

Bricktable: A Musical Tangible Multi-Touch Interface

Jordan Hochenbaum^{1,2}
jhochenb@alum.calarts.edu

New Zealand School of Music¹
Wellington, New Zealand

Owen Vallis^{1,2}
ovallis@alum.calarts.edu

California Institute of the Arts²
Valencia, CA, USA

Abstract

This paper describes how the development of a tangible multi-touch surface, Bricktable, was made possible through the use of open-source communities and tools. Subsequent work researching the musical applications of multi-touch interfaces has allowed the Bricktable to now become a resource for others. Several Bricktable projects are provided as examples.

1. Introduction

The original design for the Bricktable was to create an interface that allowed users to interact with media in ways that transcended the traditional keyboard and mouse paradigm. The concept was to allow multiple users to draw paths over a map, and use those paths to create musical compositions. In order to support this interaction, the interface would need to allow multiple users to directly access the map simultaneously. Several options were considered, including using Wiimotes¹ [1] as "light pens" on a projected wall, tracking people walking in front of or on top of a projected surface [2], or creating an interactive tangible surface similar to the reacTable[3]. Ultimately, it was decided that the ability for a user to place their hands directly onto the surface would create the most immediate and personal experience. Due to a limited budget and the lack of specialized skills required to build a multi-touch surface, the project quickly turned to the aid of free and open-source resources.

Traditionally, open-source is used to describe free software that is distributed without any licensing impediments, which also makes the source code readily available². Although the Bricktable project does rely heavily on the use of free software, the "open-source" nature of the

technology behind the hardware is of even greater importance. This extended definition of open-source encompasses the invaluable information provided by those who have documented their own multi-touch build processes. From information on ways in which to save money on material costs, to advice on particular hardware configurations, access to the freely available information enabled the Bricktable project to quickly move from the design phase to the build phase.

2. Related Work

With the potential for a high level of simultaneous control, multi-touch devices make an attractive interface for musical expression. Initial pioneering work done by Don Buchla on his Buchla Series 100³, and his Thunder midi device⁴, led to touch controls for producing and controlling sound. Although these had no visual feedback screens under the touch controls, they were early and effective touch based interfaces, and showed the potential for direct touch in musical applications. In recent years, there has been a great growth in experimentation with multi-touch devices for musical purposes [4-7]

One such project is the reacTable, a tangible multi-touch interface that allows users to perform music by placing objects called "fiducials" on to the surface. Each object can be assigned a specific function, allowing the user to interact with the process of making music within a visual environment. In addition to being a tangible musical interface, the reacTable is also powered by the open-source vision based tracking framework, reacTIVision[8].

Although the reacTable provides a rich environment for musical expression, it lacks the

¹ <http://www.nintendo.com/wii>

² <http://www.opensource.org/docs/osd>

³ <http://www.buchla.com/historical/b100/>

⁴

<http://www.buchla.com/historical/thunder/>

ability to detect the pressure of touch events, and is limited in its input rate by the frame rate of the tracking camera. Work has been done on alternate sensing methods that allow for such pressure data at extremely high sample rates[9]. However, this method still has yet to couple a visual screen with the surface.

The growth in musical multi-touch research can be attributed in part to the affordability and scalability of new methods for constructing these interfaces[10].

Additionally, researching work from over the last thirty years was also very informative as to the origins of current devices. In 1972 the PLATO IV[11] system was developed by the University of Illinois. The PLATO IV provided a 16x16 infrared touch panel for interfacing with the computer screen directly. This was an early example of pairing visual feedback with a direct touch surface. Finally, Myron Krueger's[12] work on a video desk in 1983 can be seen incorporating gestures onto the touch surface. Some of these very same gesture based multi-finger interactions are now common in many products, such as apple's iPhone and Jeff Han's work with Perspective Pixel⁵.

