FoldTronics: Creating 3D Objects with Integrated Electronics Using Foldable Honeycomb Structures

Junichi Yamaoka*
MIT CSAIL
Cambridge, MA, USA
junichiy@mit.edu

Mustafa Doga Dogan
MIT CSAIL
Cambridge, MA, USA
doga@mit.edu

Katarina Bulovic
MIT CSAIL
Cambridge, MA, USA
kbulovic@mit.edu

Kazuya Saito
The University of Tokyo
Bunkyo-ku, Tokyo, Japan
ksaito@akg.t.u-tokyo.ac.jp

Yoshihiro Kawahara
The University of Tokyo
Bunkyo-ku, Tokyo, Japan
kawahara@akg.t.u-tokyo.ac.jp

Yasuaki Kakehi
The University of Tokyo
Bunkyo-ku, Tokyo, Japan
kakehi@iii.u-tokyo.ac.jp

Stefanie Mueller
MIT CSAIL
Cambridge, MA, USA
stefanie.mueller@mit.edu

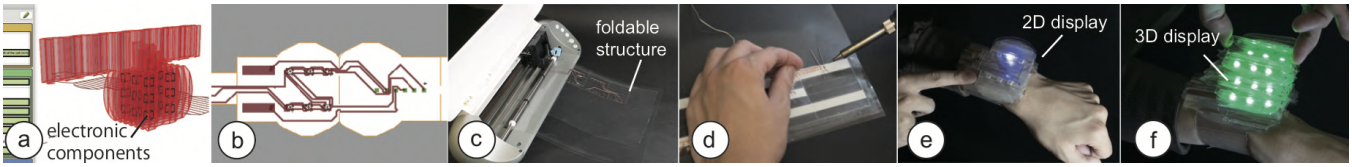


Figure 1: FoldTronics creates 3D objects with integrated electronics. Our design software allows users to: (a) Convert a 3D model into a honeycomb structure & place electronic components into it. (b) Export the 2D layers (honeycomb pattern, wiring, insulation). (c) Fabricate the layers on a cutting plotter using sheets of plastic, copper, and regular tape. (d) After soldering and folding, the object is ready to be used: (e, f) This smart watch has a 2D display when collapsed and a 3D volumetric display (3 layers, 6 LEDs each) when expanded – the LEDs and wiring are integrated *inside* the object.

ABSTRACT

We present *FoldTronics*, a 2D-cutting based fabrication technique to integrate electronics into 3D folded objects. The key idea is to cut and perforate a 2D sheet to make it foldable into a honeycomb structure using a cutting plotter; before folding the sheet into a 3D structure, users place the electronic components and circuitry onto the sheet.

The fabrication process only takes a few minutes enabling users to rapidly prototype functional interactive devices. The resulting objects are lightweight and rigid, thus allowing for weight-sensitive and force-sensitive applications. Finally, due to the nature of the honeycomb structure, the objects can be folded flat along one axis and thus can be efficiently transported in this compact form factor.

We describe the structure of the foldable sheet, and present a design tool that enables users to quickly prototype the desired objects. We showcase a range of examples made with our design tool, including objects with integrated sensors and display elements.

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**.

KEYWORDS

personal fabrication; 2D folding; electronics;

ACM Reference Format:
Junichi Yamaoka, Mustafa Doga Dogan, Katarina Bulovic, Kazuya Saito, Yoshihiro Kawahara, Yasuaki Kakehi, and Stefanie Mueller. 2019. FoldTronics: Creating 3D Objects with Integrated Electronics Using Foldable Honeycomb Structures. In *CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019), May 4–9, 2019, Glasgow, Scotland UK*. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3290605.3300858

*Also with The University of Tokyo.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CHI 2019, May 4–9, 2019, Glasgow, Scotland UK
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-5970-2/19/05…$15.00
https://doi.org/10.1145/3290605.3300858

1 INTRODUCTION

While many of today's fabrication processes allow users to fabricate passive 3D objects, fabricating 3D objects with integrated electronics that allow for interactive applications remains challenging [2]. For instance, while it is now possible to 3D print the wires of a circuit using silver ink (e.g., *Voxel8*[1]), there is only little progress in fabricating the actual electronic components themselves (see Lewis et al. for a first approach to 3D print an LED [19]).

Since 3D printing is not yet at a level to make fully functional objects, researchers are looking into alternative fabrication methods. For instance, *ObjectSkin* [12] dips the object into a water bath with a film that contains circuitry; *ActMold* [40] places electronic circuitry onto a plastic sheet prior to draw-forming a 3D structure from it, and *Foldio* [27] prints circuitry onto a 2D sheet that the user can cut and fold. However, those methods only apply electronics to the surface of an object.

Recently, *PEP* [26] showed how to integrate electronics into the *inside* of 3D objects by cutting and stacking paper sheets that were previously enhanced with electronic components. However, since *PEP* fabricates objects by stacking paper sheets, the approach is limited to thin-film electronics, such as flat coils and RFID tags, that do not interfere with the stacking process. In addition, the resulting objects are solid inside, making it difficult to apply the approach to lightweight and portable applications.

In this paper, we take a different approach: Inspired by existing methods that fold passive 2D sheets into honeycomb structures [32] to create rigid and lightweight objects, we propose a new fabrication technique that combines honeycomb structures with integrated electronics (Figure 1). Our method goes beyond existing origami-electronics applications (e.g., origami robot [9] & antenna [42]) by providing a fabrication pipeline supported by a design tool that facilitates the creation of interactive objects. Users only have to use a regular cutting plotter to cut the 2D sheets into a specific pattern, add the electronic components, and then fold the structure into its final 3D shape.

2 RELATED WORK

Our work is related to literature in HCI on making 3D structures from 2D sheets and on integrated electronics.

Fabricating Interactive 3D Objects

Several approaches have been explored to add interactivity to 3D objects. For instance, by inserting three-axis wireless accelerometers into 3D printed objects, Hook et al. [14] enable those objects to detect input from the user. By inserting a series of pipes into 3D printed objects and adding sensors and actuators to the end of the tubes, Savage et al. [34] enable objects to sense the user's interaction with the object and to actuate them accordingly. Many more approaches have been developed — a review can be found in Ballagas et al. [1]. Closest to our approach is *PrintGami* [7], which enables users to create interactive 3D printed objects by inserting paper circuits into passive printed objects during the printing process. All of these approaches are limited in the type of sensing and actuation they can integrate *inside* the 3D objects. Our approach, starting with a 2D sheet that is subsequently folded into a 3D shape, can house any type of sensor and actuator.

Cutting & Folding to Make 3D Objects

Many research projects have investigated how to make 3D objects from 2D pieces either by stacking sheets (*Autodesk 123D*[2], *LaserStacker* [36], *Layered Fabric printer* [30]), intersecting and interlocking 2D pieces (*Platener* [3], *SketchChair* [33], *FlatFitFab* [22]), or most related to our work through folding (*Making PaperCraft Toys* [23], *LaserOrigami* [24], *CurvedFolding* [17]). We use folding for our work since it requires only a single 2D sheet to make 3D objects. In addition, since we use honeycomb structures that leave a sparse volume, folding allows for application scenarios that require lightweight and portable structures (e.g., wearables, mobile applications).

2D Printing of Electronics on Sheets

One of the most popular approaches to apply circuitry onto 2D sheets is conductive inkjet printing of silver ink [11, 16]. This approach has been employed by many papers in the related work to fabricate the circuitry for displays (*PrintScreen* [28]), sensors (*Foldio* [27]), and actuators (*uniMorph* [13]). However, since inkjet-printed traces are fragile, we found that they are not robust to withstand folding. Thus, similar to the objects illustrated in *Midas* [35], we use the more robust conductive copper tape to create the circuitry of our objects and cut it into shape with a cutting plotter. *ShrinkyCircuits* [21] applies heat to plastic sheets to improve electrical characteristics of hand-drawn circuits.

Stacking Functional Layers on 2D Sheets

Similar to other related work, we do not only use a single functional layer on the 2D sheet but stack multiple layers. Related to our work is *Printem* [5], a five-layer sheet that allows for fast production of circuit boards using only a 2D printer and a UV light. Similarly, *Foldem* [6] is a sheet consisting of bendable and rigid layers that when cut selectively, allows for varying flexibility in each region of the sheet. Finally, *BlowFab* [41] is a 3-layer sheet made from Kapton Film,

[1] http://store.voxel8.com/product-catalog/standard-silver-ink-3-cc

[2] https://www.autodesk.com/solutions/123d-apps

PET, and masking tape that when selectively engraved and cut can be used to create inflatable objects. We add to this work by proposing a new multi-layer architecture to create functional foldable honeycomb structures with integrated electronics.

