $\mathcal{S}$  is composed of a set of scissor units. However, since each scissor unit is planar during its deployment. It is impossible to directly connecting all scissor units to compose a 3D scissor structure. Therefore, we split a 3D curve scissor structure into a set of scissor segments  $\mathcal{S} = \{S_i\}_{i=1}^{m-1}$  and connect them with ball-shaped joints. Each scissor segment  $S$  consists of a sequence of co-planar scissor-like units.

Each scissor segment  $S_i$  represents a line segment  $\mathbf{V}_i \mathbf{V}_{i+1}$  in a 3D contour, as Figure 2 shows. Each scissor segment lies on one plane, whose normal is  $\mathbf{n}_i$ . The unit line  $\mathbf{u}$  of the two joint scissor units connecting two adjacent segments lies on the intersection of the two segment planes. The line segment  $\mathbf{V}_i \mathbf{V}_{i+1}$  and the two unit lines  $\mathbf{u}_l$  and  $\mathbf{u}_r$  of the segment are co-planar. The angle  $\alpha$  formed by the unit line and the line segment is called *side angle*. The angle  $\theta$  between two adjacent segment planes changes during its deployment between different 3D shapes.

**Ball-Shaped Joints.** The deployment of a 2D scissor structure can be controlled by unit lines. However, during the deployment of a 3D scissor structure, not only the lengths of unit lines change, the angles between adjacent segments also change. We introduce ball-shaped joints, which allow relative 3D motion of two connecting pin nodes, to enhance the shape flexibility. A ball-shape joint consists of an inner-ball and an outer-husk, as Figure 3 shows. In order to constrain the relative movement of the inner-ball and the outer-husk,



**Fig. 2** A 3D scissor segment is composed of a chain of co-planar units to represent a line segment  $V_iV_{i+1}$  of the input shape. The side unit line  $\mathbf{u}$  lies on the intersection of the two segment planes. The side angle  $\alpha$  and the angle  $\theta$  changes during deployment.



**Fig. 3** A ball-shaped joint consists of an inner-ball (blue) and an outer-husk (cyan). The 3D deployment of two adjacent scissor segments is controlled by the guide slit on the husk.

we cut a guide slit on the husk. One pin node moves along the guide slit to change the angle between two segments.

## 4 Construction of 3D Scissor Structure

Given two 3D curves, it is non-trivial to automatically generate a deployable scissor structure to form the given shapes at two states. The challenge is that the number of scissor units and the geometric parameters of each scissor unit form a tremendous search space. Even we divide the whole scissor structure into segments, it still remains a tedious work to find good scissor units to satisfy the requirements of shape approximation and non-uniform deployment.

Inspired by the algorithm to generate 2D scissor structures [15], we propose to use a three-step approach to construct a feasible 3D scissor structure, as Figure 4 shows. Firstly, we divide the entire scissor structure into segments according to the input shapes. The normal  $\mathbf{n}_i$  of each segment plane is computed to produce a deployable scissor structure in a regular shape. Secondly, we construct the chain of scissor units for each segment to allow the deployment according to the desired 3D shapes. Thirdly, we compute the parametric guiding slits of ball-shaped joints to connect scissor segments based on their parameters.

### 4.1 Structure-Segment Construction

Since each scissor segment is composed of a chain of co-planar units to represent a line segment of the given 3D contour, we firstly compute the normals  $\{\mathbf{n}_i\}_{i=1}^{m-1}$  of segment planes to roughly construct a 3D scissor structure. The normals are computed for two states independently.

For a segment  $S_i$  whose direction is  $\mathbf{v}_i$ , its plane normal  $\mathbf{n}_i$  is perpendicular to the line segment, which means  $\mathbf{n}_i \cdot \mathbf{v}_i = 0$ . The unknown normal has one degree of freedom on the rotation around the line segment. A scissor segment is visually regular when the unit lines are perpendicular to the line segment, which means  $\mathbf{u}_i \cdot \mathbf{v}_i = 0$ , as Figure 5 shows. For two adjacent segments with directions  $\mathbf{v}_i$  and  $\mathbf{v}_{i+1}$ , the unit line is the intersection of the two segment planes. Therefore, we have  $\mathbf{u}_i \cdot \mathbf{n}_i = 0$  and  $\mathbf{u}_i \cdot \mathbf{n}_{i+1} = 0$ . Since sometimes two adjacent line segments are coplanar in our 3D shapes, they probably have the same normal. It is ill-posed to write  $\mathbf{u}_i = \mathbf{n}_{i-1} \times \mathbf{n}_i$ . In order to generate scissor segments as regular as possible, we optimize the shape regularity to compute  $\{\mathbf{n}_i\}_{i=1}^{m-1}$  and  $\{\mathbf{u}_i\}_{i=1}^m$  as

