CardBoardiZer: Creatively Customize, Articulate and Fold 3D Mesh Models

Yunbo Zhang, Wei Gao, Luis Paredes, Karthik Ramani
School of Mechanical Engineering, Purdue University
West Lafayette, IN 47907
{zhan2014, gao51, lparede, ramani}@purdue.edu

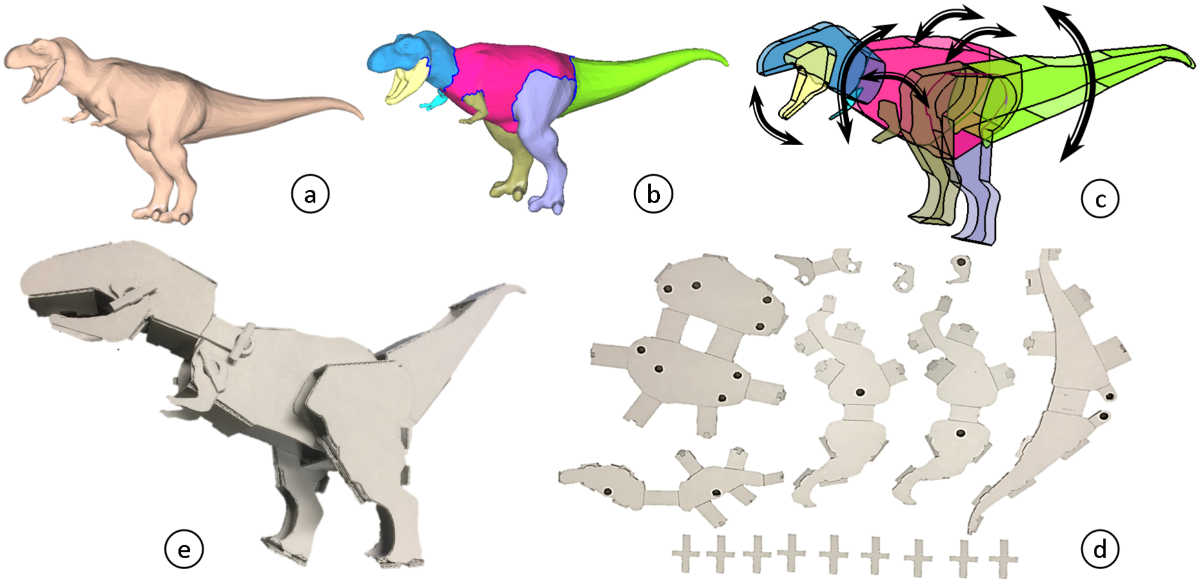

Figure 1. shows the overall building pipeline using CardBoardiZer: Given a 3D mesh T-Rex model as shown in (a), CardBoardiZer allows the user to (b) customize the segmentations at the locations where parts are desired to be articulated, (c) specify the corresponding rotational joint motions, (d) the crease-cut-slot patterns are generated for (e) the user to quickly cut, fold and assemble using cardboard.

ABSTRACT

Computer-aided design of flat patterns allows designers to prototype foldable 3D objects made of heterogeneous sheets of material. We found origami designs are often characterized by pre-synthesized patterns and automated algorithms. Furthermore, augmenting articulated features to a desired model requires time-consuming synthesis of interconnected joints. This paper presents CardBoardiZer, a rapid cardboard based prototyping platform that allows everyday sculptural 3D models to be easily customized, articulated and folded. We develop a building platform to allow the designer to 1) import a desired 3D shape, 2) customize articulated partitions into planar or volumetric foldable patterns, and 3) define rotational movements between partitions. The system unfolds the model into 2D crease-cut-slot patterns ready for die-cutting and folding. In this paper, we developed interactive algorithms and validated the usability of CardBoardiZer using various 3D models. Furthermore, comparisons between CardBoardiZer and methods of Autodesk® 123D™ Make, demonstrated significantly shorter time-to-prototype and ease of fabrication.

Author Keywords

Foldable design; articulated model; cardboard prototyping.

ACM Classification Keywords

H.5.2. [**Information Interfaces and Presentation**]: User Interface

INTRODUCTION

Current trends in democratization of fabrication make it possible for one to personalize manipulative designs through the choice of geometries, materials, and fabricate them on demand. A variety of rapid, early, but flexible prototyping techniques, such as 3D printing, laser cutting, and home milling

CHI'16, May 07-12, 2016, San Jose, CA, USA.

DOI: http://dx.doi.org/10.1145/2858036.2858362

machines, are gaining popularity among the DIY crowds [21]. As a result, common individuals now are able to fabricate artistic and personal objects without being technically trained to use sophisticated computational and production tools.

As an ancient paper craft originating from Japan, origami has been naturally contextualized into many design systems to create foldable 3D structures. The real beauty of folding lies in its elegant simplicity using 2D sheet of material to create complex 3D shapes and forms. During the last 40 years, whys, whats and hows of different origami tessellations and structures have been geometrically and symbolically described by the underlying mathematical rules, such as flat foldability [2] and folding any polygonal shape [11]. With the marriage of computational geometry and origami, systematic design tools have also been developed recently (TreeMaker [24], Origamizer [10], FPME [16], Pepakura Designer [29], Autodesk® 123D™ Make [4], and FoldMecha [33]).

We found the past applications of foldable structures and crafts are limited by the following characteristics: 1) most developments have a typical goal of achieving automation of the design process to construct deterministic shapes and structures. However, in these systems, users are not allowed to participate and customize the desired shape to be folded, determine the parts to be articulated, and decide how the parts are joined. 2) Given any single model with articulated features, it is a daunting task using traditional mechanical design approaches to synthesize and prototype interconnected joints in order to make the model movable. 3) Conventional design and manufacturing tools are highly-procedural and require elaborate training and practice before they can be effectively utilized. Such limitations of these tools impede the integration of designing and making of complex shapes for an independent tinkerer.

Inspired by the development of Do-It-Yourself (DIY) and the maker movement [1], we present a novel customizable prototyping framework, called CardBoardiZer. Using CardBoardiZer, one can repurpose any 3D model that already exists to be foldable and articulable. We aim to democratize the design and fabrication together so that designers who lack specialized knowledge can quickly prototype. It is suitable for novices in the maker movement, K-12 crafting activities, hobbyists and even college level use in prototyping and physical computing classes. The new standards for U.S. STEM education framed by the National Research Council has an explicit focus on engineering and design [41]. Our methodology encourages the design, make, and play through creating, tinkering, using widely available materials, which is essential for engagement [19]. Our approach is targeted towards the DIY community where the versatile prototyping material, cardboard, and a low-cost crafting cutter are used for cutting and then folding is done by hand.

In our work, the overall guiding design philosophy is to allow one to quickly and easily personalize desired designs, through the choice of geometries and articulations, to create foldable cardboard crafts and prototypes. Hence, the barrier to entry into 3D modeling and prototyping is lowered not only by directly repurposing the types of shapes, but indirectly by using cardboard itself as the material. The subsequent folding and assembling using their own hands become a source of pride and satisfaction. In using this guiding principle, the following detailed design elements were adopted for the CardBoardiZer system:

1)The system needs to ensure a rapid cycle of prototyping at early conceptual design stages, including folding and die cutting.