Honeycomb Structures

Honeycomb structures offer high strength and rigidity and are therefore used for various industrial products ranging from aircraft and space satellites to furniture (chairs, shelves) [4]. In addition, honeycomb structures have many functional properties that allow them to be used for heat insulation [18], sound insulation [31], rectification [4] and impact absorption [39]. Since our proposed technique and software make it possible to integrate various electronic devices into a honeycomb structure, we allow designers to build a variety of functional interfaces that harness the honeycomb's excellent mechanical and functional properties.

3 BASICS OF INTERACTIVE HONEYCOMB STRUCTURES

For our method, we employ a standard kirigami honeycomb cutting and folding method [32]. We will briefly describe how this works and how we extend it to integrate electronics.

Basic Folding Principle

Figure 2 illustrates how we can fold honeycomb cells by cutting and folding paper.

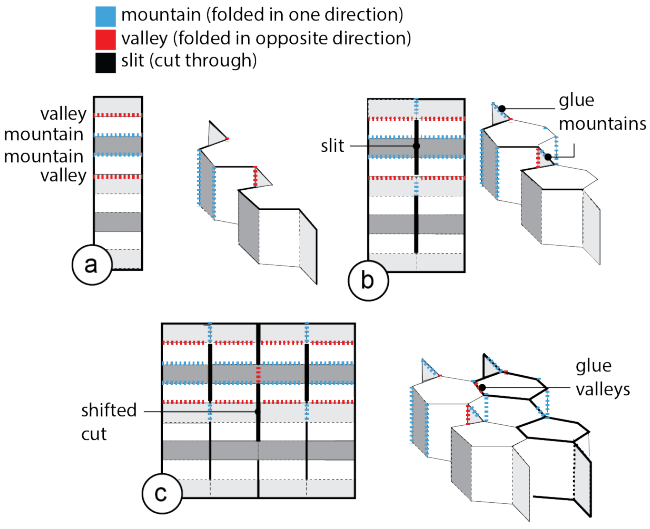


Figure 2: Folding honeycomb structures: (a) half a column of cells, (b) column of cells, (c) multiple columns.

Creating half a cell column: Each column of honeycomb cells consists of mountains and valleys. Mountains are lines that are bent in one direction, valleys are lines that are bent in the opposing direction. This causes the strip to fold into its characteristic shape. Figure 2a shows the strip pattern to create half a cell column from mountains and valleys.

Creating a full cell column: To create both sides of a honeycomb cell column, we duplicate the previously described pattern (Figure 2b). The center line connecting the two half-columns needs to be designed as follows: A slit is placed along all center line segments that do not connect to two valleys (they will form the hollow cell), all line segments connecting two valleys use a mountain line (they form glue surfaces that will hold the structure together after folding, see Figure 2b).

Creating multiple cell columns: Finally, to create a second cell column, we can use the cutting pattern shown in Figure Figure 2c. Again, we duplicate the pattern from the previous image and place it right next to the existing structure. In addition, where the two cell columns meet, we now place slits whenever the center line segments do not connect to two mountains (to form the hollow cells between the two columns, see the hollow cell in Figure 2c). All line segments that do connect to two mountains now use a valley line (they form the glue surfaces that hold the two columns together). To create more than two columns, we can repeat this pattern as many times as required.

Cell Size: The size of the cell determines the resolution of the honeycomb structure (Figure 3). This is not only important to represent the original shape of the 3D object as accurately as possible, but also for the placement and wiring of electronic components: Since the electronic components are placed inside the cells (Figure 3b), the cells need to have at least the same size as the components. Similarly, since the wires are placed alongside the walls of the cells, the larger the cell, the more wires can be routed to an electronic component. Thus, there is a trade-off between honeycomb resolution and size of electronic components. Smaller electronic components work better since they allow for a higher resolution. Each cell can also house multiple electronic components as long as they fit inside the cell.

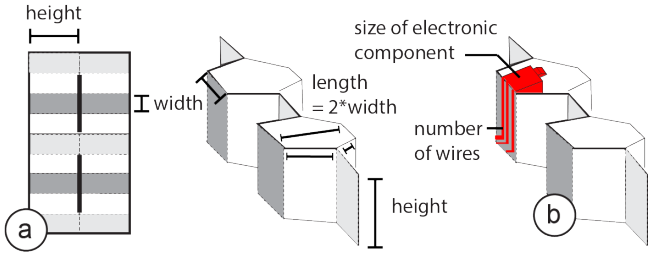


Figure 3: (a) The cell size (length, width, height) determines (b) the size of the electronic components and numbers of wires that can be placed.

Placing Electronic Components

When folding the honeycomb, some cells are formed from the top side of the sheet while others are formed from the bottom side of the sheet. Since we only support one-sided circuit boards, we can only place electronic components into those cells made from the top of the sheet (Figure 4).

In addition, electronic components can only be placed onto surfaces that are not being glued together. On the top side of the sheet, the mountain surfaces become glued and thus cannot be used for placing electronic components (see crossed out side faces in Figure 4b).

In our design tool, we highlight which cells can be used for placing electronic components. For instance, users receive a warning when placing a component as shown in Figure 4b since the component is housed in an invalid cell made from the bottom of the sheet and the wiring is on a mountain surface that would be glued when folded.

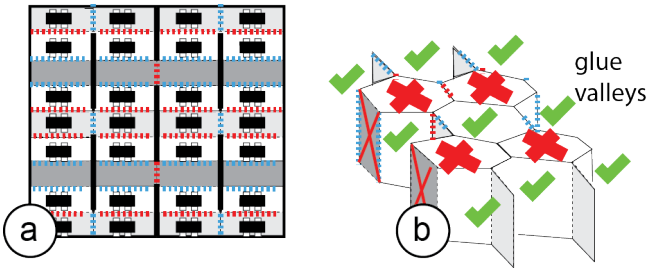


Figure 4: (a) Electronic components can only be placed on the surfaces that are not being glued, and (b) only into cells that are formed from the top of the sheet.

Circuit Layout on Honeycomb Foldable Structures

Finally, the wires to connect the electronic components can be placed onto any surface that is also suitable for an electronic component (i.e., it is neither on the back of the sheet nor being glued together). We prevent short circuiting of wires as a result of folding by adding an insulating sheet in a later step of the pipeline (see section "fabrication process").

Cross-Cell Circuit Connector: As mentioned previously, the cell size determines how many wires can be routed to a cell. For example, as shown in Figure 5a, when the width of a cell is 7 mm and the width of the wire is 1 mm, only 4 wires can reach a cell since we also need to leave a 1mm gap between each wire. To allow for more wires, we created the wiring method shown in Figure 5b: This wiring method leverages the fact that after folding, the mountains will be connected since they are being glued together. When the mountains become glued, seemingly unconnected wires become connected and the circuit is closed. Figure 5 shows the results: rather than (a) just 4 wires, our new method (b) allows for 8 wires. However, while regular wiring can be placed on any surface, our cross-cell circuit connector only works on mountains since other surfaces do not become glued together.

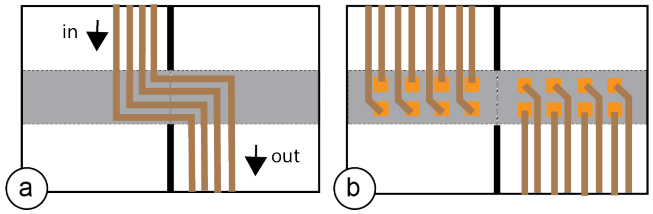


Figure 5: (a) Traditional wiring method. (b) Our new wiring allows for more wires due to folding.

4 FOLDTRONICS PIPELINE

Next, we describe how users can create interactive objects with integrated electronics using the FoldTronics design software and fabrication pipeline. We will illustrate all steps at the example of making a simple LED circuit with one cross-cell circuit connector and a battery. We chose this simple example for the step-by-step illustration and will demonstrate more complex use cases in section "Scenarios."

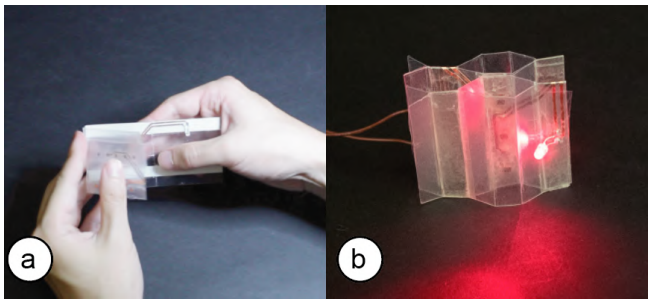


Figure 6: (a, b) Walk-through example: functional LED circuit with one cross-cell circuit connector and a battery.