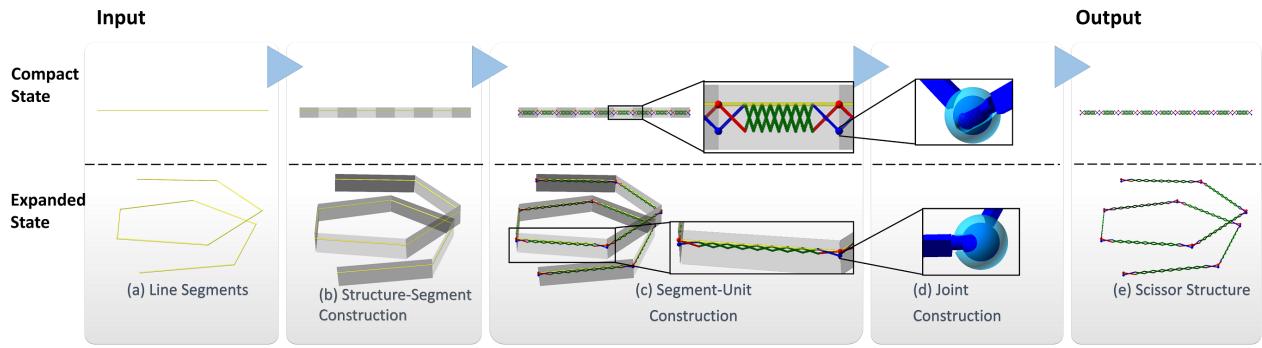
$$\begin{aligned} & \min_{\{\mathbf{n}_i\}_{i=1}^{m-1}, \{\mathbf{u}_i\}_{i=1}^m} \sum_{i=1}^{m-1} \|\mathbf{u}_i \cdot \mathbf{v}_i\| + \sum_{i=2}^m \|\mathbf{u}_i \cdot \mathbf{v}_{i-1}\| \\ & \text{s.t. } \mathbf{u}_i \cdot \mathbf{n}_i = 0, i = 1, \dots, m-1 \\ & \quad \mathbf{u}_i \cdot \mathbf{n}_{i-1} = 0, i = 2, \dots, m \\ & \quad \mathbf{n}_i \cdot \mathbf{v}_i = 0, i = 1, \dots, m-1. \end{aligned} \quad (1)$$

We use a Matlab tool of interior point method [14] to solve the above nonlinear least-squares problem. Initially, we automatically set the plane normals perpendicular to the same vector  $\mathbf{e}$  (e.g.  $\mathbf{e} = (0, 0, 1)$ ). We can also modify  $\mathbf{n}_i$  manually. As a result,  $\mathbf{n}_i = \mathbf{e} \times \mathbf{v}_i$ . For those  $\mathbf{e} \times \mathbf{v}_i = \mathbf{0}$ , we can choose  $\mathbf{n}_i$  randomly from those  $\mathbf{n}_i \cdot \mathbf{v}_i = 0$ .

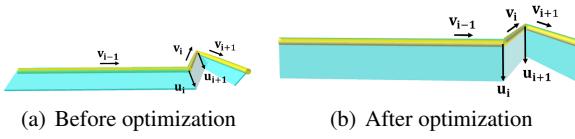
Once the normal and unit line of each segment are computed, we compute its segment parameters  $\{L_i, \alpha_i^l, \alpha_i^r, \mathbf{u}_i, \mathbf{u}_{i+1}\}$ , which are the length of the entire segment, the two side angles at the left and right ends, the two side unit lines, respectively. Given the segment parameters at the compact and expanded states, we then construct a series of scissor units for each segment.

