

# A Multi-touch Interface for Fast Architectural Sketching and Massing

Qian Sun\* Juncong Lin† Chi-Wing Fu\* Sawako Kaijima‡ Ying He\*

\*Nanyang Technological University †Xiamen University ‡Singapore University of Technology and Design

## ABSTRACT

Architectural sketching and massing are used by designers to analyze and explore the design space of buildings. This paper describes a novel multi-touch interface for fast architectural sketching and massing of tall buildings. It incorporates a family of multi-touch gestures, enabling one to quickly sketch the 2D contour of a base floor plan and extrude it to model a building with multi-floor structures. Further, it provides a set of gestures to users: select and edit a range of floors; scale contours of a building; copy, paste, and rotate a building, i.e., create a twisted structure; edit profile curves of a building's profile; and collapse and remove a selected range of floors. The multi-touch system also allows users to apply textures or geometric facades to the building, and to compare different designs side-by-side. To guide the design process, we describe interactions with a domain expert, a practicing architect. The final interface is evaluated by architects and students in an architecture department, which demonstrates that the system allows rapid conceptual design and massing of novel multi-story building structures.

## Author Keywords

Building design; multi-touch; sketching; massing

## ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: User interfaces

## INTRODUCTION

Building design is an event per excellence, involving complex considerations and requiring designers to explore a vast problem space. Such complexity tends to be beyond our perceptual capabilities, demanding multiple trial and error, and the decomposition of the problems through formal derivations. This is a rather time-consuming process and the exploration is usually limited by time and budget.

Sketching and massing are two primary design techniques widely used in the early stages of building design [10, 1]: *sketching* is a common design method used for generating

ideas and exploring options with pen and paper, while *massing* is a primary design process normally used for understanding the overall shape and size of a building in relation to its surrounding environment with approximate volumetric information. During the early stages of design, architects explore multiple scenarios to clarify their design ideas and to arrive at a coherent building solution. Sketching and massing are not merely a presentation media, but are also effective approaches that aid designer's engagement with the spatial cognition process [22]. Hence, the functionality and the quality of design media can greatly affect the development of the design.

In the early design stages, a fast feedback loop between design actions and visualization helps architects to develop their design, and facilitate communication among design teams and clients. Although there are many off-the-shelf commercial software for 3D modeling, e.g., Maya, 3Ds Max, AutoCAD, etc., most of them are exceedingly powerful and comprehensive, and are targeted mainly at general 3D modeling and highly-experienced/well-trained users, e.g., artists, engineers, designers, etc. Conversely, fast prototyping tools such as Google SketchUp are easy-to-learn and intuitive-to-use for both experts and novice users, allowing a quick mock-up of 3D prototypic structures. However, existing prototyping tools still focus mostly on general 3D shapes.

This paper is a result of a collaborative work with a domain expert who is an architect. We develop a novel, multi-touch interface design for supporting the early stages of building design, aiming to combine aspects of sketching and massing to aid the design process. In particular, we focus on the design of tall buildings due to the scale and complexity involved in such design. Moreover, we can observe the trend in tall buildings to be more expressive in its silhouette due to the level of maturity we have reached using modern construction techniques

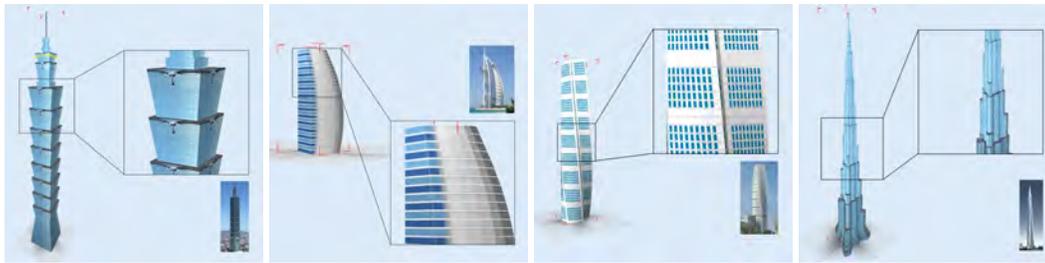


**Figure 1:** Buildings with progressively-varying and twisted structures: Burj Al Arab in Dubai, Taipei 101 in Taipei, Turning Torso in Sweden, and The Shard in London (left to right).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI'13, April 27–May 2, 2013, Paris, France.

Copyright 2013 ACM 978-1-4503-1899-0...\$10.00.



(a) Taipei 101 (183s) (b) Burj Al Arab (67s) (c) Turning torso (86s) (d) Burj Khalifa (273s)

**Figure 2:** An architect constructed prototypic models of famous buildings by using our multi-touch interface. The time taken to create these 3D building prototypes is shown below each model.

(see Figure 1). We propose a family of multi-touch gestures, where one can start by sketching and editing the 2D contour of the base floor, and then extend and edit it to create a 3D building model with fairly complex geometric expression in a matter of minutes (see Figure 2). The simple drag-and-drop texturing can further enrich the model’s appearance. In addition to 3D modeling, our interface computes and provides assorted architectural parameters for the building design, e.g., its dimensions, facade area, and volume, thereby providing useful hints for guiding the design process.

Compared to the conventional keyboard-mouse interface, our interface incorporates *hand-oriented actions* such as sketching, slicing, curving, and twisting, which allow one to quickly define the geometry and manipulate its volume (massing). Moreover, we also develop a number of novel interface tools for fast geometric modeling (such as symmetry snapping and easy texturing of image and facade geometry) on the building model, and various editing operations. Thus, our interface can contribute to the conceptual phase of contemporary building design, where building designers and architects can apply it to do quick mockups, try alternative ideas, and get fast feedback to support major design decisions. Moreover, our proposed interface can serve as a rapid 3D modeling tool for creating content in digital media and games, where we need buildings with good-looking shapes and appearance rather than a high level of detail. Such rapid 3D building prototyping is not easily achievable with conventional multi-touch 3D modeling interfaces, existing 3D sketching tools, and general 3D modeling software.

