# TiltStacks: Composing Shape-Changing Interfaces Using Tilting and Stacking of Modules

**John Tiab[a], Sebastian Boring[a], Paul Strohmeier[a], Anders Markussen[a], Jason Alexander[b], Kasper Hornbæk[a]**

[a]University of Copenhagen, Denmark
{john.tiab, sebastian.boring, p.strohmeier, amark, kash}@di.ku.dk

[b]Lancaster University, United Kingdom
j.alexander@lancaster.ac.uk

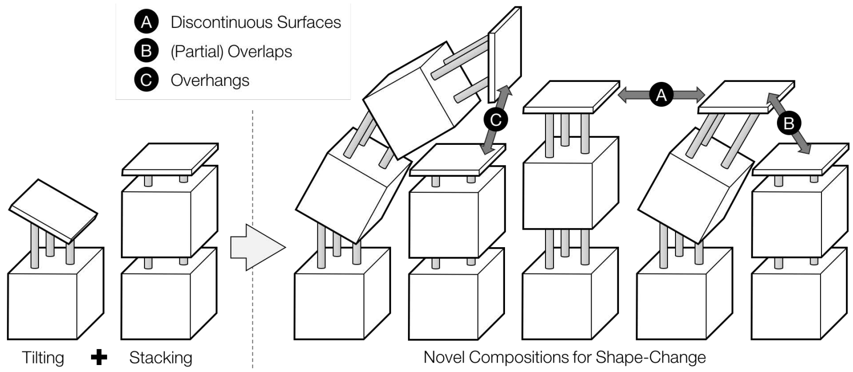


**Figure 1: TiltStacks allows making shape-changing displays from tilt-enabled modules that are stacked vertically (left). Thereby, a range of new display surfaces can be composed with discontinuous surfaces (a), partial overlaps (b), and overhangs (c).**

## ABSTRACT

Many shape-changing interfaces use an array of actuated rods to create a display surface; each rod working as a pixel. However, this approach only supports pixel height manipulation and cannot produce more radical shape changes of each pixel (and thus of the display). Examples of such changes include non-horizontal pixels, pixels that overhang other pixels, or variable gaps between pixels. We present a concept for composing shape-changing interfaces by vertically stacking tilt-enabled modules. Together, stacking and tilting allow us to create a more diverse range of display surfaces than using arrays. We demonstrate this concept through TiltStacks, a shape-changing prototype built using stacked linear actuators and displays. Each tiltable module provides three degrees of freedom (z-movement, roll, and pitch); two more degrees of freedom are added through stacking modules (i.e., planar x- and y-movement).

## CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**; *Interaction devices*; Displays and imagers



AVI '18, May 29-June 1, 2018, Castiglione della Pescaia, Italy

ACM ISBN 978-1-4503-5616-9/18/05…$15.00
https://doi.org/10.1145/3206505.3206530

## KEYWORDS

Shape-changing interfaces, tilting, stacking, compositional concept

## 1 INTRODUCTION

Self-actuated shape-changing interfaces combine physical shape-change and displayed content, affording novel interaction techniques through touch, gestures, and deformations [7,16]. Many of these interfaces are rod-based. They consist of a uniform array of rods that actuate vertically. This layout mimics the pixels of a display, allowing the interface to render physical shapes by coordinating the vertical movement of the rods. Previous work has used this layout to demonstrate dynamic capabilities for output and interaction (e.g., [2,4,5,8,9,11,15,21,22]).

Using a pixel-array layout limits the physical capabilities of shape-changing interfaces. For instance, it is not possible to have non-horizontal pixels because only height changes of the pixels is allowed. More complex shapes, which Rasmussen et al. [16] called 'non-topologically equivalent' shapes, are also not possible. For instance, pixels cannot overhang other pixels as in other forms of displays [3,19], pixels cannot partially overlap, and pixels cannot move in the plane. While these limitations are addressed in previous work [7,11,18], the array layout appears to be taken for granted. Most previous work focuses on technical solutions for achieving the array layout and does not explore new design principles (Alexander et al. provide one of few explorations [1]).

We present a concept for composing shape-changing interfaces by vertically stacking tilt-enabled modules. Through tilting and stacking, displays can move with five degrees of freedom (planar x- and y- translation, roll, pitch, and vertical movement). This enables displays with overlaps, overhangs, and discontinuous surfaces. We demonstrate this concept with TiltStacks, a shape-changing prototype built of modules of linear actuators and displays. Those modules can be stacked to generate new forms of display surfaces.

