Cubio: A Low-Budget Platform for Exploring Stackable Interactions

Marc Teyssier
MIT Media Lab
75 Amherst Ave.
marcteys@media.mit.edu

Lucas Silva
MIT Media Lab
75 Amherst Ave.
cassiano@media.mit.edu

Pattie Maes
MIT Media Lab
75 Amherst Ave.
pattie@media.mit.edu

Ira Winder
MIT Media Lab
75 Amherst Ave.
jiw@media.mit.edu

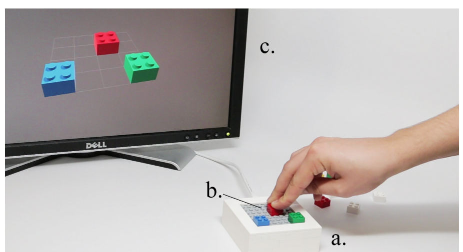

Figure 1: Prototype of Cubio, including the base (a), the modified cubes with resistors (b) and a digital representation(c)

Abstract
In this paper we introduce Cubio, a low-budget platform for LEGO bricks to enable sensing of stacking levels. Our approach consists of modifying existing LEGO blocks by adding thin layers of copper and a single resistor. By measuring the overall resistance at each point of a 8x8 Lego base plate we are able to sense the bricks location and stacking level. This creates a one to one mapping of the physical bricks with digital data allowing to explore and control virtual content using a tangible input. We present the implementation of Cubio, explaining our design and development process step by step. To explore the design space, we implemented three example applications showing the capabilities of Cubio in different contexts such as 3D modelling, urban planning and interaction with smart appliances.

Author Keywords
Stackable Interaction, Low-budget; Tangible

ACM Classification Keywords
H.5.2 [User Interfaces]: Miscellaneous.

Introduction
In recent years, tangible user interfaces (TUIs) and sensing-based interaction are used to expand computational systems inputs. [5] These new interfaces allow for the devel-

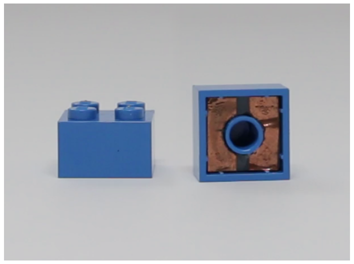

Figure 2: Our modified LEGO brick, view from left and bottom.

opment of new interaction methods based on physical manipulations, thereby engaging the user with computers. Further, tangible interfaces can enhance learning when the design of the application is coherent and fits to userâĂŹs context.[9]

In this paper, we present a low-budget platform designed to interface LEGO building blocks with virtual applications. The intention was to keep the user experience where children have building blocks to express their creativity and begin to to play with ideas from the field of organisation and architecture. Simultaneously, we allowed all the freedom and interaction level of virtual environments. An important factor is that LEGO bricks have explicit design affordance, they can be stackable constrained by rules provided by the design of the brick and also be arranged to create a wide variety of solid structures. Moreover, the original LEGO block allows a structural rigidity and geometric constraints that are fundamental for the electronic connections reliability. This allowed for having the development process that is cheap, fast and easy to reproduce.

Tangible interface experiences usually involve new devices and complex hardware solutions. In Cubio, we used common objects and digitally augment their capabilities while keeping their own affordances: the user donâĂŹt have to learn a new interactive language but can use his/hers knowledge and extend it with virtual applications. The affordances provided by the shape of the block nudges the user to stack them. To keep this affordances for the user in each brick we added a resistor which make them recognizable when stacked *(Figure 2)*. By detecting blocks as unique objects, the user is allowed to create structures with different conceptual properties, such as specific block colors possessing different resistors values which are associated to specific meanings for both user and system.

By offering a low-budget and easy to do approach, our projects contribute to spread tangible interactions through common LEGO bricks. This paper explain the development process for both hardware and software[1]. We also explored the design space of this paradigm by presenting example applications and scenarios.

Related Work

Our work is closely related to the field of Tangible Interfaces and stackable interfaces. An early approach in stackable interfaces is presented by David Anderson in [2]. In their implementation, each block embed a small computer and has to calculate his relative position. Later, *ActiveCube* by *Kitamura et Al*[7] explores the same capabilities and added a 3rd axis to the blocks. Thus this research allow a multimodal interaction, it requires a specific hardware. Rather than creating a specific device to develop tangible interaction, Cubio focuses on augmenting existing object whose affordances are known.

Existing work revealed a lot of cubic tangible interfaces. *CapStones and ZebraWidgets* [3] uses an iPad screen as input for stackable objects, using capacitive components. This system covers part of the screen and is too dependent of the screen input resolution. By choosing among multiple grid size, in our work we have more control over the grid resolution and specifications. In [11] Wolf et al. also explored a DIY low-budget approach to build tangible interfaces. However, their solution focuses more on combining paper TUIs with capacitive touchscreens.