### 4.2 Segment-Unit Construction

During the deployment of a scissor segment in 3D case, the angles between the line segment and its two side unit lines change during deployment. Similar with the 2D case [15], two symmetric angular units are used to construct the side scissor units of a scissor segment to form the side angles  $\{\alpha_0^l, \alpha_0^r; \alpha_1^l, \alpha_1^r\}$  at two states. A series of parallel scissor units change the segment length during deployment, as shown in Figure 6. The unit lines of the side scissor units connecting the parallel units are set as perpendicular to the segment direction.



**Fig. 4** Algorithm pipeline. Given two 3D shapes (a), the first step is to split the scissor structure into segments and compute the plane shape of each segment (b). Then a series of scissor units are constructed for each segment to meet the deployment and shape requirements (c). Finally, ball-shaped joints (d) are constructed to allow 3D deployment based on the 3D motion of the scissor segments. (e) The generated scissor structure.



**Fig. 5** Computing segment plane normals to maximize scissor regularity. (a) Bad normals lead to skewed segment shape. (b) After optimization, the unit lines are nearly perpendicular to the segment direction, in which case regular scissor units can be generated.

To form the side angles  $\{\alpha_o^l, \alpha_o^r; \alpha_t^l, \alpha_t^r\}$  of a segment, we construct a symmetric scissor unit given  $u_0, u_1$  and  $e_0$ , same with [15].  $e_0$  is the line length between two upper pin nodes of the joint unit. We use  $u_0 = kL_{min}$  where  $L_{min}$  is the minimum length of all segments in the input 3D contour.  $k = 0.1$  and  $e_0 = u_0$  in our experiments. They can be manually adjusted also to generate different scissor structures. Different from 2D cases where two joint scissor units are placed to approximate the input shape vertexes, we set the two end pin nodes located at the shape vertex to represent the input 3D curves, as Fig. 6(a) shows.

Because of the two side angles of a scissor segment are different, there is vertical deflection between the upper pin nodes if use the same symmetric unit at two ends. As a result, the middle part can not be connected by a chain of identical parallel scissor units, as Fig. 6(a)(a) shows. We introduce a *modified parallel unit* to allow vertical deflection and length expansion. It consists of two scissor units  $X_m = \{X_1, X_2\}$ , and each unit consists of two straight arms.  $X_2$  is a copy of  $X_1$  by rotating it around its revolute joint by  $\pi$ . Compared to the parallel scissors, modified scissor units provide variable vertical variation  $h$ , as Figure 7 shows.

For a line segment deployed from its compact state to its expanded state, after placing the two joint scissor units, we can compute the remaining lengths  $L_0$  and  $L_1$  at two states, as well as the vertical deflection  $H_0$  and  $H_1$ . Assume we use  $n$  pairs of identical modified parallel scissor units to construct the middle part, each one has to form the vertical de-

flections  $h_0 = \frac{H_0}{n}, h_1 = \frac{H_1}{n}$  at two states and deployed from  $l_0 = \frac{L_0}{n}$  to  $l_1 = \frac{L_1}{n}$ .

While the two unit side lines are perpendicular to the line segment, we build a local coordinate system with the line segment as  $x$ -axis and the unit line direction as  $y$ -axis and the top-left pin node as the origin. Denote the revolute joints as  $\mathbf{o}_0 = (x_0, y_0), \mathbf{o}_1 = (x_1, y_1)$  at two states the lengths of two arms as  $k_1 \|\mathbf{o}\|$  and  $a_2 = k_2 \|\mathbf{o} - \mathbf{p}\|$ , we have the same  $k_1$  and  $k_2$  at the two states. Given the desired unit line lengths  $u_0$  and  $u_1$ , we have  $\mathbf{p}_0 = (0, u_0), \mathbf{p}_1 = (0, u_1), \mathbf{q}_0 = (l_1, h_0), \mathbf{q}_1 = (l_1, h_1)$ , as Fig. 7 shows. As the bars have the same length at different states, we have

$$\begin{aligned} \|\mathbf{o}_0\| &= \|\mathbf{o}_1\| \\ \|\mathbf{o}_0 - \mathbf{p}_0\| &= \|\mathbf{o}_1 - \mathbf{p}_1\| \end{aligned} \quad (2)$$