We contribute (a) a compositional concept for shape-changing interfaces based on stacking and tilting, (b) a proof-of-concept prototype, TiltStacks, implementing this concept, and (c) demonstrations of new forms of displays using TiltStacks.

## 2 RELATED WORK

Diverse actuation techniques have been used to develop shape-changing interfaces, for instance mechanical actuation [2,6,12], pneumatic actuation [23], actuation using smart materials [14,17], and biological spores [24]. Many shape-changing interfaces use rod-based linear actuation [2,4,21,22]. Rod-based shape-changing interfaces render physical shapes with projected visuals or displays, often enabling touch, gestures, and deforming interactions. Previous work used rod-based interfaces for providing dynamic affordances and haptic feedback [2,5,6], remote collaboration [8], rendering material properties [13], interactive physical visualizations [21], and furniture [22].

Rod-based shape-changing interfaces have certain limitations. They are built using a planar array layout of linear actuators acting as physical pixels. The actuators have a single degree of freedom (z-movement). Rendered physical shapes are approximated using the rods, which limits their continuity and smoothness to the density of the actuators [11]. Shapes with overhangs are not possible because the shapes are generated in a vertical upward direction [18]. Moreover, more complex non-topologically equivalent shapes cannot be rendered because pixels cannot translate horizontally. Those shapes are discontinuous with holes, splits, and gaps between the pixels [16].

Little work has explored overcoming these limitations and increasing the degrees of freedom of shape-changing display interfaces. Tilt Displays [1] introduced multi-axis tilting and vertical actuation of displays using a 3×3 grid of displays where each display elevates and tilts along the x- and y-axes. However, Tilt Displays only have three degrees of freedom. Kinetic blocks [18] used a rod-based interface to actuate and assemble objects using stacking and catapulting techniques. However, the authors used static passive blocks with no displays. ShapeClip [4] is a modular tool for rapid prototyping of vertically-actuated interfaces. It was used by designers [4] and the public [20] to ideate on new configurations and layouts of self-actuated shape-changing interfaces. However, few of these configurations were implemented.

## 3 TILTING AND STACKING

The main contribution of this paper is the concept of combining tilting and stacking. Instead of a linear actuator, we suggest vertically stacking tilt-enabled modules. Tilting adds two degrees of freedom (roll and pitch) instead of a single one (z-movement) resulting in modules with three degrees of freedom (z-movement, roll, and pitch). This is similar to Tilt Displays [1].

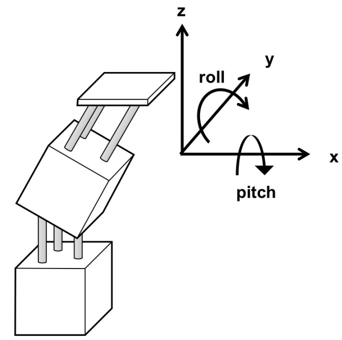


**Figure 2: Stacking two tilt-enabled modules allows the topmost surface to actuate with five degrees of freedom.**

Tilting helps in rendering smoother surfaces using physical antialiasing when the tilting displays join together. Moreover, it allows displays to face different sides enabling new interaction techniques and slope rendering.

Two additional degrees of freedom are added through stacking modules (i.e., planar x- and y-movement). Already stacking two modules (see Figure 2) adds the possibility of more complex shapes. Though stacking on its own only adds actuation height, its combination with tilting allows for many new shapes; Figure 1 shows examples of possible shapes.

Next, we discuss the parameters that affect the design of tilting and stacking modules. First, we discuss the design parameters of a single module. Afterwards, we discuss the parameters that come into play when combining and stacking several modules.

### 3.1 One Module

A single module enables three degrees of freedom (z-movement, roll, and pitch). The main parameters that affect its design are size and tilt angle, as they define the scale of the display surface and rendered shapes.

The size of the modules depends on the desired applications and the underlying hardware. Small modules could be used for mobile and tabletop sized display surfaces, where large modules could be for interactive furniture and environments. Small modules require finer actuators and designs to pack the actuators in a small confined space. Large modules do not have this problem, however, they require more energy and weigh more.

The tilt angle of the module is the amount of degrees (as roll and pitch) that the top plate of the module can move. An ideal case would be a tilt angle close to 90 degrees. The range of tilt affects the amount of overlap and gaps between modules (hence the rendered physical shapes). A small tilting range limits the possible shapes that could be rendered; stacking multiple modules increases the effective tilt angle.

