# Combining Touchscreens with Passive Rich-ID Building Blocks to Support Context Construction in Touchscreen Interactions

Chin-Yuan Lu*
National Taiwan University
chinyuanlu@cmlab.csie.ntu.edu.tw

Han-Wei Hsieh*
National Taiwan University
hanweihsieh@cmlab.csie.ntu.edu.tw

Rong-Hao Liang*
Eindhoven University of Technology
r.liang@tue.nl

Chi-Jung Lee
National Taiwan University
cjlisalee@cmlab.csie.ntu.edu.tw

Ling-Chien Yang
National Taiwan University
natalieyang@cmlab.csie.ntu.edu.tw

Mengru Xue
Eindhoven University of Technology
m.xue@tue.nl

Jr-Ling Guo
National Taiwan University
jrling@cmlab.csie.ntu.edu.tw

Meng-Ju Hsieh
National Taiwan University
mjhsieh@cmlab.csie.ntu.edu.tw

Bing-Yu Chen
National Taiwan University
robin@ntu.edu.tw

## ABSTRACT

This research investigates the design space of combining touchscreens with passive rich-ID building block systems to support the physical construction of contexts in touchscreen interactions. With two proof-of-concept systems, RFIPillars and RFITiles, we explore various schemes for using tangible inputs for context enrichment in touchscreen interactions. Instead of incorporating an electronic touchscreen module that requires per-module maintenance, this work intentionally makes each tangible object passive. We explore rear-projection solutions to integrate touchscreen interactions into these passive building blocks with capacitive touch sensing techniques and deliberate physical forgiving to retain the merits of being both batteryless and wireless. The presented research artifacts embody the interaction designs and elucidate scalability challenges in integrating touchscreen interactions into this emerging tangible user interface.

## CCS CONCEPTS

• **Human-centered computing → Touch screens**; **Systems and tools for interaction design**; • **Hardware → Emerging interfaces**; **Sensor applications and deployments**.

## KEYWORDS

RFID, stackable, touchscreen, rich-ID, building blocks, capacitive sensing, rear-projection, modular interface, tangible user interface

**ACM Reference Format:**
Chin-Yuan Lu, Han-Wei Hsieh, Rong-Hao Liang, Chi-Jung Lee, Ling-Chien Yang, Mengru Xue, Jr-Ling Guo, Meng-Ju Hsieh, and Bing-Yu Chen. 2021. Combining Touchscreens with Passive Rich-ID Building Blocks to Support Context Construction in Touchscreen Interactions. In *CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3411764.3445722

## 1 INTRODUCTION

Passive rich-ID building blocks [15], based on ultrahigh-frequency radio-frequency identification (UHF RFID) sensing, were recently proposed to support the interactive semantic construction of digital information. These rich-ID building blocks support wireless geometry resolution by leveraging a paired magnetic contact switch mechanism, which ensures a reliable low-ohmic connection and physical alignments between the modified RFID tags such that the (un)stacking events can be resolved from the concurrent presence or absence of two tags. Thanks to the batteryless and wireless UHF RFID tags, these rich-ID building blocks provide virtually unlimited IDs without a battery or microcontroller, making deployment and maintenance easier than for active building blocks when at scale.

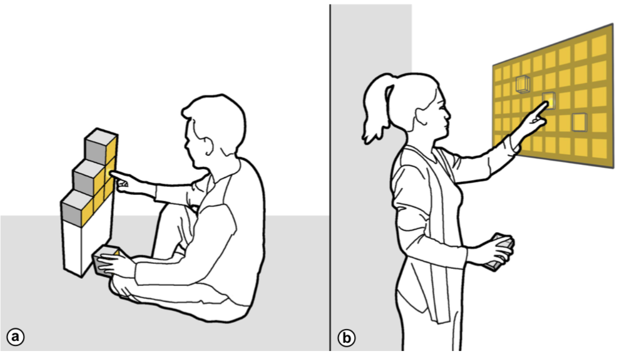

**Figure 1: Two conceptual systems that provide fluent transition between touchscreen and tangible interaction modes on passive rich-ID building blocks. (a) Constructive assemblies: an expansible touchscreen made with rich-ID cubes, allowing for a semantic screen construction by stacking them on the base station; (b) Stackable touchscreen: a tile-like touchscreen that recognizes transparent rich-ID cards stacked on it and allows for touch interactions through the cards.**

[*] Contributed equally.

*CHI '21, May 8–13, 2021, Yokohama, Japan*
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8096-6/21/05.
https://doi.org/10.1145/3411764.3445722

Nonetheless, because these passive building blocks are not touchscreens, they do not support touchscreen interactions, which are often more effective and intuitive. Interacting with these building blocks usually relies on only stacking operations; thus, the inputs are discrete, gross-grained, and effortful. Furthermore, because the building blocks do not provide a dynamic display, interaction with them usually relies on an external screen, which makes the visual output indirect. Hence, a seamless integration of touchscreen interactions that enables the performance of more fluent user experiences on this emerging tangible user interface [17] is desired.

Simply adding an active touchscreen module to the surface of each rich-ID block can be a straightforward yet effective solution, as demonstrated in PickCells [9]. However, introducing power electronics per unit increases hardware and maintenance costs, burdening further system deployment. Therefore, we explore a more challenging solution to the following research question: *How might touchscreen interactions be integrated into the building blocks in a more scalable manner?*

In this paper, we took a research-through-design [52] approach to investigating a plausible design for touchscreen integration using passive rich-ID building blocks. We realized two conceptual systems (Figure 1) to embody touchscreen interactions on passive rich-ID building blocks, either as *constructive assemblies* or as a *stackable touchscreen*, through prototyping two proof-of-concept systems, RFIPillars and RFITiles. With deliberate technical designs using rear-projection and RFID-capacitive sensor fusion techniques, we enabled a seamless and fluent transition between touch and tangible interactions without adding power electronics to each block. We constructed fantasy applications upon these prototypes to demonstrate the interaction styles.

## 1.1 Constructive Assemblies: RFIPillars

The first system, *RFIPillars* (Figure 2), comprises cubes and stations. A cube is a passive rich-ID building block augmented by a rear-projection screen with an overlain transparent capacitive electrode matrix; a station is an active unit in which a pico projector and a signal processing unit are embedded. Through the lens and mirrors built into the cubes, each station projects a dynamic visual display for each cube stacked upon the station and processes the touch input events from the cubes to enable touchscreen interactions.

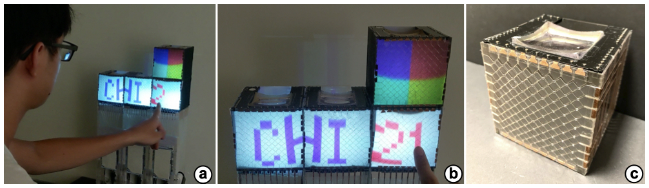

**Figure 2: (a,b) *RFIPillars* is an interactive building block system that enabled occlusion-free touchscreen interactions on rich-ID passive stackables by leveraging rear-projection. (c) passive cube.**

*Defining the Touch Interaction Context through Semantic Stacking.* The cubes in RFIPillars are in a 3D form that provides rich possibility and tangibility for interaction design [24], and they have a bezelless visual display that provides continuity. Therefore, the cubes form an expandable touchscreen that allows users to define the interaction context through semantic stacking.

Figure 3 depicts an advanced Tangible Minecraft game, which was extended from a previous work [15]. We deployed a 1×2 grid of stations as a playground and used several cubes to represent four types of building blocks: grass, tree, rock, and chicken. The user first stacks a tree on the grass and sees the tree root sink into the grass yard. The tree block displays the weather through the movement of clouds. Thereafter, the user places a stone next to the tree and puts the chicken on the rock to see a chicken nest atop the rock. The user then teases the chicken by touching the block and sees the happy chicken lay eggs into the nest.

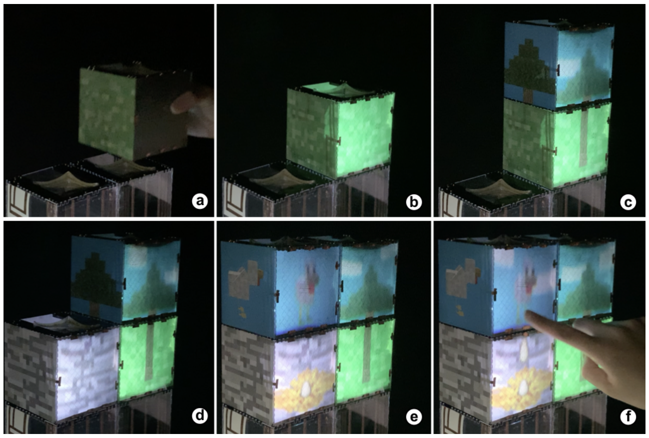

**Figure 3: In Tangible Minecraft, the touch interaction context is defined through semantic stacking operations.**

*Two-Dimensional Stacking Using Portable Stations.* The cubes can only be stacked in one dimension on each station, but the stations can be stacked side by side to extend the stacking operations to the second dimension.

Figure 4 portrays a Tangible Room Escape game. We deployed three stations as the playground and three cubes representing different rooms: one with a gate and a upward-facing ladder, one with a downward-facing ladder, and one with a wall splitting the room into two compartments, where a treasure chest is in one of them. The user stacks the rooms on the ground and touches the screen to move the character to the desired location. The user merges two stations to connect two rooms, allowing the character to walk into another room. When there seems to be no way to obtain the chest, the user finds a new route by swapping the two stations and stacking them. Finally, the user builds a path downstairs by stacking a room with a ladder down on another one with a ladder up and then walks the character out the gate to escape.