A very similar approach of using LEGO bricks is used in Cityscope[1]. Augmented reality simulation uses computer vision to detect pre-made stacks of bricks disposed on a

[1] Detailed instructions can be found at http://github.com/marcteys/cubio

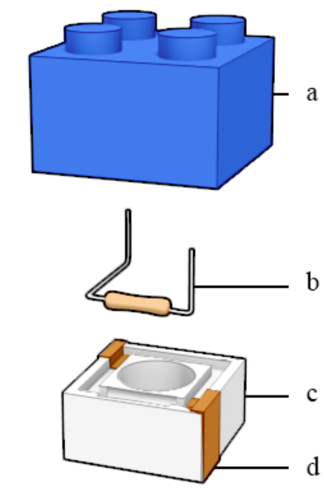

Figure 3: Original LEGO brick (a) with our addon composed of a resistor(b), the 3d printed part(c) and the copper connectors(d).

grid. Computer vision system requires specific lighting conditions and thus are not reliable in day to day use, it also requires pre-set camera angles disposed on bigger structures than our grid.This work also uses projection mapping, that can be a solution for visual output for Cubio.

Some commercial LEGO products explores the mix of real and virtual content. The LEGO Fusion[6] set and application uses augmented reality and computer vision to duplicate a front of a built house in an iPad screen. LEGO Dimension[4] uses RFID tag inside a specially designed block to detect if itâĂŹs present or not on a specific device. This is however not compatible with other LEGO bricks. To our knowledge, Cubio is the first to combine low-budget sensing capabilities with LEGO bricks to virtualize physical structures.

Implementation

Our concept consists of a squared LEGO base with a built-in Arduino combined with modified LEGO blocks with resistors inside, to enable sensing reading of stacked bricks *(Figure 4)*.

Design

To develop our prototype, we focused on the most common 2x2 LEGO blocks. Since our original intention was to keep the LEGO block unmodified, we used a combination of 3D printed piece and cheap hardware component *(Figure 3)*. We designed a piece that had to be 3d printable and fit inside a LEGO block. In order to make the platform able to identify the quantity of stacked cubes on the vertical axis, each cube was designed to have a resistor inside it and metallic copper connectors who pass through the cube vertically. Two holes are drilled through the top of each LEGO brick to close the circuit. Provided that, when stacked on top of each other, the cubes represent, on the circuit, resistors in parallel: by reading the equivalent resistance of the resistors in parallel it is possible to infer the number of cubes stacked vertically.

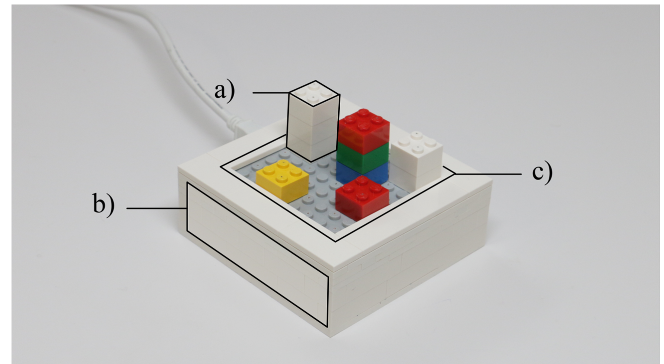

Figure 4: Cubio hardware containing the Arduino card(b), with some LEGO bricks(a) disposed on the building area(c).

Hardware

The resistance value in each cube is a key factor on this project, due to the fact that if it is too high the variation on the circuitâĂŹs voltage would be very high and voltage would be too small to read;*(Figure 5)* In other hand if the resistance is too small the system would have problems to read the small variation of voltage caused by the small equivalent resistance on this case. After few trials with different resistors values we selected the range of values that are applicable with the developed circuit, thus the current prototype uses resistors on order of 10$\hat{3}$ ohm and it is able to read up to 25 cubes, vertically. The resistors also defines the blockâĂŹs IDs, different resistances means different IDs.

The method used by this system to track the cubeâĂŹs

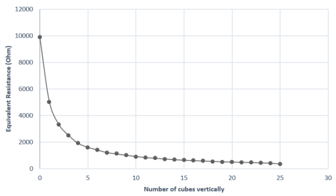

Figure 5: Real resistance variation on the circuit with 10K ohm resistors.