Creating interactive foldable 3D objects, such as the functional LED circuit with battery, requires three steps:

(1) Device Design: Users start by 3D modeling their device, then convert the 3D model into a foldable honeycomb structure using our conversion software, and place the electronics onto the cells facilitated by our software's guidance tools.

(2) Export for Fabrication: Next, users export their design for fabrication on a regular cutting plotter. All layers are exported "fabrication-ready" except for the wiring layer, which requires users to open it in the electronics design software EAGLE to use its auto-wiring function.

(3) Fabrication & Assembly: Finally, users use the cutting plotter to fabricate the different layers, then solder on the electronics and fold the shape into its 3D structure.

The software for our design tool can be found on GitHub[3]. A full list of materials and a step-by-step description of the process can also be found in our FoldTronics *Instructables* tutorial[4].

#1 Device Design using FoldTronics User Interface

To create the LED circuit, we start by creating a 3D model in the 3D editor Rhino3D for which we implemented our FoldTronics plugin (Figure 7a). After creating the basic shape of the 3D model, we convert it into a honeycomb structure by pressing the "convert" button. As soon as the algorithm split the model into the honeycomb cells, the result is displayed in the 3D view (Figure 7b).

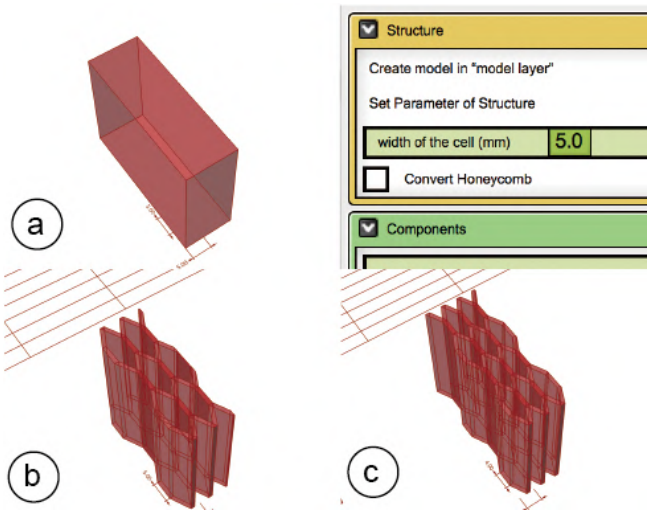


Figure 7: (a) Converting the 3D model into (b, c) honeycombs with different resolutions.

We can now vary the resolution of the honeycomb using the provided slider to find the best trade-off between higher resolution and having enough space in the cells to house the LED, the battery, and the cross-cell circuit connector. Figure 7b/c show different resolutions (5mm vs. 4mm for the width of the cells) for this purpose. The resolution slider changes both the number of columns and the number of cells simultaneously because changing the resolution for columns and rows separately would cause the final shape to differ from the original shape.

To add the LED, battery, and cross-cell circuit connector, we select them from the list of components from the menu and add them by clicking the respective button (Figure 8a). This automatically creates a 3D model of a box representing the size of the selected electronic component (Figure 8b). We can now drag the LED and other electronic components to a location in the 3D volume. In case we accidentally place a component onto a fold or a non-valid cell, it is automatically relocated to the next valid cell (Figure 8b).

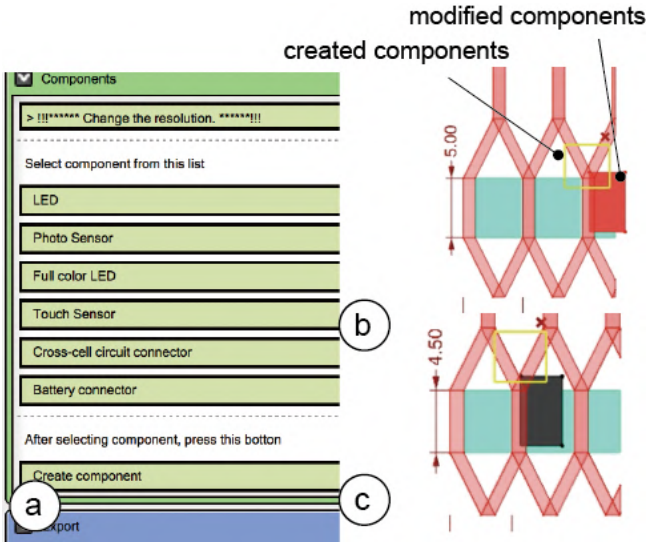


Figure 8: (a) Adding electronic components from the menu, (b) relocating components to valid cells, (c) color-coding components that are too large.

If the width of the electronic component, such as the LED, is too large (i.e., larger than the width of the cell), the color of the electronic component changes to black and a warning is displayed on the menu (Figure 8c).

We are done with placing the components. Before exporting, we use the "Collapsing Simulation" slider to preview how the design will look when compressed and expanded again.

#2 Export for Fabrication

Once we are done with placing the electronic components, we hit the "export" button to generate the layers for fabrication. As can be seen in Figure 9, on export, our 3D editor plugin creates all layers of the fabrication stack as 2D drawing files (.DXF file format) except the layer that contains the wiring, which we will create now as a seperate step in the process.

To generate the missing wiring layer, users open the 2D file of the honeycomb structure in the electronic design software EAGLE and execute our custom EAGLE ULP plugin. As can be seen in Figure 10, the plugin (a) generates a circuit board the size of the honeycomb pattern and then converts each colored square back into an electronic component (i.e. the LED, battery, and cross-cell circuit connector). (b) With the electronic components already on the sheet, users can now

[3]https://github.com/eratoimf/foldtronics
[4]https://www.instructables.com/id/FoldTronics-Creating-3D-Objects-With-Integrated-El/

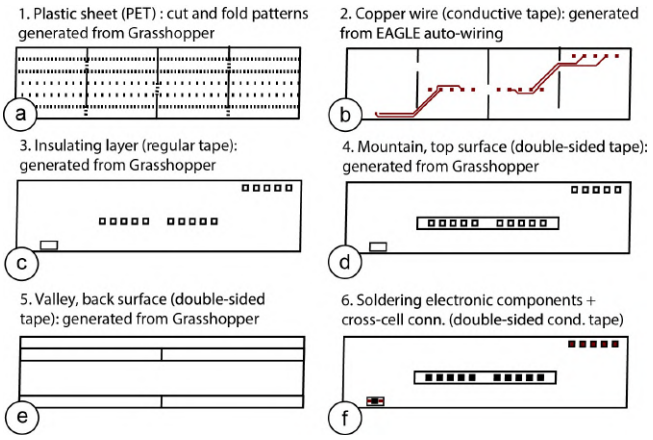


Figure 9: Individual layers generated using Grasshopper and EAGLE to create a honeycomb structure with integrated electronics.

build the schematic. (c) Finally, users can use EAGLE's auto-wiring function to create the full circuitry on the sheet, and then export the design as .DXF file, which provides the last missing layer for fabrication.

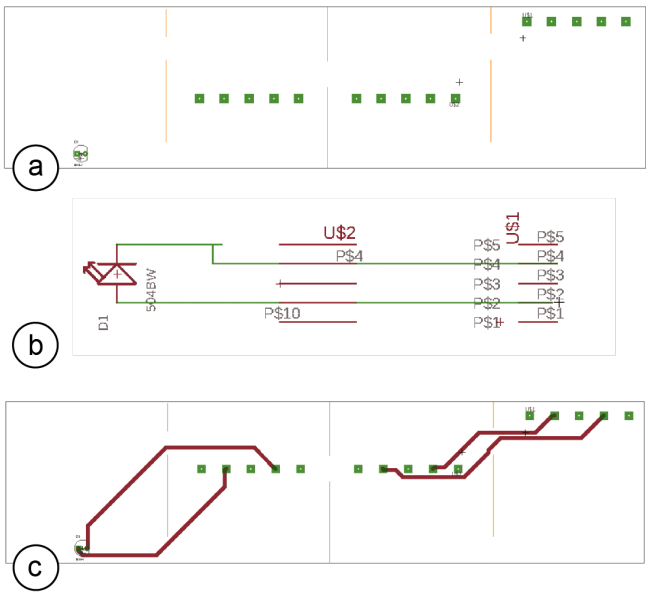


Figure 10: (a) Generated circuit board + colored squares converted to electronic components in EAGLE, (b) the user creates the logical connections, then (c) uses EAGLE's auto-wiring to create the circuit.

