

Combining Virtual Environments and Direct Manipulation for Architectural Modeling

Bruno Araújo¹, Joaquim Jorge² and José Duarte³

^{1,2}IST/INESC-ID, Portugal ³FA/UTL, Portugal

¹<http://web.ist.utl.pt/bruno.araujo>, ²<http://web.ist.utl.pt/jorgej>, ³<http://www.fa.utl.pt/~jduarte>

¹brar@vimmi.inesc-id.pt, ²jorgej@acm.org, ³jduarte@fa.utl.pt

Abstract. *In this paper we present an environment for conceptual design which mimics the interaction of designers with physical mockups. We present novel three dimensional on-and-above-the-surface interactive techniques which combine asymmetric bimanual operations with multi-touch direct manipulation on a flat surface and depth cameras. These take advantage of a continuous interaction space to create and edit 3D models in a stereoscopic environment. To allow modeling architectural shapes directly, we combine hand and finger tracking in the space above the table with multi-touch on its surface without the need to change modes. This provides an alternative design environment where users can seamlessly switch between interacting on the surface or in the space above it depending on the task, which makes it easier to model complex shapes using simple operations.*

Keywords. 3D Modeling; 3D User Interfaces; Virtual Reality; Procedural Modeling.

INTRODUCTION

We present an environment for conceptual design which mimics the interaction of designers with physical mockups. The goal of this work is to provide both architects and designers with an immersive, interactive environment in which they can explore conceptual design ideas that is free from the physical constraints associated with conventional mockups. This environment is more flexible in the sense that it allows exploring a wider range of formal solutions, while enabling the immediate visual feedback provided by real scale models. The ‘virtual materialization’ in real time of design solutions is a key to this process and it can be later extended with simulation capabilities to test structural and environmental performance among others. We describe the 3D modeling primitives from the user point of view, introducing novel operations enabled by com-

binning this interaction with procedural techniques. We start by surveying work related to 3D modeling on immersive environments. Then we present our environment describing the hardware setup and its properties. Next we describe our approach which enables creating complex objects through sketches on the surface and gestures in space. Then we present a method based on more complex editing using procedural models on the same visualization environment. Finally we present results, discuss the technique’s limitations and identify future work.

RELATED WORK

With the widespread adoption of multi-touch devices and inexpensive and intrusive tracking solutions such as the Microsoft Kinect [3], academic research on tabletops has refocused on “on and above” sur-

face interaction techniques. Mueller-Tomfelde et al. (2011) proposed different methods to use the space above the surface to provide more natural ways of interacting with 2D tabletop content. Wilson et al. (2010) proposed metaphors to interact with different displays while capturing full body posture. In this way, users can interact on or above the surface with 2D content or even between surfaces using the body to transfer virtual content from one surface to the hand or to another surface while moving their bodies in space. They also proposed to use space to interact physically with projected GUI. Our approach explores the continuous space as presented by Marquardt et al. (2011) combining it with the bimanual asymmetric model proposed by Guiard (1987). Our approach also explores sketching which is a powerful communication tool of any real conceptual design task. Research on sketch based modeling proposed approaches to take advantage of designer drawing skills as illustrated in Olsen et al. (2009). Many systems rely on gesture recognition (SKETCH), stroke beautification (Pegasus), line drawing reconstruction (SmartPaper), suggestive interfaces (Chateau), push-pull sketching (Sesame from Oh et al. (2006) similar to Google Sketchup [1]), freeform contour inflation (Teddy or ShapeShop) to make sketching as a usable alternative to traditional CAD systems.

Finally procedural modeling techniques have been explored by recent work in Computer Graphics. Parish and Müller (2001) presented City Engine an extension of L-Systems to generate cities with streets and buildings automatically using procedural modeling via scripting. Wonka et al. (2003) propose split grammars as a better approach to building description. Instead of relying on a growing process on top of lines, these grammars use shapes and rules describing how to subdivide (split) shapes into new ones. Split grammars can generate facade buildings and reuse styles from a database. This solution influenced CGA Shape grammars by Muller et al. (2006) currently available on the ESRI CityEngine [2] product and is partially supported by our interactive modeling system. Our stereoscopy multitouch

environment provides a direct modeling more flexible than 2D interactive system such as the one presented by Lipp et al. (2008).