2)The design platform should provide a workflow for different stages of customization, such as shape segmentation and modification, resolution definition, and specification of motion joints.

3)The visual interface should be integrated with the physical behaviors such as foldability, motion constraints and articulation.

4)The physical prototype needs to be easily made out of inexpensive, lightweight and readily available materials that are widely used in the physical prototypes, such as cardboard.

These design elements lead to several geometric and algorithmic challenges to create a human-in-the-loop CardBoardiZer platform. CardBoardiZer is a new genre of cardboard-based rapid prototyping system, that is designed to create new affordances for experimentation and expressiveness of designers. We create a new workflow using the customizable segmentation, shape approximation, articulation specification and unfolding design to allow rapid customization and prototyping. The geometric operations are made accessible for novice designers and use existing 3D sculptural models. Previous works such as Paper Folding 3D [39], generates foldable patterns with a small amount of folds, reducing the effort and time. However the shape approximation of such models is not satisfactory. On the other hand, complex unfolded mesh patterns created by Pepakura Designer, Autodesk® 123D™ Make (*folded panels module*) and Optimized topological surgery [43] approximate the input model well but demand a high effort and time to fold, making these methods accessible only to a few that have the expertise, manual dexterity and patience.

On the other hand, we created affordances for a new intermediate-level foldable crafting form that repurposes and abstracts existing 3D models. This design platform integrates customizable segmentation, contour extraction and approximation, geometric simplification, articulation specification and design of unfolding into a compact design environment to help one easily generate foldable patterns ready for cutting and folding. We retain the ease of foldability of the shape as a key characteristic for the user, but at the same time serve the geometric shape approximation. The alternating curve-and-straight regions (ACSR) form not only retains the curvy shape in curved regions, but also simplifies the folding process for each of the partitions. Only a small amount of straight regions have to be coupled for closing the shape. To automate ACSR, we have developed a new geometric simplification algorithm. By integrating straight portions, this algorithm also provides a basic level of structural integrity. Also by manipulating the ACSR resolution, designers can balance the shape approximation with the total time and folding effort. Several different customization cases that demonstrate the foldability and motion feasibility are discussed later in this paper.

RELATED WORK

Unfolding 3D Meshes

In computer graphics, researchers have studied different geometric processing and rendering techniques to approximate input 3D meshes to 2D patches. Takahashi et al. [43] presented a heuristic approach to unfold 3D triangular meshes without shape distortions. Variation shape approximation [8] applied a mutual and repeated error-driven optimization strategy that provides polygonal plane proxies to best approximate a given 3D shape. Mitani and Hiromasa [32] used a set of triangular strips to approximate an input 3D mesh, while Shatz et al. [37] segmented the mesh into explicitly developable parts that can be cut and glued together. Traditional mesh segmentation [3, 22] and parametrization techniques [38] also provide the implicit mapping between 3D shape and 2D facets. However, all these methods result in a large number of planar segments that are impractical or difficult to join. In addition, physical construction and assembly constraints are rarely considered in these purely digital approximation techniques.

On the commercial side, many computational design tools have been developed for the user to import a 3D textured model and unfold it into flat sheets suitable for printing, such as Pepakura Designer, Autodesk® 123D™ Make (*folded panels module*), Paper Airplane Factory [42], and Paper folding 3D, shown in Fig. 2. In addition, online supportive communities such as Instructables [23] and Robot Living [27] are bringing commercial paper crafting and shipping services directly to customers. In general, all these systems and methods try to focus on the automatic fabrication process from an original mesh model. In our work, we seek a middle ground to empower the DIY community to prototype foldable and articulable shapes.

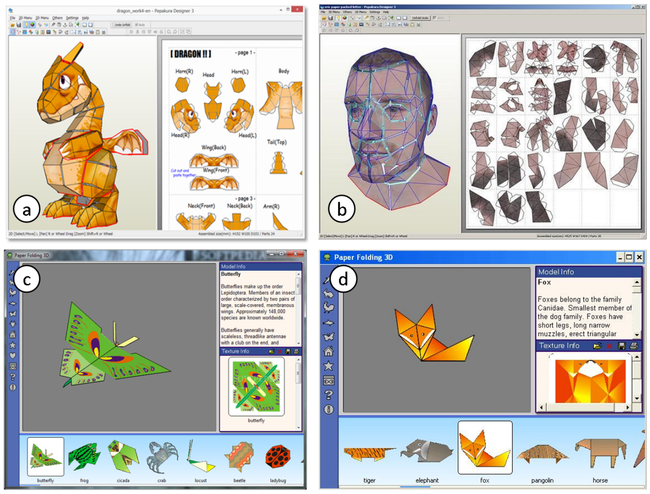

Figure 2. shows (a,b) 3D sculptural models generated using Pepakura Designer; (c,d) animal models generated using Paper folding 3D.

Fabrication-aware Shape Design

3D shape constructions using interlocked planar sections have been widely investigated for the ease of fabrication and assembly [13, 17, 36, 30, 4]. Flatfab [13] enables users to build their own models by sketching and assembling each planar slice one by one, while Crdbrd [17], Schwartzburg et al.'s [36], Slices [30] and Autodesk® 123D™ Make's *interlocked slices module* [4] can automatically convert a 3D model into planar slices. These proposed optimization algorithms derive sets of physical construction constraints to be satisfied in order to guarantee a rigid, stable and collision-free final construct. Nevertheless, the purpose of these methods is to generate a static and decorative object. In all these methods the resultant object has only one body with no joints that have motion and also they cannot house other components due to lack of interior spaces. Lau et al. [25] presented a formal grammar combined with lexical and structural analysis to generate fabricatable parts and connectors from a 3D furniture model. Another study focusses on the constraints and sequences of assemblies by creating geometric puzzles, such as Polyomino [28] and Burr puzzles [47] from a 3D model. Recently, Li et al. [26] developed an algorithm for computing paper architecture using pop-ups.

Some of the trends we notice in the recent work is the introduction of functionality into design and fabrication. Megaro et al. [31] and Coros et al. [9] proposed interactive systems for designing animated mechanical characters by kinematic synthesis based on the output trajectories or configurations specified by users, while FoldMecha [33] focuses on building linkage based toys made of paper. Tubes, ball socket joints and cuboids are embedded into given sculptural 3D models for housing of functionalities and articulation in [35], [5] and [14] and are fabricated by low cost 3D printing processes [15]. In contrast, we enable the creation of inexpensive foldable cardboard patterns for the Maker-DIY community from a wide variety of existing sculptural models.

Cardboard Crafting and Die-cutting

Cardboard, or carton board, is considered as the natural and recyclable material for constructing rapid prototypes and packaging consumer and food commodities [44, 18]. The typical structure consists of two flat panels coupled with a corrugating medium and the fibrous material to provide higher tensile strength and surface stiffness than regular craft papers. The cardboard material not only reduces the weight of the box, but also lends itself to the ease of manufacture, such as die-cutting. As the personal fabrication movement continues to lower the barrier of entry-level manufacturing systems, more and more portable desktop-scale and low-cost craft cutters and 3D printers have gained significant hobbyist, academic, and industry interest. We therefore find that cardboard is the suitable material for constructing 3D models that are found or created by users. In this work, we utilize the paper craft die-cutter to efficiently convert the digital crease patterns into flat cardboard prototypes.