#3 Fabrication & Assembly

To fabricate the layers, we only have to cut the 2D drawing of each layer (.DXF file format) in the right order using the cutting plotter. We use the FoldTronics settings sheet to set the cutting blade width, speed, and force for each layer.

Cutting / Perforating the Base Sheet: We first insert the base sheet (PET plastic) into the cutter and cut and perforate it to create the mountain, valley, and slit lines as well as the markers for the electronic components (Figure 11). The FoldTronics process only perforates the sheet from the top and differentiates between mountain and valley lines using separate visual notations (dotted lines for mountains vs. dashed lines for valleys) since they require folding into opposing directions later on (Figure 9a). Alternatively, the FoldTronics process can also perforate the sheet from both sides, i.e. perforate the mountains from the top and valleys from the bottom, however, this requires reinserting the sheet into the cutting plotter.

While all the slits are cut through, the outline of the honeycomb is only perforated to keep it connected to the main sheet, which allows us to further process the sheet with the cutting plotter in the next steps. Finally, the areas where electronic components will be soldered on are also perforated to make it easier to find out which component goes where.

For the objects used in this paper, we use PET plastic sheets, thickness 0.1mm and cut the sheets with a cutting plotter (model: Silhouette Portrait[5], settings cutting: blade 0.2mm, speed 2cm/s, force 10, settings perforating: blade 0.2mm, speed 2cm/s, force 5).

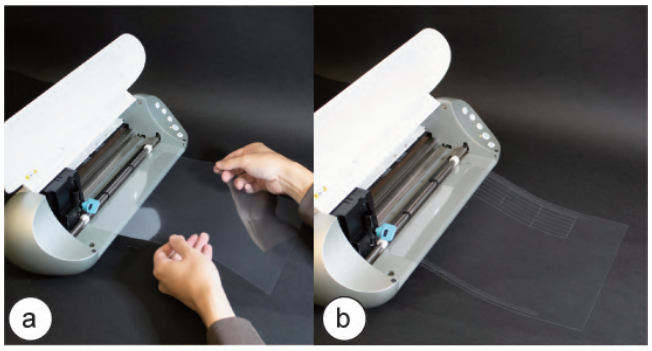


Figure 11: (a) Inserting the sheet into the plotter, (b) the cut & perforated honeycomb still connected to the sheet.

Placing the Wiring with Copper Tape: Next, we place a layer of one-sided copper tape (thickness: 0.07mm) across the entire sheet (Figure 12a). We put the sheet back into the cutting plotter with the copper side up, then execute the file to cut out the shape of the wires (Figure 9b) which is configured to make sure to not cut into the base sheet (cutting settings: blade 0.2mm, speed 2cm/s, force 13). Afterwards,

[5]https://www.silhouetteamerica.com/shop/portrait/silhouette-portrait-2-4t

we peel off the copper tape that is not part of the wiring (Figure 12b).

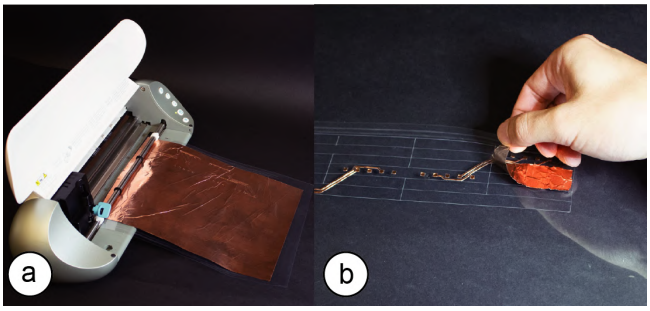


Figure 12: (a) Placing the copper sheet, (b) after cutting, we can peel the rest of the wires off.

Insulating Sheet: In order to prevent any short circuiting from wires touching after folding the base sheet, we next add an insulating layer (Figure 13a). For this, we place a layer of regular non-conductive tape across the entire sheet (thickness: 0.08mm). We put the sheet back into the cutting plotter, which removes the insulating tape only in those areas that have wire ends that either will be connected to electronic components or that use our novel cross cell circuit connector (Figure 13b) according to the drawing shown in Figure 9c. We use the cutting settings: blade 0.1mm, speed 2cm/s, force 4.

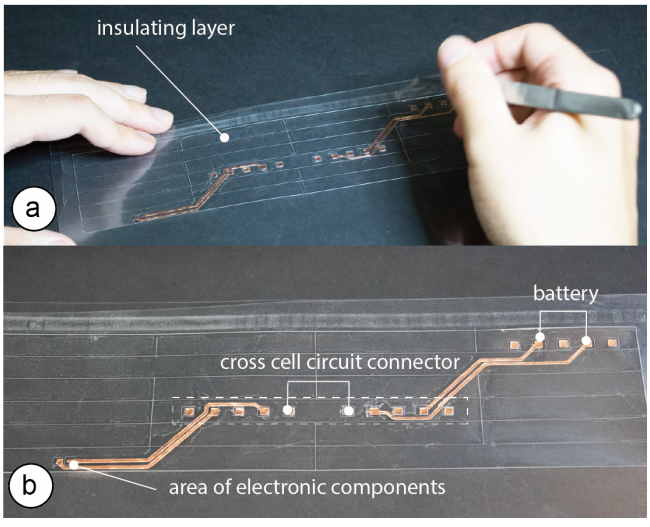


Figure 13: (a) Applying the insulating layer and cutting out areas for (b) electronic components, and the cross-cell circuit connector.

Glue mountains/valleys to hold after folding: In the next step, we apply a layer of regular double-sided tape to the sheet on both its bottom and its top (Figure 9d/e). The double-sided tape is used to connect the valleys and mountains that hold the honeycomb structure together after folding as was illustrated in Figure 2b/c (mountains get glued from the top of the sheets while valleys get glued from the bottom). After inserting the sheet into the cutting plotter, the double-sided tape is cut out in all areas that are not supposed to be taped together (cutting settings: blade 0.2mm, speed 2cm/s, force 6). In addition, for taped valleys/mountains that also carry a cross-cell circuit connector, the cutting plotter cuts out the areas needed for the electronic connections. After cutting both sides, we peel off the remaining double sided tape (Figure 14).

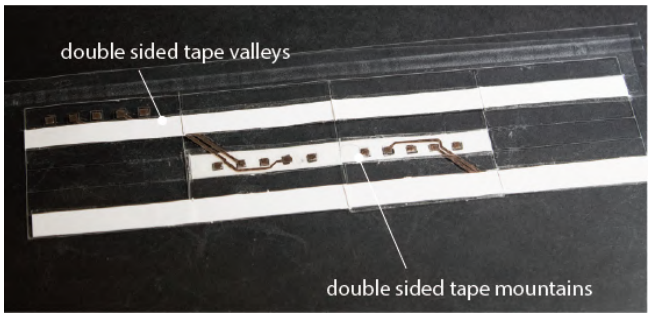


Figure 14: Applying regular double-sided tape on both sides for gluing mountains/valleys.

Soldering: In a final step before soldering, we now cut off the honeycomb pattern to disconnect it from the sheet. Next we solder the electronic components (LED, battery) onto the wires using a soldering iron. If the components are small and hard to solder, we can also use solder paste as an alternative. Since soldering the cross-cell circuit connector is difficult, we use double-sided conductive tape to create the connection.

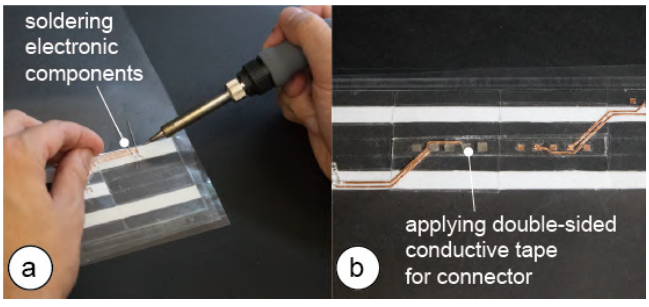


Figure 15: (a) Soldering electronic components. (b) Applying double-sided conductive tape for cross-cell circuit connector.

Folding: We now fold the honeycomb together (Figure 16). The total time for making this example, including all fabrication steps and folding was 18 minutes for us.