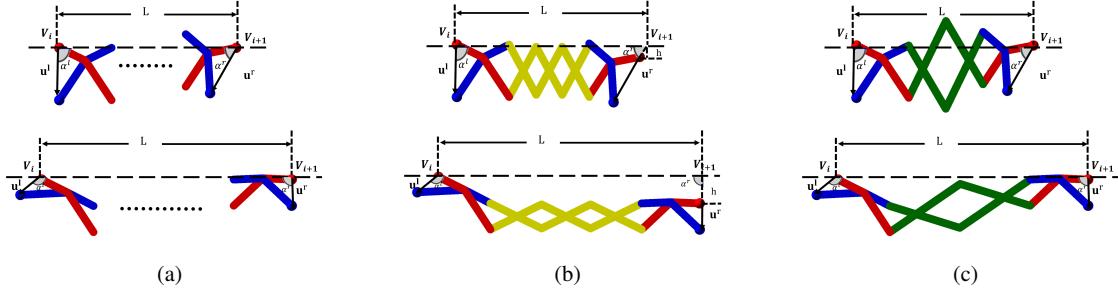
For each state, the two scissor units are symmetric, so

$$\begin{aligned} \mathbf{q}_0 - k_1 \mathbf{o}_0 &= k_2 (\mathbf{o}_0 - \mathbf{p}_0) \\ \mathbf{q}_1 - k_1 \mathbf{o}_1 &= k_2 (\mathbf{o}_1 - \mathbf{p}_1) \end{aligned} \quad (3)$$

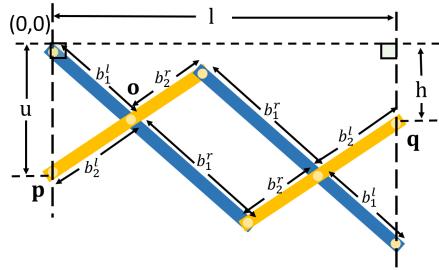
To solve the six parameters  $\mathbf{o}_0(x_0, y_0), \mathbf{o}_1(x_1, y_1)$  and  $k_1, k_2$ , we have two quadric equations (Eq. 2) and four linear equations (Eq. 3). If multiple solutions are obtained, we choose the revolute joint position that is the closest to the center.

### 4.3 Number of Scissor Units

Using modified parallel scissors, we can use a wide range of numbers of scissor units to realize the deployment of a line segment. With too many scissor, the structure become too weak to support the shape in real model. Reducing the scissor number makes the structure more stronger. It reduces the fabrication error as well because the inevitable fabrication error in real models is accumulated in each pin node connection. On the other hand, with less scissor units, the generated scissor segment might be skewed. Since we can accurately interpolate the given shape vertexes, the shape error is distributed to modified parallel scissor units. If the



**Fig. 6** Segment-unit construction. Top row: compact state; Bottom row: expanded state. (a) Construction of side units given the desired segment parameters including side angles and side unit lines. The side scissor units (blue and red arms) can change angles. (b) Segment construction in previous method [15]. (c) Our segment construction with modified parallel units (green). By placing the side unit right on the segment vertexes, the vertical deflection exists. Our scissor segments using modified parallel units approximate the desired shapes better than [15].



**Fig. 7** Construct a modified parallel scissor unit given the vertical deflections  $h_0, h_1$  as well as  $l_0, l_1$  at two states.

segment is too long, the modified parallel scissor unit will have a skewed shape with an extremely long bar and a relatively short bar. The scissor segment's shape becomes irregular, especially in the compact state. Moreover, the scissor unit will become too feeble and easily broken. Figure 8 illustrate the comparison between different number of modified scissor units. We can see that, with more scissor units, the generated scissor segment looks closer to the input shape.

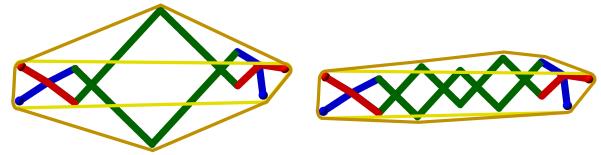
In order to balance the shape regularity and the fabrication error, we define an **energy function** combing the two aspects. To represent the shape regularity, we enclose a scissor segment with an elastic. According to Hooke's Law, the elastic energy will be  $E = \frac{1}{2}kl^2$ , where  $k$  is the elasticity coefficient and  $l$  is the incremental length of the elastic. For simplicity, we set  $k = 1$ . For a scissor segment  $S$ , the incremental length is the difference between the circumference of its convex hull  $C_{hull}(S)$  and the circumference of the quadrilateral  $C_{quad}(S)$ . We define the shape regularity energy for a scissor segment  $S_t$  at time  $t$  as

$$E_{reg}(S_t) = (C_{hull}(S_t) - C_{quad}(S_t))^2 \quad (4)$$

Combining the two desired states, we define the shape regularity energy of the generated scissor segment as

$$E_{regularity}(S) = \frac{1}{2}(E_{reg}(S_{compact}) + E_{reg}(S_{expanded})) \quad (5)$$