### 3.2 Stacked Modules

Stacking tilt-enabled modules, adds actuation points in different planes, similar to multi-joint moving objects. Therefore, actuation is not limited to a single flat surface. This enables rendering of shapes that require multiple pivot points, for instance shapes with curvatures and overhangs.

With tilting and stacking combined we produce movable displays with five degrees of freedom. The top part of the module stacked on top has roll, pitch, z-, x-, and y- movements as shown in Figure 2. This enables non-topologically equivalent shapes using horizontal planar movements. Sub-regions of the interface can split apart or partially overlap, supporting additive and subtractive interactions and visualizations. Further, using splitting animations we can generate cavities and holes resulting in discontinuous display surfaces, which imitates permeability as discussed by Rasmussen et al. [16].

The number of stacked modules on top of each other affects the complexity of rendered shapes. The more levels, the more curvature the rendered shape could have, enabling more complex shapes with discontinuous surfaces and overhangs. However, just having two stacked modules achieves five degrees of freedom enabling 3D movement of the top module. Such movement could be useful for following a user's hand or demonstrating an elliptical animation as shown in our orbiter demo (see Figure 3).

## 3.3 Homogenous vs. Heterogeneous Stacks

Stacking can be done with modules of the same size or with different sizes; where bigger modules are used for the base of the stack. Using larger modules at the bottom of the stack and smaller ones on top provides more stability and uses less actuators. However, heterogeneous stacks limits the flexibility of movement of the small modules on top. If two small modules are stacked on top of a bigger one, they follow the movement of the base module and cannot move independently. If they were stacked on two modules of the same size then they can move independently.

The number of modules on per level can also be heterogeneous. Layers might have fewer modules at the top than at the bottom. This enables multi-level shape renders that can be used for visualizing hierarchal data, dependencies, or buildings.

## 3.4 Gaps and Overlaps

An advantage of tilting and stacking is the ability to render physical shapes with gaps and overlaps. This imposes a challenge of handling possible collisions and gaps in the display surface, but also creates interesting opportunities. For example, users can interact with display surfaces on the inside of the cluster of modules. This can be used for geological applications where the inside of the display surface represents the inner layers of earth.

Gaps and overlaps can solve another limitation of interacting with tangible interfaces, which is the lack of appearance and disappearance actions of the tangible device. Prior work has emulated such interactions through projection [2,9] and moving the object away from the focus of the user [22]. Appearance and disappearance interactions can be implemented using gaps and overlaps. The stacked modules can reveal and hide tangible objects using gaps and overlaps in the display surface. The Marble Answering Machine implemented using inForm [2] can be implemented with physical appearance and disappearance of the marbles through using splitting and emerging of pixels.

## 3.5 Technical Considerations

Tilting and stacking of modules introduces technical challenges that need to be considered. Weight is an important parameter to account for when stacking. The lower level modules need to have enough power to lift the upper levels stacked on top of them. Weight distribution also affects the structural integrity of the stacked modules. Self-balancing and proper weight distribution is important.

Scalability of the stacked modules is a design limitation for tilting and stacking. Managing a lot of actuators is challenging and introducing modules with more degrees of freedom inherently means using more actuators. However, it is the compromise between rendering more complex shapes versus increasing the number of stacked modules that needs to be taken into consideration.

Implementing modules with multiple degrees of freedom could be done using a robotic arm or Stewart platform. However, using them fail short in modularity, stacking, and scalability. They do not have a uniform footprint relative to their top, which limits building a uniform display surface through stacking them. Linear actuators seem a good fit for implementing tilting and stacking as they are easily configured in a uniform module to be stacked horizontally and vertically. Therefore we used linear actuators in order to implement a proof-of-concept prototype as discussed in the following section.

# 4 TILTSTACKS IMPLEMENTATION

We implemented TiltStacks to demonstrate our composition concept of building shape-changing interfaces with five degrees of freedom using the combination of tilting and stacking. They consist of three linear stepper motors controlling the tilt and height of a horizontal plate attached to their top. To achieve 30 degrees of roll and pitch of the plate, we placed the motors in a non-collinear layout and attached them to the plate using ball joints. The top plate is then attached either to the bottom of other modules for vertical stacking or to displays for visual output.