HARDWARE SETUP

Our setup consists in a semi-immersive environment based on a stereoscopic multi-touch display 140x96 cm combined with a Microsoft Kinect [3] depth camera and two Gametrak [4] devices used to identify and precisely track the hands and fingers above the surface. A video-projector under the table displays images on the surface at 120 Hz at 1024 x 768 pixel resolution. A pair of active shutter glasses from 3D Vision NVIDIA affords stereoscopic visualization. A Kinect depth camera tracks both the position and orientation of the head, using a detected skeleton (Figure 1). The 3D models appear on top of the surface and the point of view is updated following the position and orientation of the user's head to compensate for changes in parallax. Additionally, we use a tracked Nintendo Wiimote [5] device to access to more complex modeling modes based on procedural modeling techniques. Since several operations are needed to edit procedural descriptions, our solution enables users to access complex function via different buttons and to focus on the 3D model while modifying it directly.

CREATING ARCHITECTURAL MODELS USING THE CONTINUOUS INTERACTION SPACE

We propose a direct modeling approach to create, edit and manipulate 3D models using a small set of operations. After drawing sketches, users can create 3D models by pushing and pulling faces from existing content in a manner similar to Google Sketchup. Our modeling approach takes advantage of an asymmetric bimanual interaction model where the dominant hand (DH), and non-dominant hand (NDH) are assigned different roles while the user is interacting on and above a two-dimensional display surface. This setup allows users to sketch directly on the surface and extrude faces and shapes directly on the third dimension. This is possible by using a

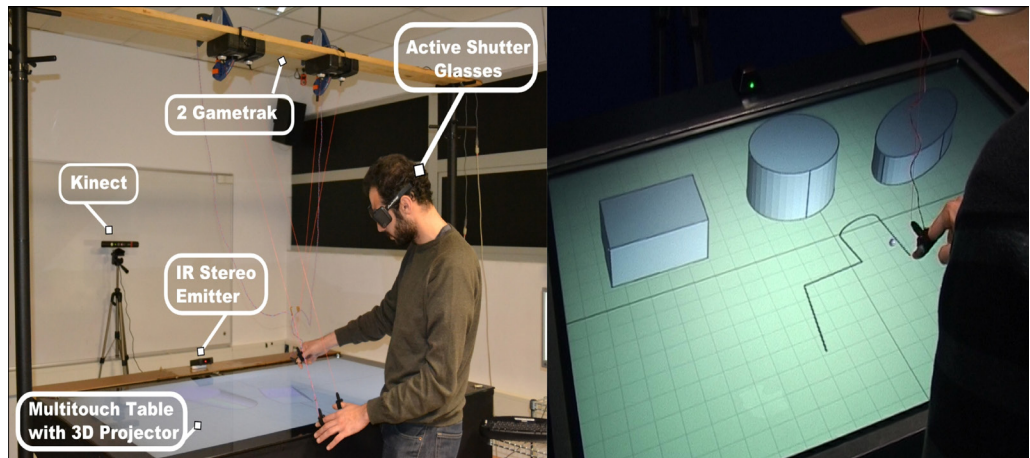
depth camera to capture the user skeleton and thus identify which hand is being used for what. Precision tracking of user's fingers allows direct manipulation operations to be issued by the dominant hand, while the non-dominant hand operates menus saving many context changes. The tabletop stereoscopic display allows us to issue direct manipulation operations in more realistic ways. This setup allows users to both create and edit complex models with few context changes, in contrast to more conventional settings where menu invocations are frequently needed and contribute to disrupt the flow of work.

Sketch based approach

We use a multi-touch surface primarily as a sketching canvas where designers draw with their fingers as depicted by (Figure 1). Users can sketch on the surface creating planar shapes from close contours using the DH, by combining multiple lines or curves issued in one or more strokes. Open strokes whose extremities are close to each other get merged automatically. While sketching, input data are fitted incrementally to either lines or cubic Bezier curves according to best fit. Our incremental fitting algorithm tries to guarantee the continuity between curves and segments by adding tangential constraints during the fitting process. When a closed contour

is detected on the surface, our algorithm creates the enclosed simple planar polygons. We perform a simple stroke beautification based on constraints detected from sketches. These constraints rely on line segments to detect parallel and perpendicular line pairs and segment pairs with equal length via a threshold on angles between segments for parallelism and perpendicularity and a threshold ratio relationship between segments with similar length. An energy function applies to each type of constraint and we perform an error minimization method to beautify user sketches accordingly. Thanks to this process, regular shapes can be created using line drawing. Regarding closed conic sections, we use a 2D shape recognizer by Fonseca and Jorge (2000) to detect circles and ellipses which are approximated by a closed piecewise curve using four cubic Bezier segments. We also use the 2D shape recognizer to detect simple gestures such as an erase commands. When an erasing gesture is recognized, the shapes or strokes overlapped by it are erased. This solution allows using open strokes as construction lines while modeling and enables us to take advantage of sketch based modeling techniques on a stereoscopic multi-touch display.

