

EverybodyLovesSketch: 3D Sketching for a Broader Audience

Seok-Hyung Bae, Ravin Balakrishnan, Karan Singh
Department of Computer Science, University of Toronto
shbae | ravin | karan @dgp.toronto.edu

ABSTRACT

We present EverybodyLovesSketch, a gesture-based 3D curve sketching system for rapid ideation and visualization of 3D forms, aimed at a broad audience. We first analyze traditional perspective drawing in professional practice. We then design a system built upon the paradigm of ILoveSketch, a 3D curve drawing system for design professionals. The new system incorporates many interaction aspects of perspective drawing with judicious automation to enable novices with no perspective training to proficiently create 3D curve sketches. EverybodyLovesSketch supports a number of novel interactions: tick-based sketch plane selection, single view definition of arbitrary extrusion vectors, multiple extruded surface sketching, copy-and-project of 3D curves, freeform surface sketching, and an interactive perspective grid. Finally, we present a study involving 49 high school students (with no formal artistic training) who each learned and used the system over 11 days, which provides detailed insights into the popularity, power and usability of the various techniques, and shows our system to be easily learnt and effectively used, with broad appeal.

ACM Classification: H.5.2 [Information Interfaces and Presentation]: User Interfaces–Interaction styles; I.3.8 [Computer Graphics]: Applications

General terms: Design, Experimentation

Keywords: Perspective sketching, 3D sketching, sketch surface, gestural interface, axis widget, learnability

INTRODUCTION

Despite a universal desire to sketch well, only a tiny percentage of people can accurately draw what they see, either in reality or in their imagination. Betty Edwards analyzes this in her book on drawing instruction [12] as follows:

The majority of adults in the western world do not progress in art skills much beyond the level of development they reached at age nine or ten. ... Most children between the ages of about nine and eleven have a passion for realistic drawing. They become sharply critical of their childhood drawings and begin to draw certain favorite subjects over and over again, attempting to perfect the image. Anything short of perfect real-

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, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

UIST'09, October 4–7, 2009, Victoria, British Columbia, Canada.
Copyright 2009 ACM 978-1-60558-745-5/09/10...\$10.00.

ism may be regarded as failure. ... Perhaps you can remember your own attempts at that age to make things “look right” in your drawings, and your feeling of disappointment with the results. ... Looking at your drawings, you may have said, as many adolescents say, “This is terrible! I have no talent for art. I never liked it anyway, so I’m not doing it anymore.”

We are interested in developing tools that can better support and revive the latent drawing skills of the broader population. Realistic depiction is based on perceptual drawing (drawing what you see) rather than conceptual drawing (drawing what you know) [10]. Perceptual drawing requires understanding the paradox that one must often draw a non-square shape to depict a shape that we know to be square [12] and overcoming the symbolic tendency to always draw a square. This skill can be learnt over time with practice, perceptual exercises [12] and the analytic rigor of perspective drawing, where a number of assisting 3D lines called a scaffold help anchor the design in 3D [36, 37].

Recently a 3D curve sketching system, ILoveSketch [3], showed that a judicious combination of 3D curve creation tools and a pen-and-paper like interface allowed design professionals trained in perspective drawing to effectively sketch 3D conceptual models. The system has generated significant interest amongst the professional design community. However, its interface was tuned to professionals, with no attention paid to the needs of a broader audience.

In this paper, we present EverybodyLovesSketch, a 3D curve sketching system that helps untrained people with a desire to draw, explore and develop design ideas directly in 3D. We first review the basics of traditional perspective drawing and analyze its use in professional practice. This

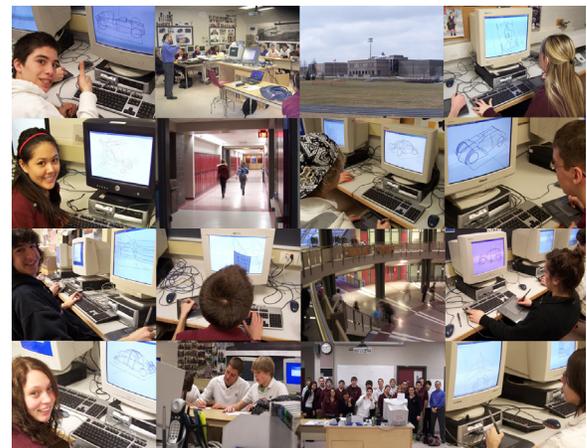


Figure 1: EverybodyLovesSketch used in a high school

analysis allows us to refine the interaction techniques of ILoveSketch to better support such professionals but also helps guide the design of a system targeted at individuals with no formal training in drawing. The result is a system built upon the ILoveSketch paradigm but with a number of novel features to support the creation of a richer set of 3D surfaces on which to project 2D strokes, curve replication through copy-and-project, and various dynamic assisting lines and perspective grids to help visualize the 3D space surrounding the evolving conceptual shape being created.