To the best of our knowledge, this is the first work we aware of to explore and develop multi-touch interactions for rapid creation and editing of tall buildings with non-trivial geometry. Here we summarize its technical contributions:

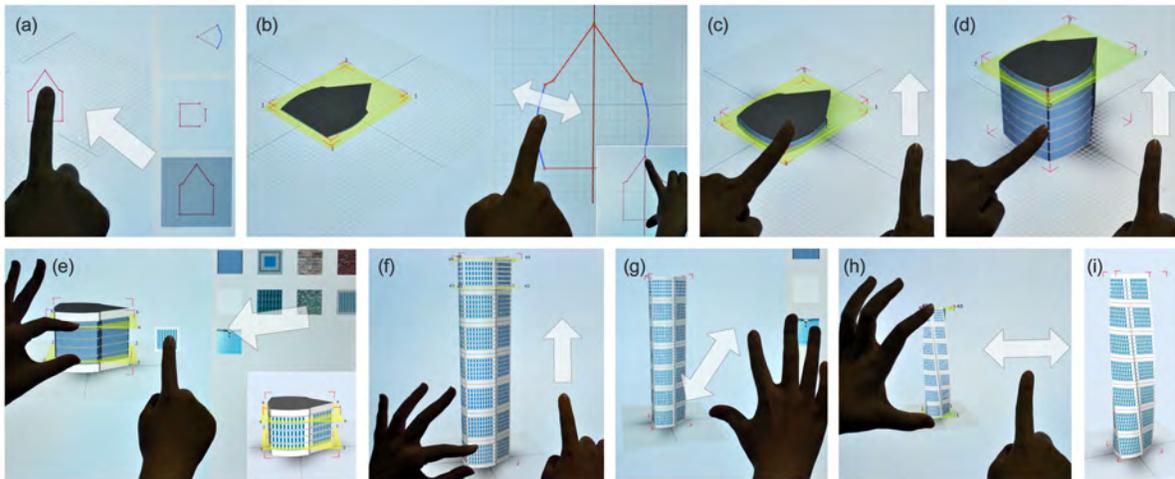
- We propose a novel multi-touch design interface, facilitating rapid creation of non-trivial contemporary buildings through stage-by-stage editing of floor, structure, and appearance. Our interface is not limited to modeling 3D buildings; it can be adopted to other types of 3D models with repeated geometric features, rotational/planar symmetry, or curved profiling.
- We propose an effective symmetry snapping method, allowing us to keep the local symmetry when editing 2D floor layouts with multi-touch. It is especially useful with imprecise finger interaction, and can be applied for editing 2D patterns and 3D models with symmetry.

- As suggested by experienced architects, we propose a set of novel multi-touch gestures; some are particularly useful for massing, e.g., multi-touch twist for progressively-varying floor structures, crop for quick editing and testing of building designs (to meet the sunlight regulations prohibiting casting shadows over neighboring buildings), etc.

## RELATED WORK

The ability to collect user input simultaneously from multiple fingers can greatly enrich our interaction. Igarashi et al. [14] proposed scenarios for how multi-touch can be used to manipulate and deform complex 2D shapes. Moscovich and Hughes [18] devised the multi-finger cursor technique, offering smoother interaction with graphical displays. Benko et al. [5] developed a new technique to use multi-touch to improve precision when selecting small targets on a touch screen. Bau et al. [4] enhanced touch interfaces by a new tactile method that used the electro vibration principle to control the electrostatic friction between the finger and surface.

Recently, a number of research works have focused on studying the use of multi-touch to manipulate 3D contents. Grossman et al. [11] presented a family of multi-touch gestures to manipulate 3D wireframe objects over a spherical volumetric display. Later, Hancock et al. [12, 13] proposed a multi-touch concept called *Sticky Fingers* to grab and manipulate 3D objects on a tabletop. Reisman et al. [20] formulated a screen-space method, allowing users to directly rotate-scale-translate 3D objects with three or more contact points. Kim et al. [15] designed multi-touch interactions on an iPhone/iPod Touch to control walking in a 3D virtual environment. Chang et al. [6] presented an interesting origami modeling interface by proposing a family of specially-designed multi-touch gestures. Fu et al. [9] proposed multi-touch gestures for controlling navigation in a large-scale astrophysical simulation shown on a wall display. Cohé et al. [8] adopted the use of 3D transformation widgets to transform 3D objects with multi-touch. Kin et al. [16] proposed multi-touch gestures for positioning, arranging, and scaling 3D objects in virtual organic environments, while Chiu et al. [7] explored document folding actions and proposed practical multi-touch gestures for various manipulation scenarios. Compared to the above work, which mainly focuses on the manipulation of 3D contents/views or navigation in virtual space, we aim to create a fast architectural sketching and massing tool for designing tall buildings. Our interface offers a rich set of novel multi-touch gestures dedicated to efficiently designing a non-trivial



**Figure 3:** System overview. Here we show our design pipeline with the Turning Torso building. The user starts by creating a 2D floor plan, which can either be selected from the tabbed panel (see (a)) or sketched by the user. After that, the user can freely edit the 2D floor contour using the 2D floor plan editor by manipulating the control points on the 2D contour. Our system provides multi-touch gestures and a novel grid&symmetry snapping mechanism for users to edit the 2D floor contour (see (b)). The inset for the two-finger anchor gesture specifies the symmetry axis. Later, the user employs one finger to select the floor plan and create the first ground-level floor (see (c)). After that, the user can create multiple floors by duplicating the ground floor (see (d)). Then, the user can apply an image texture to the selected floors by a simple drag-and-drop (see (e)). Our interface also allows the user to easily copy and paste selected floors to extend the building levels (see (f)), and zoom out using a two-finger pinch (see (g)). Finally, the user can select the whole building and use a one-finger horizontal drag to twist the building structure (see (h)) to produce the final result (see (i)).

geometry (sketching) and manipulating its volume (massing) in a matter of minutes, which has great potential for the conceptual phase of contemporary building design.

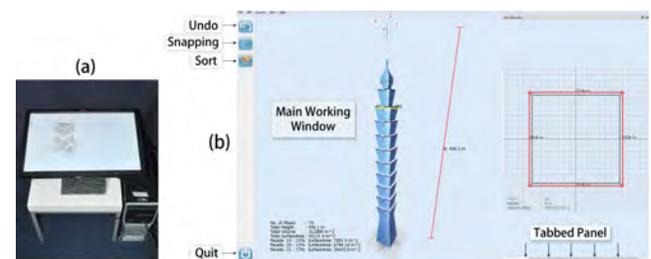
MozArt [21] is a multi-modal interface that supports speech and gesture input to Google SketchUp. Unlike MozArt, our interface does not rely on voice input, and supports a much wider range of modeling functions, allowing the users to make non-trivial shapes. TreeSketch [17] is a system for modeling complex, natural-looking trees using Palubicki et al.'s algorithm [19]. Taking advantages of a multi-touch tablet, the user can directly grow a tree structure by using a few sketches to intuitively specify the direction of the trunk and branches. All the other controls, such as changing parameters as the tree grows, interleaving controlled and autonomous growth, and editing generated forms, are still based on the conventional interface with menus, dialogs, buttons, and scrollbar. In contrast, users of our system can make a complete 3D building by multi-touch gestures alone. Bae et al. [2] presented ILoveSketch, a pen-gesture-based 3D curve sketching system, which enables professional designers to efficiently create conceptual 3D curve models. Later, they improved the sketching system by incorporating many new interactions [3]. Unlike their pen-based gesture design concept, our system offers a rich set of direct finger gestures, and allows rapid design of non-trivial modern buildings. In addition, the output of our system can be directly imported into general modeling software such as Autodesk and 3Ds Max.