Figure 1
Our 3D Modeling Setup on the left image and the user sketching on the multitouch surface on right.



Creating shapes by push and pull

The system identifies gestures with the DH above the surface either as 3D object creation or edition. We create 3D shapes by extruding planar surfaces previously sketched on the table following the push and pull modeling metaphor. Users select a surface using the DH index finger. They then perform a pinch gesture, pressing the button located on the index finger, to extrude the shape along the normal to the surface (Figure 2). The height of the extruded object is then continuously updated and co-located with the position of the finger until the button is released. Planar shapes can also be extruded along a trajectory defined in the air after this operation is selected from a menu using the NDH (Figure 2). While the user is defining the trajectory, the path is continuously re-evaluated and fitted to line segments and curve pieces similarly to what is done for strokes on the surface. Segments and curve pieces are then used to create a smooth free-form extrusion of the profile. Any topological feature on a shape (vertices, edges or faces) can be selected via the index finger of the DH either on the surface or on the space above the table. This is possible thanks to stereoscopic visualization since both the visualization and interaction spaces are co-located. This makes it possible to directly manipulate 3D models.

Alternatively, users can sketch 3D lines or curves in space if no geometrical feature is selected while doing the pinch gesture with the DH.

Visualizing and manipulating objects

While the DH primarily serves for creating and editing shapes, the NDH can issue object transformations if an object is selected, or manipulate the world (set of objects created) otherwise. While single touch gestures initiate translation, multiple-finger gestures are interpreted as translation, rotation and scale operations on objects or world following the well-known Rotation-Scale-Translate (RST) paradigm where 3D objects are constrained to movements along the plane parallel to the multi-touch surface. A gesture started with the NDH can be complemented by the DH allowing RST with both hands (Figure 3) first image. As we can see the bimanual asymmetric interaction model provides an implicit way to switch between editing and manipulation reducing the need to invoke menus and helping users to focus on their task. In addition, this model is applied continuously on either the surface or the space above allowing defining translations, rotations and scaling seamlessly. These direct 3D object manipulations appear much more efficient when compared to indirect interactions on the multi-touch surface

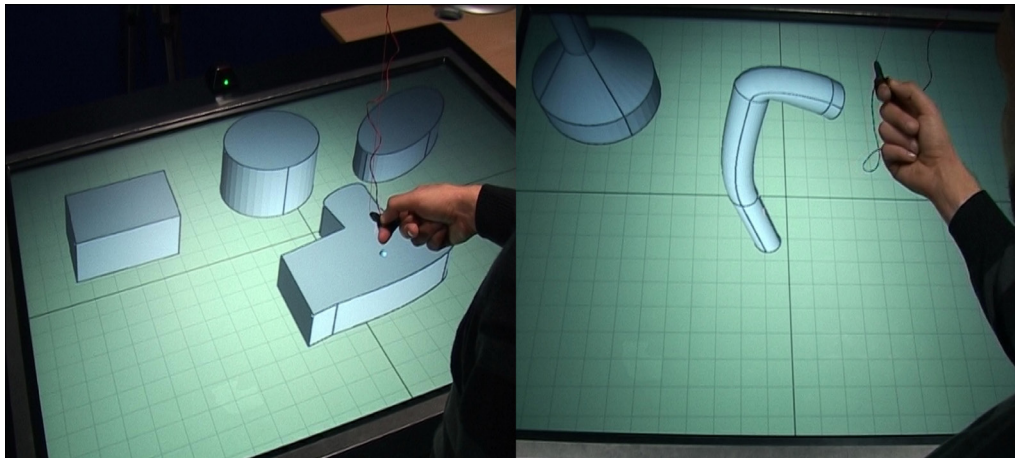


Figure 2
Extruding a Planar Shape (left image) and Curvilinear Extrusion using 3D gestures above the surface (right image).

alone and allow stacking objects over other shapes or to better exploring a 3D model as depicted by second image (Figure 3).