To penalize a large number of scissor units to save fabrication costs and reduce the fabrication error, we add the num-



**Fig. 8** The yellow line is the elastic in the zero energy state. It describe the input shape. The brown line is the convex hull of the generated scissor segment, which used to approximate the inputs shape. (a) Using less modified parallel scissor units, the resulted units are irregular and far from the desired shape. (b) A scissor segment using more modified scissor units are more regular and closer to the desired shape.

ber of scissor units  $N_{unit}$  to the final energy

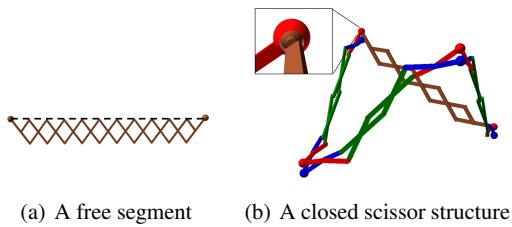
$$E(S) = E_{regularity}(S) + \lambda N_{unit}, \quad (6)$$

where  $\lambda$  is a weight proportional to  $L_{min}^2$  to make the two term is the same scale. We test a series of number of modified parallel scissor units and compute the shape energy for each resulting scissor segment. The one with the lowest shape energy is chosen to construct the scissor structure.

#### 4.4 Construction of Ball-Shaped Joints

After we construct a chain of scissor units for each segment considering their shapes at two states, we now connecting them to realize the 3D deployment while the angles between two adjacent segments change. For a pair of two adjacent segments  $S_a$  and  $S_b$ , denote their two joint pin nodes as  $\mathbf{p}_a(t), \mathbf{q}_a(t)$  and  $\mathbf{p}_b(t), \mathbf{q}_b(t)$  at time  $t \in [0, 1]$  during deployment, respectively. Their plane normals are  $\mathbf{n}_a$  and  $\mathbf{n}_b$ , respectively. Ideally,  $\mathbf{p}_a(t) = \mathbf{p}_b(t)$  and  $\mathbf{q}_a(t) = \mathbf{q}_b(t)$  at any time  $t$ . However, this can not be realized in real object because scissor bars are thick and the joint pin nodes can not be ideally connected.

We construct a ball-shaped joint for each pair of connected pin nodes from two adjacent segments according to the desired angle motion. Denote the center of the ball-shaped



**Fig. 9** A free segment (a) and a scissor structure using a free segments to represent a closed 3D curve (b).

joint  $J$  as  $\mathbf{c}$ , we fix  $\mathbf{c} = \mathbf{q}_a$  during deployment. The two revolute joints of the two adjacent joint scissor units are  $\mathbf{o}_a$  and  $\mathbf{o}_b$ . By fixing the bar  $\mathbf{o}_a\mathbf{c}$  connected to the out-husk of the ball-shaped joint, the active bar  $\mathbf{o}_b\mathbf{c}$  connecting to the inner-ball moves according to the desired plane rotation from  $\mathbf{n}_b(0)$  to  $\mathbf{n}_b(1)$ . Given the radius of the ball-shaped joint, we can compute the parametric trajectory on the out-husk according to the motion of bar  $\mathbf{o}_b\mathbf{c}$ . The guide slits are then obtained by constructive solid geometry (CSG) operation with a desired width.

For a **closed curve**, it will encounter deflection problem during its transformation, i.e. the closure of  $\mathcal{S}$  may not be preserved. In 2D cases [15], the scissor structure constructed from two desired shapes is optimized by sampling a number of states during its deployment to minimize deflections at connected pin nodes. We proposed a special scissor structure, called *free scissor segment*, to represent one line segment in a closed loop. A free scissor segment is composed of two special units at its two ends and a chain of identical parallel scissor units in the middle. This special type of units only has three bars and only one pin node to connect other scissor segments. The deflection problem is eliminated since a free segment will not add surplus constrains to the open contour scissor structure. It only serves to change the segment length. A free segment is connected to another segment by only one ball-shaped joint, as Fig. 9 shows.