## OVERVIEW

This section introduces our multi-touch user interface, and then presents its usage scenario.

**User Interface.** Our multi-touch interface consists of a main working window and a floating-tabbed panel. These are shown on the left and right sides of Figure 4(b), respectively. The main window has three buttons on the left, allowing the user to undo/redo a sequence of operations, enable/disable control point snapping when editing a 2D floor plan, and bring out different designs for comparison. Note that the current GUI layout is mainly for right-handed users, but we may re-arrange the floating tabbed panel to the left and adapt the interface for left-handed users.

The main working window shows a perspective view of the current 3D building design. See again Figure 4(b). In this window, the user can perform basic multi-touch manipulation to control the camera view as well as apply various multi-touch gestures to edit and decorate the design. The floating-tabbed panel consists of a number of windows: 1) a floor



**Figure 4:** (a) Our multi-touch display system; (b) Our GUI interface consists of a main working window (left) that presents the building design in 3D and a floating tabbed panel (right) that offers a 2D floor plan editor, a building profile editor, and various resource libraries.

plan editor (see Figure 4(b)) for one to view and edit the 2D contour of the selected floor, 2) a building profile editor for editing the profile curve, and 3) resource libraries for one to retrieve and apply pre-defined floor plans, walls, window images, and geometric facades by simple drag-and-drop.

**Usage Scenario.** There are three major stages to mock up a 3D building design, as illustrated in Figure 3(a)-(i):

1. **Initial Design.** First, we begin by constructing the 2D contour of the ground level floor. The user can either pick an existing floor plan from the related resource library (see (a)) and/or sketch&edit his/her own contour using the 2D floor plan editor (see (b)). Then, the user drops the floor plan into the main working window, creates a single floor (see (c)), adjusts its height, and then duplicates it to create many other floors and form an initial design (see (d)).
2. **Structure Editing.** After that, we enter the structure editing stage, where the user edits the building structure by various multi-touch gestures. Here, the user applies various multi-touch gestures (e.g., copying (see (f)), scaling (see (g)), twisting (see (h&i)), and slicing, etc) to edit the entire building or a range of selected floors. Various building parameters including the facade area and volume are recomputed and presented interactively to guide the design.
3. **Appearance Enhancement.** Lastly, the user decorates a building design by dragging and dropping image textures or geometric facades (see (e)) from the library resources.

Note that we group the interactive operations into these three stages to facilitate our understanding of design strategy, but in practice, one can perform them in any order as long as the operation being performed is appropriate to the current condition of the building design. For example, we can drag-and-drop image texture (see Figure 3(e)) onto the initial ground floor before the structure editing stage. As shown in Figure 3, a user can rapidly create a 3D building design with non-trivial geometry in a matter of minutes. However, such a task could be very tedious and time consuming, even when performed by experienced users with general 3D modeling software.

Besides the above major stages, the user can also record multiple building designs and compare them side by side visually in the main working window with their key architectural parameters. Moreover, the user can load a surrounding environment, put the current building design into it, and examine its relation to the surroundings. These features, as suggested also by an architect, allow one to clarify the design and to arrive at a more coherent building solution.

## DESIGN CONSIDERATIONS

Our system is implemented on a 3M M2256PW (22") multi-touch display system (see Figure 4(a)). We tilted the display to a near horizontal orientation, which allowed the users' elbows to rest on the display border during the interaction. In the followings, we present the interface design considerations that guide and direct our user interaction design:

**1) Sufficient modeling capability.** The interface should provide sufficient modeling capability for one to design contemporary buildings, so that we can create different forms of

structures and shapes that can be found in various existing buildings, see Figure 1, e.g.,

- Curved profile, see *Burj Al Arab*, Dubai;
- Disrupted floor plans, see *Taipei 101*, Taiwan;
- Twisted structure, see *Turning Torso*, Sweden;
- Symmetric floor plans, see *Burj Al Arab*, Dubai;
- Slanted side, see *The Shard*, London.

Regarding this, we surveyed a wide range of contemporary buildings and examined their shapes and structures before formulating the interaction design.

**2) Fast architectural sketching and massing.** The interface should support the rapid design of tall buildings. Regarding this, we identified three major stages needed in the overall building prototyping process: *initial building*, *structure editing*, and *appearance enhancement*, as well as assorted modeling and editing operations. The interface also allows the user to compare all the designs side by side and put a design into the target surrounding environment, so that the user can easily identify both the merits and defects, and make further improvements if necessary. Besides the editing capability, the interface also shows the key building parameters, such as dimensions, facade area, and volume, to help guide the design.

**3) Easy-to-learn gestures.** As a fast prototyping interface, the gestures should also be easy to learn, so that users can quickly understand the interface and perform the necessary operations to rapidly develop and evolve their designs. Regarding this, we organize relevant interactions by their types and operational stages to facilitate a consistent sequence of interaction. As users usually prefer a small number of gestures, it is highly desirable to re-use the gestures (especially the standard ones) in different scenarios without ambiguity. Furthermore, the most frequently used gestures should be designed to be simple to perform.

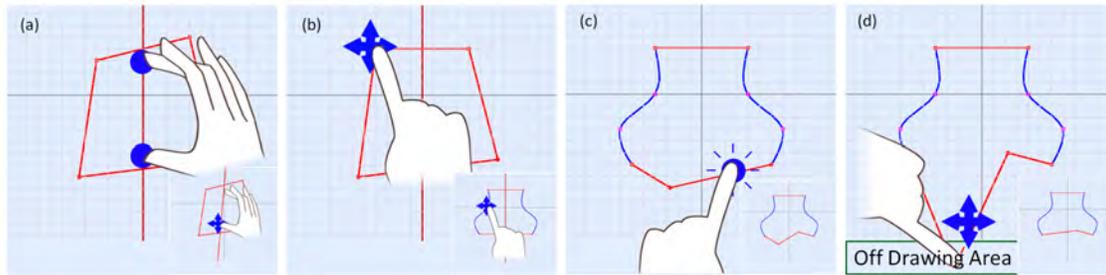
**4) Taking advantage of direct manipulation.** Structure editing is usually the most critical and challenging part of the whole design process. To enable the users to focus on this task, we provide them with a direct-manipulation interface, where all structure editing operations, e.g., viewpoint navigation, floor selecting, and facade editing, can be done with multi-touch gestures. By this, the users can fully focus their attention on the main working window.