These examples demonstrate seamless transition between the modes of touchscreen and semantic stacking interaction. The users use rich-ID building blocks to define the context and perform fine-grained touch inputs on the focus. The building blocks afford semantic constructions that make screen expansion meaningful. The

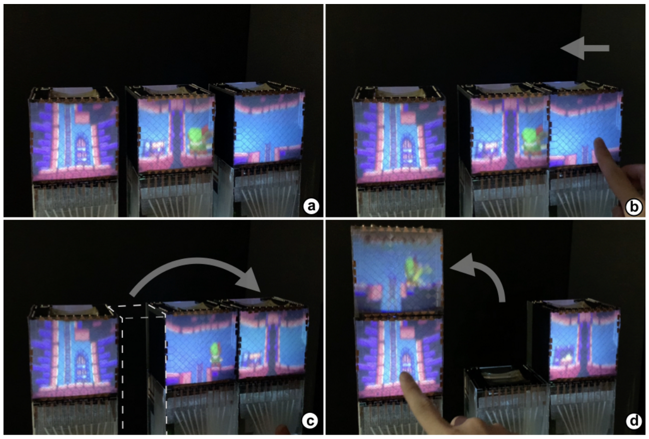

**Figure 4: In Tangible Room Escape, users solve a puzzle by stacking the cubes in one direction, and stacking the stations in another direction.**

rear-projection and capacitive touch sensing support occlusion-free touchscreen interactions.

## 1.2 Stackable Touchscreen: RFITiles

The second system, *RFITiles* (Figure 5), is an improvement of the previous RFIDesk [14] system that is aimed at reducing the visual parallax problem and providing higher-resolution touch interaction. The RFITiles system comprises cards and tiles. Each card is a thin-form passive rich-ID building block augmented by an overlain transparent capacitive electrode matrix. Tiles are tiled on a rear-projection surface and connected to a signal processing unit. The rear-projection interactive surface provides a dynamic visual display for each card and tile stacked upon it, and the system processes the touch input events from the capacitive electrode matrix to enable touchscreen interactions through the cards and tiles.

*Upgrading Modes Through Semantic Stacking.* The cards are stacked at the same location, allowing for the upgrading of the modes according to the cards used in the semantic stacking. Figures 5 and 6 are images of an improved tangible version of a grid-based tower defense game, Plants vs. Zombies [1], that has been implemented in RFIDesk [14]. In the game, users can deploy, upgrade, and combine weapons and defenses by stacking the ID cards at the grid position and can still perform touch inputs to trigger events. For instance, the user can upgrade a weapon by stacking the same cards or fuse different weapons by stacking different ID cards. Notably, the rich-ID stackables proposed in this work provide better visual experiences and higher touch input resolution than does the previous implementation, which was disadvantaged by notable visual parallax and tag antenna occlusions.

*Rich Touch Interactions through the Stack.* Input methods may vary between input modes. Figure 7 demonstrates the greater variety of touch interactions in a cooking game that were made possible by stackables. The user can stack the passive cards to bring materials into the game. The user can stack the passive cards to bring

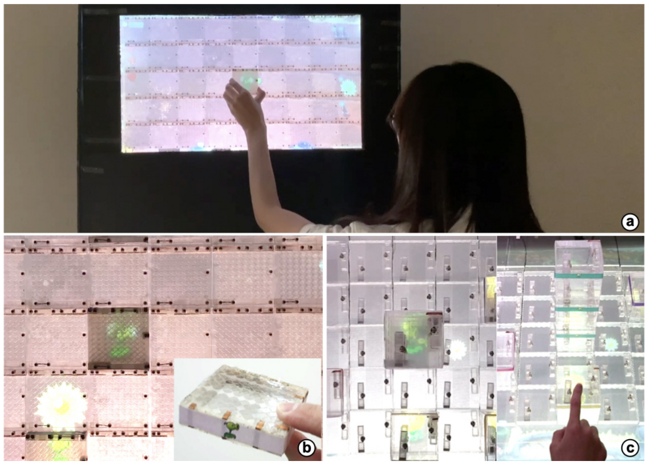

**Figure 5: (a) *RFITiles* is a tile-like touchscreen that support touch inputs through a stack of rich-ID cards on the tiles. (b) The card implementation in RFITiles provides better visual experiences and higher touch input resolution than do (c) the block implementation in the previous RFIDesk system [14].**

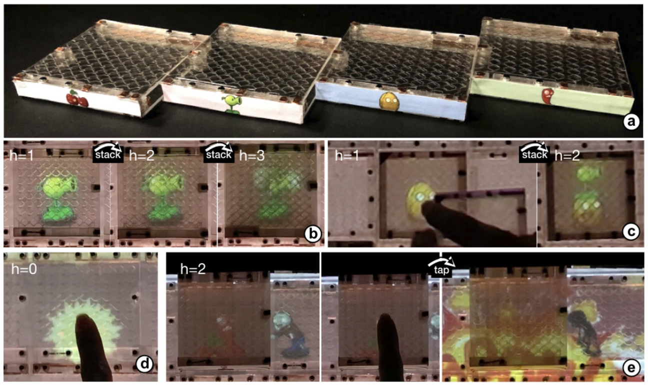

**Figure 6: In Tangible Tower Defense, the user incrementally updates the mode of touch inputs at the same location by stacking rich-ID cards. The order and combination of the stacked cards enrich the context.**

the materials onto the production line and then modify their parameters (e.g., quantity, frequency, and amplitude) through direct touch inputs, which allow the user to complete these tasks more efficiently. For instance, instead of stacking three tomato cards, the user can achieve the same result by calling out a numpad with a double-click gesture and then typing the number (Figure 7c and 7d). The tap and swipe operations also make the discrete interactions more engaging (Figure 7e) and fluent (Figure 7f). More importantly, these touch inputs are volatile. Thus, they do not occupy the stack, and users can fully utilize the rich-ID stackables for semantic construction. Users can bring a stack of a sorted list of materials to the stage and perform the operation in a batch, as in the tangible programming example demonstrated in RFIBricks [15].

[1] https://www.ea.com/studios/popcap/plants-vs-zombies

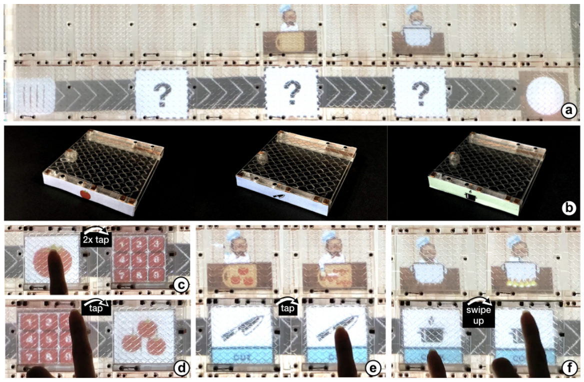

**Figure 7: In Tangible Programming applied to a cooking task, the user performs various touch input methods after building the context by stacking rich-ID cards.**

### 1.3 Summary and Contributions

Stack inputs are discrete and effortful after an extended period of use, especially with graspable objects such as RFIBricks [15]. Therefore, leveraging GUI and touchscreen inputs on the surface of stackables can increase the fluidity and expressivity of interactions. Rich-ID blocks do not require further GUI configurations on the user side because every unit is unique. The bindings can be preprogrammed; thus, they are perfectly suitable for building the background context of foreground touchscreen interactions. To fully leverage the scalability of rich-ID systems, we made these blocks passive, maintenance-free, and as calm as Lego bricks for ubiquitous computing environments [44].

The main contribution of this paper is two-fold. 1) The realization of a novel interaction system that supports a smooth transition between touchscreen and semantic stacking interactions. Users of this system can construct the touchscreen interaction context using rich-ID building blocks and then interact with the focused content through rich and efficient touch interactions in a post-WIMP interaction scheme [19]. 2) The knowledge created in the process and the results of building these research artifacts help the HCI community better understand the solution space of reconfigurable touchscreen designs. We disclose the practical limitations of this passive approach through a series of technical evaluations, which also inform directions for future development. We further reflect on our assumptions by comparing our rear-projection approach with the electronic one, thereby providing a relatively nuanced design prescription for deployment scales.

The remainder of the paper is organized as follows. First, we discuss the related work. Then, we present the system design and proof-of-concept implementation with a series of technical evaluations. Finally, we discuss the limitations and design implications and draw a conclusion.

## 2 RELATED WORK

### 2.1 Tangible User Interfaces With Stackables

Tangible user interfaces (TUI) seamlessly couple digital information with physical forms [17] and exploit existing cognitive and spatial knowledge to embody and comprehend the digital information. Stacking is a common task that we perform in spatiotemporal organization, such as when grouping or ordering things. Piling is a lightweight, casual activity with little cognitive overhead [29]. Compared with 2D clustering, stacking is also a space-saving method to extending the property of an object, which also makes the object's location perpetually stable, which is especially important in token+constraint systems [43] in which locations are typically meaningful. Stackables are also types of constructive assemblies that support a prolonged tangible interaction lifecycle [26], which engages users in not only HCI but also processes of physical creation, usage, tweaking, storage, and destruction.

Researchers have enabled stackable TUIs through two approaches. The first is the detection of stack events by using embedded electronic circuits and sensors, such as connectors [21], motion sensors [13], a single IR sensor array [1], or conductive dot patterns [8]. Nonetheless, the deployment and maintenance costs of the power electronics may limit their scalability and sustainability. The second is based on passive sensing techniques involving the use of, for example, optical markers [2], markered fiber-optic bundles [3], capacitive footprints [5], pressure images [25], or magnetic-field images [27, 28] to allow the stacking state to be detected by an external sensor. However, the ID space and stacking height are limited in these solutions.

RFID technologies provide a virtually infinite ID space. Therefore, with RFIBricks [15], a UHF RFID tag was modified into contact switches. Stack events and user inputs on widgets can therefore be identified through physical contact. RFIDesk [14] further incorporates capacitive touch sensing to allow users to interact with the display surface where the blocks are placed through touching the block surface. Nonetheless, the tokens used in these systems are too thick and not desirably transparent; the touch sensing is not localized on the blocks. The visual parallax problem causes a severe bottleneck in higher-resolution touchscreen interactions.