Furthermore, commercial products, such as the Lemur by Jazz Mutant⁶, also serve as an inspiration for using multi-touch technology in audio applications. The concept of a modular interface for manipulating audio closes the gap between performer and machine, while opening the potential for unique, project specific interfaces. Lastly, the forum provided by NUI group⁷ is an indispensable resource for learning the techniques and technology used to create the hardware behind tangible multi-touch surfaces.

Although the work above shows great potential for a more direct connection between the performer and the music, it was felt that the musical applications could be broadened in new directions. To this end, the Bricktable project focused on creating not only real-time performance instruments, but also multi-user sound installations as well. The idea being that not only could the interface be tangible and modular, but the musical context could be as well.

3. Hardware

3.1 Techniques

There are four primary methods for vision tracking in tangible multi-touch systems. These include: Diffused Illumination (DI), Frustrated Internal Reflection (FTIR), Laser-Light Plane (LLP), and Diffused Surface Illumination (DSI). Each method exhibits positive and negative factors in its technology. The requirements of particular projects were the deciding factors in the choice to use DI for the Bricktable.

3.1.1 Diffused Illumination (DI)

Diffused Illumination is the method of vision tracking at the core of the Bricktable. DI was chosen for several reasons: cost, scalability, finger tracking, and object tracking. Other traditional methods of vision-based tracking systems for multi-touch control (such as FTIR & LLP) only permit the ability to track fingers. The nature of the final works in which the Bricktable would act as an interface (see section 4.2) required both the ability to track touches, as well as interactions with tangible objects on the table's surface.

DI works using the following process: An image is projected onto a screen that is typically made from either clear acrylic or glass. Infrared light is shined up at the screen, and is diffused by a material that is placed on the top or bottom of the screen. When a finger is pressed on to the screen's surface, it reflects more light than the rest of the diffused material around it, which is tracked by a camera underneath the screen. An infrared band pass filter is placed over the camera lens in order to permit the vision-based tracking, while also serving as a means to separate the projected image from the tracking input. Additionally, the rear illumination allows for the camera to recognize shapes as well as fingers. This is unique to the DI method, and allows for individual markers to be recognized.

⁵ <http://www.perspectivepixel.com>

⁶ <http://www.jazzmutant.com>

⁷ <http://www.nuigroup.com>

3.2 Construction

3.2.1 Brick I

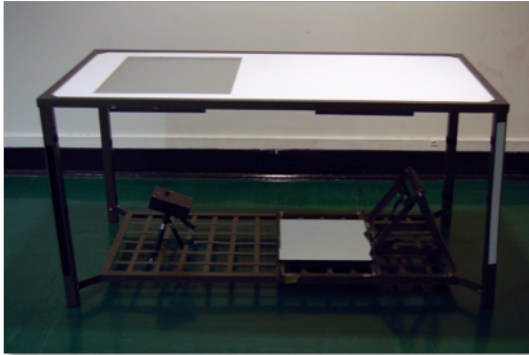


Figure 1 - Inside view of Brick I

The first step in designing Brick I was deciding on which open-source tracking software to use. reacTIVision was chosen because it was cross-platform, and the other option, Touchlib⁸, was not. Once the software was decided upon, it was necessary to design a cost effective hardware solution. After following build-logs within the NUI community, a parts list was finalized. The table would require an outside shell, an IR LED array to illuminate the inside of the table, a projector and mirror to produce the image, a webcam to track the images, and a rear projection surface.

The table frame was purchased from IKEA. An outdoor patio table was chosen, as it provided an inexpensive metal frame on which an enclosure could be built. After much trial and error, Side panels were securely fastened with Velcro strips; however, the top panel required welding small pieces of angle iron in order to keep it in place. A small 2' x 1.5' opening was cut out of the top to provide the space for the projection area. The final piece of the enclosure was a false floor to support placement of the projector, mirror, and webcam. This floor ensured that any movement of the table by users did not change the crucial spatial relationship of the individual hardware components.