CREATIVE DESIGN FRAMEWORK

CardBoardiZer produces customizable, articulated, and foldable prototypes directly from any digital 3D model. As shown in Fig. 3, our computational design platform workflow unfolds as follows: the designer (1) inputs a desired 3D mesh model, (2) customizes the segmented parts within the model to be articulated, (3) approximates the shape of each partition using a planar contour or an extruded volume, (4) augments the relative articulated movement between partitions, and then (5) develops the crease-cut-slot patterns ready to be die-cut and folded. The user is able to carry their creativity and intent towards the control of the number of articulated partitions, feature details, motion complexity, and the corresponding foldability.

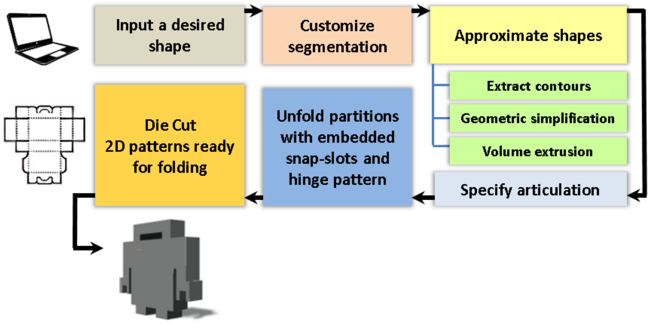

Figure 3. Building process of CardBoardiZer: given a 3D mesh model provided by user, the user customizes the partitions as desired, and enables each partition foldable using planar contour or extruded volumetric representation. The joint motion of articulated partitions are then specified and the system unfolds the crease-cut-slot patterns ready for die-cutting and folding.

Customizable Segmentation

Wong et al. [45] first developed the interactive mesh segmentation approach by specifying points on the cutting boundary and finding the shortest path connecting the points. Later, foreground / background snapping methods such as graph-cut [7], geodesic curvature flow [49], and region growing [46] also provides cutting boundaries to closely follow designers' specification.

In CardBoardiZer, we apply "dot scissor" [50] to capture local concave shape features using concavity-aware harmonic fields, and to select the best cutting boundaries using a voting scheme. The designer first specifies a stroke on the model where the partitioning curves are expected to pass through. A concavity-aware harmonic field is then computed by using the user's specified stroke as constraints. A set of candidate curves are computed upon the harmonic field via extracting iso-value curves. The same voting scheme as [50] is adopted to select the best partitioning curves according to their length and distance to user's stroke. Compared with original "dot scissor" approach, our system with strokes rather than dots representation, affords more of the user intent to be captured by the design process. Besides, the system also allows the users to continue interacting with strokes in the same region without actually partitioning the model. This scheme also increases the flexibility and convenience in using our tool.

Contour Representation

Contours are the basic representation of object shape since it contains explicit and dominant characteristics for determining an object's shape. As seen in Fig. 4, we provide a planar section extraction tool in our platform to cut each partition with a plane and obtain the resultant cross-sectional contour. To generate the initial cutting plane, the system applies the principal component analysis (PCA) so that the plane is created by taking the principal axis with the smallest eigenvalue as the normal and passing through the geometric center. If the initial plane is not satisfactory, a widget-based tool can be used to manipulate the cutting plane until it represents the shape as desired. The tool consists of two circular widgets to rotate the plane and a moving widget to translate the plane. Once the contour is selected, one can either choose to extrude the contour section along the normal vector of the cutting plane to create a prismatic model, or to retain the original planar shape. CardBoardiZer also allows for symmetrically tilting the prism surface with a non-uniform thickness. A sketch completion tool is provided to close the open contour and therefore a generated contour is not necessary to be a closed loop.

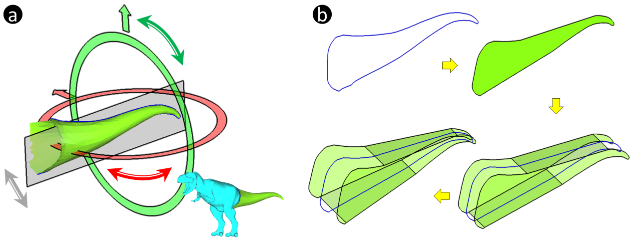

Figure 4. (a) The representative contour of the tail partition can be generated by a widget-based interactive tool, where two circular widgets are used to adjust the normal of a cutting plane. The cutting plane can be translated as well. (b) Once the contour is selected and closed, an extrusion operation is used to generate volumetric models. The tilting operation is for generating a model with non-uniform thickness.

Geometric Simplification for Leveraging Foldability and Shape Approximation

When considering unfolding, volumetric models that are extruded from highly curvy contours usually result in massive

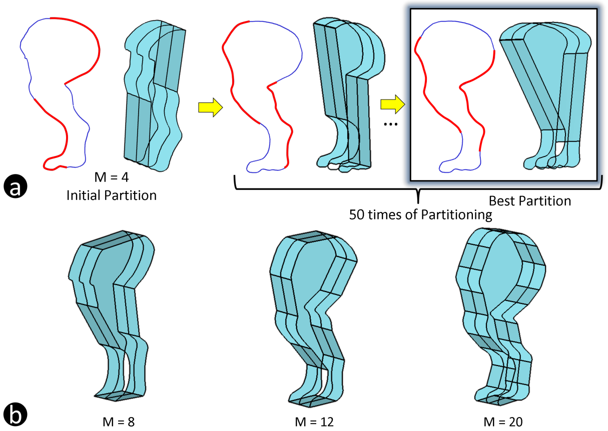

Figure 5. shows (a) when setting M to 4, our simplification generates a region partition with best score according to Eq.1 in the right upper corner. (b) by increasing M to 8, 12, and 20, the simplification results with improved shape approximation are obtained.

number of folding lines and makes the fabrication and assembly impractical. Therefore, it is necessary to geometrically simplify the contours before extrusion. The objectives of our simplification are twofold: 1) generate as little folding edges as possible to alleviate the construction burden, (2) approximate the shape of original curve as much as possible. Existing simplification algorithms such as [20, 40] gives a rough shape approximation and a limited number of retained edges. In these methods, the shape is isotropically coarsened with straight and curvy regions. In this section, we present a geometric simplification algorithm that simultaneously leverages the foldability and shape approximation.

The key idea of the algorithm is to classify the whole contour curve into ACSR, as shown in Fig. 5. Each straight region is approximated by a single line segment and will be extended later with connecting "side walls" to close up the volume, while the curvy regions are left open to preserve the curvy features of the contour. To ensure the folded model to be structurally integral, as a heuristic, we evenly distribute the straight and curved regions along the whole contour length.

To leverage the foldabilty and shape approximation, our simplification algorithm is summarized as follows:

1. **Initial region classification** All the points on each contour are parameterized using an arc length parameterization. Based on this parameterization, by inserting M evenly distributed anchor points, the whole contour can be divided into M regions from R_1 to R_M. The regions are specified into straight and curvy regions alternatively. If R_i is assigned as straight region, R_{i+1} $(i, i+1 \in [1, M])$ will be the curvy regions, and vice versa. This initial classification guarantees the distribution and length of each straight and curvy regions to be the same.