We then evaluate this system in a study involving 49 high school students who each learned and used the system over 11 days (Figure 1). Their impressive learning curve and resulting 3D sketches show that EverybodyLovesSketch is a learnable and usable system enabling non-professional users to explore new ideas in 3D while being encouraged to sketch with a fun interface that also improves their sense of 3D space and ability to draw in perspective.

RELATED WORK

Both traditional perspective drawing and 3D curve sketching view the 3D world under a 2D perspective projection, where sketched 2D strokes are resolved in 3D by anchoring them to parts of an evolving 3D scaffold of assisting lines or by projecting them on virtual 3D sketch surfaces. Effective 3D curve sketching thus requires that the viewer has a good spatial understanding of the 3D scaffold or sketch surface in the given view. Orthographic planes [15, 42], rotated vertical planes [11], oblique planes [5, 21, 33], extruded surfaces [16, 42] and freeform meshes [19, 22, 31] are easily understandable 3D sketch surfaces, that have been extensively used in 3D curve sketching systems. ILoveSketch uses a context-sensitive axis widget to define orthographic sketch planes and extruded sketch surfaces [3]. We present a richer set of sketch surfaces with easier invocation by using the axis widget and tick marks – a gesture popularly used in traditional perspective drawing.

Alternatively, the user’s spatial understanding can be aided by using special hardware to correlate the 3D design space and the physical workspace. The user can draw on a 6DOF drawing tablet whose position and orientation directly map to a plane in the virtual 3D space [38, 41], or by attaching the horizontal drawing surface to a mechanical structure that can move vertically [25].

Systems aimed at novice users often use feedback and feedforward mechanisms to inform the user of current and expected states of the system. Chateau [20] and the image guided suggestive drawing system [42] let users select a desired action or curve among some candidate results, which are inferred based on the user’s input as a hint, and displayed on the main or thumbnail windows. However, the user has to spend time to search for a correct result among visually distracting candidates whether or not it has actually been suggested [43]. In our system, the user’s intention to select a specific sketch surface and interactive grid is precisely inferred based on explicit gestures that are

cohesively related to the drawing context. Once a sketch surface is defined, to avoid visual clutter, only information related to sketching on the surface, such as assisting lines and intersection points, is dynamically displayed.

TRADITIONAL PERSPECTIVE DRAWING

We now review traditional perspective drawing to distill aspects that can augment a 3D sketching system, and determine aspects that can be automated or are unnecessary for a system that allows frequent 3D view changes.

Basics of Linear Perspective

Linear perspective, also known as the visual field [13], has been used widely in art and design [7, 8, 10, 18, 35] despite it being a bit different from human processing of visual information, also known as the visual world [13].

- Visual field: relates to the use of one eye looking straight at the horizon, parallel to the ground plane, with a narrow cone of vision, while standing still (Figure 2).
- Visual world: is based on the use of two eyes looking all around with peripheral vision, while moving dynamically.

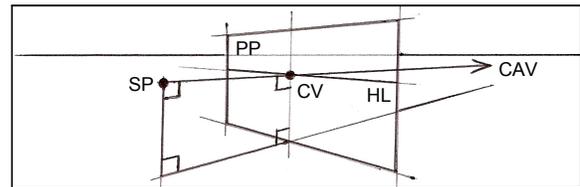


Figure 2: Linear perspective system: picture plane (PP), station point (SP), central axis of vision (CAV), center of vision (CV), horizon line (HL).

Convergence. Parallel lines apparently converge toward a common vanishing point as they recede [7, 8, 10, 18]. Figure 3 explains important convergence rules.

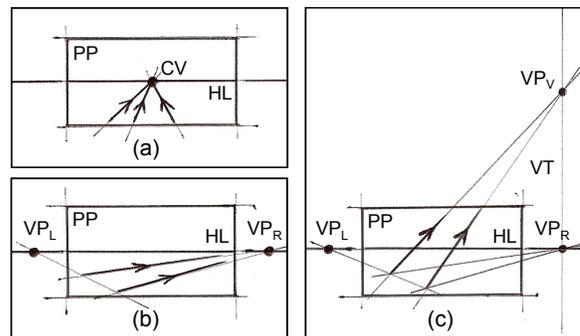


Figure 3: Convergence: (a) parallel lines perpendicular to PP appear to converge at CV, (b) horizontal parallel lines oblique to PP appear to converge at a vanishing point (VP) on HL, (c) non-horizontal parallel lines oblique to PP appear to converge at a VP on the vanishing trace (VT).