The next component was a custom built LED array to illuminate the inside of the table. Ledtech⁹ UT188X-81-940IR LEDs were configured in a large series parallel network. The total current draw was so great that ceramic

resistors were required in order to dim the array without burning out components. Unfortunately, the narrow viewing angle of the LED's created problematic hot spots, and the 940nm wavelength's proximity to the visible spectrum made image tracking difficult. Both these issues were addressed in designing Brick II.

The project budget only permitted the use of a pre-owned InFocus LP290 projector; however, the LP290's older technology resulted in a somewhat degraded image quality, and inconsistent operating performance. The projector was placed on a metal mount that enabled a repeatable image angle-adjustment. Finally, the image was reflected off of a front-side mirror, allowing for a clean image to be projected onto the multi-touch surface.

Lastly, the Unibrain Fire-i firewire camera comes without any IR blocking filter, and so it was an ideal candidate for tracking images in infrared. In order for the Fire-i to ignore the projected image, two pieces of exposed film negatives were used as an IR low pass filter.

With the inside components complete, various projection materials were considered. Again researching on forums led to several different methods. Vellum, and tissue paper were both discarded, despite their good performance in producing a visible image. The initial installation was to be open to the public, and it was agreed that tissue paper was not only in high risk of ripping, but also not the most "finished" looking solution. Frosted acrylic was decided on, and although it looked nice initially, many shortcomings arose. To begin with, the frosting did not diffuse the image sufficiently. This created a concentrated bright spot in the center of the surface. Worse still, the image began to quickly fade away the further it was from the center of view. This required viewers to physically move around in order to see the whole image. Finally, the frosting mixed with natural oils from human contact and became slightly translucent. Several modifications were made to the frosted acrylic to fix these issues. The final surface combined both an additional frosted coating on top of the existing one in order to create a more diffuse surface, and an additional thin piece of clear acrylic on top as a protective layer.

Once completed, this table predominantly relied on the use of tangible objects for interaction with the software. This heavily influenced the development of the software that was designed and used on the Bricktable I.

⁸ <http://nuigroup.com/touchlib/>

⁹ <http://www.ledtechusa.com/>

3.2.2 Brick II

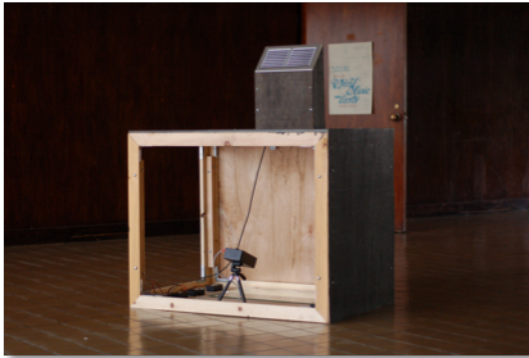


Figure 2 - Inside view of Brick II

The second Bricktable was an attempt to solve several issues with the initial design. This included increasing portability and ease of building, improve the IR source, increasing the touchable surface, and improving the quality of the overall projected image. All of these concerns needed to be addressed within a budget of \$500.00.

The first task was to create a table that was more portable than the original design. The original design used a heavy metal frame from an IKEA table to act as the basic structural frame. The frame required special hex keys and hardware to put together, and was rather large in size. Additionally the combined weight of the five acrylic panels plus the metal frame was substantial. Both of these factors made for a difficult time in transporting the table by car, or shipping the table, to and from events. Lastly, the shape of the table placed extreme restrictions on the overall size of the touchable surface area. After researching other table designs on the NUI group forums, Brick II was designed using lightweight wood frames and wing nuts that allowed for a compact break down and quick assembly that required no additional tools.

With the ability to control the dimensions of the table, projection travel distance was optimized to create a 50" diagonal touchable surface with a 16:9 aspect ratio. This was a great improvement over the 30" diagonal 4:3 screen on Brick I. These changes greatly increased the potential for multi-user input, while at the same time created a much larger and vibrant surface. Additionally, several different rear projection/diffusion materials were researched and 225 Neutral Density Frost made by Lee

Filter¹⁰ was chosen. This material provided a much more evenly illuminated image while acting as an effective diffuser for the IR. The Lee filter was glued to the acrylic using 1 part white glue to 10 parts water. This provided a transparent bond between the two materials; however, after repeated flexing of the acrylic during moving, it is necessary to re-glue the filter to the acrylic from time to time.