## MULTI-TOUCH GESTURES

This section presents our interaction design and the related implementation. Our design includes four sets of multi-touch gestures organized as follows:

### 1) Initial Building

In the process of prototyping a building, our approach is to first create an initial 3D building design and then further refine it by editing its 3D structure (next stage: *structure editing*) and enhancing its appearance (another stage: *appearance enhancement*). To sketch an initial design, we basically need to create a floor plan and then extrude it to form an initial building. In summary, we propose the following operations relevant to this stage:



**Figure 5:** Multi-touch gestures for editing 2D floor plan contour. (a) Two fingers to specify the axis for symmetry snapping; while we hold the fingers on screen, we can still move them to adjust the symmetry axis (see the inset); (b) We can freely move a control point with a finger and it could snap to a mirror location of a point on the other side of the symmetry axis; see the inset for making a symmetric shape; (c) To add a control point, we can double-tap on a line/curve segment. The inset shows the result; (d) To remove a control point, we can drag it off the drawing area.

**ii) Sketching and editing a floor plan.** In our system, a floor plan is represented by a 2D floor plan contour, or simply a floor contour, which is a closed curve made up of polylines and spline curve segments (we use interpolatory splines). Hence, we can create non-trivial shapes for floor plans, see Figure 3. In detail, the user can pick a built-in floor plan from the library resource, or sketch a closed curve as a floor contour by free-style sketching using the floor plan editor. With the editor, the user can directly use fingers to manipulate the control points on the floor contour to adjust the contour shape, see Figure 3(b). Furthermore, double-tapping on a line segment can create a new control point for refining the contour shape, see Figure 5(c); double-tapping on an existing control point can toggle its status between straight and curved; while dragging a control point off the drawing area can remove it, see Figure 5(d). Lastly, the user can also store an edited 2D floor plan in the library resource as a future asset.

**ii) Grid snapping.** To improve precision in editing the floor contours, our interface provides a grid snapping function that allows the user to precisely snap and put a control point on grid points. This function is enabled by default but can be toggled by the snapping button on the left hand side of the interface, see Figure 4(b).

**iii) Symmetry snapping.** Like most everyday objects, contemporary buildings usually contain strong symmetric structures. To improve the editing efficiency, we allow users to preserve the symmetry (local or non-local) of floor plans during the multi-touch interaction by our novel multi-touch technique: *symmetry snapping*. The user can start with a *two-finger anchor gesture* on the display - holding two fingers on the floor plan editor for two seconds to create a straight line, see the inset in Figure 5(a). Once a symmetrical line is formed, the user can fine-tune its position (possibly snapping its control points to the grid also) by moving the two fingers on the screen. After that, symmetry snapping is enabled with the constructed line serving as the symmetry axis:

- When the user manipulates a control point, he/she can move it close to the mirror image position of points on the other side of the axis. After releasing his/her finger, our interface snaps the control point to a mirror image position.
- Furthermore, once two points are snapped together, the two points are said to be binded, and can be moved together to

maintain the local symmetry, see Figure 5(b). If one wants to break this symmetry, one just needs to use another finger to hold one of the two points and move the other. This is a novel and efficient technique for creating and editing symmetric patterns in multi-touch manipulation - a process that usually involves imprecise finger movements.

Lastly, to ensure consistency in successive symmetry snapping, our interface remembers all symmetry axes created so far per floor plan. Thus, if a new axis is found to be close to a previous axis, our interface will snap it to the previous axis. Finally, dragging the symmetry axis out of the main working window can take it out and disable symmetry snapping.

**iv) Initial building from a floor plan.** Once an edited floor plan is ready in the floor plan editor or we find a desired floor plan in the library resource, we can drag and drop the floor plan to the main working window with a single finger. After that, we can use one finger to touch and select a floor plan, and then use another finger to drag up to control the height of this initial ground floor. The floor height will be used for every single floor in the entire building design unless the user rescales some floor heights later. After we have created the first floor, we can touch the resulting floor to select it and then apply the copy gesture (to be described later) to duplicate this floor to form all other floors above it, see detail in the next subsection. Finally, an initial building can be readily modified by the structure editing operations in the next stage.

## 2) Structure Editing

Our interface provides a family of multi-touch gestures, allowing one to quickly edit and mock up a design that mimics a contemporary building structure, see previous section. In short, one can first select a range of floors (or a particular floor) to be edited, and then perform the relevant gestures to modify the structure/geometry/appearance of the selected range of floors. Typically, the user of this approach will first employ the fingers of one hand to make the selection, and will then use the fingers of the other hand to do the editing. If no floors are selected, the editing applies to the entire building.

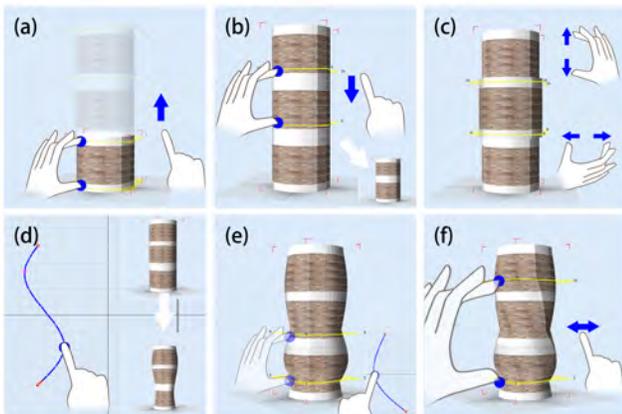
**i) Floor selector.** Whenever the user puts a finger on the building, a floor selector is created for the associated floor being touched, and is presented as a semi-transparent yellow thin-plate tagged with a floor number, see Figure 3 (b,c&d).

Now, the user can use his/her finger to drag the selector up/down to choose a particular floor. To select a range of floors, he/she can use two fingers to touch two different floor levels. In this way, a pair of floor selectors can be created simultaneously with all floors in-between (inclusively) selected, see Figure 3(e). Once a floor selector is created and shown, we can still touch it and drag it to a different floor level on the building. To de-select the floor(s), the user can simply tap on a screen region outside the building. If there are no selected floors, the entire building is considered to be selected and the user edits will be applied to the entire building. Using this selection mechanism, the user can hold down the floor selection, allowing us to disambiguate the reuse of drag and pinch for object manipulation in the same window.