### 2.2 Reconfigurable Displays

Several works have succeeded in designing reconfigurable modular displays to support constructive tangible interactions [31, 32] or to extend the display area [23], but they have not realized support for touchscreen interactions in their display area. PickCells [9] is the system of modular touchscreen modules that is most related to our work. Each single-touch display module of PickCells supports 2D localization when connected, without relying on additional object-tracking infrastructure. However, PickCells neither allows z axis stacking (e.g., in RFITiles) nor supports whole-stack operations (e.g., in RFIPillars). Regarding touch inputs, the self-capacitance touch sensing matrix on RFITiles and RFIPillars units support more expressive gestures once the researcher has full access to the 2D touch sensing raw data. The size of RFITiles and RFIPillars also affords a greater variety of touch gestures. Regarding deployment cost, six PickCells units cost £300 [9], thus, building a larger screen, such as the 9×5 RFITiles, is less economical with PickCells. Building PickCells with larger touchscreens, such as that in the cubes in RFIPillars, also increases its cost per unit. By contrast, our passive cubes can be fabricated cheaply with an epoxy mold, vinyl-cut copper, and indium tin oxide (ITO) sheets, RFID tags, and magnets. Hence, our rear-projection method pays off in an upscaled deployment that features more and/or larger modules.

## 2.3 Sensing Touch on Interactive Surfaces

Cameras and projectors are commonly used for sensing touch interactions on a surface [11, 18, 49, 50]. However, the placement of the camera and the projector is critical because of line-of-sight problems and, therefore, may require a rear-projection setting [10, 30] or the use of multiple cameras and projectors to compensate for shadows and occlusions. Instead of using cameras, several works have used conductive patterns [5, 12, 20, 27, 40] or dielectric materials [51] to enable touch through the passive objects by using capacitive touchscreens or dense capacitive sensor grids [6, 37].

## 2.4 Transparent Tangibles on Screen

Researchers have exploited transparent tangible interactions to allow for more visual feedback to be perceived on the screen [4]. These transparent tangibles have been made into tokens or tools [42], tiles [22, 38], and haptic proxies [33, 39, 47] for facilitating HCI with digital information. With on-screen visual labeling, a transparent object can be bound with digital information as a generic handle and as a rich-haptic tabletop controller [46], which can even be actuated on the tabletop for two-way interactions [45]. Despite transparent tokens being widely applied in facilitating touchscreen interactions, they were mainly designed to provide parametric control (e.g., as a knob) rather than for stacking.

# 3 RFIPILLARS

## 3.1 Design and Implementation

The RFIPillars system comprises two parts: 1) the cubes, which are the building blocks, and 2) the station, which is the display and signal processing unit. Figure 8 depicts the cube and station hardware implementation. Both were made with laser-cut acrylic sheets. The dimensions of each cube are 85 (W)×85 (L)×85 (H)$mm^3$, which allows users to grasp it in one hand comfortably. The dimensions of the station are 85 (W)×85 (L)×272 (H)$mm^3$.

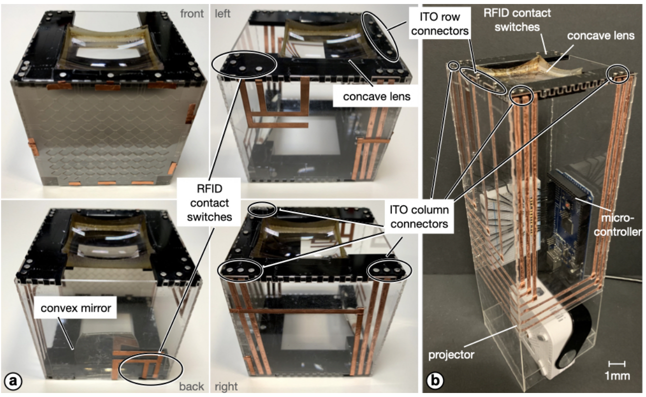

**Figure 8: Hardware overview of a 3×3 touch resolution. (a) Cube and (b) station.**

### 3.1.1 *Sensing Touch Inputs.*

To enable touch interactions on the cubes, we deployed a rear-projection screen with a transparent, ITO-made 9-mm-pitch 9×9 DiamondTouch [6] pattern that can support an up to 9×9 resolution of touch input without further interpolation. We implemented cubes and stations with 3×3 (Figure 8) and 9×9 (Figure 9) resolutions.

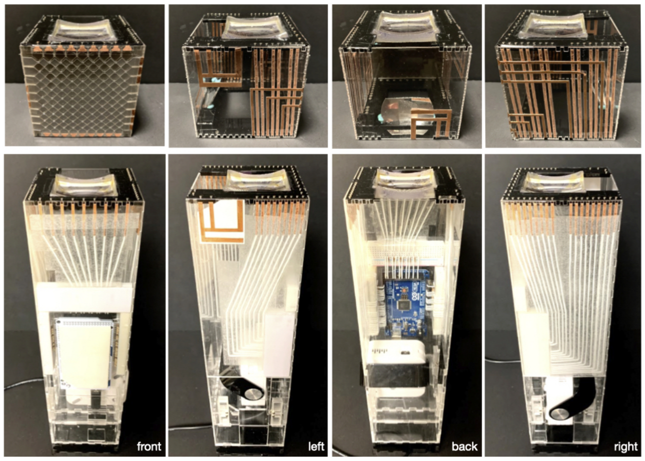

**Figure 9: Cube and station implementation with 9×9 resolution of touch input.**

For the sensing of touch inputs on the cube front surface, we connected ITO sensors to an Arduino board through magnets. We designed and implemented a signal-passdown circuitry to enable stackable 2D touch sensing with copper tape. Figure 10 depicts the circuitry design on a cube and a station. The circuitry aggregates signals from all the rows and columns of the stacked cubes to the station. The Arduino board is used for touch signal processing. Touch events are detected using the CapSense library in self-capacitance mode, which outputs analog signals for later system analysis. Touch events are sent to the application through a serial connection. Unity is used to implement the application software. Consequently, the 2D touch event of an entire stack of cubes can be resolved (Figure 11).

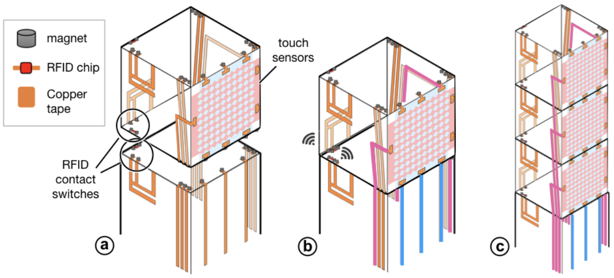

**Figure 10: Sensor deployment of an example RFIPillar with 3×3 touch sensing resolution. (a) Cube and station, (b) stacking a cube on a station activates two RFID switches simultaneously, and (c) signal pass-down circuitry aggregates all the sensors on the stack.**

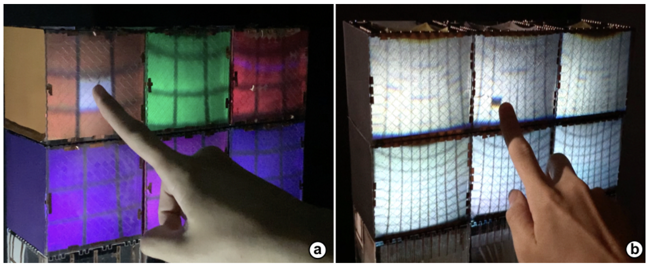

**Figure 11: (a) Two-layer stacks of 3×3-resolution cubes on a 1×3 grid of stations. (b) 9×9 resolution cubes on a 1×3 grid of stations.**

*3.1.2 Visual Display.* An L-mix DLP mini projector, which provides a 1500–2000-lumen 1080p display, is fixed at the bottom of the station for bottom-up projection. To display visual content on cubes stacked on the station's front surface, we used a combination of a mirror and a concave lens. The mirror deflects visuals from the station to the front surface, and the concave lens restores the graphic source and passes it to the upper layer. In our prototype, we first determined the type and placement of the mirror and the lens through an iterative optimization process. We then unified the manufacturing of cubes with laser-cut acrylic supports. Next, we calibrated the visuals on every layer of the stack by using a checkerboard pattern, a camera, and MATLAB software. After the camera's intrinsic parameters were determined, we used the homogeneous matrix obtained to unwarp the distortion caused by the concave lens, with the results as shown in Figure 12.

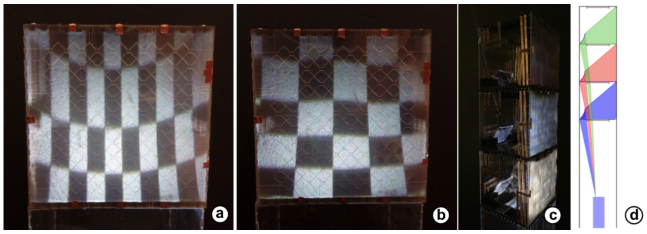

**Figure 12: Visual display. (a) before and (b) after calibration, (c) results, and (d) projection path.**

*3.1.3 Sensing Stacking Operations.* The system can resolve cube-to-cube, cube-to-station and station-to-station stacking events.

*Cube-to-Cube and Cube-to-Station Stacking.* To identify and localize stacking events between the cubes and the stations, we applied RFID contact switches [15], which were modified from conventional UHF RFID tags. The RFID contact switch comprises two parts: a (normally off) Alien Higgs 3 RFID chip and a UHF RF antenna. Each of the two terminals had an attached magnetic connector made of a convex magnet whose size at the bottom was 4 mm (diameter; $\phi$)×1 mm (T) and, at the top, 3 mm ($\phi$)×2 mm (T). When stacking a cube onto a station or another cube, the two switches placed at the corresponding locations of the two surfaces activate simultaneously. Then, the combination of two unique IDs of the two switches represents the location and the stacking event [15].

*Station-to-Station Stacking.* To identify the station's relative position, we also applied the RFID contact on two sides of the stations. After one station connects to another, the two switches placed at the two corresponding surface locations activate simultaneously. The combination of the two unique IDs of the two switches represents the relative location in a horizontal stack that can also be resolved by using the same techniques proposed in RFIBricks [15]. Therefore, stations can be deployed as an arbitrary 1D array.

*Implementation.* An Impinj Speedway Revolution R420 UHF RFID reader and two AANT925SMA circularly polarized antennas were fixed behind the stations for the sensing of the interaction events generated by the UHF RFID tags affixed on the stations and cubes. The RFID reader's signal band was configured between 902 and 928 MHz, and the signal amplitude was set to 32.5 dB. RFID tag events are sent to the Unity application through WebSocket.