GUI based operations

We still rely on menu-based commands to distinguish between modeling modes such as linear and curvilinear extrusion or to provide additional operations such as shape copying. To this end we use contextual menus presented under the NDH after a shape or a part of it are selected with the DH. The operations presented in the contextual menu correspond to those available in the current mode associated to the operation performed by the DH (Figure 3) third image. For example if an edge is selected only moving will be possible, if a planar face is selected extrusions (along the normal or along a trajectory) and face snapping become available. Finally if a shape is selected highlighting its bounding box cloning the complete shape will be made available. To avoid visual clutter, the contextual menu transparency is adjusted based on the distance between the NDH and the surface. Above 15 cm, the menu is fully transparent and becomes progressively opaque as the NDH approaches the surface. To improve the accessibility, the contextual menu follows the NDH but its location is progressively fixed as the NDH comes closer to the surface to avoid spatial in-

stabilities and reduce errors while selecting an item. Since the menu follows the position of the NDH, it can be used to define the location where clones appear when the cloning operation is selected by the user. The cloning is available when any shape is selected and it duplicates the entire shape as illustrated in (Figure 3) fourth image.

Switching between 2D and 3D interaction space

Creating 3D planar shapes in space remains an operation difficult to perform due to lack of physical constraints to guide the hand. We propose a snapping operator to easily switch between the surface and space allowing using sketches on the surface or gestures in 3D space at convenience. Snapping is available through the contextual menu accessible on the NDH to snap on or back on any selected face (Figure 4) first image. It works by computing a transformation matrix to align the 3D scene to the visible grid defined as a representation of the table surface. A simple linear animation between the two orientations is rendered to help the user understand the new orientation of the model. Furthermore, it allows sketching details on existing shapes [Figure 4] second image or guaranteeing that new shapes are created on top of an existing shape.

Figure 3
Moving Objects, Rotating and Scaling in Space, Contextual Menu while selecting, Copying Objects using Menus.

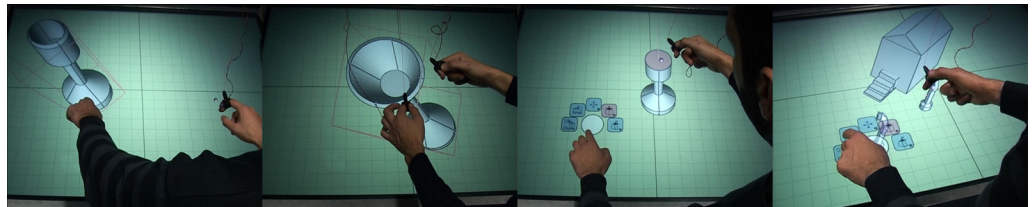
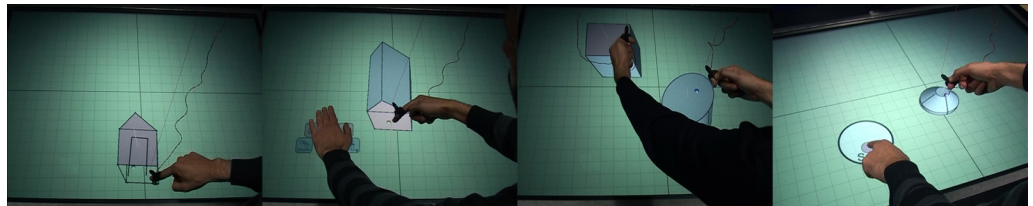


Figure 4
Snapping a face and Rotated Mode, Constraining the height of an extrusion, Scaling the profile while Extruding.



Constraining 3D editing

Since most of 3D editing operations are performed using only the DH, we decided to use the free NDH to enrich our 3D operators and constrain both sketching and 3D modeling to create more rigorous and controlled shapes. The simplest constrained operation allows sketching symmetrical shapes on the surface. First, the user sketches a straight line defining a mirroring plane which can be selected by touching it with the NDH. While the mirroring plane is selected, sketches using the DH are automatically mirrored and are considered as additional strokes if the selection remains active at the end of the sketch. By creating a closed shape formed by a stroke and its mirrored version, users can create symmetrical shapes. It can also be used to add symmetrical details to an existing stroke or shape. 3D operations above the surface can also be constrained. For example, while an object is being extruded with the DH, the NDH can select a face of an object to define a maximum or minimum height constraint [Figure 4] third image. Once the constraint is defined, the user continues to move his DH until the maximum or minimum height is reached. Further movements along the preceding direction do not continue to update the height of the object. This allows the user to also define that the height of an object should not be higher or lower than the height of another object. Finally the extrusion can be also constrained allowing scaling the profile while extruding a shape [Figure 4] fourth image. This allows the creation of