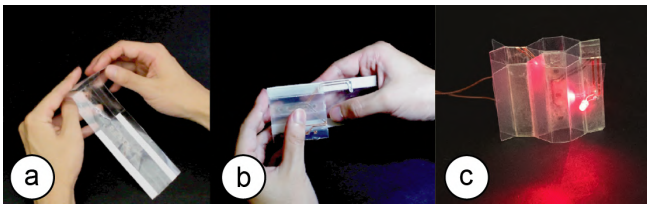


Figure 16: (a, b) Folding the honeycomb. (c) Assembled functional LED circuit.

The fabrication took 2 minutes for cutting the plastic sheet, 3 minutes for cutting the copper foil, 1 minute for cutting the insulating sheet, and 2 minutes total to cut the double-sided tape on each side. After fabricating all layers, it took us 10 minutes to assemble the structure. For less experienced users, the process will take longer. While assembly time is a limitation of our process since it scales linearly with the number of cells, our work enables functionality which cannot be achieved in other ways.

5 SCENARIOS

Next, we illustrate several scenarios that benefit from having integrated electronics in foldable honeycomb structures.

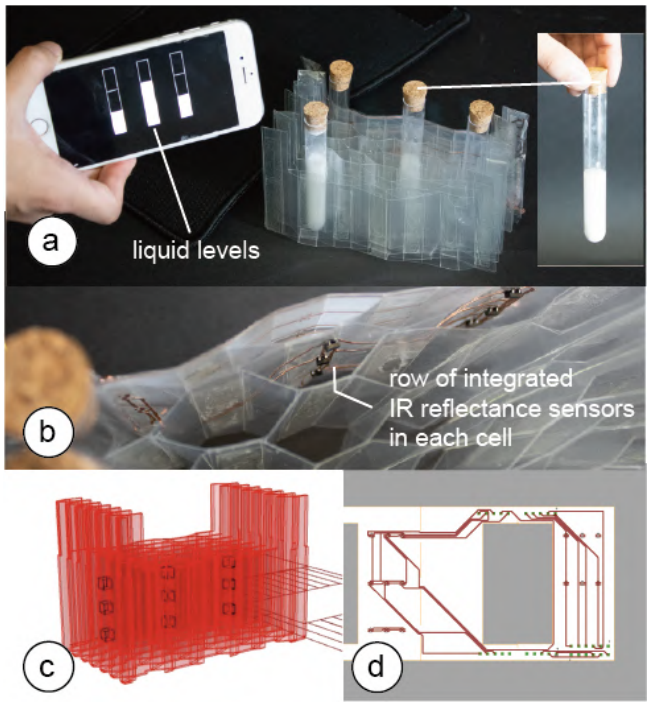


Figure 17: (a, b) Tube case with sensors. (c) 3D model data, (d) circuit layout.

#1 Tube Case with Liquid Levels (Integrated Sensing)

Figure 17 shows a portable test tube case that can detect if tubes are inserted or removed, the type of liquid they contain, and the current liquid level. After measuring, the portable test-tube case displays the tube's content quantity and color in real time on the user's phone.

To allow for sensing inside the 3D structure, we placed three IR reflectance sensors per cell onto the 2D base sheet prior to folding the structure into its three-dimensional shape. Once folded, the sensors can detect a test tube inserted into one of the honeycomb cells.

Since we use a honeycomb structure, the test-tube case is portable, i.e. when users walk to the test site, they can easily store it in their pocket and expand it on site (thickness when collapsed 5mm vs. 45mm when expanded (9x)). In addition, since the honeycomb structure is hollow, the test-tube case is very lightweight and thus easy to carry around.

#2 2D + 3D Smartwatch Display (Integrated Displays)

Figure 18 shows a prototype of a smart watch: (a) By default, the smart watch provides a regular 2D display (for simplicity of prototyping we use a display made from 2x3 LEDs). (b) However, when the user pulls the watch face upwards, the 2D display transforms into a volumetric 3D display for viewing additional content.

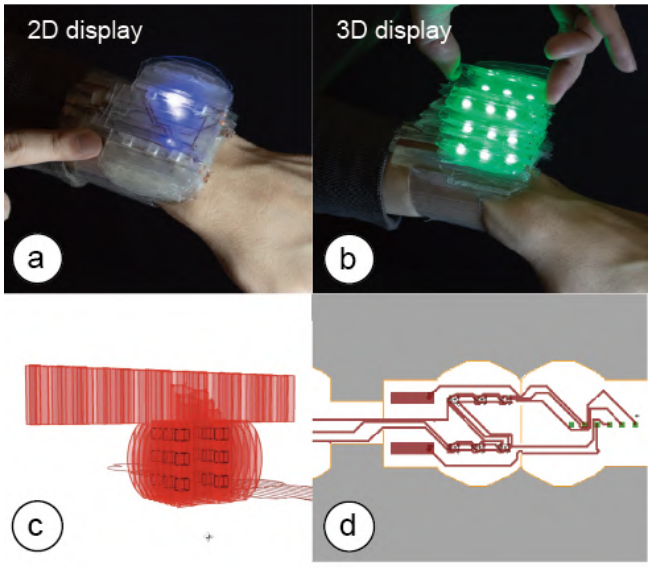


Figure 18: (a, b) Expandable smart watch. (c) 3D model data, (d) circuit layout.

While traditional fabrication approaches make it difficult to arrange LEDs inside a 3D structure, our method of folding a 2D sheet reduces the 3D placement problem to a 2D layout problem, which we can solve with our design software.

To detect when the display state changes from 2D to 3D and the additional LEDs need to be turned on, we also integrated a sensor that can detect when the display is in its flat 2D state. The key idea is to place conductive wires in a way that they touch when the display is compressed, and do not touch as the display gets expanded. We support this "touch sensor" in our design software as a separate electronic component.

In summary, this example made use of several benefits of the honeycomb structure: The 2D-to-3D folding facilitated the placement of the LEDs in the 3D volume, the capabilities of cells to expand and contract enabled the interaction of transforming the display, and since the honeycomb structure is sparse the resulting object was also very lightweight, which is especially important for wearable applications. More ways of extending screen space of such a folded smartwatch are shown by Fuchs et al. [10].

#3 Dynamic Game-Controller (Integrated Sensing)

Building onto the functionality explained above, we also build a game-controller that users can transform into different shapes to represent different game states, e.g., from a regular controller to a gun-shaped controller as shown in Figure 19. The controller includes two of the touch-buttons explained in the previous example: one touch button operates the controller and allows the user to shoot objects, whereas the other touch button detects the state of the game controller (when user compresses the controller to state 1, the touch button activates; when the user expands the controller, the touch button deactivates and we transform the controller to state 2).

In summary, foldable honeycomb structures with integrated electronics that enable interactive 3D objects are useful in a variety of scenarios ranging from mobile applications that require effective storage for transportation (scenario 1: mobile tube case) to deformable and shape-changing objects (scenario 2: smart watch as well as scenario 3: dynamic game controller).

6 IMPLEMENTATION / CODE

Figure 20 shows a flowchart of our implementation pipeline. As mentioned previously, the main design tool is implemented in the 3D editor Rhino3D as a Grasshopper extension. Grasshopper directly exports the layers for the honeycomb sheet, insulating tape, and mountain/valley assembly. In addition, to generate the wiring, we implemented a ULP plugin to the electronic design software EAGLE, which exports the wiring layer – making the stack of layers complete.

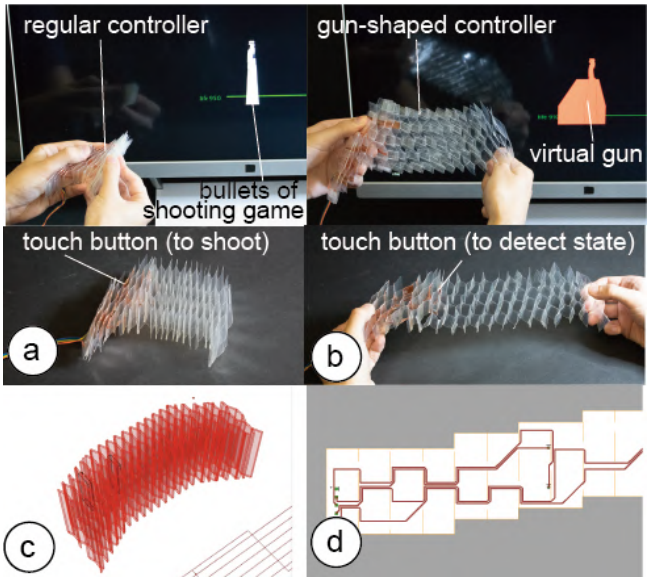


Figure 19: Shape changing controller. (c) 3D model data, (d) circuit layout.