Measurement. Distant objects appear smaller than identical objects closer to the picture plane. Once an initial depth judgment is established, subsequent depth judgment can be easily done in proportion to the first (Figure 4). Techniques for this initial judgment, however, are mechanical and complicated [8] (not shown here).

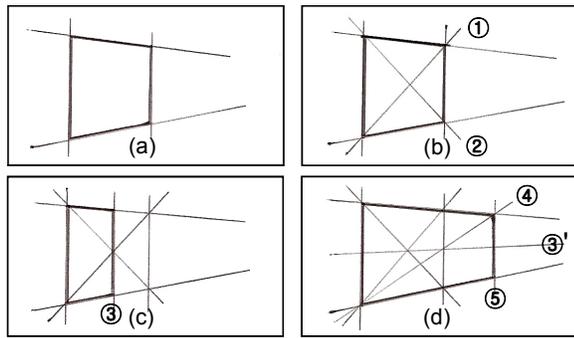


Figure 4: Depth measurement by the diagonal method: (a) initial perspective square, (b) sketching diagonals, (c) subdivision in half, (d) extension double.

Perspective Drawing in Practice

Designers use sketching for graphic thinking [26], where it is important to quickly visualize a new idea to crystallize it before it slips away [18]. Drawing these 3D shapes with precision in correct perspective requires first an understanding of convergence and measurement techniques and then considerable time and effort to use this knowledge to construct a scaffold in which to situate the 3D curves. Many professionals take a hybrid approach, using a coarse scaffold along with shortcuts for rapid, yet plausible results.

Perspective Line Estimation. In many cases, it is not possible to precisely construct perspective lines to a vanishing point (VP) because it lies outside the paper (Figure 3). Professionals approximate this by practicing line drawing in proper perspective based on existing lines [18] (Figure 5).

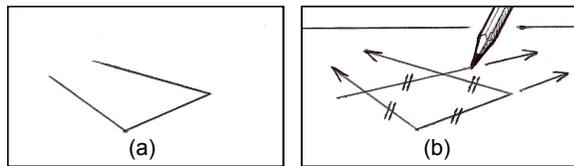


Figure 5: Perspective drawing based on estimation.

Measurement Estimation. Professionals often subdivide and extend perspective depth similarly, based on eye estimation.

Initial Perspective Guidelines. Interior designers use simple perspective grids [7, 8]; car designers use a box, cylinder, computer-generated perspective grid and wheels [35].

Frequent Checking. It is common to see the canvas with a fresh eye from a different distance and orientation, to discover incorrect proportion and perspective [10].

Analyzing a Professional Perspective Drawing Session

To better understand and characterize how professionals draw, we analyzed concept artist Scott Robertson's demo videos on perspective concept sketching [36, 37] (Table 1).

He created quick initial perspective lines for every sketch. Except for an example of one-point perspective (where the VP lays within the canvas), perspective line estimation was used instead of drawing lines to explicitly intersect a VP.

A lot of perspective lines were drawn effortlessly based on estimation and experience, repeating things done before that had proved successful [40]. The number of assisting lines was over half the number of design curves. But he did not create every line for precise perspective drawing; many measurements were done by simply guessing. In addition to providing a 3D space in which a sketch can be anchored scaffold lines are often used to provide connectivity to 3D design curves that are spatially related but topologically unconnected, such as a design detail in the interior of the surface of a 3D model.

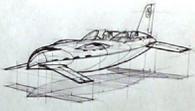
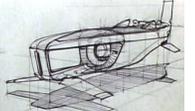
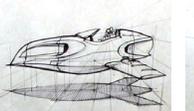
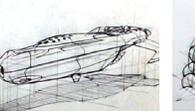
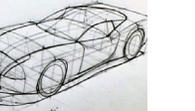
Tick marks were used as a visual reminder on existing sketched marks including lines and curves. He utilized ticks to depict the start and end position of a line to draw, to adjust arm movement so as to move the pen tip to pass through them, to indicate interpolation positions for smooth curve drawing, to emphasize important design points, and to mark intersection points of an existing line and ghost diagonal to accelerate the speed of the diagonal method and avoid clutter on paper. The number of tick marks was about 60% of the number of design curves on average.

Reposition and reorientation of the canvas was also very frequent: occurring for about 75% of the design curves. Navigation was more important when precision was required as in the car example, where relatively more line drawing between two constraint points was done.

DESIGN GOALS

The essence of perspective drawing thus resides in the construction of assisting lines that help guide the perspective width, height, and depth of the 3D shape to be sketched [8]. Professionals create these lines almost subconsciously and spend more time in creative activities [6], whereas non-professionals do not create these lines due to a lack of perspective drawing knowledge, or waste time creating them unnecessarily. The ultimate goal of our system is to give everybody and especially non-professionals, a drawing experience without the burden of perspective drawing the-

Table 1: Analysis of a professional concept artist's perspective sketches (images: © Scott Robertson).