a cone or a frustum from a circle or a quadrilateral planar face respectively. The scaling factor can be controlled dynamically using a 2D overlay menu accessible by the NDH while extruding the shape. Doing so, we take advantage of the increase of expressiveness provided by bimanual interaction techniques. Furthermore, we hypothesize that this definition of constraints on the fly allows to improve the flow of interaction and better fits constraint based modeling in conceptual design stages.

A PROCEDURAL INTERACTIVE EDITOR

While the previous approach focuses on sketching and 3D gestures, we propose an additional mode accessible through the usage of a tracked Nintendo Wiimote device to allow interactive procedural editing in the same environment. The main advantage of relying on procedural modeling is to provide meaningful editing operations such as resizing the height of a building which will add automatically new floors or easing the resize of a floor adding new windows automatically. We choose to support the CGA shape grammar description supported by ESRI CityEngine [2] in our application. We enable the user to load previously created rule based grammars using textual description into our application as depicted by [Figure 5]. These procedural models can then be loaded into our stereoscopic multitouch environment and manipulated mimicking physical mock-ups lying on top of the table. [Figure 5] presents the user interacting with a tracked Nintendo

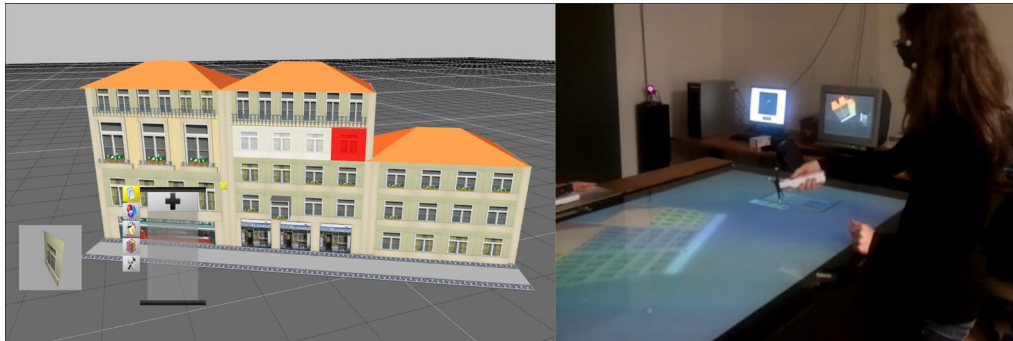


Figure 5
Procedural generated 3D
model example (left image)
in our Interactive proce-
dural setup using the Wiimote
(right).

Wiimote device selecting and editing 3D objects taking advantage of the procedural description without having to deal with scripting which is the common approach.

Interactive selection of procedural models

Procedural models are generated starting from a planar shapes and applying rules from a grammar using an inference method. This computed generated method defines a hierarchical representation of the model where each level corresponds to the application of a given rule. Editing of procedural models is done changing rule attributes i.e. its precondition, the sequence of rules or attributes of operator used by the rule description. This methodology requires the user to identify which rule needs to be change in order to perform a modeling change on the 3D model. Instead of relying on textual description, our approach enables the user to select rules and perform changes interacting directly with the 3D model through direct manipulation on the virtual geometric representation presented on the table. The tracked Wiimote device is used as a virtual wand to select part of the model and navigate through model hierarchy using the buttons of the device.

Two selection mechanisms are provided to the user. The first is a Hierarchical Selection (Figure 6 right) which enables to navigate through the model hierarchy selecting parent nodes or children nodes from a given selected part of the 3D model (using the “UP” and “DOWN” arrow Wiimote buttons) . The second method is a Semantic Selection [Figure 6] left which allows selecting sibling nodes from the same hierarchical level, i.e. corresponding to nodes with the same semantic or using the same rule (using the “LEFT” and “RIGHT” arrow buttons).The “A” button allows confirming the selection defining the primary selection (red color) of a given modeling operation. Selecting a different part of the model allows defining a secondary selection (yellow color) if needed depending of the operation. Pressing the button “A” using the virtual hand on an empty space i.e. not intersecting any part of the model, will deactivate the primary selection if it is the only one or the secondary selection if it exists.

