Constructing Virtual 3D Models with Physical Building Blocks

Ricardo Jota
VIMMI / Inesc-ID
IST / Technical University of Lisbon
1000-029 Lisbon, Portugal
jotacosta@ist.utl.pt

Hrvoje Benko
Microsoft Research
One Microsoft Way
Redmond, WA 98052
benko@microsoft.com

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.
ACM 978-1-4503-0268-5/11/05.

Abstract

Constructing virtual 3D models typically requires specialized desktop modeling tools (e.g., CAD tools), which, while very powerful, tend to require a lot of precision, time, and expertise from the user. We present *StereoBlocks,* a system that combines a Kinect depth camera with 3D stereoscopic projector to allow the user to build complex virtual 3D models from available physical objects. By treating the camera information as a continuous 3D digitizer, we are able to capture the details of the real world and re-project virtual objects side-by-side to real objects. The user is able to visualize such mixed reality model through stereoscopic projected imagery tightly aligned with the real world. In our system, it is literally possible to build the entire virtual castle, using only a single physical brick piece. We discuss our prototype implementation and report on early feedback from the four users that evaluated our system.

Figure 1 - *Top*: Workbench overview. *Bottom left*: Examples of physical objects available to the users. *Bottom right*: Wireless number pad used to manipulate the model and capture object information.

ACM Classification Keywords
H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General Terms
Design, Human Factors

Keywords
Depth cameras, 3D modeling, freehand interactions, stereoscopic 3D interfaces.

Introduction
StereoBlocks aims to create a seamless interface between physical and virtual worlds. Rather than relying on CAD tools to synthesize virtual models we were guided with a simple idea to let the user build virtual models using physical objects. Our motivation was to provide a building experience, similar to scale models, where a virtual model could be build, step by step, using simple everyday objects and visualized side-by-side to the construction blocks.

Our system provides a workbench where users can capture everyday objects to compose a complex virtual scene, mixing physical construction with virtual stereoscopic visualization. The system was tested with four users to understand what kind of emerging interaction can come out of a system with limited building blocks. Early findings on how users interact are discussed.

Related work
Tangibles, as a way to build virtual models, have been the target of previous research. In Illuminating clay, Ishii *et al.* presents a tabletop where users can change the topology of a clay landscape model and have the projected geometry adapt in real-time [8]. In the Bricks system [2], Fitzmaurice *et al* explored how tangible blocks could act as analog physical controllers to virtual objects. Similarly, the Luminous Table uses objects as physical elements that include virtual shadows and affect wind and traffic visualizations [5]. However, the physical objects have to be defined beforehand, which limits the number of objects available for authoring. Our approach removes this limitation and allows any object to build models, not just pre-defined objects.

Both Bricks and the Luminous Table enable a user to construct a model by abstracting the analog properties of a physical artifact. For example, changes in the artifact's shape are not recognized by the system. Grossman uses changes in the object's shape as a rich input channel in ShapeTape [3]. He presents an interaction device that directly controls the shape and position of a virtual curve. In *StereoBlocks*, we take a similar approach to Grossman and capture as many input characteristics as possible from a physical objects. However, we go on step further and support multiple objects *and* composition of objects.

In general, the previous examples focus on single objects that are never composed to form new objects. Raffle *et al*. introduces a 3D constructive assembly system to quickly assemble biomorphic forms [9]. Although they assemble a model, it is not possible generate a model more complex than the number of assembly pieces. Hosokawa, Anderson and Kitamura introduce systems that also compose physical objects to build a virtual model [1][4][6]. Hosokawa and Kitamura, however, only support pre-defined building blocks. Anderson *et al.* go a step further and support clay objects that can express different shapes. In each

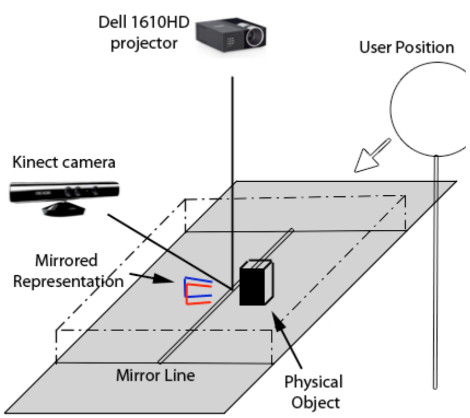

Figure 2 – StereoBlocks uses a 3D ready projector and a Kinect camera as input device. The projector is setup on top of the table surface and projects down to a 50x50cm area. The Kinect camera is setup on a 45 angle in relation to the table and is used to track objects on top of the table and user movement. The user is required to wear 3D active shutter glasses and experiences a correctly projected 3D virtual scene that blends with physical objects positioned on the workbench.

of these examples, construction and visualization are disjoint tasks the user completes during construction (they build and then visualize). We differ from these examples by combining the space for construction, using any physical object available, with visualization, by offering the user a stereoscopy projection in the same workbench used for construction (they build and visualize the result at the same time).