## 5 Fabrication

In order to validate the scissor structures generated by our approach, we make real models using 3D printing. However, more fabrication factors have to be considered, including materials, mechanics and physical constraints.

The thickness of scissor arms in fabrication leads to that the two arms of a scissor unit are not ideally coplanar. In fact, we have two layers of arms in fabricated models. Screws and nuts are used to connect arms between up and down layers. They provide enough friction for the model to stay stable. When printing ball-shaped joints, both the outer husk and the inner ball are printed simultaneously with the scissor arms connected to them. The part of arms intersecting with the ball is designed as a cylinder and its radius is the



**Fig. 10** Calibration sets for testing 3D printer parameters for achieving optimal friction, structure strength, material saving and structural integrity preservation. For this purpose we vary the following parameters: gap size, arm thickness, joint size and screw hole radius etc. Here we only show a small portion of them.

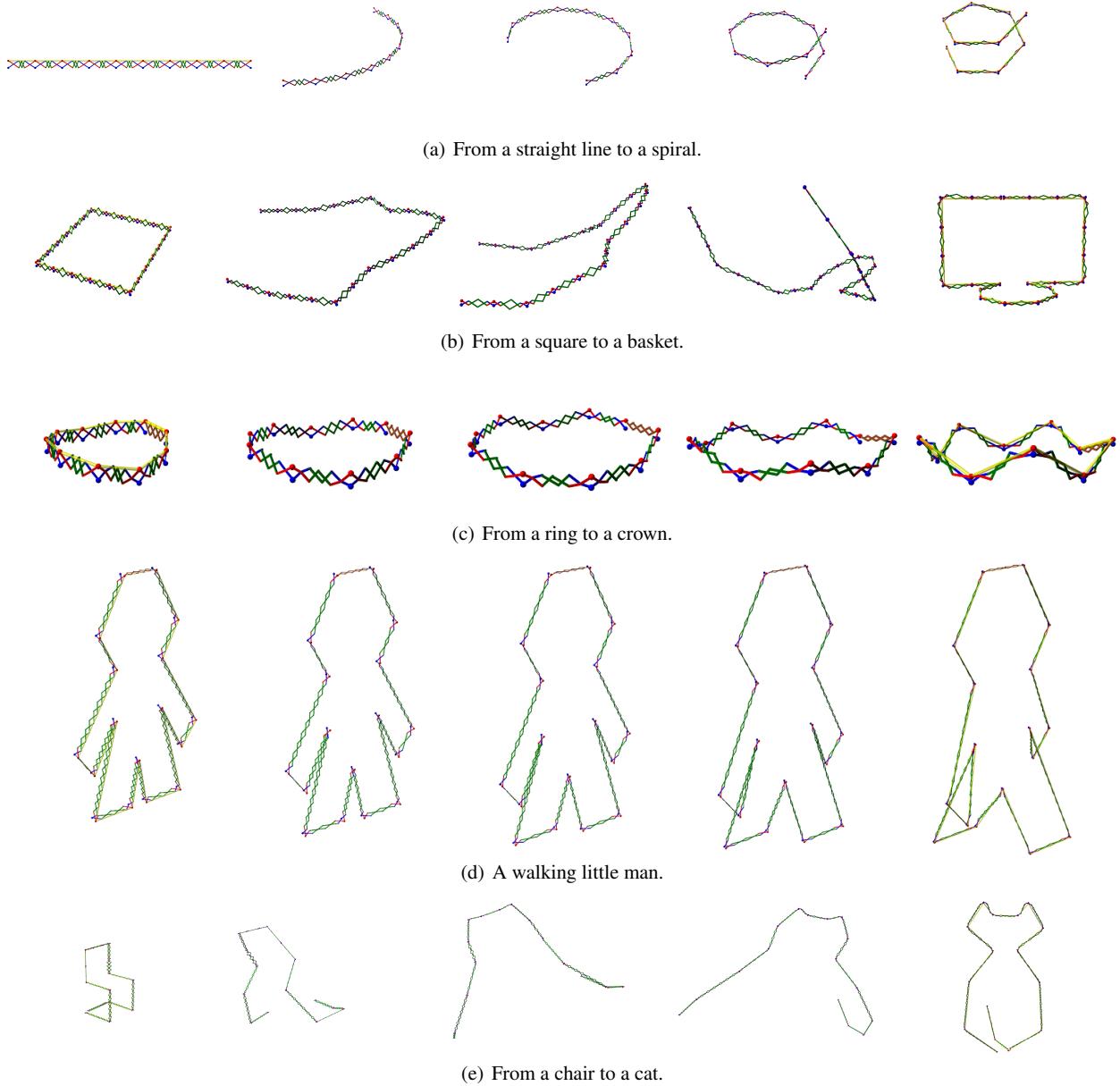
same as orbit in order to keep it in the orbit during movement. To ensure that individual parts are movable and not fused during printing, we set two device-dependent minimal distances for joints. One is the gap between husks and balls. The other is the gap between arms and their guide slits.