2. **Best classification search** Based on the results of initial region classification, we applied a search algorithm to find out the best region classification that preserves the original shape best. An error evaluation metric is defined as follows:

$$Score = \frac{\sum_{j\in[0,M]} \overline{chordal}(R_j)}{\sum_{k\in[0,M]} \overline{chordal}(R_k)}, \qquad (1)$$

   where R_j is a curvy region and R_k is a straight region. $\overline{chordal}(\cdots)$ measures the average chordal length error of each region. The classification search algorithm is designed to find out a classification with max score evaluated by Eq.1. By rotating the contour every small step δ in the parametric domain, we recursively parameterize the same contour and evaluate the score to select the best region classification. Note that, the searching stops when the rotation reaches $\frac{2\pi}{M}$ degrees due to the rotational symmetry of the region classification. The rotation step angle δ is set to $\frac{0.04\pi}{M}$ for balancing the classification quality and speed.

3. **Contour simplification** After a best region classification is selected, we perform the contour simplification by simply linking the starting and ending points of each straight regions. After extruding, the volumetric model will only have Ms set to be under 20, meanwhile the shape of the model is well preserved.

By selecting a different number for M, different levels of detail of the simplified models can be obtained. In this paper, we set M to be the power of 2 up to 16, thereby three levels of details of the simplified models. Besides, a pair of snap-slot patterns are added along each straight region to enclose the volumetric partitions.

Articulation Specification and Motion Hinge Synthesis

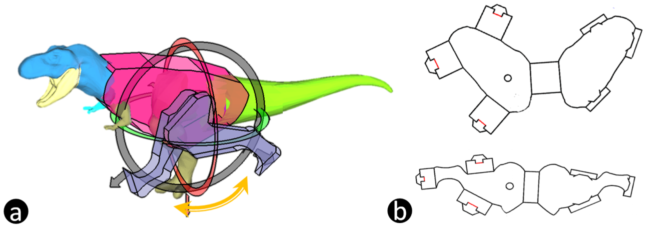

Figure 6. shows (a) a relative rotation specified using our widget based tool by selecting P_b, P_m and rotation axis on widget R. (b) the pre-synthesized motion hinge patterns are automatically added onto unfolded patterns.

Setting up arbitrary axes or pivots using traditional mechanical approaches is typically tedious and time-consuming. However, it is often the case that the desired manipulation constraint of an object exists in the candidate constraints of another scene object. Our system supports a simple interaction to let users to specify the relative motion between two parts. First, the user interactively specifies a partition P_b that serves as the fixed base, and the moving part P_m that rotates with respect to P_b (see Fig. 6 (a)). The rotation control widget R enables the user to specify which axis they like P_m to rotate about (shown in Fig. 6 (a)). The mating surface S_b on P_b and S_m on P_m can be determined by finding the closest surfaces on two parts. To ensure the relative motion, our system automatically adjusts the orientation of P_m such that two mating surfaces S_b and S_m are coplanar. Currently, our system relies on visual feedback for collision detection during the relative motion between two parts. Once the motion is defined, we assemble each pair of articulated parts P_b and P_m with a synthesized modular and easy-to-assemble motion hinge kit. (see Fig. 6 (b)).

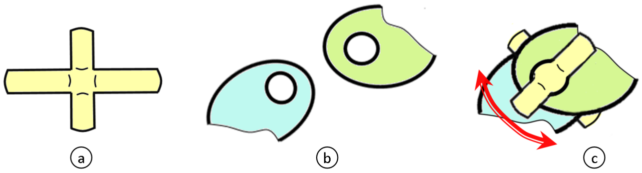

Figure 7. shows a cross-shape stripe (a) for connecting adjacent partitions (b) with revolute joint motion. (c) shows the complete assembly.

Each motion-hinge kit consists of a cross-shape 2D stripe and two circular holes on the patterns of adjacent partitions to be articulated, shown in Fig. 7. In order to generate a revolute motion between two partitions, the user needs to overlap the two patterns together where the holes are aligned to each other, bend the two opposite tips of the stripe towards the

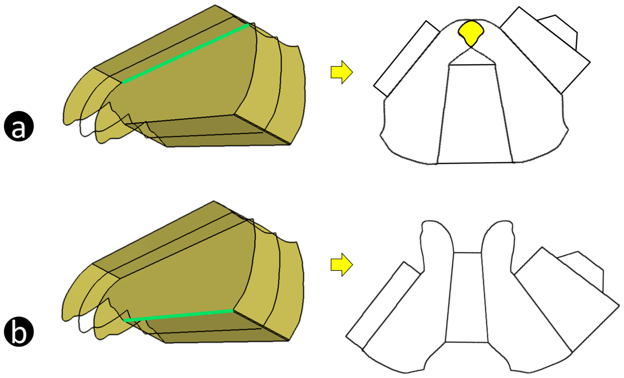

Figure 8. shows by selecting different straight line and curvy regions, overlapping issue occurs in case (a), but not in case (b)

middle, thread them into holes, and then release the tips on the other side.

Design of Unfolding

Our system unfolds each extruded volumetric shape into a 2D pattern with the motion hinges and snap-slot patterns. The design rationale for the unfolded pattern is twofold: 1) to ease the effort of assembly, the 2D pattern for each individual partition needs to be a single connected patch and 2) the pattern requires to be self-overlap-free so that all facets are cuttable. Our unfolding algorithm separates all the ACSR on the contour and leaving one pair of straight regions for connecting facets. During unfolding, the extruded volume can always be flattened ([12]) due to the foldability of polyhedra. Note that, the selection of unseparated straight regions cannot be arbitrary because the self-overlapping of facets might occur (see Fig. 8(a)). We thereby develop a separation algorithm to ensure the a self-overlap-free unfolding result. The algorithm consists of the following two steps:

1. **Edge sorting** All pairs of straight regions are sorted inside a queue Ψ in a descending order of the edge length.
2. **Edge separation & unfolding** One pair of straight region popped up from Ψ is labeled as unseparated while others as separated. Existence of self-overlapping are checked after unfolding.

Step 2 is repeated until Ψ is empty or there exists the unfolding without self-overlapping. In the scenario where self-overlapping areas can not be avoided, the algorithm assigns all straight regions as separated and thus the 2D pattern is separated into two pieces.

PROTOTYPE AND RESULTS

A 1.6 mm-thick single corrugated wall board was used as the substrate material and is shown in Fig. 9 after the 2D pattern is generated and cut. We selected this thickness to facilitate low-cost cutting and folding. We used the 24” Silver Bullet Die cutter Fig. 9(a) along with the 60-degree long blade to cut the crease and motion-hinge pattern. For the overall cutting setup, we use a relatively lower speed (200 mm/s) and higher force (588 grams) for the contour through-cutting, and a higher speed (800 mm/s) and lower force (470 grams) for depth-cutting of folds. Our system requires manual user involvement such as folding and assembling to construct the final prototype (Fig. 9(b)).