We used stepper motors (Nema 11 linear stepper motors, 100 mm travel) for their high holding torque and good positioning. This helps in lifting the weight of the stacked modules and achieving accurate tilt orientations. Each motor weighs 130 g, has 1.8 degrees step angle and a holding torque of 8 Ncm. They are controlled by three A4988 motor drivers connected to an Arduino. For visual output we use four 1.5'' displays per module to prototype a square-shaped display surface. Images were stored on the SD cards and updated using an Arduino. A software solution

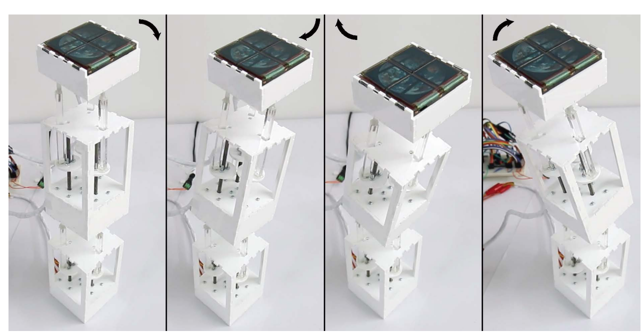

**Figure 3: Orbiter. The display surface moves in a planar orbit while showing a picture of planet earth.**

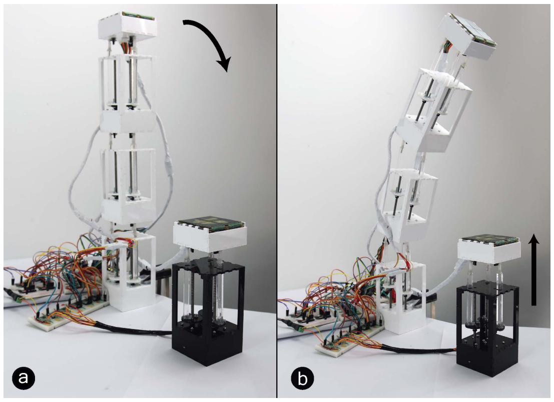

**Figure 4: Watering flowers. The white stack moves to form a curved display with an overhang on top of the black module. (a) start state. (b) end state.**

was implemented to split images into tiles before storing them on the SD cards of the displays.

## 5 DEMONSTRATIONS

We implemented four applications demonstrating displays with overhangs, planar movements, non-continuous surfaces, and combining non-continuous surfaces with planar movements.

**Watering Flowers:** We use TiltStacks to generate a shape-changing interface with curvature and overhang. The stack of the white modules curves towards the black module, mimicking the movement of pouring water on a flower as shown in Figure 4. The black module reacts to the curving movement by increasing its height, showing a blooming flower (see Figure 4 (b)).

**Orbiter:** We demonstrate a shape-changing display interface that moves in a planar circular path. TiltStacks shows the movement of planet earth in orbit as shown in Figure 3. Tilting of the two modules combined provides five degrees of freedom for the top display. This enables the display to move in a planar circular movement.

**Battleships Game:** We demonstrate the ability of TiltStacks to build a shape-changing interface with a non-continues display for playing Battleships. Battleships is a two-player guessing game where each player tries to guess the arrangement of the fleet of the other player. Battleships gameplay requires that the two players have a private part of the display so as not to reveal their fleet layout. TiltStacks starts as a single continuous display, then splits into two halves where each half moves towards a player to provide privacy for the players as shown in Figure 6.

**Newton's cradle:** We mimic the movement of Newton's cradle (a series of swinging balls that demonstrates the conservation of momentum and energy) to demonstrate the ability of TiltStacks to produce shape-changing display interfaces with non-continuous surfaces and planar movement. We use two adjacent stacks of TiltStacks where each stack consists of two modules as shown in Figure 5. The stacks demonstrate a continuous swinging animation showing silver spheres on their displays. The stack on the left-hand side swings towards the right, colliding with the other stack (see Figure 5 (a, b)). Afterwards, the

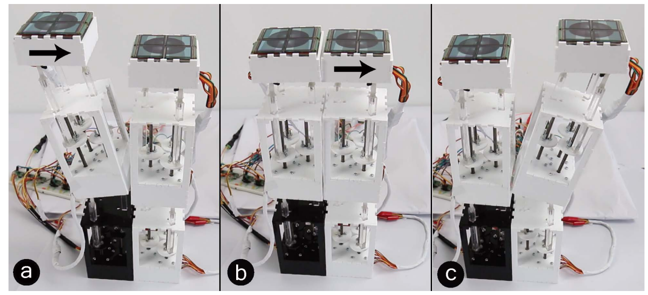

**Figure 5: Newton's cradle. (a) Left TiltStack swings from the left. (b) Left TiltStack hits the right one. (c) Right TiltStack swings to the right.**

stack on the right-hand side continues the swinging movement towards the right (see Figure 5 (c)).