With Brick II, infrared emitter issues were also addressed. Researching the NUI forums and build-logs of other members of the community, it became clear that the IR LEDs in Brick I were providing a very narrow and focused beam of IR. This created an uneven light source within the table, and made reliable object tracking very difficult. Research on NUI group led to the use of Osram Opto Semiconductors¹¹ SFH 426 LEDs in the new Brick II design. These LEDs emit an IR beam at a wavelength of 880nm, at a dispersion angle of 120°, providing a much more even field of IR while using much fewer LEDs. The need to dim the IR LED's was also addressed. In the original design of Brick I, the series/parallel LED network pulled so much current that the only effective way of resisting that current was to use bulky ceramic resistors. With Brick II the opposite approach was explored and a custom-built variable voltage regulator design found on the DIY website <http://www.instructables.com> was implemented. This made it possible to vary the voltage on the front end, in turn lowering the amount of current available to the LED network. Switching from film negatives to Lee Filter's Infrared, No. 87 provided a more controlled low pass IR filter and greatly improved the overall response to objects and fingers.

Finally, a switch was made from the tangible oriented tracking software reactIVision, to tBeta¹², an open-source and cross platform vision tracking system started by Seth Sandler. This provided a more robust multi-touch finger tracking system; not only did this improve the ability to track fingers as events, but through it's built in automatic calibration mode, greatly improved the time and difficulty in setting up the touchable surface. The major draw back was the loss of tangible object support. Experimentation showed that digital filter settings for finger

¹⁰ <http://www.leefiltersusa.com/>

¹¹ http://www.osram-os.com/osram_os/EN/

¹² <http://ccv.nuigroup.com/>

tracking were not effective for object tracking and vice versa. Effective integration of both tracking techniques would require separate filter paths for both fingers and objects. As tBeta is open-source, there is already work being done to add support for these features.

3.2.3 Brick III

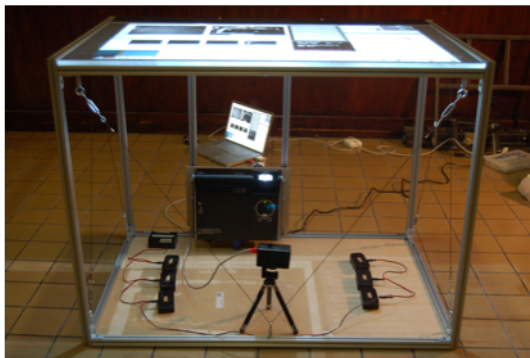


Figure 3 - Inside view of Brick III

While Brick II offered great improvements over the Brick I design, further improvements were still necessary in an attempt to make Brick a highly portable and stable tangible multi-touch interface. Brick II could easily break down and be reconfigured quickly, stacking to fit inside almost any car trunk space; however, transporting Brick II to destinations that required shipping proved to be both extremely expensive and unreliable. Furthermore, in order to address poor visual representation of our artistic ideas, as well as to streamline and improve the actual table design, Brick III required a new projector. In addition to these concerns, it was important to reduce the effect of ambient IR interference, which would greatly improve the overall Bricktable setup, functionality, and usability.

Brick III was designed to be easily constructed out of a ridged and lightweight aluminum material called 80/20TM¹³. Known as the "industrial erector set," its t-slotted profile (see figure 3) enables the ability to easily configure and adjust a reliable mounting system for the projector, camera, IR led system, mirror and laptop. Not only does this framing system allow for an easily configurable and adjustable multi-touch table construction, it breaks down into multiple 1"x1" bars, with the maximum length being 43.5". This allows the entire table to

conveniently break down into a duffle bag or suitcase, which can easily be traveled with regardless of distance. For times when shipping the permanent side panels and acrylic top is not feasible, solutions that enable even more portability have been devised. Inexpensive arrangements for ordering the acrylic at the destination can be made, and a cloth-based side panel system has been created. This greatly reduces the cost of traveling with the Bricktable anywhere in the world, and has finally fully addressed the issue of portability that plagued previous Bricktable designs.