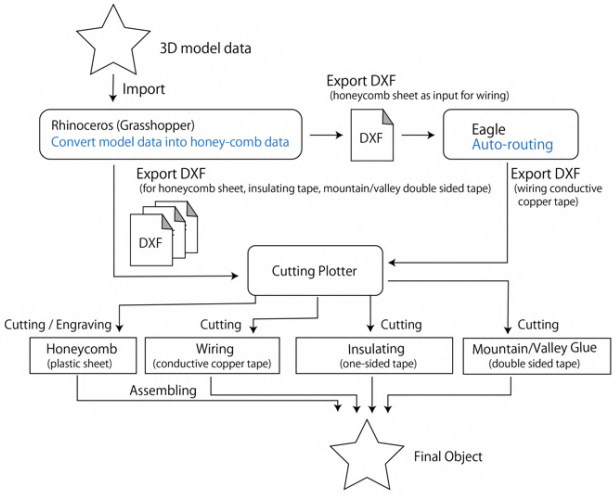


Figure 20: Flowchart of the system.

Layer #1: Converting the 3D Model into a 2D Structure

In the current version of our software, we can only convert a specific type of 3D model, i.e., shapes whose bottom and top are flat surfaces. There are no constraints on the other sides of the object, i.e. they can be slanted or curved or have an arbitrary shape (Figure 21).

Generating the Honeycomb Columns: We start the conversion process by splitting the model into 2D planes that will form the walls of the honeycomb cell (Figure 22a/b).

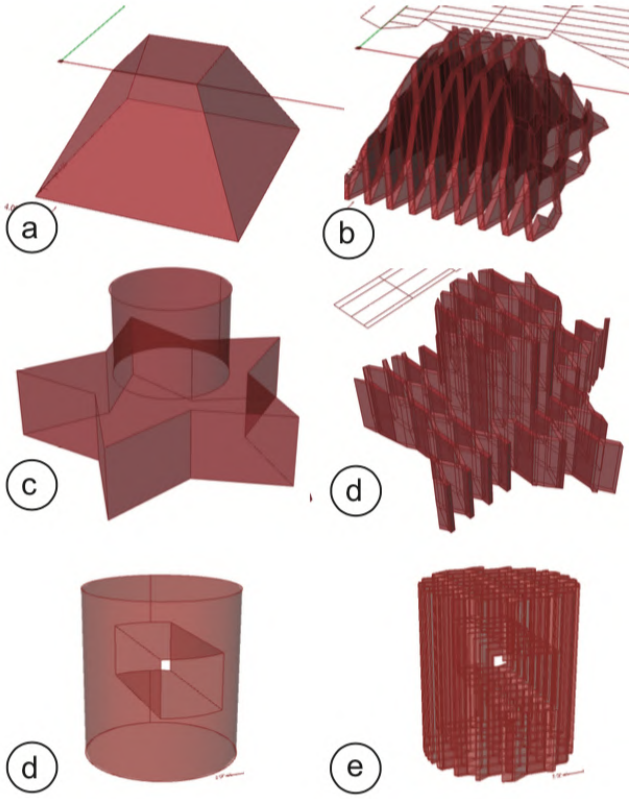


Figure 21: Examples of models our algorithm can convert.

To determine the position of each 2D plane, our algorithm first computes the bounding box of the model using Grasshopper's *DeBox()* function and then divides the width of the bounding box by the number of slices (which the user set using the "resolution" slider). To generate each 2D plane, we now split the model with a large cutting plane at each plane's position using Grasshoppers *Brep()* function (Figure 22b).

Generating the Honeycomb Cell Segments: As can be seen in Figure 22c the next step is to split each 2D plane into the honeycomb cells segments, which will form the mountains and valleys. To find the position of each cell segment, we compute the bounding box of each 2D plane, compute its width, and then divide it by the cell resolution, which the user had previously set using the resolution slider. In the final step illustrated in Figure 22d, we duplicate each 2D plane and its cell segments to have one available for folding each of the connecting cells (see also Figure 23 for how this is represented in the final 2D layout).

Generating the 2D Layout: To convert the 2D planes with cell segments into a 2D drawing, we proceed as follows: First, we project each 2D plane onto the sheet starting with plane

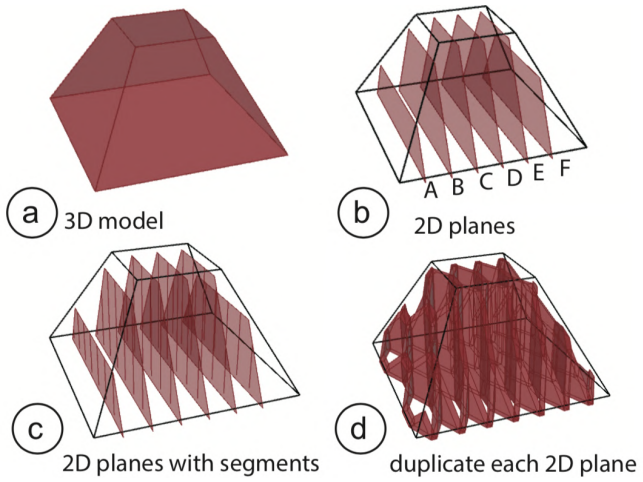


Figure 22: (a) 3D model, (b) splitting into honeycomb columns, and (c) cells. (d) Duplicating each column to form both sides of a cell.

A, plane B, plane C and so on. When placing a 2D plane, we also have to create a duplicate vertically mirrored since each 2D plane is used for both the left and right connecting cell as mentioned previously (compare Figure 23 and Figure 22d).

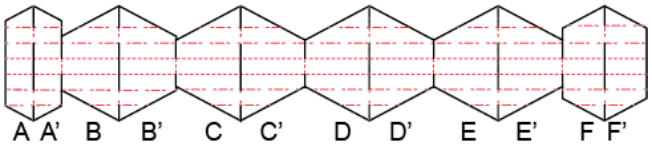


Figure 23: Converting the model from Figure 22 into a 2D layout.

After placing all planes and generating the line layout, we now determine which lines will be cut and which perforated.

For this, our algorithm iterates over the cells starting with the top left cell and processes all the cells in a column before moving to the next one. While going over the cells, our algorithm executes the layout that we explained at the beginning of this paper and that is shown in Figure 2c, i.e. our algorithm tags each line as either a mountain, a valley, a slit, or as part of the honeycomb outline. In the 2D drawing, all slits are colored black for cutting, all mountains and valley lines are colored red for engraving.

Placing Electronic Components: In addition, we also compute the position of electronic components on the 2D layout. For this, we identify which valley/mountain the electronic component is attached to and then identify that valley/mountain on the 2D sheet. We draw the outline of electronic component as a rectangle onto the 2D sheet in the

respective position. Each electronic has a different color assigned to it (except red and black), which allows us to identify each component in EAGLE.

Layer #2: Wiring of Components with EAGLE

After the user imported the base-layer into EAGLE, our EAGLE User Language Program (ULP) plugin first generates a circuit board the size of the honeycomb pattern by parsing the .DXF file and going over the outside contour of the honeycomb using the coordinates in the file. Next, it converts the colored squares according to their color into matching electronic components. First, we extract the position of electronic components from the .DXF file, then load the matching electronic component from EAGLE's component library using EAGLE's *add()* function and provide the respective component ID as parameter. Using the generated board and electronic component, the user can now create the desired schematic and then use EAGLE's standard auto-routing function to generate the wiring layout.

Layer #3: Insulating Layer

The insulating layer is exported from Grasshopper when the 2D honeycomb layout is generated. The insulating layer is simple in that it covers all cells except those that have an electronic component or a cross-cell circuit connector placed on them, which we can easily identify when we place the colored squares for the components on the sheet.

Layer #4: Valley/Mountain Glue Layer

Similarly, the valley/mountain glue layers that use double-sided tape to hold the honeycomb together are generated when Grasshopper generates the 2D honeycomb layout. Since this follows a pre-defined cell-pattern (see Figure 2c), we generate this layer when we are defining which cells are mountains and which valleys on the base layer.

7 EXTENSIONS FOR HONEYCOMB CIRCUITS

Next, we describe extensions to our approach that allow for additional functionality.

Double-Sided Circuit

We can also create double-sided circuit boards by applying our fabrication process on both sides (wiring + insulating sheet). However, at this point our design software only supports one-sided boards and thus users would have to create the layout manually. If the user prints the circuit on the back side of the sheet, it is necessary to connect the circuit of the top side and the back side using eyelets (Figure 24).