Prototypical Results

According to the CardBoardiZer results on a number of humanoid, animal and man-made 3D objects, our creative design framework allows different articulated features of the model to be quickly customized, folded and assembled using cardboard material. Fig. 10 shows our segmentation and prototypical results using 9 demonstrated examples, including 6 sculptural models (T-Rex, Apatosaurus, Michelangelo’s David, Stanford bunny, tree frog and tank) and 3 real life objects (desk lamp, clock and plier). In Fig. 11, we show the T-Rex model with 3 different levels of resolutions: M=4, M=8, M=12, respectively. The head and leg are illustrated with the major resolution differences due to many small curvy feature along the contours of this two parts.

Prototyping Time Evaluation

In order to understand the construction efficiency and folding capability using our system, we studied the speed of cutting, folding and assembly using cardboard as a sheet of material. A corrugated cardboard contains 3 sheets of Kraft paper attached together with each other. The internal I-beam structure provides rigidity and strength over the corrugations, while blade cutting and leaving only the bottom substrate layer makes it easy to fold. The 24” Silver Bullet Die cutter enables the through-cuts completed with 0.00625 s/mm for contours, and depth-cuts completed with 0.00125 s/mm. When carefully investigating the folding motion, we observe that typically one first bends the sheet along the pre-cut crease lines and presses on the overlapping side. Our experimental results shows these two motions typically take 1 to 2 seconds. Then the user quickly slides along the crease line within 0.2 ~ 0.4 seconds to complete the folds. Folding each snap into the slot for closing the volumetric partition takes an average of 2 ~ 4 seconds, while assembly the motion hinge takes 8 ~ 9 seconds. Table. 1 shows the time statistics of design, die-cutting, and folding (assembly) of our 11 demonstrated craft prototypes.

Usability Discussion and Comparison

Using CardBoardiZer, the designer is able to customize through the choice of geometries, articulation, joint motions and resolutions, quickly fabricate the patterns using cutters on demand, and complete the model through simple manual

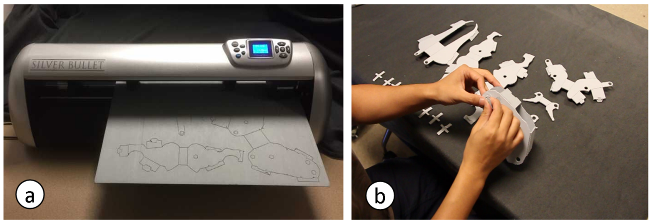

Figure 9. shows (a) 24” Silver Bullet Die cutter, and (b) the 2D T-Rex patterns when folding and assembling

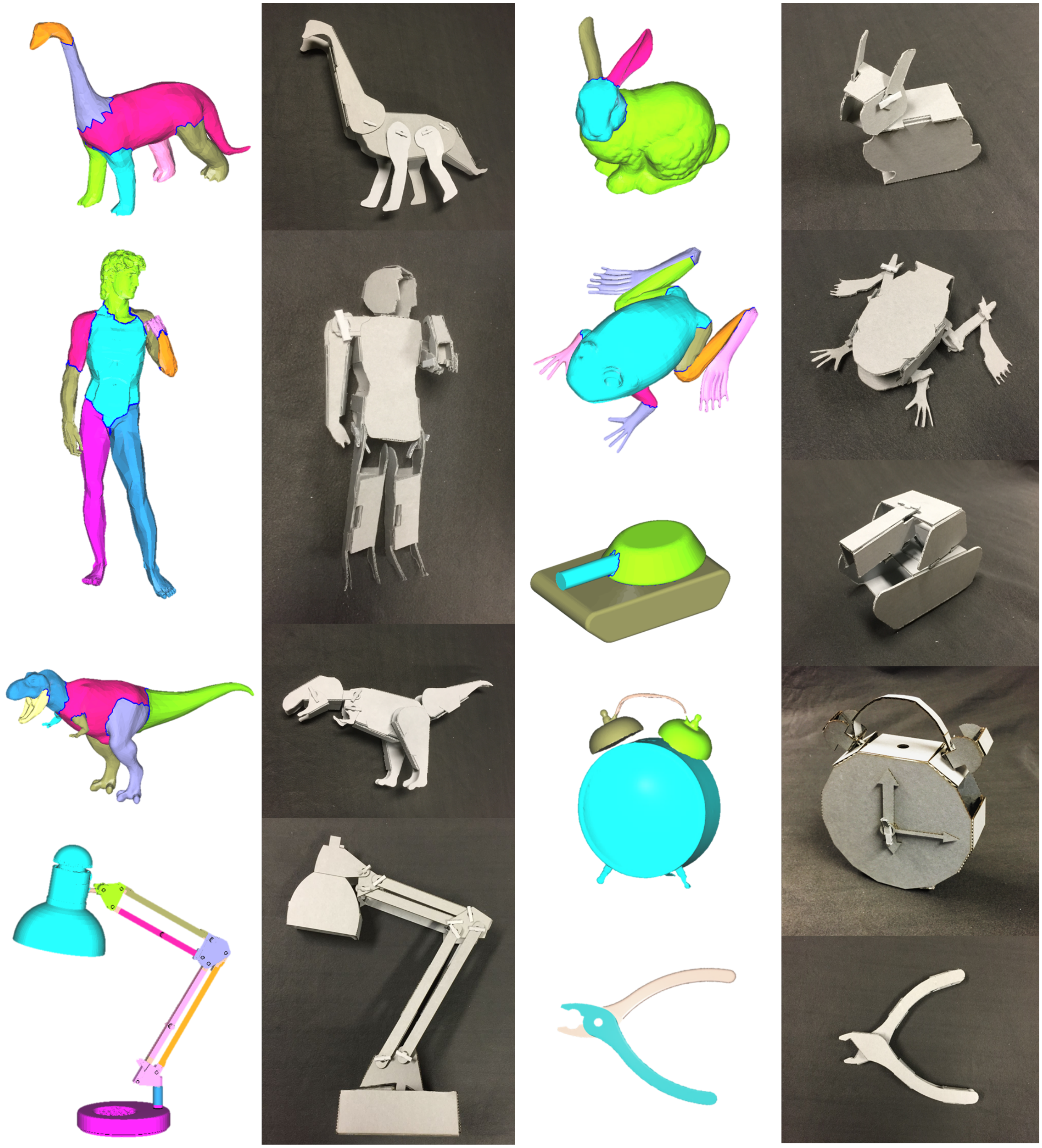

Figure 10. shows prototypical results generated by CardBoardiZer. First column from top down: Apatosaurus, Michelangelo's David, T-Rex and Desk lamp; and second column from top down: Stanford bunny, tree frog, tank, clock and plier.

folding and assembly. It is fast, friendly to use where the users only need to load the digital 3D model, segment the partitions as desired, specify the motions, and finally the system generates the 2D crease-cut-slot patterns ready for cutting, folding and articulation. Compared to traditional manual origami crafting methods, our method allows fast customization of desired shapes and augmented motion features, and fast prototyping by using die-cutting and folding approaches.