**ii) Copy floors.** To facilitate the modeling of tall buildings, we offer a simple gesture to copy blocks of floors. After selecting a single floor(or a range of floors) and holding down the selection, one can apply a one-finger vertical drag up to copy the selected floor(s) above the selected floors. The number of copies is proportional to the vertical distance between the initial and end touch points. Since the 3D building design can change interactively with the finger drag, the user can observe the visual changes in the prototyping design and adjust the number of copies interactively from the visual feedback, see Figure 3 (b&c) for copying a single floor and Figure 3(f) and Figure 6 for copying a range of floors.

**iii) Removing floors.** As well as creating more floors, one can also remove the selected floors by a one-finger vertical drag-down gesture, which is a reversed version of copying floors, see Figure 6(b).

**iv) Rescaling floor height.** Since rescaling floor height is a rather infrequent action, we assign a more complicated gesture for it: a two-finger vertical pinch open gesture. By repeating this two-finger pinch open gesture, one can continue to increase the height of the selected floor(s); reversing it (after a slight pinch open) with a pinch close can decrease the floor height.



**Figure 6:** Multi-touch gestures for structure editing: (a) copying floors; (b) removing floors; (c) scaling floors (heights or contours); (d) editing the building profile for the entire building; (e) editing the building profile locally on selected floors; and (f) twisting.

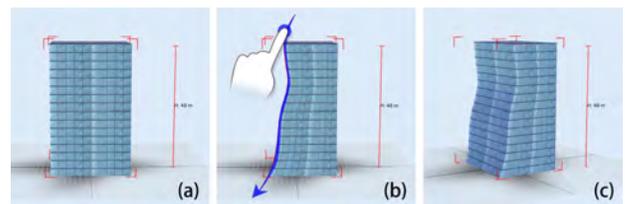
**v) Scaling floor plan.** In addition, one can uniformly scale the contours of a range of floors simultaneously to quickly try out different shapes on the building design. This can be done by a simple two-finger horizontal pinch on the screen after selecting one or more floor(s), see Figure 6(c). Note that to allow a more comfortable execution of this gesture and prevent possible strain to the wrist, we do not require the two-finger gesture to be performed horizontally to the display. As long as the pinch angle is  $45^\circ$  degrees or below on the display, we regard it as a horizontal pinch.

**vi) Twisting.** Modern structural engineering allows the construction of complicated buildings with creative profiles. One of these is the twisted structure, see again the Turning-Torso building in Figure 1. To create a design with twisted appearance, one can first select a range of floors (at least two floors), and then use a one-finger horizontal drag to control the amount of twisting, see Figure 6(f).

In our implementation, we first compute the total twisting angle relative to the finger drag displacement, and divide it by  $n - 1$  to obtain the twisting angle between successive floors, where  $n$  is the total number of selected floors. Like the copy gesture, one can again interactively perform this gesture to observe the results while moving one's finger(s). Lastly, since buildings normally cannot be twisted to a large angle due to structural constraints, we employ six degrees as the upper limit of the twisting angle between successive floors.

**vii) Editing profile curve.** Another interesting feature of contemporary buildings is the curved profile shape, such as the Burj Al Arab Hotel (see again Figure 1). We can use the building profile editor in the tabbed panel to edit a profile curve to control the amount of scaling on the selected range of floors, see Figure 6(d)-(e). As shown in the building profile editor view, such a profile curve is constrained, and must start from the top and end at the bottom, corresponding to the top most and bottom most floors currently selected. The user can interactively manipulate the control points in ways similar to those used in the floor plan editor, and can observe the visual changes on the building design as his/her fingers move.

**viii) Cropping the facade.** Lastly, our interface also offers a handy editing tool for producing a slanted side, as in the modern building *The Shard*. As illustrated in Figure 7, the user can tap four fingers on any region outside the region in the main window to enable cropping. Then he/she can sketch a freeform curve on the building silhouette and cut out a portion of the building. Note that apart from artistic concerns, a number of real building designs have to be modified by such a cropping to allow sufficient sunlight to reach surrounding buildings in accordance with some building construction reg-



**Figure 7:** Cropping the facade.

ulations. This interface feature allows us to quickly modify the building and observe its relation to the surroundings.

### 3) Appearance Enhancement

With the aforementioned gestures, the user can quickly make a building design with non-trivial geometry. However, the design lacks surface details, such as its facade. Our interface provides libraries of image textures and geometric facades for the user to quickly drag and drop to apply to the building.

The more complicated operation here is in fact the underlying implementation behind the user interaction. To map an image texture onto the building, our interface first computes the  $u$  and  $v$  texture coordinates for each vertex, where the  $v$  coordinate is simply interpolated along the height of a floor, whereas the  $u$  parameter can be found by an arc-length parameterization of each floor contour. Regarding a geometric facade, which is in fact a 3D geometric texture (e.g., windows), our interface can compute a volumetric space with the  $w$  texture coordinate in addition to  $u$  and  $v$ , where  $w$  is defined along the local normal direction on the building surface. In this way, we can appropriately map and attach a 3D geometry texture by deforming each geometric facade and fit them one by one onto the building surface like a 3D image.

### 4) Camera View

Camera view controls offer basic but essential operations for users to see their designs from different views. However, instead of allowing unrestricted control as in general 3D modeling software, we constrain the camera view to center only on locations along the building's central vertical axis from the ground to its tip. As a sketching and massing interface, this control is sufficient and brings greater efficiency since our view always focuses on the design without taking off.

In summary, our interface has the following camera-view gestures, see also Figure 8:

- First, we can rotate using a one-finger drag (left): the horizontal component of the drag rotates the building about its central vertical axis while the vertical component of the drag updates the camera's elevation angle, whose range is restricted from 0 (i.e., the camera cannot go below the ground) to 90 degrees (i.e., the camera cannot tilt beyond the building top);
- Second, a two-finger drag (middle) can move the view up and down by translating the view center (look-at) along the central vertical axis of the building;
- Last, users can also zoom in/out using a standard two-finger pinch (right), or double-tap to quickly zoom out (reset) to see the entire building design.

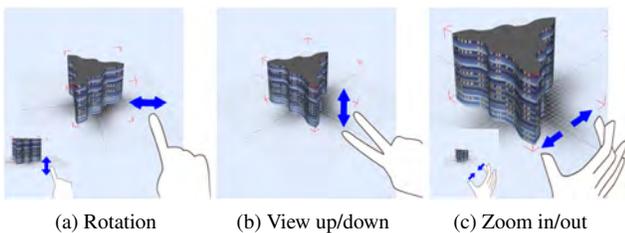


Figure 8: Gestures for camera view controls.