In theory, these parameters may be derived from known material parameters and the specification of the 3D printing process. We found that the printed results are different even with the same process. Thus, we take a calibration process to determine the above parameters. Before we print entire scissor structures, we run a test on each printer. A calibration set of scissor units and joints are shown in Fig. 10. For the material and strength consideration, the thickness of scissor bars is set to be 3mm while the width is set to be 7mm. The gap between the outer ball and inner ball is set to be 0.3mm, which is the minimal gap distance allowed. The radius of the screw hole is set to be 1.55mm, which is 0.05mm larger than the radius of screw.

## 6 Results and Discussions

In order to evaluate our model, we present a series of simulation and fabrication results, given a wide range of 3D contour pairs. Without considering the fabrication effort, we first simulate several scissor structures given pairs of complex 3D contours, as shown in Fig. 11. Their deployment processes are shown under different points of view. We can see that, our method is able to construct scissor structures for both closed 3D contours and open 3D contours. More detailed deployment can be found in the supplementary video.

For fabricated results, because ball-shaped joints require high accuracy to support small gaps, we use a high-accuracy printer ProJet ®3500 HDMax with Visijet Crystal to generate joints. For the remaining parts, we use HT480S with GP Plus, which is much more cheaper with higher strength, to fabricate the real models. We show two real fabricated scis-

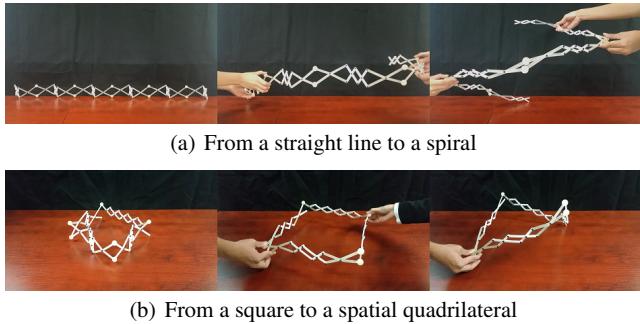


**Fig. 11** The simulated 3D curve scissor structures given different pairs of 3D contours. All above simulation are sampled at the procedure of  $t = 0, t = 0.25, t = 0.5, t = 0.75, t = 1$  from left to right respectively. (a) an example of open 3D curve structure. It can transform from a straight line to a spiral line with varying radius (i.e. the middle of the spiral is slimmer than the two sides of it). (b) A basket can be stored in the form of 2D square. (c) A closed 3D curve structure that transforms from a 3D ring to a crown. Since the 3D ring is not planar shape, this example also shows that our method is able to generate a scissor structure which is able to transform between two 3D lines. (d) A little 3D walking man. (e) An example that transforms from a chair to a cat with a curly tail.

sor structures in Fig. 12. The first one is able to transform from a straight line to a 3D spiral. The second example is to generate a closed 3D scissor structure, which transforms from a square to a spatial quadrilateral curve. The design of free segments is a practical solution for closed shapes, without introducing much complicated computation.