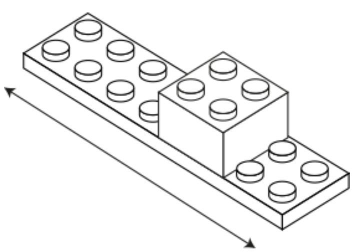

Figure 6: Moving the cube on a specific axis allows the user to create a custom slider input.

position on the XY plane the system uses a mesh of voltage and ground lines, having positive and negative poles for each point on the Grid perpendicularly positioned.

This way the system reads one position per time, activating both connectors related to this position and putting the others on a high-impedance state. The system also differentiates when a tower is built then placed on the building area. To read one position per time increases negatively the reading time of the entire grid, on the other hand this allows the system to pass all current through the single current position on grid, thus making it possible to have a higher vertical reading range.

We based our grid on an Arduino because it's low-budget, easily accessible, widely spread in the DIY community. The reading of the stacked cubes equivalent resistance on a given position is made by reading a microcontrollerâĂŹs analog pin connected to the output signal from the voltage changed by the cubes resistances. We have build a 4x4 grid which can handle up to 400 unique blocks, 25 blocks on each grid position. This hardware is scalable up to a 16x16 grid or even bigger by using multiplexers. The current grid system allow us to read up to 5 different IDs, however on our first prototype the multiple IDs interactions work only with the first layer.

In combination to the hardware, we developed a bridge to connect the platform with the computer and with our applications. The Arduino board is connected via serial communication to our C# API. We are reading an array of values, representing the equivalent resistance on each gridâĂŹs position, which give us the position and the ID of each brick.

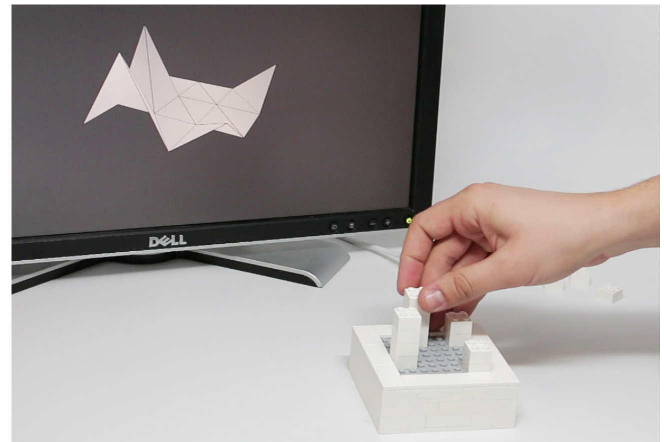

Figure 7: User interacting with the surface modelling application. The surface on the screen result of the position of the blocks.

Software

Interaction

In Cubio, the physical blocks can be rapidly switch from a off-grid environment to an online environment. A user could for example, build a structure behind the scene and then connect it to the grid. This flexibility in representation and transitioning between modalities enables new interaction capabilities, swapping from LEGO's regular constructions to applicative metaphor.

Using an external platform allow us to not obstruct or overlay existing screens with physical objects: Cubio can therefore be used as an additional input device, allowing the user to expand screen capabilities and mix interactions between physical and virtual world.

We used transposed metaphor and physical affordances to expand interaction capabilities. Users can create physi-

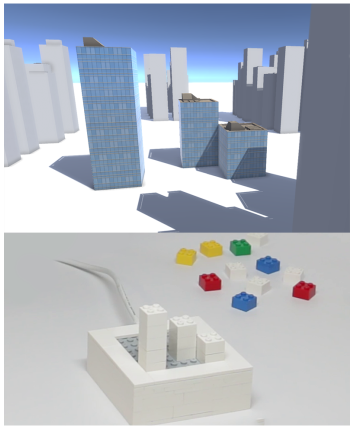

Figure 8: City building application, with the screen-shot of the real-time view (top) and what the user has build (bottom).

Figure 9: The purple color is set by two physical sliders, red and blue.

cal controls which are used to control virtual inputs.For example, building user interfaces elements such as buttons, handles or sliders can be rendered graphically on the application*(Figure 6)* . By using bricks with different aspect (shape or color), we can detect cubes as an individual entity and expand user's creative possibilities: a red or blue bricks can have different meanings and be used to trigger different actions.

Applications

Surface modelling

Some tangible interfaces are trying to solve the challenging problem of 3D surface modeling on a 2D screen [8]. In this application *(Figure 7)*, we combine object stacking and NURBS (Non-Uniform Rational Basis Spline) surfaces to explore 3D modelling. Connecting and moving block stacks affects the virtual surface. (figure) Using a 3D object-based input rather that a 2D mouse input help the user to understand and visualize shapes in addition to provide an efficient manipulation of 3D content [10] thanks to haptic feedbacks and more degree of freedom through brick control.