Table 1. Design, cutting and folding time statistic for different 3D models using CardBoardiZer

Models		Design Time (mm:ss)	Cutting Time (mm:ss)	Fold & Assembly Time (mm:ss)	Total Time (mm:ss)
T-Rex (47cm x 19cm x 9cm)	M=4	4:25	4:03	5:06	13:34
	M=8	5:38	7:51	7:33	21:02
	M=12	7:03	10:27	12:10	24:37
Apatosaurus (41cm x 44cm x 6cm)		2:19	2:12	1:58	6:29
Tree frog (14cm x 3cm x 18cm)		6:22	5:43	6:09	18:14
Stanford bunny (16cm x 16cm x 7cm)		2:24	1:47	2:01	4:51
Tank (18cm x 8cm x 8cm)		2:01	2:05	1:30	4:36
Michelangelo's David (25cm x 9cm x 4cm)		6:56	6:39	7:13	20:48
Desk lamp (54cm x 35cm x 17cm)		7:23	8:41	10:12	26:16
Clock (19cm x 15cm X 6cm)		1:17	2:20	1:40	5:17
Plier (27cm x 20cm x 3.5cm)		1:02	5:12	2:20	8:34

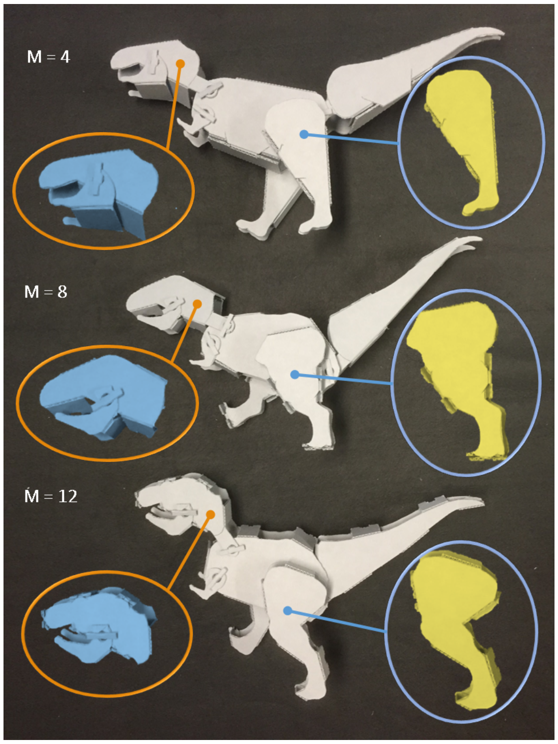

Figure 11. shows the time statistics for cutting and folding the T-Rex model with three different levels of resolutions (top down: M=4, M=8, M=12, respectively).

Our system is applicable for both novice and experienced designers who have basic computer operation skills. The interaction tools of CardBoardiZer are designed for ease of use and enable users to access complex geometric operations. Operations such as segmentation, contour generation and articulation specification, and shape control can be easily performed by simply stroking on model, adjusting a control widget and using a slider bar for different resolutions.

The selection of cardboard as our building material enables accessibility, experimentation and expressiveness by novice users. Cardboard is a low-cost everyday material that users are familiar with and can be easily accessed by novice users. Compared with other materials such as plastics, cardboard can be cut by low cost die-cutters instead of a more expensive laser cutter. The objects generated by CardBoardiZer are tinkerable in many ways: the objects can be easily adjusted and enhanced by users using color pens, scissors, glue and Velcro to paint, cut, make holes and attach other objects or decorative materials (e.g. wheels, levers, textiles, electronics and LED). Tinkering with objects generated by CardBoardiZer and other objects has multiple benefits for both learning and expression [34] as it invites broader participation and deepens the learning outcomes by allowing for a range of new solutions. Cardboard as a material for toys has gained significant attention from the maker community and is also gender neutral [48]. In 2005, cardboard box was inducted by Toy Hall of Fame, Strong National Museum of Play in Rochester and noted to be recognized, respected and remembered while having profoundly changed play or toy design [6].

Autodesk® 123D™ Make provides a variety of rapid prototyping functions for the user to fabricate cardboard-based designs with simplified shapes, including *interlocked slices* and *folded panels*. We herein carefully compare the workflow, ease of interaction, fabrication statistics as well as the final prototypical results using the *interlocked slices* (Fig. 12(a)) and *folded panels* (Fig. 12(b)) of Autodesk® 123D™ Make and our CardBoardiZer (Fig. 12(c)). The T-Rex model was selected for all three methods with identical scales (47cm x 19cm x 9cm) and comparable resolutions. The interlocked planar sections were designed to approximate the shape using successive orthogonal cross-sections. In Autodesk® 123D™ Make, the user is allowed to quickly manipulate the total number and orientation of the planar sections with slots. During the assembly, we found that the Autodesk® 123D™ Make system generates multiple slots along the concave shape regions, which makes it difficult to assemble. Overall, the *interlocked slices* method takes 6 mins to design the pattern, 18 mins to die cut the planar sections, and 38 mins to complete the assembly. Another function in Autodesk® 123D™ Make, *folded panels*, is similar to Pepakura and creates foldable patterns of a model by unfolding its 3D meshes into multiple patches and stripes. It is designed to approximate the shape by generating 2D folding patterns with a high number of folds (198 folds for T-Rex). However, it demands a very high effort and time to fold, assemble, as well as manual dexterity and patience. For example the T-Rex takes, in total 5 mins to design the pattern, 23 mins to die cut the patches, and 3 hours and 32 mins to complete

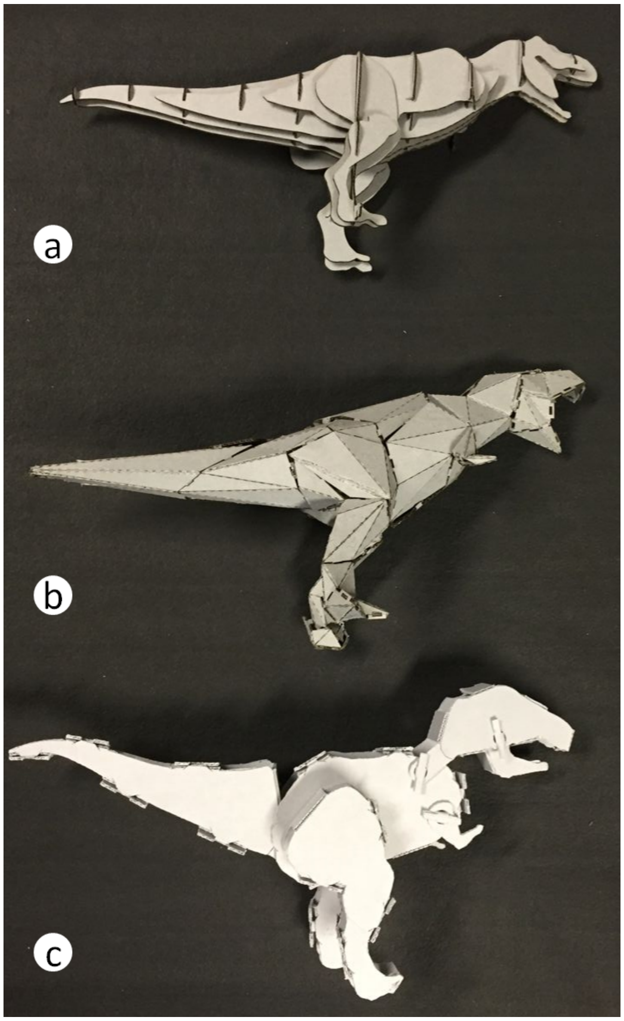

Figure 12. shows 3 fabricated T-Rex models, with identical scale and comparable resolutions, by (a) Autodesk® 123D™ Make *interlocked slices module* taking 6 mins for designing pattern, 18 mins for die cutting and 38 mins for assembly (in total 1 hr), (b) Autodesk® 123D™ Make *folded panels module* taking 5 mins for designing pattern, 23 mins for die cutting and 3 hrs and 32 mins for assembly (in total 4 hrs), and our CardBoardiZer taking 5 mins for designing pattern, 8 mins for die cutting and 7 mins for assembly (in total 20 mins).