### 5) Design Coherency

In the building design process, one may have come up with various designs with (slightly) different variations. With our interface, the user can touch any screen region outside the building with three fingers. Then all existing designs can be displayed for comparison. Similar to the camera view control for a single building, the user can explore *all* buildings simultaneously. To delete a model, the user can simply drag it out. Furthermore, the user can touch the Sort button to re-order all the buildings according to a selected architectural parameter such as the facade area. After identifying the model that requires further editing, the user can double-tap it and put it into the normal editing mode. This comparison feature is desirable to allow the users to clarify their design ideas and to arrive at a coherent building solution, see Figure 9.

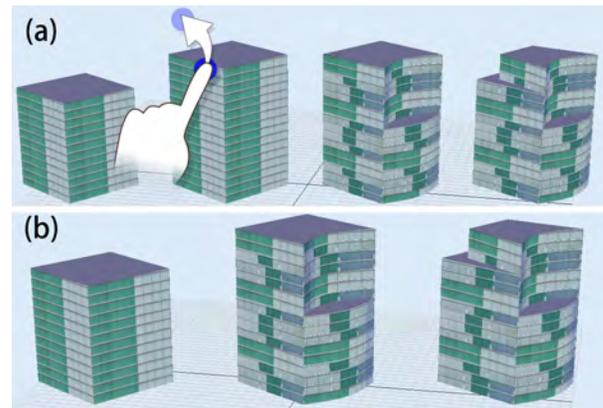


Figure 9: Side-by-side comparison of designs is desirable for users to clarify their design ideas and to arrive at a coherent building solution. To delete a design, one can simply drag-and-drop it (a); the view can be updated (b).

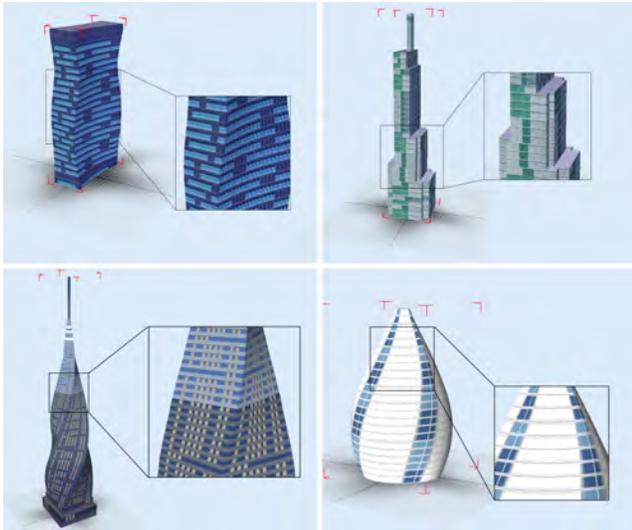
### IMPLEMENTATION AND RESULTS

Our multi-touch building prototyping interface was built on a Dell workstation T3500 with a 2.67GHz Xeon CPU, 12GB memory, and a GeForce GTX295 (1792MB) graphics card, as well as a 3M M2256PW (22") multi-touch display system, see Figure 4(a). The software is written in C++ and Qt4.7.

With our multi-touch interface, users can quickly design 2D floor plans with non-trivial shapes, as well as 3D buildings with non-trivial appearance by twisting, profiling, cropping, and local non-uniform scaling. Figure 2 shows a collection of building models that mimic famous contemporary buildings around the world. All these models can be constructed in a matter of minutes by a 3D modeling artist, who only needs some pictures of each building, see Figure 2.

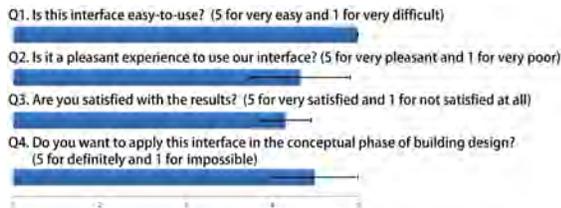
### EXPERT REVIEW

We invited three architects to evaluate our multi-touch interface: two with 10+ years experience and one with three years experience. All were very familiar with multi-touch devices, such as the iPad, iPhone, etc. The purpose of this expert review was to evaluate our interface from the architects' perspective, e.g., could our system meet their requirements? Was our interface easy-to-use? Did the architects enjoy using our interface and did they want to use it in their case studies?



**Figure 10:** Some results from the architects who participated in the expert review. All models were made in a few minutes.

The architects first watched a 5-minute video tutorial that demonstrated the interaction gestures for 3D navigation, initial modeling, structure editing, appearance enhancement, and design comparison. The tutorial also contained step-by-step instructions for constructing the Turning Torso building. Next, they were provided with a gesture table, and were given 10 minutes to practice all the gestures. Then, each of them had 30 minutes to freely make building designs with our tool. All the participants were impressed by our multi-touch interface, and made various non-trivial designs, some of which are shown in Figure 10. Lastly, each participant was asked to complete a questionnaire and rate our interface on a scale from 1 to 5, see results in Figure 11.



**Figure 11:** Questionnaire in the expert review.

Some gestures in our interface are also common in other multi-touch applications, e.g., the two-finger pinch for camera zoom in/out, and the one finger drag to change view (see Figure 8(a) and (c)). To reduce the user's learning curve, we also extend the usage of these gestures to other scenarios. For example, during the structure editing, when a range of floors is selected, a two-finger pinch vertically (resp. horizontally) scales the floor plan (resp. height), see Figure 6(c)).

The architects also commented that our gesture set was easy to use. We observed that after the 15-minute training period (watching the video tutorial and practicing for 10 minutes), all the architects had been able to learn the gestures well. Furthermore, the gestures were also easy to recall, as evidenced by the fact that the architects seldom needed to look at the gesture table during the building design stage. Our system

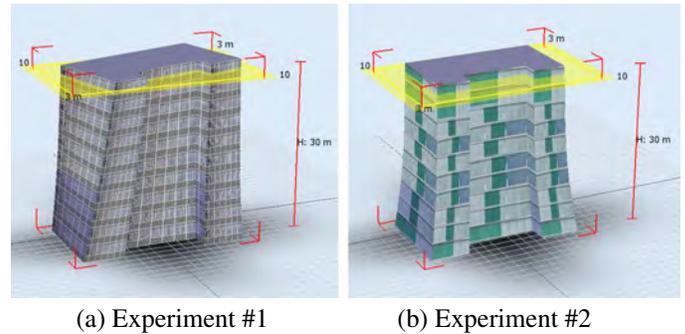
also recorded the frequency of each gesture when the architects performed them during the design stage. Statistics show that the most frequently used gestures were camera rotation (26%), floor selection (25%), editing 2D floor plan (17%), and applying the texture (12%). The least frequently used gestures were floor duplication (1%), floor scaling (3%), and initial building from the ground floor plan (3%).