Perceptive Workbench is perhaps the closest to our system. Starner *et al.,* are able to reconstruct any object positioned on the table[10]. Their infrared setup requires accurate calibration and does not provide texture information. Our system improves on this work by offering a smaller setup, more robust to lighting conditions, that reconstructs multiple objects with richer information of the object's shape and texture.

StereoBlocks

Stereoblocks is a tabletop environment composed of an input workbench and stereoscopic visualization (Figure 2). It uses a Dell 1610HD projector to project a 50x50 cm stereoscopic 3D scene onto a desk. To achieve stereoscopic visualizations the system requires the user to wear active 3D shutter glasses, which are synchronized to the projector's frame rate using DLP-link technology. The Kinect camera serves a dual purpose in our system. First, it captures the physical objects on top of the workbench and synthesizes them into a virtual copy (a 3D texture-mapped virtual model). Second, Kinect tracks the user's head position to generate a correct perspective view of the virtual content to the user. The end result is a 3D scene imaged by the user that seamlessly mixes real and virtual objects (Figure 4).

We support two types of interactions in *StereoBlocks*. First, the user can build a part of the model, by simply constructing it from any available physical objects, including their hands (Figure 4), and the system automatically displays how such objects would appear when virtualized. Second, once the user captures a scene, they can adjust the position and orientation of the virtual object using keys on a small number pad keyboard (see Figure 2 for the system setup).

User Interface

The basic sequence of interactions in *StereoBlocks* follows a simple set of steps. Users can compose a scene by (1) placing some physical object(s) to the workbench, (2) capturing the initial virtual scene, (3) moving and rotating that virtual scene, (4) adding, removing or adjusting the physical objects and (5) capturing new objects which end up augmenting the existing virtual scene. To build a complex scene with limited physical pieces, the user can capture the workbench information multiple times. Furthermore, being able to move and manipulate the virtual scene in 3D space enables the user to build larger or higher models. This is particularly helpful for adding to the models in the vertical dimension, where the user can use the workbench surface as a working plane and, for example, simply lower the virtual model into the physical surface to build a higher floor.

To provide a clean workbench, we map the buttons of a number pad controller to facilitate all virtual scene manipulations (e.g., moving the scene in all three dimensions, rotate the scene along the Z-axis). Two additional buttons are used to capture real object information into the virtual scene and to undo the last capture. Individual buttons for these actions are shown

Figure 3 – Number pad used as interface device for the workbench.

in Figure 3. Note that the buttons in black (with no tags) have no functionality and are not mapped.

Mixing real and virtual objects

The workbench is a 3D projection that includes representation of the physical objects currently on top of the workbench plus a virtual scene that includes all previously captured objects. Any object inside the workbench's interactive area has a live-feed mirrored visual representation (Figure 1 - Top). A mirrored representation simply means that any object (or hand) the user places on the table has an instantaneous virtual copy appear as if coming directly from the opposite side of the table (Figure 2). Such mirrored visualization was selected for two reasons. First, it provides visualization offset so that physical objects or hands do not occlude virtual content. Second, through mirror representation, the users can easily visualize how their physical pieces modify the final captured result without having to capture the scene and remove the physical objects.

The user can either move the physical objects to align with the existing virtual model, or use the number pad interface to reposition captured information. Note that navigation only affects previously captured information; it does not affect live-feed virtual representations of real objects currently on top of the workbench.

Capturing real objects

When the prototype is running, models are captured as follows: first, the depth information is captured and the background information is subtracted. Secondly, the resulting 3d points are converted to real-world coordinates, using the offline calibration, and information outside the pre-defined workbench area (50x50x30 cm region) is removed. This effectively removes all background and most user position depth information, leaving only real objects and user information (hand and arms) inside the workbench area. Third, using the remaining points, a mesh is created and a texture is applied, with the color information captured by the depth camera. Finally, the mesh is projected using stereoscopic projection in a mirrored position. Figure 1 shows multiple blocks being captured and projected in stereo.