In addition to the new aluminum framing, various projector concerns were also addressed. Brick III uses the Toshiba TDP-EW25P extreme short throw projector, solving the throw distance issue of projecting a 50" diagonal image. Previously, the InFocus LP290 was encased above the table to reflect an image 3ft. down and 3ft. up off a mirror in order to throw the demanded screen size. This not only had physical and aesthetic implications on the Bricktable, but also acted as a barrier for full, 360° multi-user interactivity around the table's surface. Using the new short throw projector, Bricktable III is now barrier free, and provides more space to encourage multiple user interaction and creativity. In addition, the Toshiba TDP-EW25P provides much higher image resolution and quality, improving the image across the board, including, color, clarity, contrast and sharpness.

4. Software Development

Applications for the Bricktable have been developed in many programming environments, both commercial & open-source; however, as the Bricktable project has matured, software has increasingly been programmed in open-source languages. Applications are developed around an open framework for multi-touch data passing/parsing called TUIO[13].

TUIO is a standardized protocol with an active community, and has been a crucial component of the Bricktable's software applications. Additionally, the TUIO framework is supported in a plethora of programming languages, including C++, Max/MSP, Java, Processing, and Flash. This broad support provides the freedom to explore developing in a variety of languages while using a common architecture; furthermore, the common framework allows us to rapidly develop and

¹³ <http://www.8020.net/>

implement new ideas with others. In fact, in 2008 Brick II saw a software programming collaboration that took place between multiple people on different computers, using different operating systems (Windows and Mac OS X), and on different continents. Our collaborator developed without even having a multi-touch surface to test on.

The community has created an open-source TUIO simulation software that allows developers to write applications without requiring hardware. This simulator acts as virtual tangible multi-touch table, allowing the computer mouse to imitate fingers and tangible objects. Without TUIO's standard framework making it ease to collaborate with other developers, beyond geographic, physical, and hardware limitations, Roots would not have been possible.

4.1 Installation

4.1.1 Weather Report

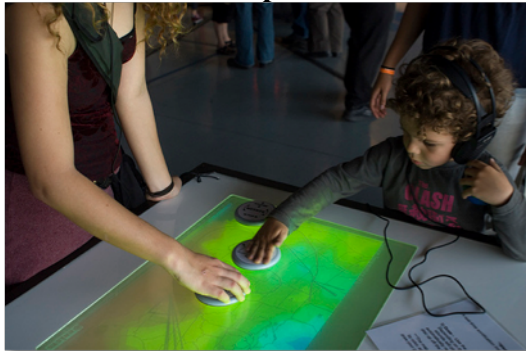


Figure 4 – "Weather Report" at Yuri's Night Bay Area Festival 2008

Weather Report utilizes the Bricktable as a sonification instrument. Users are invited to move tangible objects around a surface displaying a colorized map of the United States (see figure 4). This map details the current surface temperature, which is updated hourly over the Internet. Various samples are associated with geographic locations around the map, and as users move the objects, they trigger and compose the sounds. This interaction leaves behind a visual "trail" of the paths, and when the user stops moving the objects, the path in which they traveled will automatically begin to loop. The current temperature of the regions in which the objects travel affect the timbral qualities of the sounds, and so as the temperature changes throughout the day, week, month, year...etc, the arrangements created by the user morph and

develop. Additional musical elements are added through the use of specific objects which when rotated, create rhythmic layers. Weather Report was developed using reacTIVision, Max/MSP, and Jitter programming environments.