## 7 Conclusion

We present an automatic framework for constructing 3D deployable contours using scissor structure. Given two 3D contour shapes, a scissor structure that can deploy from one shape to the other one can be generated. We propose a feasible solution to find a deployable structure by decomposing the problem to structure-segment construction and segment-



**Fig. 12** Fabricated real models. From left to right: compact state, intermediate state and expanded state.

unit construction. Ball-shaped joints are proposed to realize angle changes in 3D contours. Moreover, we consider the fabrication problem using 3D printing. Both the simulation results and the fabricated examples illustrate that our method is able to generate deployable scissor structures for a wide range of 3D contour shapes. With the flexibility of designing scissor structure, our framework has applicable potential in various areas, such as architecture, furniture design.

Currently, we only study the construction of scissor structures to deploy between two 3D contour shapes. The designing method of scissor structure for arbitrary 3D surfaces is out of scope of our paper. There are a large number of loops between vertexes in 3D shapes, while our method works for open contours well. Collision is not handled currently. We simply set the unit line short enough to avoid collision. A future work is to integrate collisions as well as the physical stability into consideration during the construction of a scissor structure.

## References

1. Akgn, Y.: A novel transformation model for deployable scissor hinge structures. Universitt Stuttgart dissertation (2011)
2. Alegria Mira, L.: Design and analysis of a universal scissor component for mobile architectural applications. Vrije Universiteit Brussel. Ph.D. dissertation (2007)
3. Bächer, M., Bickel, B., James, D.L., Pfister, H.: Fabricating articulated characters from skinned meshes. ACM Trans. Graph. (Proc. SIGGRAPH) **31**(4) (2012). DOI 10.1145/2185520.2335398
4. Cali, J., Calian, D.A., Amati, C., Kleinberger, R., Steed, A., Kautz, J., Weyrich, T.: 3d-printing of non-assembly, articulated models. ACM Trans. Graph. **31**(6), 130:1–130:8 (2012). DOI 10.1145/2366145.2366149
5. De Temmerman, N.: Design and analysis of deployable bar structures for mobile architectural applications. Vrije Universiteit Brussel. Ph.D. dissertation (2007)
6. Gruen, A., Akca, D.: Least squares 3d surface and curve matching. ISPRS Journal of Photogrammetry and Remote Sensing **59**(3), 151–174 (2005). DOI 10.1016/j.isprsjprs.2005.02.006
7. Hanaor, A., Levy, R.: Evaluation of deployable structures for space enclosures. International Journal of Space Structures **16**(4), 211–229 (2001). DOI 10.1260/026635101760832172
8. Hoberman, C.: <http://www.hoberman.com>
9. Hoberman, C.: Reversibly expandable doubly-curved truss structure. United States Patent No. 4, 942, 700 (1990)
10. Langbecker, T.J.: Kinematic analysis of deployable scissor structures. International Journal of Space Structures **14**(1), 1–15 (1999). DOI 10.1260/0266351991494650
11. Ovadya, A.: Origami transformers: Folding orthogonal structures from universal hinge patterns. Massachusetts Institute of Technology B.S. dissertation (2011)
12. Sebastian, T.B., Klein, P.N., Kimia, B.B.: On aligning curves. IEEE Trans. on Pattern Recognition and Machine Intelligence **25**, 116–125 (2003)
13. Stehmann, E.: Expanding lamp (2016). URL <http://erikstehmann.nl/work/expand1.html>. Accessed July 18, 2016
14. Waltz, R., Morales, J., Nocedal, J., Orban, D.: An interior algorithm for nonlinear optimization that combines line search and trust region steps. Mathematical Programming **107**(3), 391–408 (2006). DOI 10.1007/s10107-004-0560-5
15. Zhang, R., Wang, S., Chen, X., Ding, C., Jiang, L., Zhou, J., Liu, L.: Designing planar deployable objects via scissor structures. IEEE Transactions on Visualization and Computer Graphics **22**(2), 1051–1062 (2016). DOI 10.1109/tvcg.2015.2430322
16. Zhao, J.S., Chu, F., Feng, Z.J.: The mechanism theory and application of deployable structures based on SLE. Mechanism and Machine Theory **44**(2), 324 – 335 (2009). DOI 10.1016/j.mechmachtheory.2008.03.014. URL <http://www.sciencedirect.com/science/article/pii/S0094114X08000815>
17. Zheng, C., Sun, T., Chen, X.: Deployable 3d linkages with collision avoidance. In: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 179–188. Eurographics Association (2016)