All the participants agreed that our interface was easy to use, and was efficient for designing buildings. They were also satisfied with the results produced by our interface. In addition, they were very pleased with the capability of our interface to allow the entire building to be made just with the fingers, without any other form of input. They also appreciated that all the frequently used editing and navigation operations could be done completely by gestures, and that there were very few buttons for undoing, sorting and snapping. They could use drag-and-drop texturing to flexibly apply texture to a range of floors or to the entire building, which was very effective for improving the appearance. Such a texturing procedure is often tedious with conventional 3D modeling tools. Moreover, our multi-hand gestures, such as twisting, scaling, and duplication, were very powerful for 3D editing, and they had not found their equal in any other multi-touch applications. However, they also pointed out several limitations of our current prototype system. The interface was limited to designing a *single* tall building, thus, it could not be used for the Petronas Twin Towers in Kuala Lumpur or a group of buildings. Moreover, the interface could not support the design of *freeform* structures, such as the Milan Trade Fair Complex.

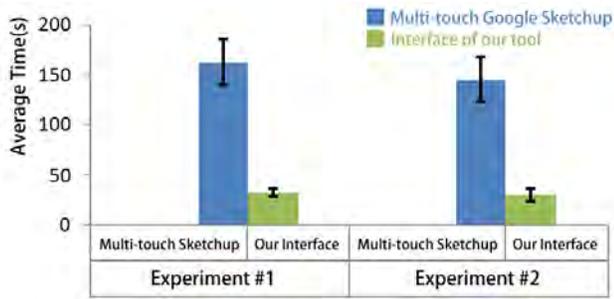
## USER STUDY

To quantitatively evaluate the performance of our interface, we compared our interface with Google Sketchup, a popular general-purpose 3D prototyping tool. Furthermore, we installed a multi-touch plug-in<sup>1</sup> on Google Sketchup, so that users could perform simple multi-touch gestures, such as pinching. We recruited twenty participants, 10 males and 10 females aged from 18 to 26. All were undergraduate or post-graduate students in a university architecture department.

<sup>1</sup>Sketchup MultiTouch project by Sharma, Solyga, and Bintahir: <http://code.google.com/p/sketchupmultitouch/>



**Figure 12:** User study. We designed two experiments for buildings with non-trivial geometry to compare the efficiency of our multi-touch interface against Multi-touch Google Sketchup in terms of fast building prototyping.



**Figure 13:** User study results. The vertical axis shows the average time taken for the participants to complete each experiment. The black vertical line on each bar is the standard deviation. Time was measured in seconds.

We designed two experiments to compare our interface with multi-touch Google Sketchup. The first experiment was to create a 10-storey building with a concave floor contour and curved/twisted profile (see Figure 12(a)). Each participant had to first perform the initial modeling gestures to create a 2D floor contour and then extrude the floor plan to make a 3D building. Next, the participant had to non-uniformly scale only the building profile for the top levels, edit the related profile curve, and then twist it. Finally, the participant had to attach image textures to improve the appearance. The second experiment was to further improve the appearance of a building (produced from the first experiment) by applying image textures to it (see Figure 12(b)). There were three facade textures available to the participants, and we also provided a table that specified the texture to be applied to each floor.

At the beginning of the user study, we introduced our *multi-touch* tool and tutored each participant in the same way as we did in the expert review process. After being given the figures that show the buildings in experiments #1 and #2, each participant was asked to complete the buildings, and our interface recorded the time taken by each of them. After that, we introduced multi-touch Google Sketchup to the participants. As a general purpose 3D modeling tool, Google Sketchup has many buttons, thus posing extra difficulty for beginner users. Though the plug-in enables participants to use Google Sketchup with simple multi-touch gestures, modeling operations such as extruding a face can only be done by first tapping a button. To reduce the learning curve for the participants and thus allow a fairer comparison, we discussed which buttons were necessary for the participants in order to complete the experiments. To assist the participants to tap these buttons, we reset the screen resolution to enlarge the buttons. The participants were given 15 minutes to familiarize themselves with these functionalities. Finally, they were asked to repeat the experiments with multi-touch Google Sketchup.

As shown in Figure 13, the average times taken to complete the two experiments with our multi-touch interface were 33 and 30 seconds, respectively, whereas the times taken with multi-touch Google Sketchup were 163 and 146 seconds, respectively. To analyze the user study data, we conducted a paired *t*-test with the null hypothesis  $H_0$ : the mean values of the task completion time using the two interfaces are equal.

We used a significance level of 0.005, given  $H_0$  is true. The resultant *t* values for experiments one and two were 27.6 and 26.2, respectively. Both values were larger than the critical value, i.e., 2.86, (with a degree of freedom  $DOF=19$ ) obtained from the *t*-test table; hence, we could reject  $H_0$ , and showed that our interface can help users to complete the tasks faster.

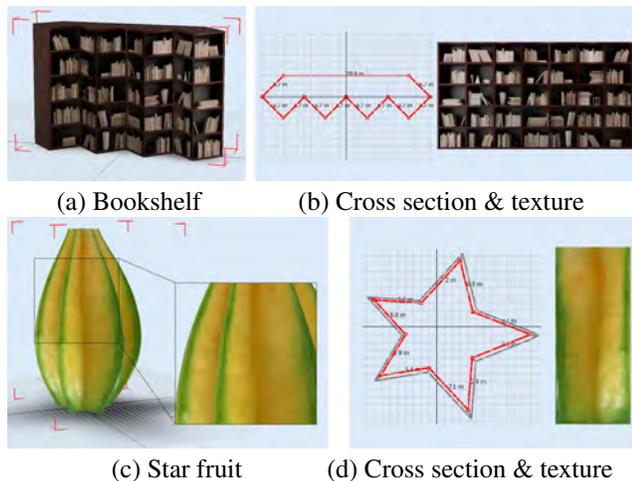
Multi-touch Google Sketchup is slower than our interface for the following reasons. First, although users can use basic multi-touch gestures in Google Sketchup, they have to frequently tap various buttons to perform the different modeling operations needed to complete the tasks; this is not only tedious but also distracting to the users. Second, our drag-and-drop texturing function allows the users to quickly apply textures to a selected range of floors (or the entire building). Although multi-touch Google Sketchup also allows users to apply textures to selected faces (i.e., windows in a floor), such selection is done on polygon levels (rather than semantic levels on the buildings); an approach, which is typically tedious and prone to error.