## 6 DISCUSSION AND CONCLUSION

We presented a composition concept that combines stacking and tilting to achieve shape-changing display interfaces with overhangs, overlaps, and non-continuous surfaces. Our main contribution is not technical, but a principle of composition not limited to our implementation. Combining tilting and stacking achieves five degrees of freedom and can thus be used to generate shapes that no previous rod-based display has been able to generate because they almost exclusively have used arrays of linear actuators only [2,10,15,22].

Our prototype is intended as an initial illustration of modules that tilt and stack. As such, it is limited in terms of tilt angle and speed of motion. Interactivity was omitted, displays used pre-stored visuals, and connectivity is achieved with wires. The vision of TiltStacks, however, assumes numerous interactive modules at any scale with tilting capabilities close to 90°, and display surfaces that cover the entire top of each module. This will create novel configurations such as curved displays that can expand their screen-real estate or make cylindrical or ball-shaped surfaces; displays that can be physically split aside with new display modules appearing underneath. In doing so, TiltStacks opens a new world of opportunities for shape-changing display interfaces.

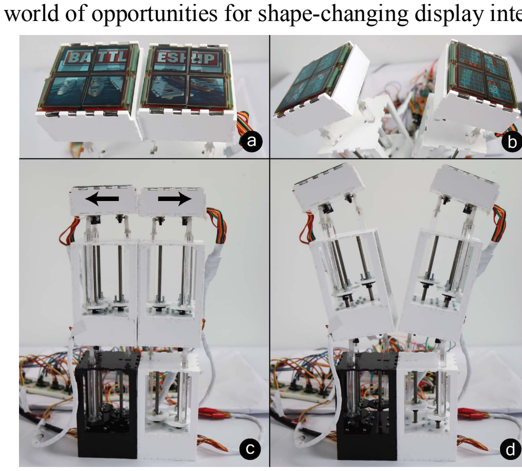

**Figure 6: Battleships game. (a, b) Top view of the displays (c) TiltStack shows a continuous display when the game starts. (d) TiltStack is split to provide privacy for each player.**

## REFERENCES

1. Jason Alexander, Andrés Lucero, and Sriram Subramanian. 2012. Tilt Displays: Designing Display Surfaces with Multi-axis Tilting and Actuation. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services* (MobileHCI '12), 161–170. https://doi.org/10.1145/2371574.2371600

2. Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. inFORM: Dynamic Physical Affordances and Constraints Through Shape and Object Actuation. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (UIST '13), 417–426. https://doi.org/10.1145/2501988.2502032

3. Antonio Gomes, Calvin Rubens, Sean Braley, and Roel Vertegaal. 2016. BitDrones: Towards Using 3D Nanocopter Displays As Interactive Self-Levitating Programmable Matter. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (CHI '16), 770–780. https://doi.org/10.1145/2858036.2858519

4. John Hardy, Christian Weichel, Faisal Taher, John Vidler, and Jason Alexander. 2015. ShapeClip: Towards Rapid Prototyping with Shape-Changing Displays for Designers. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15), 19–28. https://doi.org/10.1145/2702123.2702599

5. Hiroo Iwata, Hiroaki Yano, Fumitaka Nakaizumi, and Ryo Kawamura. 2001. Project FEELEX: Adding Haptic Surface to Graphics. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (SIGGRAPH '01), 469–476. https://doi.org/10.1145/383259.383314

6. Sungjune Jang, Lawrence H. Kim, Kesler Tanner, Hiroshi Ishii, and Sean Follmer. 2016. Haptic Edge Display for Mobile Tactile Interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (CHI '16), 3706–3716. https://doi.org/10.1145/2858036.2858264

7. D. Leithinger, S. Follmer, A. Olwal, and H. Ishii. 2015. Shape Displays: Spatial Interaction with Dynamic Physical Form. *IEEE Computer Graphics and Applications* 35, 5: 5–11. https://doi.org/10.1109/MCG.2015.111

8. Daniel Leithinger, Sean Follmer, Alex Olwal, and Hiroshi Ishii. 2014. Physical Telepresence: Shape Capture and Display for Embodied, Computer-mediated Remote Collaboration. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (UIST '14), 461–470. https://doi.org/10.1145/2642918.2647377