Projective Texture

To provide correct perspective and stereoscopic visualizations, our system is calibrated so that the camera and the projector position and orientation are known in the real-world coordinate system. To calibrate the system we follow the similar camera and projector calibration methods as outlined in the LightSpace project by Wilson and Benko [11]. At start time, we also capture an empty scene depth map to use as a background baseline to segment all objects and user body parts inside the scene.

The goal of our visualization is to present virtual objects as direct copies of their physical counterparts. This means that these virtual objects are correctly aligned with the scene and appear to be of the same size and depth, from the perspective of our user. In order to enable this, we synthesize the projector imagery through the use of projective texture that takes into account the position and orientation of the users head. A projective texture is a method of texture mapping described in Segal [7] that executes two GPU render passes. The first render pass creates a texture of what objects are seen, from the user perspective, and the second pass re-projects the texture given the

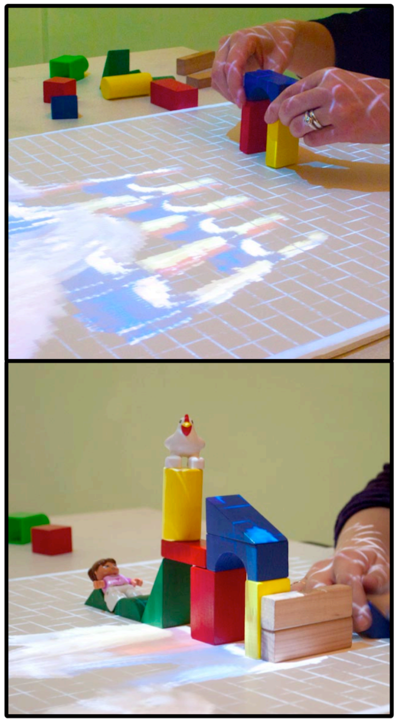

Figure 4 - Users capture less objects at a time when asked to re-create a model (left). When free forming (right), users seem to capture more objects at the same time.

projector and user eye position. This means that virtual objects react to user movements, for example they appear elongated if the users lower his head.

System Limitations

The system, in its current form, has a number of limitations. A stationary single Kinect camera can only capture objects from a single side. Multiple cameras reduce the issue, but require algorithms to match object information and combine the meshes.

Currently, we only support a single user due to the need to present a correct perspective stereo projection. However, we have considered connecting two *StereoBlocks* setups in a remote conferencing system, which would offer additional interactive capabilities.

The captured resolution of our virtual models is rather low and their surfaces are noisy as this information depends on the available resolution of the Kinect camera. Higher camera resolutions would provide better textures, but additional improvements could be achieved by temporally averaging the information across many camera frames. Lastly, projecting virtual content intermixed with the physical objects can result in unwanted artifacts recorded on the objects texture. This issue could be addressed by briefly suppressing the projection during capture.

Preliminary Feedback

We conducted a preliminary usability investigation to gather feedback on the performance of our system. The system was evaluated with four users: three females and one male, ages ranged between 30 and 40 years old. Two of the users were 3D modeling experts (architects) and two users were non-experts (computer science professionals with previous 3D interactive experience), users were rewarded for their time and effort with $25 gift certificate. We asked the users to build three pre-existing models, for each model the user was given a printed example (see Figure 5 – left).

Early evaluation of the system allows us to observe how users interact with the system. An interesting finding is how users combine virtual scene interaction with physical objects position. Using the system interface to clone objects, users can, to achieve the same results, either (1) position an object and capture it, move the virtual scene, press capture again, without reposition the object, or (2) position an object and capture it, reposition the object and press capture again. Our evaluation shows that users learn both techniques and apply them in different situations. Whenever cloning does not require rotations (such as cloning the temple columns one by one) all users favor moving the virtual scene while keeping the object stationary. Whenever rotations are required (e.g., in the coliseum section model), all users preferred to move the physical object, instead of the scene, only using scene translation to move the scene up or down. Moreover, in our system, holding the pieces in mid-air is unpractical because it might cause arm strain and results in hand information being forcefully captured. When users are questioned whether they prefer to move the physical pieces or translate the model, all four stated that they decided to use the solutions that best fitted the task. This suggest that both solutions are relevant when constructing models.