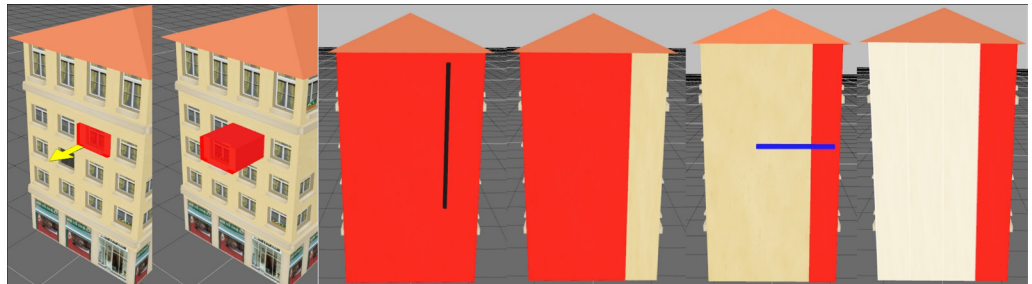
Interactive procedural editing

Depending of the selection, several editing operations can be done. Users can edit the grammar creating model complex grammars applying grammar

Figure 6
Hierarchical selection and semantic selection.



Figure 7
Editing operation examples: extrusion, split and repeat.



operations which will generate new shapes. [Figure 7] presents the three operators which can be done on an active selection using the virtual wand and Wiimote buttons. Pushing or Pulling a face will perform an extrusion, drawing a line on face will perform a split operation. Finally crossing the limit of a shape drawing a line will repeat a split pattern subdividing a shape. Regarding attribute editing of shapes illustrated by (Figure 8), we combined the wand interaction with menus projected on the multi-touch surface. The menu can be invoked using a triangular shape gesture by multitouch on the surface. Its content is adapted to the active selection allowing the user to change colors or reuse textures from a collection. It is also possible to copy an active selection into a clipboard like area of the menu. The clipboard will store the sequence of rules uses by the selection. Previous grammar templates can be also selected on the menu and apply to selected shapes. Both templates and clipboard rules are presented using a snapshot of the generated 3D model. When selected over a face, the subset of rules is applied to the selected part of the model generating new geometry. Doing so, the user is able to edit the procedural model without having to rely on textual

description of rules. The selection provides a direct access to the grammar definition interacting directly with the 3D model on our stereoscopic environment using the Wiimote Virtual Wand.

RESULTS

Our prototype was implemented to demonstrate our modeling approach in C++ using OpenGL and OpenSG for stereoscopic visualization. We used an Intel I7 920 2.67 GHz processor with 3 Gb of memory RAM and an NVidia Quadro 4000 graphics card running Microsoft Windows 7 64-bit operating system. Regarding the sketching and gesture interface for creation and edition are following the push and pull modeling metaphor, it was informally tested to assess the different design choices and iteratively improve the design of the interface. In total, 20 undergraduate and graduate students in Computer Science with variable experience with CAD applications tested the application. Our observations suggest they quickly learned how to use the interface. However, we noticed that some confused the usage of the two hands at the beginning: sometimes participants wanted to move an object using their dominant hand. While sketching is beneficial to surface-based interactions, beautification is a must to create more rigorous shapes for manufacturable objects.

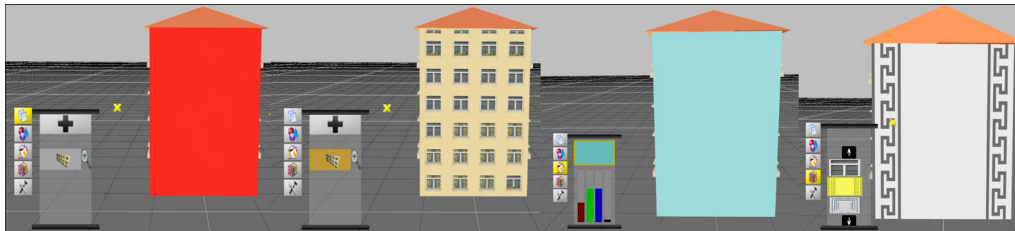


Figure 8
Editing shape attributes:
reusing template models,
changing colors, assigning
textures.



Figure 9
Examples using our system
and a traditional CAD.

Snapping provides a way to transition between the two. Moreover, bimanual asymmetric interaction provides an implicit switch between modeling and manipulation, letting the user focus on his design tasks.