City building

Differently stacked, few blocks can create a various range of structures. This modality is used on this application to create sections of cities. By mapping a LEGO brick to a floor of a building, these physical objects are now building with specific location and height. Different piece are used to represent different kind of buildings. This application *(Figure 8)* can be used for city planning : by adding buildings users can explore in real-time how the shape and position of buildings affects the environment.Physical elements allow performing more quickly a specific task and explore in a playful way serious topics. This concept can be extend with projection mapping as output for city-based data visualisation and manipulation.

User defined tangible inputs

In this example, we explored how Cubio can be used with internet-connected objects. *Philips Hue* Lamps don't have physical switches and rely only on digital inputs on smartphones. By defining 4 sliders on the grid, we can individually change the red, green, blue and alpha channels to set the color of the ambient light *(Figure 9)*.Creating tangibles elements to materialize virtual functions allow the user to modify and appropriate his object. With our platform we can easily create physical sliders or buttons. There are simple to develop and scalable. The color of the bricks are used to strengthen specific parameters. This example explore transposition mapping of objects representation to personify real physical space.

Limitation and future work

In our current implementation the copper connectors are not rigid enough to maintain a reliable electrical connection. This issue is going to be fixed in the next implementation by redesigning the 3d printed addon and connect a spring system. In addition to the upgraded hardware, we will more focus on applications, including more educational experiences for children, and augmenting Cubio with AR thus allowing a multi-users remote experience. To solve the multi-id reading problem we will implement a voltage divider inside each block, thus change the total resistance depending of its vertical position.

Conclusion

In this work we introduce Cubio, a low-budget modification of LEGO bricks to enable sensing capabilities. We presented the concept and implementation of Cubio explaining the full design process and offering the detailed building instruction as open source for the DYI community We implemented three example applications showing how to use Cubio for scenarios such as 3D modelling, urban planning and interaction with smart appliances. By offering Cubio as an open platform we encourage researchers and the DIY community to use it as a quick prototyping platform for tangible user interfaces.

REFERENCES

1. Tarfah Alrashed, Almaha Almalki, Salma Aldawood, Tariq Alhindi, Ira Winder, Ariel Noyman, Anas Alfaris, and Areej Alwabil. 2015. An observational study of usability in collaborative tangible interfaces for complex planning systems. *Procedia Manufacturing* 3 (2015), 1974–1980.
2. David Anderson, James L. Frankel, Joe Marks, Darren Leigh, Eddie Sullivan, Jonathan Yedidia, and Kathy Ryall. 1999. Building Virtual Structures with Physical Blocks. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology (UIST '99)*. ACM, New York, NY, USA, 71–72. DOI: http://dx.doi.org/10.1145/320719.322587
3. 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 *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2189–2192.
4. LEGO Dimensions. 2015. Virtual Games. Website. (10 September 2015). Retrieved September 10, 2015 from http://www.lego.com/en-us/dimensions/.
5. A.M.B. Dourado, L.C. Botega, and R.B. Araujo. 2014. TTUI-SM: A New Specification Model for Tabletop Tangible User Interfaces. In *Virtual and Augmented Reality (SVR), 2014 XVI Symposium on*. 238–245. DOI:http://dx.doi.org/10.1109/SVR.2014.30
6. LEGO Fusion. 2015. Real Build. Virtual Games. Website. (10 September 2015). Retrieved September 10, 2015 from http://www.lego.com/en-us/fusion/.
7. 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 (CHI EA '01)*. ACM, New York, NY, USA, 355–356. DOI: http://dx.doi.org/10.1145/634067.634277
8. 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*. ACM, 1441–1450.
9. Paul Marshall. 2007. Do Tangible Interfaces Enhance Learning?. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. ACM, New York, NY, USA, 163–170. DOI: http://dx.doi.org/10.1145/1226969.1227004
10. M.F. Tunali and M. Gokturk. 2015. Effects of HMD on 2D object interaction in 3D virtual environments. In *Signal Processing and Communications Applications Conference (SIU), 2015 23th*. 1825–1828. DOI: http://dx.doi.org/10.1109/SIU.2015.7130210

11. Katrin Wolf, Stefan Schneegass, Niels Henze, Dominik Weber, Valentin Schwind, Pascal Knierim, Sven Mayer, Tilman Dingler, Yomna Abdelrahman, Thomas Kubitza, Markus Funk, Anja Mebus, and Albrecht Schmidt. 2015. TUIs in the Large: Using Paper Tangibles with Mobile Devices. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*. ACM, New York, NY, USA, 1579–1584. DOI: http://dx.doi.org/10.1145/2702613.2732863