Although users were aware that hand information was being captured (it appeared in the live-feed), most were still surprised that hand information was register

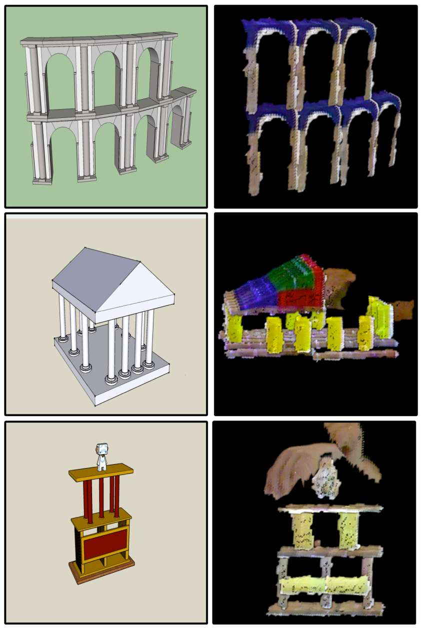

Figure 5 – Left: images of models given to the users. Right: models that the users constructed in StereoBlocks. From top to bottom: coliseum section, temple and trophy models. Note: the user's hands, if not removed at the time of capture, are also captured above the model.

along with object's information (Figure 5 shows a captured hand). Whenever this happened users would undo the last capture and re-capture the object information, without capturing their hands.

Users required instruction regarding the mirrored representation. Once instructed, however, they would gauge the piece position by looking at the mirrored representation and only capturing when the mirrored representation was in the correct position. On the other hand, the mirrored representation dictated user behavior: users would use the bottom half as a workbench and the top half (where the mirrored representations appear) as the visualization area.

Figure 5 shows the target models next to the end result users achieved. Users were able to reproduce the models with details such as: coliseum slight rotation; matching the number and position of the columns or build the trophy level by level without stacking physical objects. Users were very enthusiastic about the system and commented that the system was easy to learn.

Conclusion

StereoBlocks enables the user to build virtual models using typical physical objects captured by the depth camera and visualized through the real-world-aligned stereoscopic 3D projections. Early results indicate that this setup enables interesting novel interactive scenarios and might provide an alternative to modeling tools. In its current form, one can envision such system being used by children to build basic models. In the future we look forward to extending this work to enable the user to use their hands or other physical tools (e.g., a knife) to interact with the virtual objects in order to select, copy or even carve or sculpt the virtual objects.

References

[1] Anderson D, Frankel J, Marks J, Agarwala A, P. Tangible interaction + graphical interpretation: a new approach to 3D modeling. *SIGRAPH. 2000*;p393-402.

[2] Fitzmaurice G, Ishii H, Buxton W. Bricks: laying the foundations for graspable user interfaces. *Proceedings of the SIGCHI*. 1995;p442-449.

[3] Grossman T, Balakrishnan R, Singh K. An interface for creating and manipulating curves using a high degree-of-freedom curve input device. *Proceedings of the SIGCHI '03*. 2003;p185.

[4] Hosokawa T, Takeda Y, Shioiri N, Hirano M. Tangible design support system using RFID technology. *Proceedings of the TEI* 2008; p75-78.

[5] Ishii H, Ben-Joseph E, Underkoffler J, Yeung L, Chak D, Kanji Z, and Piper B. 2002. Augmented Urban Planning Workbench: Overlaying Drawings, Physical Models and Digital Simulation. In *Proceedings of the* ISMAR. 2002. p203-.

[6] Kitamura Y, Itoh Y, Kishino F. Real-time 3D interaction with ActiveCube. *CHI '01 extended abstracts on Human factors in computing systems*. 2001; p355.

[7] Mark Segal, et al. Fast shadows and lighting effects using texture mapping. In *Proceedings of SIGGRAPH '92*, p249-252.

[8] Piper B, Ratti C, Ishii H. Illuminating clay: a 3-D tangible interface for landscape analysis. *Proceedings of the SIGCHI*. 2002. p355-362.

[9] Raffle H, Parkes A, Ishii H. Topobo: A Constructive Assembly System with Kinetic Memory. *Proceedings of the SIGCHI*. 2004;p647-654

[10] Starner T, Leibe B, Minner D, Westyn T, Hurst A, Weeks J. The perceptive workbench: Computer-vision-based gesture tracking, object tracking, and 3D reconstruction for augmented desks. *In Journal of Machine Vision and Applications vol 14. P59-71. 2003*

[11] Wilson A, Benko H. Combining multiple depth cameras and projectors for interactions on, above and between surfaces. *Proceedings of the UIST'10, p273-28*