## 3.2 Technical Evaluation

A series of measurements were collected to understand the system performance of our RFIPillars implementation.

*3.2.1 Session 1: Stack Sensing Capability.* The first session was focused on measuring stack sensing capability, determined by the number of stacked layers detected on a grid of stations when touch sensing was operational.

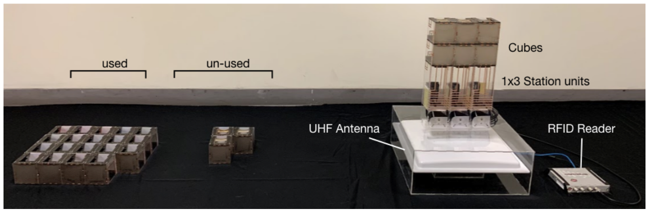

**Figure 13: Experimental apparatus in the technical evaluation of the RFIPillars.**

*Apparatus.* Figure 13 presents the experimental apparatus, which was installed in a 3 (W)×5 (L)×3 (H) $m^3$ empty space. During the measurements, a 1×3 grid of stations, placed on a table (height: 32 cm), served as a measurement platform and was arranged in the center of the room. A total of 28 cubes were prepared and applied in this session. The ANT925SMA circular polarized antenna was fixed vertically at 32 cm under the RFID tags mounted on the station. The antenna was wire-connected to the Impinj Speedway Revolution R420 UHF RFID reader placed under the antenna. The touch sensors were connected to Arduino mega boards embedded in each station for signal processing.

*Procedures.* The cubes were placed in three locations: in a *used* pool of cubes that had been tested, in an *unused* pool for those that had not been tested, and on the testing platform. These three locations were separated by 1 m. During each round of measurement, one of the unused cubes was randomly selected to be stacked on the center position, and the process was repeated until the stacking event was not detected. Subsequently, these used cubes were separated and moved to the used pool to avoid interference. Once the unused pool was empty, the used pool cards were transferred to the unused pool. In total, 10 rounds of measurements were conducted during this session.

*Results.* The results revealed the cubes can reach an average stack height of 4.3 layers (SD = 0.48) on the 1×3 station. The 95% CI indicates that the 1×3 station can reliably detect three levels of cubes in a full stack.

*3.2.2 Session 2: Touch Sensing Capability of 3×3-Resolution Touch Sensors.* The second session focuses on understanding the relationship between the sampling times and accuracy of capacitive touch sensing when the stack sensing is operating. We started from the one with 3×3-resolution touch sensors.

*Apparatus and Procedures.* The apparatus was similar to that in session 1. Nine cubes were stacked on the 1×3 station as a 3-level full stack. During the measuring of each touch sensing point, the user touched each point 100 times. The duration of each touch was at least 0.4 s, as determined by a pilot test. Within the 0.4 s, the capacitive sensing results for all touch sensors were collected eight times. In total, 64,800 records (100 touches×81 touch points×8 samples) of measurements were collected during this session. Accuracy was determined by the row and column reported by the system regarding actual finger position.

*Results.* As Figure 14a indicates, touch sensing reached 100% accuracy with seven samples, which required 0.35 s. The mean accuracy was 95.85% with only two samples, which required 0.1 s.

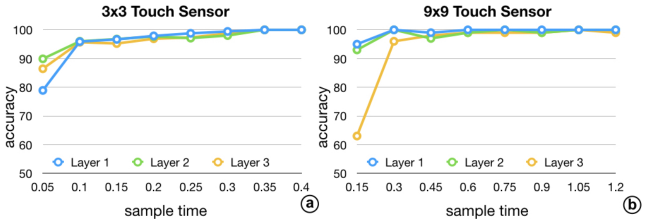

**Figure 14: Response time vs. accuracy. (a) 3×3 resolution and (b) 9×9 resolution.**

*3.2.3 Session 3: Touch Sensing Capability of 9×9-Resolution Touch Sensors.* Following the results of session 2, the aim of session 3 was understanding latency when the number of electrodes was increased to 9×9. Because the touch sensors were the same as those used in session 2, we conducted a simplified experiment.

*Apparatus and Procedures.* Three cubes were stacked on one station as a 3-level full stack. During the measuring of each touch sensing point, the user touched the center of each cube 100 times for at least 1.2 s, as determined by a pilot test. Within the 1.2 s, the capacitive sensing results of all touch sensors were collected eight times. In total, 2,700 records (100 touches×3 touch points×8 samples) of measurements were collected during this session. Accuracy was determined by the row and column reported by the system regarding actual finger position.

*Results.* As Figure 14b shows, touch sensing reached 100% accuracy at seven samples, which required 1.05 s. However, the mean accuracy was 98.67% with only two samples, which required 0.3 s.

*3.2.4 Session 4: Display Quality.* Session 4 was focused on understanding the display quality, that is, the illuminance and resolution of the hardware. A 1×3 station with a 3-level full stack of cubes was used for the measurement. To determine the maximum illuminance, we used a TES1330 meter, which measures a range of illuminance between 0 and 2000 lumens, as the sensing apparatus. We first displayed a fully white frame on each cube and attached the sensor to the center of the display surface. The sensor readings of each layer were taken in a block box. The results indicated that the first, second, and third layers of the cubes provided 81.7, 14.2, and 1.9 lumens, respectively. These values can be used to calibrate the illuminance of each layer of projection. However, the illuminance of the third layer was weak, thus, we recommend using the third layer only in a completely dark room.

To measure the resolution, we displayed the calibrated checkerboard pattern on each cube and measured the size of the rectangles to ascertain the ratio of the magnification. The first, second, and third layers of the cubes provided the pixel densities of 24.3–47.9, 9.7–19.2, and 3.9–7.7 ppi, respectively. This implies that the graphical quality of the third layer of cubes is approximately six times worse than that of the first layer. Thus, it can only support lower resolution content, such as visual feedback.

## 3.3 Summary

The RFIPillars provide a bezel-less visual display that provides continuity. The proposed touch sensing mechanism can also be generalized to a higher resolution. Nonetheless, the visual display quality decreases with the number of stacked layers. A projector should be embedded in each station unit, which would increase hardware costs when the system is deployed at scale.

# 4 RFITILES

## 4.1 Design and Implementation

The RFITiles system comprises two parts: 1) cards, which function as passive stackables, and 2) the surface, which is a layer of 9×5 tiles for stack and touch sensing. The cards and the surface support both vertical and horizontal uses.

*4.1.1 Cards and Tiles.* Four-edge and two-edge designs of cards and tiles are proposed.

*Four-Edge Cards and Tiles.* Figure 15 shows the hardware implementation of four-edge cards and tiles, which are both made using laser-cut acrylic sheets. The dimensions of each four-edge card are 62.5 (W)×62.5 (L)×10 (H) mm$^3$; this size allows users to grasp them comfortably. The dimensions of each four-edge tile are 62.5 (W)×62.5 (L)×4 (H) mm$^3$.

On the top of each four-edge tile and four-edge card, we used transparent ITO to make a 5 mm-pitch 9×9 DiamondTouch [6] electrode matrix that can support up to 9×9 touch sensing points without further interpolation. In our prototype, we simplified the connection for sensing 3×3 touch inputs for ease of implementation. The touch sensors on the tiles were connected to several Arduino boards via copper tape. The single-touch events were resolved with self-capacitance touch sensing using the CapSense library, which outputs analog signals for later system analysis. Touch events are

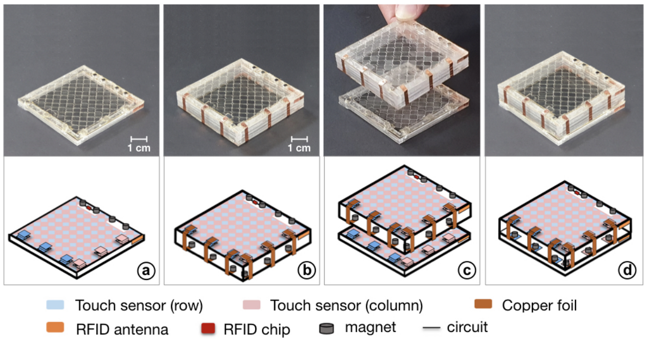

**Figure 15: Hardware overview of and touch sensing on four-edge cards and tiles. (a) Tile, (b) card, (c) stacking a card on a tile, and (d) results.**

sent to the Unity application through a serial connection. Unity is used to implement the application software.

We applied RFID contact switch pairs [15] to identify and localize the stacking events between the cards and the tiles on the surface. Because the transparency of the stackables is critical for touchscreen interactions, we propose a new design that uses 3 mm-width low-ohmic 3M 1181 copper tape to implement the T-match antenna pattern [35]. We deployed the antenna on one side of the four-edge card and tile (see Figure 16). As such, the top surface of the window of the four-edge cards and four-edge tiles was no longer occupied by the antenna. Consequently, each four-edge tile and card provided an area of interest of 52.5 (W)×52.5 (H) $\text{mm}^2$, which was the same size as a 3×3 grid in our explorative study, and thus provided a high-resolution visual display. We used 4-mm-diameter cylindrical neodymium magnets as connectors to both force the alignments of a stack and to retain a reliable electrical connection for both RFID and capacitive touch sensing.

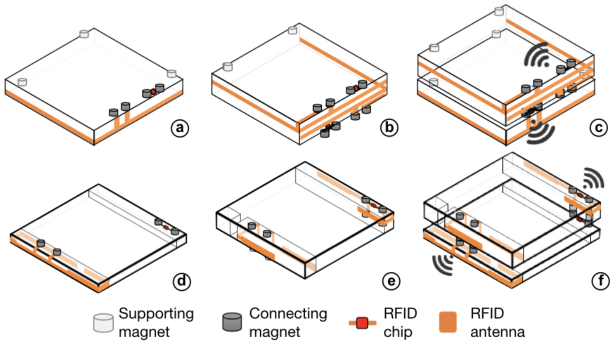

**Figure 16: Antenna design and stack sensing in two-edge and four-edge designs. (a) four-edge tile, (b) four-edge card, (c) stacking a four-edge card on a four-edge tile, (d) two-edge tile, (e) two-edge card, and (f) stacking a two-edge card on a two-edge tile. The stacking operations switch on two UHF RFID tags simultaneously.**

*Two-edge Cards and Tiles.* We also developed two-edge cards and tiles, as depicted in Figures 17a and 17c, to increase the continuity of the display when they are tiled on a screen. By optimizing the layout of connectors and reducing the size of the RFID tag antenna, we reduced the connector of the left edge. Therefore, the system can provide continuous touch bars in rows, with just a few point-size occlusions in the middle.