4.1.2 Roots

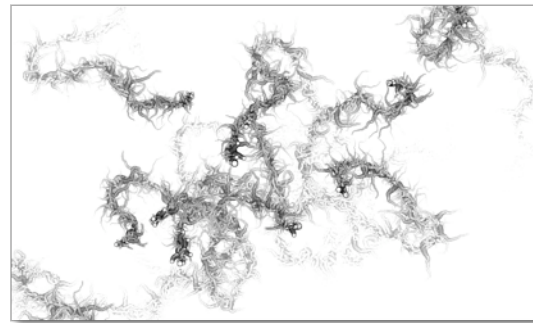


Figure 5 - Screenshot of "Roots" Installation

Roots is an interactive installation which invites multiple users to create music in a generative, semi-generative, and or completely composed musical environment. When a user touches the table surface, a vine-like "branch" is generated. This branch randomly grows and maneuvers around the surface, scrubbing through various audio buffers and generatively recomposing the musical material.

In order to give the user more control over the chaotic environment, users can decide to introduce tangible "force field" objects that turn the system from completely generative into a semi-generative environment. When the objects are placed on the Bricktable's surface, a visual "force field" is emitted, either attracting or repelling the roots, depending on the rotation of the object. The further rotated clockwise from zero degrees, the stronger the attraction force, and likewise, the further the objects are rotated counter-clockwise, the greater the repelling force. Setting force field objects around the surface, users introduce influence over where and how the vines maneuver, and how the audio is generated.

For complete control over the audio, the user simply has to press their finger on the surface, and the audio scrubbing will directly correlate with their movement in a 1:1 relationship. This allows Roots to operate either completely generative, semi-generatively, and/or fully controlled.

Roots is a collaborative project. While brainstorming ideas for a new installation, we

discovered a web application called "Eerie and Drippy". Because the software was written in Processing¹⁴, an open-source Java based programming language, we were able to compile and source all required libraries easily. We realized the potential for enabling "Eerie and Drippy" to work with the Bricktable by implementing TUIO support in the program. After contacting the original creator of "Eerie and Drippy", work began on developing the installation.

More so than any other Bricktable project, Roots embodies the potential for collaborative work possible through using open-source software.

4.2 Performance

4.2.1 Spaces



Figure 6 - User Interacts with "Spaces" Interface

Designed as a minimalist interface to free musicians from traditional compositional markers such as frets and keys, Spaces enables musicians to compose intuitively through immediate visual and sonic feedback. The screen is divided into four instruments over sixteen lanes. This gives each instrument four controls, three to control various parameters such as pitch, filters...etc and one to control instrument volume. Users moving their finger from top to bottom receive visual feedback as the lanes morph colors from blue to pink. Instead of the labels and markers used on traditional instruments, Spaces forces the performer to use their ears to determine the musical effect of their actions. Spaces uses the open-source software tBeta for finger-tracking, Processing for visual feedback, and Native Instruments Reaktor¹⁵ for audio synthesis.

4.2.2 Robotics

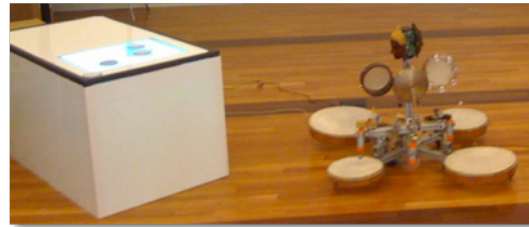


Figure 7 - Brick I & MahaDeviBot

The MahaDeviBot[14] is a 12-armed, solenoid driven percussion robot developed by Dr. Ajay Kapur, director of the MTIID¹⁶ program at California Institute of the Arts. The Bricktable has interfaced with the MahaDeviBot in three unique ways. The first applications used tangible objects. Moving the objects along the x-axis selected various rhythms, moving along the y-axis drove manipulations of a global tempo, and the object's rotation angle controlled the velocity of each rhythm. The next group of applications used images on the surface to directly trigger the individual drums in real-time; however, due to the 30fps rate of the Unibrain Fire-I camera, the overall response was found to be too slow for useful musical expression. As a result of this experiment, less direct methods of robotic drum control have now been explored. This has lead to a third approach of controlling musical robotics with the Bricktable—using Conway's Game of Life to generatively control drum patterns.