## CONCLUSION

This paper proposes a novel multi-touch interface comprising a rich set of direct finger gestures to facilitate rapid prototyping of modern building structures. Using our interface, users can quickly mock up, in a matter of minutes, reasonably complex building structures that can be found in many renowned modern buildings around the world.

In our approach, we divide the prototyping process into three stages, and devise effective multi-touch gestures dedicated to relevant modeling and editing needs in different working stages. Among them, our work contributes particularly to a number of novel modeling and editing techniques, including symmetry snapping for constrained multi-touch 2D editing, interface design for efficient multi-touch modeling of buildings, as well as a collection of effective gestures for quick refinement of building designs, e.g., copy and collapse, non-uniform deformation, profile curve editing, and twisting. These interaction gestures are not only efficient and relatively easy to learn and use, but also have great potential for supporting more general 2D/3D modeling and editing in other application scenarios. Finally, we developed a working system, and also evaluated its effectiveness with a number of users. The expert review justified that our interface has the potential for use in the conceptual phase of contemporary building design.

**Generalization.** Existing multi-touch techniques to interact with 3D contents focus mainly on 3D navigation, 3D object manipulation, physical simulation, and virtual scene design. However, our work advances the multi-touch interaction technology with a comprehensive and novel set of 3D modeling and editing gestures. With this, we can enhance the capability of multi-touch for designing 2D shapes as well as 3D models with non-trivial structures involving twisting, profiling, and local non-uniform scaling. It is worth noting that the proposed multi-touch interface is not limited to designing 3D buildings. In addition, it can be used for other types of geometric designs: 1) for 3D models with repeated geometric features, e.g., bookshelves, cabinets, etc., see Figure 14(a)(b),



**Figure 14:** Our interface can be used with other types of 3D design, where the 3D models have repeated geometric features, planar/rotational symmetry, or curved profiling.

which shows a bookshelf created with our interface; we first sketched the cross-section of the bookshelf using the 2D floor plan editor, and then adjusted the number of levels by the floor select/extruding feature in our tool; and 2) for 3D models with progressively varying and symmetric cross-sections, e.g., vases, bottles, bowls, etc. See the star fruit 3D model shown in Figure 14(c)(d). It can be quickly made with our interface in just two minutes.

**Acknowledgment.** We would like to thank the reviewers for their various helpful comments, and the architects and students (from SUTD and NUS, Singapore) who participated in our expert review and user study. Photos shown in Figure 1 are from Flickr, courtesy of photographers James Cridland, My Hourglass, NASA Videographer, and failing\_angel (left to right). This project is supported in part by NRF2008IDM-IDM004-006 and the MOE Tier-1 Grant (RG 29/11).

## REFERENCES

1. Akin, O., and Moustapha, H. *Strategic use of representation in architectural massing*. Elsevier, 2003.
2. Bae, S.-H., Balakrishnan, R., and Singh, K. ILoveSketch: as-natural-as-possible sketching system for creating 3D curve models. In *ACM UIST '08* (2008), 151–160.
3. Bae, S.-H., Balakrishnan, R., and Singh, K. EverybodyLovesSketch: 3D sketching for a broader audience. In *ACM UIST '09* (2009), 59–68.
4. Bau, O., Poupyrev, I., Israr, A., and Harrison, C. TeslaTouch: Electro-vibration for touch surfaces. In *ACM UIST '10* (2010), 283–292.
5. Benko, H., Wilson, A. D., and Baudisch, P. Precise selection techniques for multi-touch screens. In *ACM CHI '06* (2006), 1263–1272.
6. Chang, S. H.-H., Stuart, L., Plimmer, B., and Wünsche, B. Origami simulator: A multi-touch experience. In *ACM CHI Extended Abstracts* (2009), 3889–3894.
7. Chiu, P., Liao, C., and Chen, F. Multi-touch document folding: Gesture models, fold directions and symmetries. In *ACM CHI '11* (2011), 1591–1600.
8. Cohé, A., Décle, F., and Hachet, M. tBox: A 3D transformation widget designed for touch-screens. In *ACM CHI '11* (2011), 3005–3008.
9. Fu, C.-W., Goh, W.-B., and Ng, J. A. Multi-touch techniques for exploring large-scale 3D astrophysical simulations. In *ACM CHI '10* (2010), 2213–2222.
10. Gänschert, C. *Tools for Ideas: An Introduction to Architectural Design*. Birkhäuser GmbH, 2007.
11. Grossman, T., Wigdor, D., and Balakrishnan, R. Multi-finger gestural interaction with 3D volumetric displays. In *ACM UIST '04* (2004), 61–70.
12. Hancock, M., Carpendale, S., and Cockburn, A. Shallow-depth 3D interaction: Design and evaluation of one-, two- and three-touch techniques. In *ACM CHI '07* (2007), 1147–1156.
13. Hancock, M., ten Cate, T., and Carpendale, S. Sticky tools: full 6DOF force-based interaction for multi-touch tables. In *ITS '09* (2009), 133–140.
14. Igarashi, T., Moscovich, T., and Hughes, J. F. As-rigid-as-possible shape manipulation. *ACM TOG* 24, 3 (2005), 1134–1141.
15. Kim, J.-S., Gracanin, D., Matkovic, K., and Quek, F. iPhone/iPod touch as input devices for navigation in immersive virtual environments. In *IEEE VR '09* (2009), 261–262.
16. Kin, K., Miller, T., Bollensdorff, B., DeRose, T., Hartmann, B., and Agrawala, M. Eden: A professional multitouch tool for constructing virtual organic environments. In *ACM CHI '11* (2011), 1343–1352.
17. Longay, S., Runions, A., Boudon, F., and Prusinkiewicz, P. TreeSketch: interactive procedural modeling of trees on a tablet. In *SBIM '12* (2012), 107–120.
18. Moscovich, T., and Hughes, J. F. Multi-finger cursor techniques. In *Graphics Interface '06* (2006), 1–7.
19. Palubicki, W., Horel, K., Longay, S., Runions, A., Lane, B., Měch, R., and Prusinkiewicz, P. Self-organizing tree models for image synthesis. *ACM Trans. Graph.* 28, 3 (2009), 58:1–58:10.
20. Reisman, J. L., Davidson, P. L., and Han, J. Y. A screen-space formulation for 2D and 3D direct manipulation. In *ACM UIST '09* (2009), 69–78.
21. Sharma, A., Madhvanath, S., Shekhawat, A., and Billingham, M. MozArt: a multimodal interface for conceptual 3D modeling. In *ICMI '11* (2011), 307–310.
22. Tversky, B. *The Cambridge Handbook of Visuospatial Thinking*. Cambridge University Press, 2005, ch. Functional Significance of Visuospatial Representations.