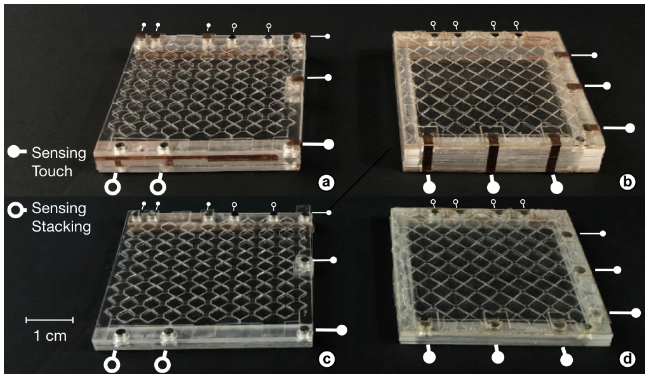

**Figure 17: A comparison of the two-edge and four-edge designs. (a) two-edge card, (b) four-edge card, (c) two-edge tile, and (d) four-edge card.**

*Integrating Capacitive Touch and RFID Stack Sensing.* The integration of the two sensing techniques was not straightforward because the performance of UHF RFID sensing was interfered with by the high density of the electric field generated by capacitive touch sensing. The dense electric field blocked the RF signal and disabled the tag tracking. Referring to previous work [14], we applied time-division multiplexing sampling on the capacitive touch sensing. Odd and even sensing lines of rows and columns were then activated sequentially, so no adjacent sensing lines were activated simultaneously. In this manner, the RF tracking operated normally.

### 4.1.2 *Interactive Surface.*

Figures 18 and 19 present the hardware design and implementation of the interactive surface; these enable touch interactions through rich-ID stackables in either vertical or horizontal settings. We attached a grid of 9 (W)×5 (H) tiles to a rear-projection screen, forming a 56.25 cm (W)×31.25 cm (H) interaction area. This display's height allows users to adapt their body posture to align their line of sight to the target when they are sitting in front of a horizontally mounted screen or standing in front of a vertically mounted screen. A RICOH PJ WX4152N short-throw projector is used for projection.

*Implementation.* An Impinj Speedway Revolution R420 UHF RFID reader and two AANT925SMA circularly polarized antennas, each one covering a 5×5 tile grid, were fixed behind the display for sensing the interaction events generated by the UHF RFID tags, which were affixed on the tiles and cards. Therefore, the entire 9×5 grid was within the signal coverage of the RF Antenna. The RFID reader's signal band was configured to be between 902 and 928 MHz, and the signal amplitude was set to 32.5 dB. Unity was used to implement the application software. RFID tag events were sent to the Unity application through WebSocket.

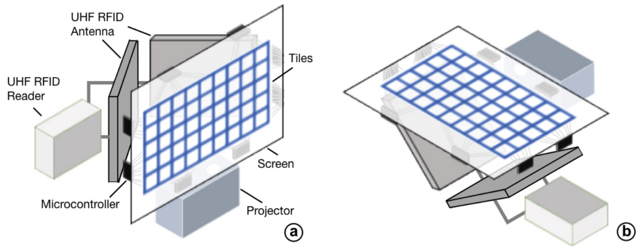

**Figure 18: Hardware design and example deployment of the interactive surface for touch interactions through rich-ID stackables. (a) Vertical setting, (b) horizontal setting.**

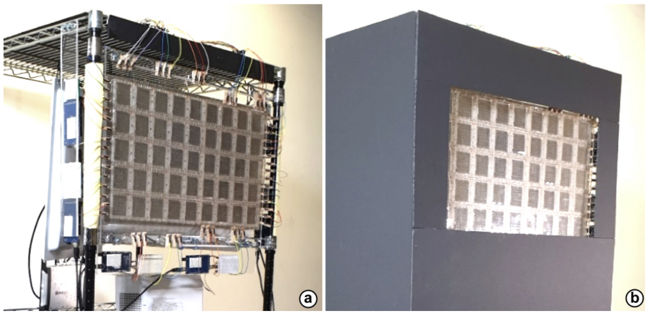

**Figure 19: Implementation of a vertical setting. (a) Overview, (b) results.**

## 4.2 Technical Evaluation

Four measurements were conducted to understand the system performance of our proof-of-concept implementation.

*4.2.1 Session 1: Stack Sensing Capability of Four-Edge Design.* The first session was focused on testing the stack sensing capabilities and the number of stacked layers that can be detected in a different area when touch sensing is operational in the four-edge design.

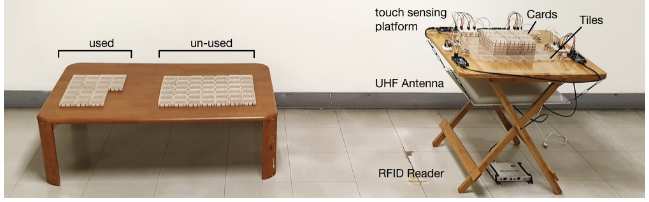

**Figure 20: Experimental apparatus of the technical evaluation of four-edge cards and tiles.**

*Apparatus.* Figure 20 is an image of the experimental apparatus, which was installed in a 3 (W)× 5 (L)×3 (H) $m^3$ empty space. During the measurement, a gridded 5×5 set of four-edge tiles served as the test platform, mounted at the center of a 32-cm wooden table in the middle of the room. An ANT925SMA circular polarized antenna was fixed vertically 32 cm under the RFID tags mounted on the station. The antenna was wire-connected to the Impinj Speedway Revolution R420 UHF RFID reader, which was placed under the antenna. The touch sensors were connected to Arduino mega boards embedded in each station for signal processing. A total of 75 four-edge cards and 25 four-edge tiles were prepared.

*Procedures.* The cards were placed in three locations: in a *used* pool for tested cards, in an *unused* pool for those not tested, or on the testing platform. These three locations were separated by 1 m. During each round of measurement, one unused card was randomly selected to be stacked on the center position, and the process was repeated until the stacking event was not detected. Thereafter, the used card was separated and moved to the used pool to avoid interference. Once the unused pool was empty, the cards in the used pool were moved to the unused pool. Three sizes of card grids, 1×1, 3×3, and 5×5, were tested. In total, 30 rounds (10 iterations×3 sizes) of measurements were conducted.

*Results.* The mean stack heights of 6.2 (SD = 0.63), 3.4 (SD = 0.52), and 2 (SD = 0) layers could be reached on a single card, 3×3 grid of cards, and a 5×5 grid of cards, respectively. The 95% CI of the results indicate that the system reliably detected up to two layers when 5×5 cards were stacked on 5×5 tiles when capacitive touch sensing was operational.

*4.2.2 Session 2: Stack Sensing Capability of Two-Edge Design.* The second session was focused on determining the stack sensing capability of stacked layers in a different area when the touch sensing was operational on the two-edge designs.

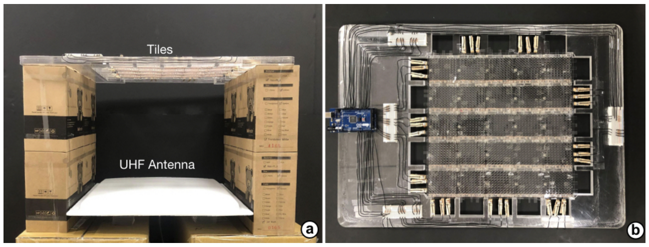

**Figure 21: Experimental apparatus in the technical evaluation of two-edge cards and tiles.**

*Apparatus.* Figure 21 depicts the apparatus, which was similar to that of session 1. A total of 75 two-edge cards and 25 two-edge tiles were prepared and applied. A gridded 5×5 set of two-edge tiles served as the test platform, and it was mounted at the center of a 32-cm wooden table in the middle of the room. The touch sensors of the rows and columns of the gridded tiles were connected to Arduino boards for signal processing.

*Procedures.* The procedure was also similar to that in session 1. Two-edge cards were selected and tested in rotation. Three sizes of two-edge card grids, 1×1, 3×3, and 5×5, were tested. In total, 30 rounds (10 iterations×3 sizes) of measurements were conducted.

*Results.* The results indicate that the cards can reach the mean stack heights of 4.9 (SD = 0.32), 3 (SD = 0), and 1.3 (SD = 0.48) layers on 1×1, 3×3, and 5×5 stack of cards, respectively. In all 10 iterations, the system reliably detected at least one layer of fully stacked 5×5 two-edge cards on 5×5 two-edge tiles when the capacitive touch sensing was operational.

### 4.2.3 *Session 3: Touch Sensing Capability.*

The third session was focused on understanding the relationship between sampling time and accuracy of capacitive touch sensing when stack sensing was operational.

*Apparatus.* The sensing apparatus was similar to the that in session 1. Because the four-edge and two-edge designs comprised the same number of electrodes, only four-edge cards were used for the test. A 5×5 tile grid was tested with 50 four-edge cards. New cards did not replace used ones after each measurement. Touch measurements were only executed after each layer was fully covered with one or more stacks. For instance, when the layer was set to $n = \{0, 1, 2\}$, $25n$ cards were stacked on the tiles.

*Procedures.* During the measurement of each touch sensing point, an example user touched each point 100 times for at least 0.4 s, as determined by a pilot test. Within the 0.4 s, the sensing results of each of the (15×15 = 225) touch sensors were collected eight times. In total, 64,800 records (100 touches×81 touch points×8 samples) of measurements were collected during the session. Accuracy was determined by the coordinate reported by the system versus the actual position of the finger.

*Results.* Figure 22 presents the results. The touch sensing attained 100% accuracy with seven samples, which required 0.35 s. The mean accuracy was >95% with two, five, and six samples in zero, one, and two layers, respectively.