Using Different Honeycomb Patterns

Beside the honeycomb pattern we used for this paper, there are other structures with different properties available. For instance, by using a different size for mountains than for valleys, we can create structures of negative Poisson's ratio (NPR), which can be stretched in two directions [25]. With recent inventions in meta-materials, such structures could in the future even potentially be expanded to three-dimensional expansion and contraction [29]. Finally, by generating heterogeneous patterns, in a way similar to the ones shown in *Hybrid Bricolage* [8], we can generate honeycombs that better approximate an object's geometry. Properties of irregular honeycomb structures have also been explored in computer graphics [15], mechanical engineering [20], and material science [43].

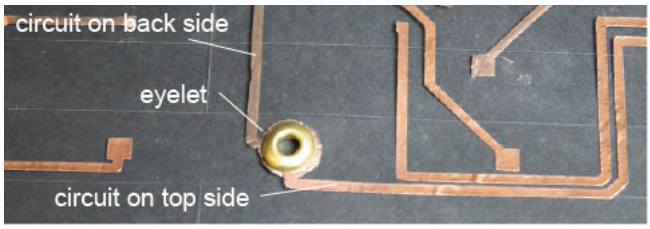


Figure 24: Double-sided circuit and connection.

Stacking Multiple Honeycomb Structures

At the moment, we use a single layer of honeycomb structure only, which limits us to very simple shapes. However, moving forward we can stack multiple honeycomb layers on top of each other for more complex geometries.

Other Base-Sheet Materials: Opaque, Translucent etc.

For the objects illustrated in this paper, we used a transparent PET plastic sheet. However, other materials than PET can be used as the base as long as they comply with the maximum thickness of 0.8mm (thicker sheets are hard to cut and hard to fold). For example, by using a translucent plastic sheet we can diffuse the light of LEDs for a lamp, or by using a black plastic sheet we can conceal the internal parts. Similarly, paper can also be used for low cost prototyping.

8 DISCUSSION

Finally, we will discuss our experiences while building honeycomb circuits over the course of several months.

Conductive Inkjet Printing vs. Conductive Tape

We initially considered the use of conductive inkjet printing to print the circuits directly onto the sheet rather than applying a layer of conductive tape and then peeling of the non-wire areas. However, conductive inkjet printing turned out to be not as reliable since the wire traces easily broke during folding. Conductive tape, in contrast, was robust enough to withstand folding over longer periods of time.

Robustness of Prototypes

We found that with respect to the robustness of the constructed circuitry, the weakest point are the wire traces. When the lines are thin (<1mm) and folded several hundred times, such as when expanding and contracting the transformable smart watch display, traces start to break. We can reduce this effect by using thicker wire traces. However, this results in fewer wires per cell. For the cross-cell circuit connector, solder paste helped eliminate these issues since it "glues" both sides together more reliably. However, the paste needs to be applied manually while the tape can be cut using the plotter. We also noticed that the blade of the cutting plotter gets dull over time. However, for the entire project, we only replaced the blade once ($9 per blade).

Debugging Folded Circuitry

During assembly, we found that debugging the circuitry was challenging once the circuit was folded into its 3D shape since the honeycomb-cells were glued/taped together and thus could not be easily unfolded again. To facilitate this, we envision a version of our fabrication process that uses hook-and-loop fasteners (both regular and conductive) instead of tape to connect the mountains/valleys. When debugging is required, the hook-and-loop fasteners allow users to pull the two surfaces apart to reveal the circuitry, diagnose the problem, and then re-connect the surfaces.

Learning Curve when Using Honeycomb Circuits

Most of the fabrication steps (e.g., applying tape and aligning sheets, automated cutting using the plotter) require little expertise and have limited complexity. Soldering is the hardest task but our target users (makers and other technical inventors with expertise in electronics and prototyping) tend to be familiar with this. The steps in the electronic design software EAGLE are not more difficult than for regular 2D electronics. The most error-prone step from our own experience is soldering the right components into the right place, which could be further improved by cutting visually distinguishable marks into the sheet at the correct locations. The cutting and layering could be further automated using layered-object manufacturing (automatically aligns, cuts, laminates layers) or a machine similar to the one used during the honeycomb manufacturing process shown in Wadley et al. [37].

9 CONCLUSION

In this paper, we presented a novel fabrication technique that allows users to integrate electronic parts into foldable 3D structures. Since our technique is based on using a plotting cutter and a plastic sheet, users can create foldable prototypes fast and at low cost. To facilitate the design, we developed a layer-based fabrication method and developed software that automatically generates the required layers for folding, circuitry, and insulating. We demonstrated how our technique can create a variety of 3D objects with integrated functions, such as a smart-watch, a test tube case and a shape-changing controller. For future work, we plan to investigate how to reduce the number of layers required by using conductive inkjet printing for printing traces that are robust to folding. In addition, we plan to automate the folding process using a feeder and a folding machine [38]. We also plan to develop a circuit layout algorithm that avoids potential short circuiting when folding which would allow us to reduce the need for having an insulating sheet.

10 ACKNOWLEDGMENT

This work was supported by JST ERATO Grant Number JPMJER1501, Japan.

REFERENCES

[1] Rafael Ballagas, Sarthak Ghosh, and James Landay. 2018. The Design Space of 3D Printable Interactivity. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 2, Article 61 (July 2018), 21 pages. https://doi.org/10.1145/3214264

[2] Patrick Baudisch and Stefanie Mueller. 2017. Personal Fabrication. *Foundations and Trends® in Human-Computer Interaction* 10, 3-4 (2017), 165–293. https://doi.org/10.1561/1100000055

[3] Dustin Beyer, Serafima Gurevich, Stefanie Mueller, Hsiang-Ting Chen, and Patrick Baudisch. 2015. Platener: Low-Fidelity Fabrication of 3D Objects by Substituting 3D Print with Laser-Cut Plates. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1799–1806. https://doi.org/10.1145/2702123.2702225

[4] Tom Bitzer. 2012. *Honeycomb technology: materials, design, manufacturing, applications and testing.* Springer.

[5] Varun Perumal C and Daniel Wigdor. 2015. Printem: Instant Printed Circuit Boards with Standard Office Printers & Inks. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 243–251. https://doi.org/10.1145/2807442.2807511

[6] Varun Perumal C and Daniel Wigdor. 2016. Foldem: Heterogeneous Object Fabrication via Selective Ablation of Multi-Material Sheets. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5765–5775. https://doi.org/10.1145/2858036.2858135

[7] Claudia Daudén Roquet, Jeeeun Kim, and Tom Yeh. 2016. 3D Folded PrintGami: Transforming Passive 3D Printed Objects to Interactive by Inserted Paper Origami Circuits. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16)*. ACM, New York, NY, USA, 187–191. https://doi.org/10.1145/2901790.2901891

[8] Tamara Anna Efrat, Moran Mizrahi, and Amit Zoran. 2016. The Hybrid Bricolage: Bridging Parametric Design with Craft Through Algorithmic Modularity. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5984–5995. https://doi.org/10.1145/2858036.2858441

[9] Samuel Felton, Mike Tolley, Erik Demaine, Daniela Rus, and Robert Wood. 2014. A method for building self-folding machines. *Science* 345, 6197 (2014), 644–646. https://doi.org/10.1126/science.1252610 arXiv:http://science.sciencemag.org/content/345/6197/644.full.pdf

[10] Alexandra Fuchs, Miriam Sturdee, and Johannes Schöning. 2018. Foldwatch: Using Origami-inspired Paper Prototypes to Explore the Extension of Output Space in Smartwatches. In *Proceedings of the 10th Nordic Conference on Human-Computer Interaction (NordiCHI '18)*. ACM, New York, NY, USA, 47–59. https://doi.org/10.1145/3240167.3240173

[11] Nan-Wei Gong, Amit Zoran, and Joseph A. Paradiso. 2013. Inkjet-printed Conductive Patterns for Physical Manipulation of Audio Signals. In *Proceedings of the Adjunct Publication of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13 Adjunct)*. ACM, New York, NY, USA, 13–14. https://doi.org/10.1145/2508468.2514932

[12] Daniel Groeger and Jürgen Steimle. 2018. ObjectSkin: Augmenting Everyday Objects with Hydroprinted Touch Sensors and Displays. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 134 (Jan. 2018), 23 pages. https://doi.org/10.1145/3161165

[13] Felix Heibeck, Basheer Tome, Clark Della Silva, and Hiroshi Ishii. 2015. uniMorph: Fabricating Thin Film Composites for Shape-Changing Interfaces. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 233–242. https://doi.org/10.1145/2807442.2807472