the whole assembly. Our method, CardBoardiZer is designed to abstract the 3D shapes and approximate the individual bodies using a simple extruded cross-section. Compared to *folded panels*, CardBoardiZer reduces both the number of folds and time to fold. It also generates articulated features for the model that cannot be achieved using *interlocked slices*. Overall, CardBoardiZer requires 5 mins to design the pattern, 8 mins to die cut, and only 7 mins to fold up the model. The main difference is thus seen in the die cutting, assembly and the folding time. In particular the dexterity and patience to fold or assemble become extraordinarily high that most novices will not attempt crossing this barrier to entry. In addition the users need to keep track of a large number of individual parts and their location and sequence to be assembled.

LIMITATIONS, CONCLUSIONS AND FUTURE WORK

In this paper we present CardBoardiZer, a rapid design and prototyping system that allows a designer to customize, articulate, and fold everyday sculptural models. The computational design platform and prototyping pipeline are explicitly represented to ensure a rapid digital fabrication system. We demonstrate the usability of our system with a number of sculptural models. Currently the system is more suitable for processing shapes with reflectional symmetry, and preferably approximated with extruded features. Models with revolute bodies and massive curvy and protruding features limit our approach towards a good shape approximation. Because CarBoardiZer create shapes via extrusion from contours, models that are spherical-like or those created by revolution operations cannot be approximated well by CardBoardiZer. While repurposing is considered a part of the sharing DIY culture, the transformation from an original 3D model to the paper craft created by CardBoardiZer does raise potential ownership issues.

Several directions are possible for the future work such as understanding the learning that comes from the quick prototyping at early design, algorithmic enhancements to increase functionality, and enabling electromechanical functionalities. We plan to conduct a formal evaluation to further understand how CardBoradiZer helps the novice as well as expert designers to quickly design early prototypes and understand spatial and size constraints, elementary physics such as stability, motion and kinematics through iterative prototyping. CardBoardiZer affords options for further tinkering such as decoration, coloring, crafting and changing the prototype and these aspects will also be studied. Currently, we deal with structural integrity of the objects by heuristically setting the straight and curved regions to be evenly distributed. A more sophisticated strategy for structural stability analysis will further enhance the tool. Also the motion joints are not stable and the individually connected bodies cannot hold relative positions with respect to each other. We would like to address these problems by adding hinges. Another expansion option is towards systematic embedding strategies so that modular electronic components can be preassembled on the flat and encased inside the construction when folded. Integrating different gait patterns for the locomotion of the models would also make an interesting addition. Our CardBoardiZer design platform can serve as an enabler and inspiration for many derivative research works such as cardboard mechanotronics where circuits and sensors can be printed onto the substrate surfaces.

ACKNOWLEDGMENTS

The authors thank all the reviewers for providing valuable insights and suggestions that have helped in substantially improving this paper. This work was partially supported by the NSF IGERT on Sustainable Electronics (DGE 1144842) and by the Donald W. Feddersen Chair Professorship of the School of Mechanical Engineering, Purdue University. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors. The authors also would like to acknowledge Youyi Zheng from ShanghaiTech

University for sharing the code of “dot scissor”, and Ke Huo and Diogo Nazzetta from C Design Lab, Purdue University for their support.

REFERENCES

1. Chris Anderson. 2010. The new industrial revolution. *Wired magazine18* 2 (2010).
2. Esther M Arkin, Michael A Bender, Erik D Demaine, Martin L Demaine, Joseph SB Mitchell, Saurabh Sethia, and Steven S Skiena. 2004. When can you fold a map? *Computational Geometry* 29, 1 (2004), 23–46.
3. Marco Attene, Sagi Katz, Michela Mortara, Giuseppe Patané, Michela Spagnuolo, and Ayellet Tal. 2006. Mesh segmentation-a comparative study. In *Shape Modeling and Applications, 2006. SMI 2006. IEEE International Conference on*. IEEE, 7–7.
4. Autodesk. 2014. Autodesk 123D Make. (2014). Retrieved September 17, 2015 from `http://www.123dapp.com/make`.
5. Moritz Bächer, Bernd Bickel, Doug L. James, and Hanspeter Pfister. 2012. Fabricating Articulated Characters from Skinned Meshes. *ACM Trans. Graph.* 31, 4, Article 47 (July 2012), 9 pages.
6. James Barron. 2015. New Inductees at Toy Hall of Fame Join the Ball, the Box and the Barbie Doll. *New York Times* (2015), Page A18.
7. Steven Brown, Bryan Morse, and William Barrett. 2009. *Interactive part selection for mesh and point models using hierarchical graph-cut partitioning*. Canadian Information Processing Society.
8. David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun. 2004. Variational shape approximation. *ACM Transactions on Graphics (TOG)* 23, 3 (2004), 905–914.
9. Stelian Coros, Bernhard Thomaszewski, Gioacchino Noris, Shinjiro Sueda, Moira Forberg, Robert W. Sumner, Wojciech Matusik, and Bernd Bickel. 2013. Computational Design of Mechanical Characters. *ACM Trans. Graph.* 32, 4, Article 83 (July 2013), 12 pages.
10. E Demaine and Tomohiro Tachi. 2010. Origamizer: A practical algorithm for folding any polyhedron. (2010).
11. Erik D Demaine, Martin L Demaine, and Joseph SB Mitchell. 1999. Folding flat silhouettes and wrapping polyhedral packages: New results in computational origami. In *Proceedings of the fifteenth annual symposium on Computational geometry*. ACM, 105–114.
12. M. P. do Carmo. 1976. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, NJ.
13. Flatfab. 2014. Flatfab. (2014). Retrieved September 17, 2015 from `http://www.flatfab.com`.
14. Wei Gao, Yunbo Zhang, Diogo C. Nazzetta, Karthik Ramani, and Raymond J. Cipra. 2015a. RevoMaker: Enabling Multi-directional and Functionally-embedded 3D Printing Using a Rotational Cuboidal Platform. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST ’15)*. ACM, New York, NY, USA, 437–446.
15. Wei Gao, Yunbo Zhang, Devarajan Ramanujan, Karthik Ramani, Yong Chen, Christopher B. Williams, Charlie C.L. Wang, Yung C. Shin, Song Zhang, and Pablo D. Zavattieri. 2015b. The status, challenges, and future of additive manufacturing in engineering. *Computer-Aided Design* 69 (2015), 65 – 89.
16. Steven Gray, Nathan Zeichner, Vijay Kumar, and Mark Yim. 2011. A Simulator for Origami-Inspired Self-Reconfigurable Robots. In *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education*. CRC Press, 323.
17. Kristian Hildebrand, Bernd Bickel, and Marc Alexa. 2012. crdbrd: Shape fabrication by sliding planar slices. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 583–592.
18. Dennis Hine. 1999. *Cartons and cartoning*. Pira International.
19. Margaret Honey and David E Kanter. 2013. *Design, make, play: Growing the next generation of STEM innovators*. Routledge.
20. Pu Huang, Charlie CL Wang, and Yong Chen. 2013. Intersection-free and topologically faithful slicing of implicit solid. *Journal of Computing and Information Science in Engineering* 13, 2 (2013), 021009.
21. Scott E Hudson and Jennifer Mankoff. 2006. Rapid construction of functioning physical interfaces from cardboard, thumbtacks, tin foil and masking tape. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*. ACM, 289–298.
22. Dan Julius, Vladislav Kraevoy, and Alla Sheffer. 2005. D-Charts: Quasi-Developable Mesh Segmentation. In *Computer Graphics Forum*, Vol. 24. Wiley Online Library, 581–590.
23. Squid Labs. 2005. Instructables. (2005). Retrieved September 17, 2015 from `http://www.instructables.com`.
24. Robert J Lang. 1998. Treemaker 4.0: A Program for Origami Design. (1998).
25. Manfred Lau, Akira Ohgawara, Jun Mitani, and Takeo Igarashi. 2011. Converting 3D furniture models to fabricatable parts and connectors. In *ACM Transactions on Graphics (TOG)*, Vol. 30. ACM, 85.
26. Xian-Ying Li, Chao-Hui Shen, Shi-Sheng Huang, Tao Ju, and Shi-Min Hu. 2010. Popup: automatic paper architectures from 3D models. *ACM Transactions on Graphics-TOG* 29, 4 (2010), 111.
27. Robot Living. 2007. Robot Living. (2007). Retrieved September 17, 2015 from `http://www.robotliving.com`.