5. Conclusion

The open nature of the various software used to power the Bricktable has allowed us to create highly customized and project-specific applications. With each of these applications, we have explored new potential musical implementations of touch surfaces, enabling us to contribute back to the community. In this way, we hope to expand the potential uses of tangible multi-touch surfaces as interactive, collaborative, and compositional tools for future musical explorations.

¹⁴ <http://www.processing.org>

¹⁵ <http://www.native-instruments.com>

¹⁶ <http://music.calarts.edu/~mtiid/>

6. Acknowledgments

We would like to thank Memo Akten for both his work on the roots installation, as well as his creative input. We would also like to thank Dimitri Diakopoulos and Jim Murphy for their work on the spaces application. Additionally, we would like to thank Seth Sandler and the rest of those who have helped from the NUI group forums. Lastly, we would like to thank Ajay kapur for his guidance and mentorship.

7. References

1. Bradshaw, D. and K. Ng, *Tracking Conductors Hand Movements Using Multiple Wiimotes*, in *Proceedings of the 2008 International Conference on Automated solutions for Cross Media Content and Multi-channel Distribution*. 2008, IEEE Computer Society.
2. Delbr, T., et al., *A tactile luminous floor for an interactive autonomous space*. Robot. Auton. Syst., 2007. **55**(6): p. 433-443.
3. Jordà, S., et al. *The reacTable*. in *Proceedings of the International Computer Music Conference 2005*. Barcelona, Spain.
4. Davidson, P.L. and J.Y. Han, *Synthesis and control on large scale multi-touch sensing displays*, in *Proceedings of the 2006 conference on New interfaces for musical expression*. 2006, IRCAM Centre Pompidou: Paris, France.
5. Patten, J., et al., *Sensetable: a wireless object tracking platform for tangible user interfaces*, in *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2001, ACM: Seattle, Washington, United States.
6. Wessel, D., M. Wright, and J. Schott, *Intimate musical control of computers with a variety of controllers and gesture mapping metaphors*, in *Proceedings of the 2002 conference on New interfaces for musical expression*. 2002, National University of Singapore: Dublin, Ireland.
7. Newton-Dunn, H., H. Nakano, and J. Gibson, *Block jam: a tangible interface for interactive music*, in *Proceedings of the 2003 conference on New interfaces for musical expression*. 2003, National University of Singapore: Montreal, Quebec, Canada.
8. Kaltenbrunner, M. and R. Bencina, *reactIVision: a computer-vision framework for table-based tangible interaction*, in *Proceedings of the 1st international conference on Tangible and embedded interaction*. 2007, ACM: Baton Rouge, Louisiana.
9. Jones, R. and A. Schloss, *Controlling a physical model with a 2D force matrix*, in *Proceedings of the 7th international conference on New interfaces for musical expression*. 2007, ACM: New York, New York.
10. Han, J.Y., *Low-cost multi-touch sensing through frustrated total internal reflection*, in *Proceedings of the 18th annual ACM symposium on User interface software and technology*. 2005, ACM: Seattle, WA, USA.
11. Schneck, M.A. *Plato Implementations And Evaluations: New Behaviors And Objectives For Education*. in *EdCompCon*. 1984.
12. Krueger, M.W., T. Gionfriddo, and K. Hinrichsen, *VIDEOPLACE--an artificial reality*, in *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1985, ACM: San Francisco, California, United States.
13. Kaltenbrunner, M., et al. *TUIO - A Protocol for Table Based Tangible User Interfaces*. in *Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation 2005*.
14. Kapur, A., *Digitizing North Indian Music: Preservation and Extension using Multimodal Sensor Systems*, *Machine Learning and Robotics*. 2008: VDM Verlag. 276.