9. Daniel Leithinger, Sean Follmer, Alex Olwal, Samuel Luescher, Akimitsu Hogge, Jinha Lee, and Hiroshi Ishii. 2013. Sublimate: State-changing Virtual and Physical Rendering to Augment Interaction with Shape Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13), 1441–1450. https://doi.org/10.1145/2470654.2466191

10. Daniel Leithinger and Hiroshi Ishii. 2010. Relief: A Scalable Actuated Shape Display. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction* (TEI '10), 221–222. https://doi.org/10.1145/1709886.1709928

11. Daniel Leithinger, David Lakatos, Anthony DeVincenzi, Matthew Blackshaw, and Hiroshi Ishii. 2011. Direct and Gestural Interaction with Relief: A 2.5D Shape Display. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (UIST '11), 541–548. https://doi.org/10.1145/2047196.2047268

12. Ken Nakagaki, Sean Follmer, and Hiroshi Ishii. 2015. LineFORM: Actuated Curve Interfaces for Display, Interaction, and Constraint. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (UIST '15), 333–339. https://doi.org/10.1145/2807442.2807452

13. Ken Nakagaki, Luke Vink, Jared Counts, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. 2016. Materiable: Rendering Dynamic Material Properties in Response to Direct Physical Touch with Shape Changing Interfaces. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (CHI '16), 2764–2772. https://doi.org/10.1145/2858036.2858104

14. 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), 223–232. https://doi.org/10.1145/2807442.2807494

15. Ivan Poupyrev, Tatsushi Nashida, Shigeaki Maruyama, Jun Rekimoto, and Yasufumi Yamaji. 2004. Lumen: Interactive Visual and Shape Display for Calm Computing. In *ACM SIGGRAPH 2004 Emerging Technologies* (SIGGRAPH '04), 17–. https://doi.org/10.1145/1186155.1186173

16. Majken K. Rasmussen, Esben W. Pedersen, Marianne G. Petersen, and Kasper Hornbæk. 2012. Shape-changing Interfaces: A Review of the Design Space and Open Research Questions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '12), 735–744. https://doi.org/10.1145/2207676.2207781

17. Anne Roudaut, Abhijit Karnik, Markus Löchtefeld, and Sriram Subramanian. 2013. Morphees: toward high shape resolution in self-actuated flexible mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 593–602. Retrieved March 21, 2014 from http://dl.acm.org/citation.cfm?id=2470738

18. Philipp Schoessler, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. 2015. Kinetic Blocks: Actuated Constructive Assembly for Interaction and Display. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (UIST '15), 341–349. https://doi.org/10.1145/2807442.2807453

19. Julian Seifert, Sebastian Boring, Christian Winkler, Florian Schaub, Fabian Schwab, Steffen Herrdum, Fabian Maier, Daniel Mayer, and Enrico Rukzio. 2014. Hover Pad: Interacting with Autonomous and Self-actuated Displays in Space. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (UIST '14), 139–147. https://doi.org/10.1145/2642918.2647385

20. Miriam Sturdee, John Hardy, Nick Dunn, and Jason Alexander. 2015. A Public Ideation of Shape-Changing Applications. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces* (ITS '15), 219–228. https://doi.org/10.1145/2817721.2817734

21. Faisal Taher, John Hardy, Abhijit Karnik, Christian Weichel, Yvonne Jansen, Kasper Hornbæk, and Jason Alexander. 2015. Exploring Interactions with Physically Dynamic Bar Charts. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15), 3237–3246. https://doi.org/10.1145/2702123.2702604

22. Luke Vink, Viirj Kan, Ken Nakagaki, Daniel Leithinger, Sean Follmer, Philipp Schoessler, Amit Zoran, and Hiroshi Ishii. 2015. TRANSFORM As Adaptive and Dynamic Furniture. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems* (CHI EA '15), 183–183. https://doi.org/10.1145/2702613.2732494

23. Lining Yao, Ryuma Niiyama, Jifei Ou, Sean Follmer, Clark Della Silva, and Hiroshi Ishii. 2013. PneUI: Pneumatically Actuated Soft Composite Materials for Shape Changing Interfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (UIST '13), 13–22. https://doi.org/10.1145/2501988.2502037

24. Lining Yao, Jifei Ou, Chin-Yi Cheng, Helene Steiner, Wen Wang, Guanyun Wang, and Hiroshi Ishii. 2015. bioLogic: Natto Cells As Nanoactuators for Shape Changing Interfaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15), 1–10. https://doi.org/10.1145/2702123.2702611