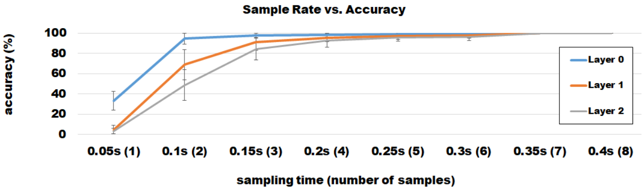

**Figure 22: Touch sensing performance by layers.**

### 4.2.4 *Session 4: Display Quality.*

The fourth session was designed to understand the transparency of the hardware when illuminated. We used a TES1330 meter as the sensing apparatus. We first displayed an entirely white frame on the projection surface. We then placed a tile at the center of the surface and attached a sensor to the center of the stack's top surface. We dimmed the projector to set the baseline illuminance to 1,043 lumens, which was within the measurement range of the apparatus and because further attenuation would thus be observable. The sensor readings of each layer were taken in a block box. The results indicate that, from 0 to 4 layers of stacking, the illuminance linearly decreased from 1,042 lumens to 190 lumens, with a mean decrease ratio of 34.62% (SD = 2.24%) per layer. We used the values to calibrate the illuminance of each layer of projection.

## 4.3 Summary

Due to the differences in RFID tag antenna design, the four-edge design provides better stackability, whereas the two-edge design provides better visual experiences. The delay in touch sensing was still noticeable, but this can be improved through further optimization. Transparency was reduced as stacked layer height increased, however, this can be mitigated by better manufacturing processes for the ITO electrodes. Overall, as a proof-of-concept system, the current system serves its purpose as a tool for developing and exploring potential applications.

# 5 DISCUSSION

In this section, we reflect on the scalability concerns exposed in our design exploration and discuss the possible generalization of our presented systems in the context of our preliminary results.

## 5.1 Rethinking Scalability

In this work, we investigated a more scalable solution that realizes the potential of the proposed cubes and cards; this solution comes in a new form of an RFID artifact that affords touchscreen interactions. Therefore, we intentionally included no electronics in the cube and card designs. The results suggest that this passive approach is viable for those who prefer not to maintain a reconfigurable system with tens or hundreds of electronic touchscreen modules.

Nonetheless, in RFIPillars, the projectors embedded in each station unit entail higher hardware costs when the size of the station matrix is increased. Using a single projector instead of a DLP-projector array can reduce such cost. However, this would also require a more sophisticated calibration and would reduce flexibility in the physical arrangement of stations.

For small-scale deployment (e.g., with fewer than 10 cubes used in the system), embedding power electronics in the cubes remains an economical option. For instance, the lenses, mirrors, and projectors in the RFIPillars can be replaced with capacitive touchscreens mounted on every cube, as in the configuration previously investigated by Foxels [34]. For stacking, each block's ID can serve as the $I^2C$ address of each cube, and the station can power the entire stack of cubes. This alternative solution can further improve the quality of touchscreen interactions after stacking. Also, deploying more than one display on each face of cubes is possible because the routing of optical projection is no longer a design constraint. Future developers can consider the costs and the scale of deployment for their applications.

## 5.2 Generalization and Preliminary Results

*Higher Resolution Touch Inputs.* The spatial resolution of a touch input can be increased with a more sophisticated algorithm design or a denser matrix of touch sensing electrodes. Our current implementation achieved continuous finger tracking with a bicubic interpolation, but the current size of the electrode results in slightly jittery performance. Therefore, we report the sampling time–accuracy relationship on segmented positions instead and leave further touchscreen optimizations for future work. Regarding electrode density, Figure 23 presents the 9×9 touch sensing pad that we implemented to demonstrate the feasibility of our system. Although three-times finer-grained touch inputs are realized, RF sensing performances are also affected because the electric field is also three times denser. Therefore, time-multiplexed sampling at a higher precision is required to maintain the responsiveness of the touch sensing. The transparency on the frame is also decreased due to the increased number of connectors.

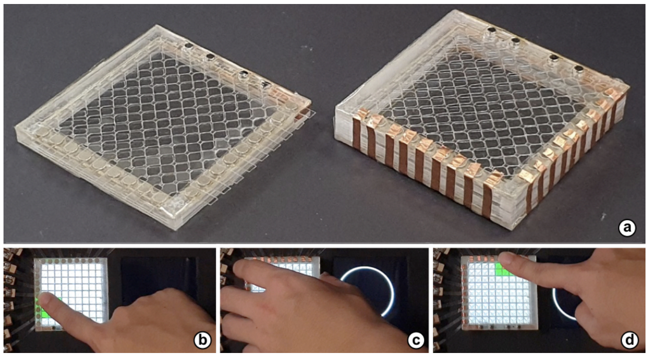

**Figure 23: (a) Tile and card that support 9×9 touch inputs. (b) Touch before stacking, (c) stacking, (d) touch after stacking.**

*Multi-Touch and Advanced Touch Sensing.* Although the Arduino platform allows for easy prototyping, implementing multi-touch functionality was not straightforward. Recent solutions such as the use of a multi-touch kit [36] do not yet support ITO electrodes because of their high impedance. Fortunately, emerging mutual-capacitance multi-touch sensing hardware such as the MuCa [41] board is user-friendly for prototyping. Therefore, implementing multi-touch inputs in our platform will be easier in future work.

Beyond rudimentary touch inputs (e.g., touches and swipes), the self-capacitance touch sensing matrix on RFITiles and RFIPillars units support more types of touch input as shown in Figure 24, such as multi-finger selection (from the bounding box of two fingers), palm and fist inputs (represented in terms of the size of the touch), ulnar border inputs (represented in terms of the bounding box aspect ratio), and dynamic gestures (e.g., swipes). The form factors of RFITiles and RFIPillars also allow more touch gestures to be performed on them.

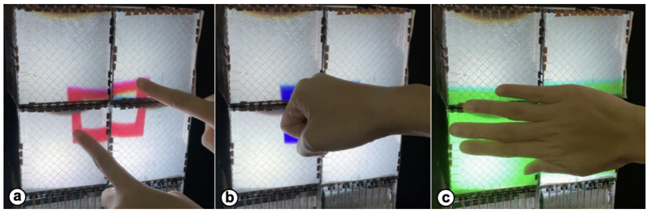

**Figure 24: Advanced touch gestures supported by the self-capacitance touch sensing matrix: (a) multi-finger selection, (b) fist, and (c) palm.**

*Better Visual Quality After Stacking.* In both systems, visual display quality decreases as more layers are stacked. With regard to RFITiles, one plausible solution to making cards and tiles more transparent is using touch sensors that have not been cut through. Figure 25 depicts an implementation similar to ours but is processed by a laser engraving machine that removed conductive material from the touch sensing surface, which achieves high transparency even after stacking. Figure 26 provides an example of using this sheet for both touch and stacking sensing. Although producing this sensor is not cheap, the cost could be reduced by mass production.

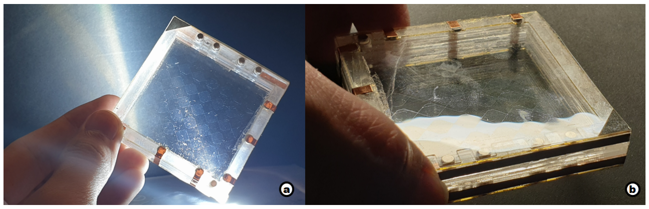

**Figure 25: Improving the transparency of touch electrodes through laser engraving. (a) Results. (b) A stack of Cards.**

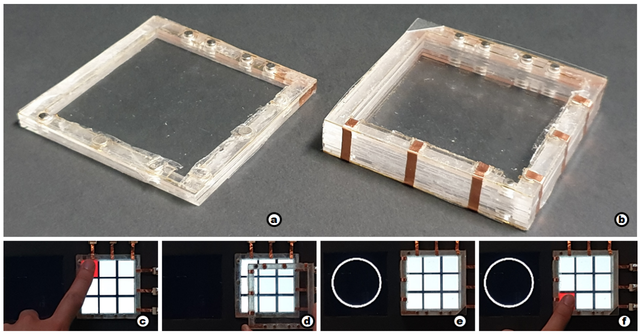

**Figure 26: (a) Tile and (b) Card that are made of the laser-engraved transparent touch sensor. (c) Touch before stacking. (d) Touch after stacking.**

Regarding RFIPillars, the problem lies mainly in distortion and defocusing along the light propagation path. Using laser projectors with a higher quality lens and mirror can mitigate these two problems, but resolution may be greatly decreased. Future work may consider using optic fibers as an alternative to a lens and a mirror to increase the spatial resolution by better utilizing the unused volume of cubes for non-line-of-sight light propagation. The optic fibers can be printed [48] and/or installed with jigs and supports for standardization. Nonetheless, 3D printing optic fibers of a desirable quality using such an additive manufacturing method may increase the cost and complexity of cube manufacturing.

*Opaque Components.* Opaque components were also noted to downgrade the user experience. With RFIPillars, copper tapes are visible from the front surface of the cube; this can be mitigated through more careful finishing. However, for RFITiles, reducing the visibility of the magnets and RFID on each two-edge tile and card is not trivial. With this limitation, the current size of tiles and cards is more suitable for grid-based GUI applications such as calendars and board games. Enlarging the frame to the size of a pad would support Windows-based applications, as with a standard tablet PC but doing so would also decrease stacking resolutions and would require two-handed manipulation.

*Rotation Operation.* Rotation operations can be added to both RFITiles and RFIPillars systems. Regarding RFITiles, by adding one more pair comprising an RFID contact switch and contact points to one more side to the card, 180° rotation of the card can be realized, (Figure 27). The different IDs can recognize the card's orientation, and the touch sensing works in both directions. By enabling the sensing card's orientation, a tangible token can express a greater range of representations, such as different interface languages. As with the case of increasing resolutions, the frame's transparency would also be decreased due to the increased number of connectors.