28. Kui-Yip Lo, Chi-Wing Fu, and Hongwei Li. 2009. 3D polyomino puzzle. *ACM Transactions on Graphics (TOG)* 28, 5 (2009), 157.

29. Tama Software Ltd. 2014. Pepakura Designer. (2014). Retrieved September 17, 2015 from `http://www.tamasoft.co.jp/pepakura-en`.

30. James McCrae, Karan Singh, and Niloy J Mitra. 2011. Slices: a shape-proxy based on planar sections. *ACM Trans. Graph.* 30, 6 (2011), 168.

31. Vittorio Megaro, Bernhard Thomaszewski, Damien Gauge, Eitan Grinspun, Stelian Coros, and Markus Gross. 2014. ChaCra: An Interactive Design System for Rapid Character Crafting. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '14)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 123–130.

32. Jun Mitani and Hiromasa Suzuki. 2004. Making papercraft toys from meshes using strip-based approximate unfolding. In *ACM Transactions on Graphics (TOG)*, Vol. 23. ACM, 259–263.

33. Hyunjoo Oh, Mark D. Gross, and Michael Eisenberg. 2015. FoldMecha: Design for Linkage-Based Paper Toys. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15 Adjunct)*. ACM, New York, NY, USA, 91–92.

34. Kylie Peppler. 2013. STEAM-powered computing education: Using e-textiles to integrate the arts and STEM. *Computer* 9 (2013), 38–43.

35. Valkyrie Savage, Ryan Schmidt, Tovi Grossman, George Fitzmaurice, and Björn Hartmann. 2014. A Series of Tubes: Adding Interactivity to 3D Prints Using Internal Pipes. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 3–12.

36. Yuliy Schwartzburg and Mark Pauly. 2013. Fabrication-aware Design with Intersecting Planar Pieces. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, 317–326.

37. Idan Shatz, Ayellet Tal, and George Leifman. 2006. Paper craft models from meshes. *The Visual Computer* 22, 9-11 (2006), 825–834.

38. Alla Sheffer, Emil Praun, and Kenneth Rose. 2006. Mesh parameterization methods and their applications. *Foundations and Trends® in Computer Graphics and Vision* 2, 2 (2006), 105–171.

39. Softonic. 2015. Paper Folding 3D. (2015). Retrieved September 17, 2015 from `http://origami-windows-8.en.softonic.com/download?ex=SWH-1608.5`.

40. Milan Sonka, Vaclav Hlavac, and Roger Boyle. 2014. *Image processing, analysis, and machine vision*. Cengage Learning.

41. NGSS Lead States. 2013. Next Generation Science Standards: For States, By States. (2013).

42. North Shore-LIJ Health System. 2015. Paper Airplane Factory. (2015). Retrieved September 17, 2015 from `http://www.ihi.org/education/ihiopenschool/resources/Pages/Activities/PaperAirplaneFactory.aspx`.

43. Shigeo Takahashi, Hsiang-Yun Wu, Seow Hui Saw, Chun-Cheng Lin, and Hsu-Chun Yen. 2011. Optimized topological surgery for unfolding 3D meshes. In *Computer Graphics Forum*, Vol. 30. Wiley Online Library, 2077–2086.

44. Diana Twede and Ron Goddard. 1998. *Packaging materials*. Pira International Leatherhead.

45. Kevin Chun-Ho Wong, Tommy Yu-Hang Siu, Pheng-Ann Heng, and Hanqiu Sun. 1998. Interactive volume cutting. In *Graphics Interface*, Vol. 98.

46. Huai-Yu Wu, Chunhong Pan, Jia Pan, Qing Yang, and Songde Ma. 2007. A sketch-based interactive framework for real-time mesh segmentation. In *Computer graphics international*.

47. Shiqing Xin, Chi-Fu Lai, Chi-Wing Fu, Tien-Tsin Wong, Ying He, and Daniel Cohen-Or. 2011. Making burr puzzles from 3D models. *ACM Transactions on Graphics (TOG)* 30, 4 (2011), 97.

48. Sang Ho Yoon, Ansh Verma, Kylie Peppler, and Karthik Ramani. 2015. HandiMate: Exploring a Modular Robotics Kit for Animating Crafted Toys. In *Proceedings of the 14th International Conference on Interaction Design and Children (IDC '15)*. ACM, New York, NY, USA, 11–20.

49. Juyong Zhang, Chunlin Wu, Jianfei Cai, Jianmin Zheng, and Xue-cheng Tai. 2010. Mesh snapping: Robust interactive mesh cutting using fast geodesic curvature flow. In *Computer Graphics Forum*, Vol. 29. Wiley Online Library, 517–526.

50. Youyi Zheng, Chiew-Lan Tai, and Oscar Kin-Chung Au. 2012. Dot scissor: a single-click interface for mesh segmentation. *Visualization and Computer Graphics, IEEE Transactions on* 18, 8 (2012), 1304–1312.