[14] Jonathan Hook, Thomas Nappey, Steve Hodges, Peter Wright, and Patrick Olivier. 2014. Making 3D Printed Objects Interactive Using Wireless Accelerometers. In *Proceedings of the Extended Abstracts of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 1435–1440. https://doi.org/10.1145/2559206.2581137

[15] Caigui Jiang, Jun Wang, Johannes Wallner, and Helmut Pottmann. 2014. Freeform Honeycomb Structures. *Computer Graphics Forum* (2014). https://doi.org/10.1111/cgf.12444

[16] Yoshihiro Kawahara, Steve Hodges, Benjamin S. Cook, Cheng Zhang, and Gregory D. Abowd. 2013. Instant Inkjet Circuits: Lab-based Inkjet Printing to Support Rapid Prototyping of UbiComp Devices. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '13)*. ACM, New York, NY, USA, 363–372. https://doi.org/10.1145/2493432.2493486

[17] Martin Kilian, Simon Flöry, Zhonggui Chen, Niloy J. Mitra, Alla Sheffer, and Helmut Pottmann. 2008. Curved Folding. *ACM Trans. Graph.* 27, 3, Article 75 (Aug. 2008), 9 pages. https://doi.org/10.1145/1360612.1360674

[18] Kantha Konka, Jayathirtha Rao, and Kumar Swamy Avss Gupta. 2015. Heat Insulation Analysis of an Aluminum Honeycomb Sandwich Structure. *Journal of Thermal Engineering* 1 (03 2015), 210. https://doi.org/10.18186/jte.25885

[19] Jennifer A Lewis and Bok Y Ahn. 2015. Three-dimensional printed electronics. *Nature* 518 (feb 2015), 42. https://doi.org/10.1038/518042ahttp://10.0.4.14/518042a

[20] Ke Li, Xin-Lin Gao, and Jun Wang. 2007. Dynamic crushing behavior of honeycomb structures with irregular cell shapes and non-uniform cell wall thickness. *International Journal of Solids and Structures* 44, 14 (2007), 5003 – 5026. https://doi.org/10.1016/j.ijsolstr.2006.12.017

[21] Joanne Lo and Eric Paulos. 2014. ShrinkyCircuits: Sketching, Shrinking, and Formgiving for Electronic Circuits. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 291–299. https://doi.org/10.1145/2642918.2647421

[22] James McCrae, Nobuyuki Umetani, and Karan Singh. 2014. FlatFitFab: Interactive Modeling with Planar Sections. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 13–22. https://doi.org/10.1145/2642918.2647388

[23] Jun Mitani and Hiromasa Suzuki. 2004. Making Papercraft Toys from Meshes Using Strip-based Approximate Unfolding. *ACM Trans. Graph.* 23, 3 (Aug. 2004), 259–263. https://doi.org/10.1145/1015706.1015711

[24] Stefanie Mueller, Bastian Kruck, and Patrick Baudisch. 2013. LaserOrigami: Laser-cutting 3D Objects. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 2851–2852. https://doi.org/10.1145/2468356.2479544

[25] Taketoshi Nojima and Kazuya Saito. 2006. Development of Newly Designed Ultra-Light Core Structures. *JSME International Journal Series A Solid Mechanics and Material Engineering* 49, 1 (2006), 38–42. https://doi.org/10.1299/jsmea.49.38

[26] Hyunjoo Oh, Tung D. Ta, Ryo Suzuki, Mark D. Gross, Yoshihiro Kawahara, and Lining Yao. 2018. PEP (3D Printed Electronic Papercrafts): An Integrated Approach for 3D Sculpting Paper-Based Electronic Devices. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 441, 12 pages. https://doi.org/10.1145/3173574.3174015

[27] Simon Olberding, Sergio Soto Ortega, Klaus Hildebrandt, and Jürgen Steimle. 2015. Foldio: Digital Fabrication of Interactive and Shape-Changing Objects With Foldable Printed Electronics. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 223–232. https://doi.org/10.1145/2807442.2807494

[28] Simon Olberding, Michael Wessely, and Jürgen Steimle. 2014. PrintScreen: Fabricating Highly Customizable Thin-film Touch-displays. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 281–290. https://doi.org/10.1145/2642918.2647413

[29] Johannes T. B. Overvelde, James C. Weaver, Chuck Hoberman, and Katia Bertoldi. 2017. Rational design of reconfigurable prismatic architected materials. *Nature* 541 (2017), 347–352.

[30] Huaishu Peng, Jennifer Mankoff, Scott E. Hudson, and James McCann. 2015. A Layered Fabric 3D Printer for Soft Interactive Objects. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1789–1798. https://doi.org/10.1145/2702123.2702327

[31] Edoardo Alessio Piana, Candida Petrogalli, Diego Paderno, and Ulf Carlsson. 2018. Application of the Wave Propagation Approach to Sandwich Structures: Vibro-Acoustic Properties of Aluminum Honeycomb Materials. *Applied Sciences* 8, 1 (2018). https://doi.org/10.3390/app8010045

[32] Kazuya Saito, Sergio Pellegrino, and Taketoshi Nojima. 2014. Manufacture of Arbitrary Cross-Section Composite Honeycomb Cores Based on Origami Techniques. *Journal of Mechanical Design* 136, 5 (March 2014), 051011–051011–9. https://doi.org/10.1115/1.4026824

[33] Greg Saul, Manfred Lau, Jun Mitani, and Takeo Igarashi. 2011. SketchChair: An All-in-one Chair Design System for End Users. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, New York, NY, USA, 73–80. https://doi.org/10.1145/1935701.1935717

[34] 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. https://doi.org/10.1145/2642918.2647374

[35] Valkyrie Savage, Xiaohan Zhang, and Björn Hartmann. 2012. Midas: Fabricating Custom Capacitive Touch Sensors to Prototype Interactive Objects. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 579–588. https://doi.org/10.1145/2380116.2380189

[36] Udayan Umapathi, Hsiang-Ting Chen, Stefanie Mueller, Ludwig Wall, Anna Seufert, and Patrick Baudisch. 2015. LaserStacker: Fabricating 3D Objects by Laser Cutting and Welding. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software &*

Technology (UIST '15). ACM, New York, NY, USA, 575–582. https://doi.org/10.1145/2807442.2807512

[37] Haydn N.G Wadley. 2006. Multifunctional periodic cellular metals. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 364, 1838 (2006), 31–68. https://doi.org/10.1098/rsta.2005.1697

[38] Lijun Wang, Kazuya Saito, You Gotou, and Yoji Okabe. 2017. Design and fabrication of aluminum honeycomb structures based on origami technology. *Journal of Sandwich Structures & Materials* (06 2017), 109963621771464. https://doi.org/10.1177/1099636217714646

[39] Enboa Wu and Wu-Shung Jiang. 1997. Axial crush of metallic honeycombs. *International Journal of Impact Engineering* 19, 5 (1997), 439 – 456. https://doi.org/10.1016/S0734-743X(97)00004-3

[40] Junichi Yamaoka, Yasuaki Kakehi, and Yoshihiro Kawahara. 2017. ActMold: Rapid Prototyping of Electronic Circuits on 2.5D Objects with Interactive Vacuum Forming. In *ACM SIGGRAPH 2017 Studio (SIGGRAPH '17)*. ACM, New York, NY, USA, Article 2, 2 pages. https://doi.org/10.1145/3084863.3084870

[41] Junichi Yamaoka, Ryuma Niiyama, and Yasuaki Kakehi. 2017. BlowFab: Rapid Prototyping for Rigid and Reusable Objects Using Inflation of Laser-cut Surfaces. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, New York, NY, USA, 461–469. https://doi.org/10.1145/3126594.3126624

[42] Shun Yao, Xueli Liu, Stavros V. Georgakopoulos, and Manos M. Tentzeris. 2014. A novel reconfigurable origami spring antenna. *2014 IEEE Antennas and Propagation Society International Symposium (APSURSI)* (2014), 374–375.

[43] Qiancheng Zhang, Xiaohu Yang, Peng Li, Guoyou Huang, Shangsheng Feng, Cheng Shen, Bin Han, Xiaohui Zhang, Feng Jin, Feng Xu, and Tian Jian Lu. 2015. Bioinspired engineering of honeycomb structure – Using nature to inspire human innovation. *Progress in Materials Science* 74 (2015), 332 – 400. https://doi.org/10.1016/j.pmatsci.2015.05.001