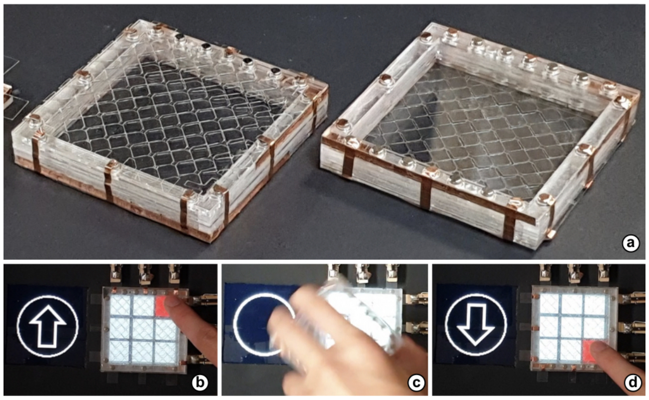

**Figure 27: (a) Tile and card that support 180-degree rotation. (b) Touch before rotation, (c) rotating the card, and (d) touch after rotation**

Regarding RFIPillars, block rotation can be added by adapting the four-sided pattern designs in RFIBricks [15]. As Figure 28 suggests, a four-sided, partial display is possible with a more sophisticated optical routing design, which includes four convex mirrors, one concave lens, and four pairs of RFID contact switches. Although the display of each block does not connect in the stack, this system can support interactive gaming experiences, such as rotating a cube in pattern matching. A larger playground can be created by tiling the station in 2D. More sides for touch sensing can be added by increasing the density of the capacitive circuitry (see Figure 10).

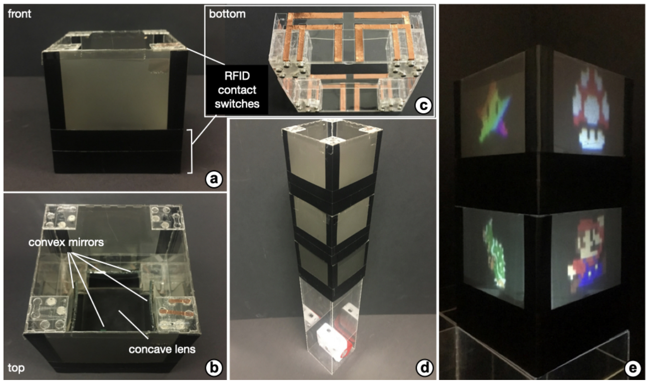

**Figure 28: Four-sided display cube that supports rotation operations. (a–c) overview, (d) 3-layer stack of cubes on a station, and (e) result in the form of a pattern matching game.**

*Visual Affordance of Transparent Cards.* The transparent card must provide effective visual affordance [7] to avoid confusion during use. Figure 29 provides an example of a minimally designed visual cue that does not occlude or distract from the central area of the user's interest. A mark is added at the upper left corner of the top surface for the user to identify the card's orientation. Graphical labels are added on the sides of the nearly transparent tangibles, which users can glimpse and thus recall their identity.

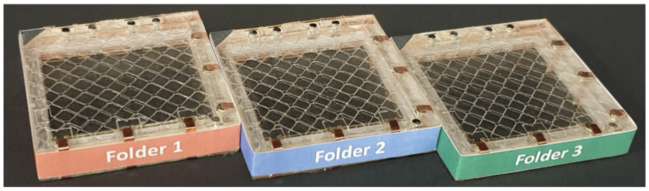

**Figure 29: Visual cue for using the transparent cards.**

*Preliminary User Experiences.* This paper provides a technical HCI contribution in its design and engineering, not unlike Lumino [3], and proof-of-concept implementation and technical evaluation are considered viable forms of validation [16]. Nonetheless, we informally tested the two prototypes systems with participants in informal sessions after we developed the applications shown in Figures 3, 4, 6, and 7. All the participants were able to complete the application tasks, which indicates that the systems are robust enough to deliver such experiences. In these applications, participants were able to compare the pros (e.g., rich tangibility and an engaging nature) and cons (e.g., the need to acquire tokens before use) of the experience with their previous touchscreen experiences with the applications (e.g., the Plants vs. Zombie game). We also received feedback from the participants. Some mentioned the noticeable latency in touch sensing, the reduced touch precision on a 3-layer stack of cards in RFITiles, and the problem of fatigue experienced in vertical settings after an extended period of use. However, most of these usability issues are known ergonomic problems in touchscreen HCI and can be addressed with an advanced implementation (e.g., a more responsive, tilted touchscreen that better aligns with the user's perspective), and we can leave future work to conduct usability studies.

## 6 CONCLUSION

We presented a novel approach to combining touchscreens with rich-ID building block systems to support the physical construction of contexts in touchscreen interaction. To fully exploit the scalability of passive rich-ID systems, we extended the designs of previous systems [14, 15] by integrating position sensing, touch sensing, and visual feedback in the rich-ID blocks, turning them into modular touchscreens and thus making them passive. Therefore, users can physically construct the context of touchscreen interaction by stacking rich-ID touchscreen cubes or by incrementally updating the touch input modes through stacking rich-ID transparent tiles that allow for touch input through the stack. We constructed two proof-of-concept systems based on the design philosophy and demonstrated and evaluated rudimentary inputs (stacking and touch) to the systems. The results provide preliminary yet concrete evidence of their good performance in operation. Furthermore, the knowledge generated in the deliberate design, implementation, and technical evaluation elucidate the scalability issues in the solution space of reconfigurable touchscreen systems.

We also discuss plausible research directions with our initial attempts to overcome current limitations to help future researchers scrutinize the design space.

## ACKNOWLEDGMENTS

We sincerely acknowledge Associate Chairs and the anonymous reviewers for the helpful comments. This research was supported in part by the Ministry of Science and Technology of Taiwan (MOST109-2634-F-002-032, 109-2218-E-002-026, 109-2218-E-011-011), and National Taiwan University.

## REFERENCES

[1] Masahiro Ando, Yuichi Itoh, Toshiki Hosoi, Kazuki Takashima, Kosuke Nakajima, and Yoshifumi Kitamura. 2014. StackBlock: Block-shaped Interface for Flexible Stacking. In *Proceedings of the Adjunct Publication of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST'14 Adjunct)*. ACM, New York, NY, USA, 41–42. https://doi.org/10.1145/2658779.2659104

[2] Tom Bartindale and Chris Harrison. 2009. Stacks on the Surface: Resolving Physical Order Using Fiducial Markers with Structured Transparency. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces* (Banff, Alberta, Canada) *(ITS '09)*. ACM, New York, NY, USA, 57–60. https://doi.org/10.1145/1731903.1731916

[3] Patrick Baudisch, Torsten Becker, and Frederik Rudeck. 2010. Lumino: Tangible Blocks for Tabletop Computers Based on Glass Fiber Bundles. In *ACM CHI 2010 Conference Proceedings* (Atlanta, Georgia, USA). 1165–1174.

[4] Wolfgang Büschel, Ulrike Kister, Mathias Frisch, and Raimund Dachselt. 2014. T4-transparent and translucent tangibles on tabletops. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*. ACM, 81–88.

[5] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. 2012. CapStones and ZebraWidgets: Sensing Stacks of Building Blocks, Dials and Sliders on Capacitive Touch Screens. In *ACM CHI 2012 Conference Proceedings* (Austin, Texas, USA). 2189–2192.

[6] Paul Dietz and Darren Leigh. 2001. DiamondTouch: A Multi-user Touch Technology. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology* (Orlando, Florida) *(UIST '01)*. ACM, New York, NY, USA, 219–226. https://doi.org/10.1145/502348.502389

[7] James J Gibson. [n.d.]. The theory of affordances. ([n. d.]).

[8] Audrey Girouard, Aneesh Tarun, and Roel Vertegaal. 2012. DisplayStacks: Interaction Techniques for Stacks of Flexible Thin-film Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. ACM, New York, NY, USA, 2431–2440. https://doi.org/10.1145/2207676.2208406

[9] Alix Goguey, Cameron Steer, Andrés Lucero, Laurence Nigay, Deepak Ranjan Sahoo, Céline Coutrix, Anne Roudaut, Sriram Subramanian, Yutaka Tokuda, Timothy Neate, Jennifer Pearson, Simon Robinson, and Matt Jones. 2019. PickCells: A Physically Reconfigurable Cell-Composed Touchscreen. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3290605.3300503

[10] Jefferson Y. Han. 2005. Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology* (Seattle, WA, USA) *(UIST '05)*. Association for Computing Machinery, New York, NY, USA, 115–118. https://doi.org/10.1145/1095034.1095054

[11] Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. 2011. OmniTouch: Wearable Multitouch Interaction Everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. ACM, New York, NY, USA, 441–450. https://doi.org/10.1145/2047196.2047255

[12] Florian Heller, Simon Voelker, Chat Wacharamanotham, and Jan Borchers. 2015. Transporters: Vision & Touch Transitive Widgets for Capacitive Screens. In *CHI '15: Extended Abstracts of the SIGCHI Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea). 1603–1608. https://doi.org/10.1145/2702613.2732707

[13] Toshiki Hosoi, Kazuki Takashima, Tomoaki Adachi, Yuichi Itoh, and Yoshifumi Kitamura. 2014. A-blocks: Recognizing and Assessing Child Building Processes During Play with Toy Blocks. In *SIGGRAPH Asia 2014 Emerging Technologies* (Shenzhen, China) *(SA '14)*. ACM, New York, NY, USA, Article 1, 2 pages. https://doi.org/10.1145/2669047.2669061

[14] Meng-Ju Hsieh, Rong-Hao Liang, Jr-Ling Guo, and Bing-Yu Chen. 2018. RFIDesk: An Interactive Surface for Multi-Touch and Rich-ID Stackable Tangible Interactions. In *SIGGRAPH Asia 2018 Emerging Technologies* (Tokyo, Japan) *(SA '18)*. Association for Computing Machinery, New York, NY, USA, Article 11, 2 pages. https://doi.org/10.1145/3275476.3275491

[15] Meng-Ju Hsieh, Rong-Hao Liang, Da-Yuan Huang, Jheng-You Ke, and Bing-Yu Chen. 2018. RFIBricks: Interactive Building Blocks Based on RFID. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3173574.3173763

[16] Scott E Hudson and Jennifer Mankoff. 2014. Concepts, values, and methods for technical human–computer interaction research. In *Ways of Knowing in HCI*. Springer, 69–93.