[Figure 9] presents different models built using the interface by an expert user. The second model from the left was built in 5'20" while the fourth took one of us 2'45" to complete. An expert user took 5'41" and 3'46" respectively for the same models using Rhino3D; these are rendered in Figure 12 using a white background. More rigorous tests should yield more exact measures, while direct editing of curves should reduce retrials.

CONCLUSION

We have described an approach to model 3D scenes using semi-immersive virtual environments through a synergistic combination of natural modalities afforded by novel input devices. While early experiments and informal assessments of our system show promise and seemingly validate some of these assumptions, we plan to run formal evaluations with both novice and expert users to highlight and explore both the strengths and the weakness of our modeling interface. On a positive note, the continuous interaction provides plausible gestures for extrusion and easy to define 3D trajectories leveraging the best features of the surface and space above it. Thanks to stereoscopic visualization, both modeling modes provide co-location between user hands and virtual objects adapted to direct modeling methods and offering a direct access to procedural rules while interacting with the virtual model. We are considering further exploring multiple fingers tracking as an alternative using non ambiguous start and end gestures.

ACKNOWLEDGEMENTS

We would like to thank Luis Lopes for implementing the interactive procedural modeler. This work was funded by Fundação para a Ciência e Tecnologia through "Digital Alberti" project (PTDC/AUR/

AQI/108274/2008), INESC-ID multiannual funding through PIDDAC Program funds and project PTDC/EIAEIA/104031/2008 (MIVIS). Bruno Araújo was supported by doctoral grant SFRH/BD/31020/2006

REFERENCES

- Fonseca, M and Jorge, J 2000, Using fuzzy logic to recognize geometric shapes interactively, *Proceedings of IEEE Fuzzy Systems 2000*, vol. 1 ,pp 291-296.
- Lipp, M, Wonka, P, Wimmer, M 2008, Interactive visual editing of grammars for procedural architecture, *Transactions on Graphics*, 27(3):102:1-10.
- Guiard, Y 1987, Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model, *Journal of Motor Behavior*, 1987, 19, 486-517.
- Marquardt, N, Jota, R, Greenberg, S, and Jorge, J 2011, The continuous interaction space: interaction techniques unifying touch and gesture on and above a digital surface, *Proceedings of INTERACT'11*, Springer-Verlag, Berlin, Heidelberg, pp. 461-476.
- Mueller-Tomfelde, C, Hilliges, O, Butz, A, Izadi, S, and Wilson, A 2010, Decision Interaction on the tabletop: Bringing the physical to the digital, *Mueller-Tomfelde (eds), Human-Computer Interaction Series*, Springer London, pp. 189-221.
- Müller, P, Wonka, P, Haegler, S, Ulmer, A and Van Gool, L 2006, Procedural modeling of buildings, *ACM Transactions on Graphics*, 25(3), pp 614-623.
- Oh, J-Y , Stuerzlinger, W and Danahy, J 2006, Sesame: towards better 3d conceptual design systems, *Proceedings of the 6th conference on Designing Interactive systems*, DIS '06, pages 80-89, NY, USA.
- Olsen, L, Samavati, F, Sousa, M, and Jorge, J 2009, Technical section: Sketch-based modeling: A survey, *Computers & Graphics*, 33,;pp. 85-103.
- Parish, Y and Mueller, P 2001, Procedural modeling of cities, *ACM SIGGRAPH '01*, pages 301-308, New York, NY, USA, ACM Press.
- Wilson, A and Benko, H 2010, Combining multiple depth cameras and projectors for interactions on, above and between surfaces, *Proceedings of UIST'10*, ACM Press, New York, NY, USA , pp. 273-282.
- Wonka, P, Wimmer, M, Sillion, F, and Ribarsky, W 2003, Instant architecture, *ACM SIGGRAPH 2003 Papers*, pages

669-677, New York, NY,USA, ACM.

- [1] Google Sketchup <http://sketchup.google.com/> (retrieved June 2012)
- [2] Esri CityEngine 2012, <http://www.esri.com/software/cityengine/index.html>
- [3] Microsoft Kinect, 2011, <http://www.xbox.com/kinect>.
- [4] Gametrack, 2004 <http://en.wikipedia.org/wiki/Gametrak> (retrieved June 2012)
- [5] Nintendo 2010 Wii remote <http://www.nintendo.com/wii> (retrieved June 2012)