[17] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '97)*. ACM, New York, NY, USA, 234–241. https://doi.org/10.1145/258549.258715

[18] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, and Andrew Fitzgibbon. 2011. KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. ACM, New York, NY, USA, 559–568. https://doi.org/10.1145/2047196.2047270

[19] Robert J.K. Jacob, Audrey Girouard, Leanne M. Hirshfield, Michael S. Horn, Orit Shaer, Erin Treacy Solovey, and Jamie Zigelbaum. 2008. Reality-Based Interaction: A Framework for Post-WIMP Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 201–210. https://doi.org/10.1145/1357054.1357089

[20] Kunihiro Kato and Homei Miyashita. 2015. ExtensionSticker: A Proposal for a Striped Pattern Sticker to Extend Touch Interfaces and Its Assessment. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. ACM, New York, NY, USA, 1851–1854. https://doi.org/10.1145/2702123.2702500

[21] Yoshifumi Kitamura, Yuichi Itoh, and Fumio Kishino. 2001. Real-Time 3D Interaction with ActiveCube. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems* (Seattle, Washington) *(CHI EA '01)*. Association for Computing Machinery, New York, NY, USA, 355–356. https://doi.org/10.1145/634067.634277

[22] Hideki Koike, Wataru Nishikawa, and Kentaro Fukuchi. 2009. Transparent 2-D markers on an LCD tabletop system. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 163–172.

[23] Jinha Lee, Yasuaki Kakehi, and Takeshi Naemura. 2009. Bloxels: Glowing Blocks as Volumetric Pixels. In *ACM SIGGRAPH 2009 Emerging Technologies* (New Orleans, Louisiana) *(SIGGRAPH '09)*. Association for Computing Machinery, New York, NY, USA, Article 5, 1 pages. https://doi.org/10.1145/1597956.1597961

[24] Kevin Lefeuvre, Soeren Totzauer, Michael Storz, Albrecht Kurze, Andreas Bischof, and Arne Berger. 2018. Bricks, Blocks, Boxes, Cubes, and Dice: On the Role of Cubic Shapes for the Design of Tangible Interactive Devices. In *Proceedings of the 2018 Designing Interactive Systems Conference*. ACM, 485–496.

[25] Jakob Leitner and Michael Haller. 2011. Geckos: Combining Magnets and Pressure Images to Enable New Tangible-object Design and Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. ACM, New York, NY, USA, 2985–2994. https://doi.org/10.1145/1978942.1979385

[26] Joanne Leong, Florian Perteneder, Hans-Christian Jetter, and Michael Haller. 2017. What a Life! Building a Framework for Constructive Assemblies. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction* (Yokohama, Japan) *(TEI '17)*. Association for Computing Machinery, New York, NY, USA, 57–66. https://doi.org/10.1145/3024969.3024985

[27] Rong-Hao Liang, Liwei Chan, Hung-Yu Tseng, Han-Chih Kuo, Da-Yuan Huang, De-Nian Yang, and Bing-Yu Chen. 2014. GaussBricks: Magnetic Building Blocks for Constructive Tangible Interactions on Portable Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 3153–3162. https://doi.org/10.1145/2556288.2557105

[28] Rong-Hao Liang, Han-Chih Kuo, Liwei Chan, De-Nian Yang, and Bing-Yu Chen. 2014. GaussStones: Shielded Magnetic Tangibles for Multi-Token Interactions on Portable Displays. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 365–372. https://doi.org/10.1145/2642918.2647384

[29] Thomas W. Malone. 1983. How Do People Organize Their Desks?: Implications for the Design of Office Information Systems. *ACM Trans. Inf. Syst.* 1, 1 (Jan. 1983), 99–112. https://doi.org/10.1145/357423.357430

[30] Nobuyuki Matsushita and Jun Rekimoto. 1997. HoloWall: Designing a Finger, Hand, Body, and Object Sensitive Wall. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology* (Banff, Alberta, Canada) *(UIST '97)*. Association for Computing Machinery, New York, NY, USA, 209–210. https://doi.org/10.1145/263407.263549

[31] David Merrill, Jeevan Kalanithi, and Pattie Maes. 2007. Siftables: Towards Sensor Network User Interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction* (Baton Rouge, Louisiana) *(TEI '07)*. Association for Computing Machinery, New York, NY, USA, 75–78. https://doi.org/10.1145/1226969.1226984

[32] David Merrill, Emily Sun, and Jeevan Kalanithi. 2012. Sifteo Cubes. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI EA '12)*. Association for Computing Machinery, New York, NY, USA, 1015–1018. https://doi.org/10.1145/2212776.2212374

[33] Rafael Morales González, Caroline Appert, Gilles Bailly, and Emmanuel Pietriga. 2016. Touchtokens: guiding touch patterns with passive tokens. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 4189–4202.

[34] Florian Perteneder, Kathrin Probst, Joanne Leong, Sebastian Gassler, Christian Rendl, Patrick Parzer, Katharina Fluch, Sophie Gahleitner, Sean Follmer, Hideki Koike, and et al. 2020. Foxels: Build Your Own Smart Furniture. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Sydney NSW, Australia) *(TEI '20)*. Association for Computing Machinery, New York, NY, USA, 111–122. https://doi.org/10.1145/3374920.3374935

[35] M. Philipose, J. R. Smith, B. Jiang, A. Mamishev, Sumit Roy, and K. Sundara-Rajan. 2005. Battery-free wireless identification and sensing. *IEEE Pervasive Computing* 4, 1 (Jan 2005), 37–45. https://doi.org/10.1109/MPRV.2005.7

[36] Narjes Pourjafarian, Anusha Withana, Joseph A. Paradiso, and Jürgen Steimle. 2019. Multi-Touch Kit: A Do-It-Yourself Technique for Capacitive Multi-Touch Sensing Using a Commodity Microcontroller. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 1071–1083. https://doi.org/10.1145/3332165.3347895

[37] Jun Rekimoto. 2002. SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 113–120.

[38] Jun Rekimoto, Brygg Ullmer, and Haruo Oba. 2001. DataTiles: A Modular Platform for Mixed Physical and Graphical Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Seattle, Washington, USA) *(CHI '01)*. ACM, New York, NY, USA, 269–276. https://doi.org/10.1145/365024.365115

[39] Toshiki Sato, Haruko Mamiya, Hideki Koike, and Kentaro Fukuchi. 2009. PhotoelasticTouch: Transparent Rubbery Tangible Interface Using an LCD and Photoelasticity. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology* (Victoria, BC, Canada) *(UIST '09)*. ACM, New York, NY, USA, 43–50. https://doi.org/10.1145/1622176.1622185

[40] Martin Schmitz, Mohammadreza Khalilbeigi, Matthias Balwierz, Roman Lissermann, Max Mühlhäuser, and Jürgen Steimle. 2015. Capricate: A Fabrication Pipeline to Design and 3D Print Capacitive Touch Sensors for Interactive Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software &#38; Technology* (Charlotte, NC, USA) *(UIST '15)*. ACM, New York, NY, USA, 253–258. https://doi.org/10.1145/2807442.2807503

[41] Marc Teyssier, Gilles Bailly, Catherine Pelachaud, Eric Lecolinet, Andrew Conn, and Anne Roudaut. 2019. Skin-On Interfaces: A Bio-Driven Approach for Artificial Skin Design to Cover Interactive Devices. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 307–322. https://doi.org/10.1145/3332165.3347943

[42] Brygg Ullmer and Hiroshi Ishii. 1997. The metaDESK: Models and Prototypes for Tangible User Interfaces. In *ACM UIST 1997 Conference Proceedings* (Banff, Alberta, Canada). 223–232.

[43] Brygg Ullmer, Hiroshi Ishii, and Robert JK Jacob. 2005. Token+ constraint systems for tangible interaction with digital information. *ACM Transactions on Computer-Human Interaction (TOCHI)* 12, 1 (2005), 81–118.

[44] Mark Weiser. 1991. The Computer for the 21 st Century. *Scientific american* 265, 3 (1991), 94–105.

[45] Malte Weiss, Florian Schwarz, Simon Jakubowski, and Jan Borchers. 2010. Madgets: Actuating Widgets on Interactive Tabletops. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology* (New York, New York, USA) *(UIST '10)*. ACM, New York, NY, USA, 293–302. https://doi.org/10.1145/1866029.1866075

[46] Malte Weiss, Julie Wagner, Yvonne Jansen, Roger Jennings, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. 2009. SLAP Widgets: Bridging the Gap Between Virtual and Physical Controls on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. ACM, New York, NY, USA, 481–490. https://doi.org/10.1145/1518701.1518779

[47] Cary Williams, Xing Dong Yang, Grant Partridge, Joshua Millar-Usiskin, Arkady Major, and Pourang Irani. 2011. TZee: Exploiting the Lighting Properties of Multi-touch Tabletops for Tangible 3D Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. ACM, New York, NY, USA, 1363–1372. https://doi.org/10.1145/1978942.1979143

[48] Karl Willis, Eric Brockmeyer, Scott Hudson, and Ivan Poupyrev. 2012. Printed Optics: 3D Printing of Embedded Optical Elements for Interactive Devices. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. ACM, New York, NY, USA, 589–598. https://doi.org/10.1145/2380116.2380190

[49] Andrew D. Wilson. 2005. PlayAnywhere: A Compact Interactive Tabletop Projection-vision System. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology* (Seattle, WA, USA) *(UIST '05)*. ACM, New York, NY, USA, 83–92. https://doi.org/10.1145/1095034.1095047

[50] Robert Xiao, Scott Hudson, and Chris Harrison. 2017. Supporting responsive cohabitation between virtual interfaces and physical objects on everyday surfaces. *Proceedings of the ACM on Human-Computer Interaction* 1, EICS (2017), 12.

[51] Kentaro Yasu. 2019. Magnetact: Magnetic-sheet-based Haptic Interfaces for Touch Devices. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, Article 240, 8 pages. https://doi.org/10.1145/3290605.3300470

[52] John Zimmerman, Jodi Forlizzi, and Shelley Evenson. 2007. Research through Design as a Method for Interaction Design Research in HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '07)*. Association for Computing Machinery, New York, NY, USA, 493–502. https://doi.org/10.1145/1240624.1240704