Number of						Total
Design curves	75 (13:58)	101 (15:20)	88 (11:10)	96 (16:50)	129 (19:14)	489 (76:32)
Guide lines	37 (49 %)	50 (50 %)	44 (50 %)	48 (50 %)	79 (61 %)	258 (52 %)
Ticks	70 (93 %)	59 (58 %)	30 (34 %)	55 (57 %)	75 (58 %)	289 (59 %)
Paper adjustments	78 (104 %)	47 (46 %)	29 (33 %)	64 (67 %)	151 (117 %)	369 (75 %)

Parentheses in design curves row: sketching time (MM:SS); other parentheses: percentage with respect to the number of design curves.

ory and practice, allowing easy exploration and design in 3D space. We thus identify our design goals as:

- Easy 2D stroke drawing
- Easy 2D/3D navigation
- Easy scaffolding or sketch surface definition
- Dynamic visualization of a perspective scaffold
- Easy syntax

SYSTEM OVERVIEW

We find from experience that the pen-and-paper like 2D multi-stroke based drawing and the two-handed simultaneous 2D/3D navigation method of ILoveSketch [3] generally suffice to leverage existing sketching skills and develop a new way of seeing that can be applied back to traditional drawing. The two-handed clutch based modal navigation method of ILoveSketch guarantees fast mode switching [27] and quickly becomes a subconscious action.

We eschew the explicit creation of scaffolds since they are unintuitive to an untrained user. Instead, we focus on providing a rich variety of sketch surfaces that are easy to specify and predictable to sketch on. These sketch surfaces are defined by a new tick gesture in conjunction with an augmented axis widget [3]. Inspired by perspective drawing, the tick is a powerful addition to the ILoveSketch gestural vocabulary. We also need the ability to spatially relate topologically distinct 3D design curves in the absence of a scaffold and do this by allowing users to select a closed loop of 3D curves to create a temporary NURBS surface, and to sketch 3D curves on it. Coupled with an easy-to-use interface, asking the user to explicitly set sketch surfaces can provide a better 3D sketching workflow with less cognitive burden than inference-based 3D curve sketching [2, 9, 22, 23, 29]. Note that while it would be easy to surface the entire 3D model in this way, we relegate that to downstream techniques specialized in surfacing, since the definition of surfaces early in conceptual design often biases one’s visual thinking [21, 33].

Drawing convergent scaffold lines and perspective measurements are second nature to professionals, but are difficult for novice users. Given that a 3D sketching system already has this information for the existing 3D geometry and the current view, it can be automatically created to aid professionals and educate novices. It is, however, important to manage this dynamic display to minimize visual distraction and clutter. EverybodyLovesSketch uses the active sketch surfaces and current view to provide a dynamic 3D scaffold comprising: assisting lines, a perspective grid on the sketch surfaces, the intersection points of sketch surfaces and existing 3D curves, and the sketchability [3] of the active sketch surfaces.

Syntax design can significantly affect the quality (learnability, error frequency and nature, skill retention, task performance speed) of the user interface of an interactive system [6]. The grammar of EverybodyLovesSketch consistently follows chunking [24] and a noun-verb form [34]

based on selection-action phrasing [6]. Every verb command for sketching is invoked by applying pen gestures originating from hand-marks used in drawing (Figure 6).



Figure 6: Gestures: (a) scratch-out, (b) roll, (c) tick, (d) small lasso, (e) tap, (f) span, (g) flick, (h) angled flick.

To keep the gesture vocabulary for sketching compact, we provide a crossing menu [1] for a few less-frequently used functions including opening and saving files, gesture calibration, and toggling symmetry. The crossing menu appears only when the user presses a button on the graphics tablet or keyboard with the non-dominant hand. Crossing to select items is similar to sketching, making the menu fluidly compatible with sketching programs (Figure 7).

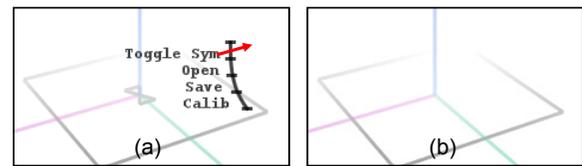


Figure 7: Crossing menu: (a) selecting the *toggling symmetry* item in the symmetric sketching mode, depicted by two triangles on the floor plane facing each other, meaning one is a mirror image of the other, (b) asymmetric sketching mode (no triangles).

SKETCH SURFACE INTERACTION

We now explain how to define various sketch surfaces and create 3D curves in detail.

Ticks

In professional perspective sketching most ticks are used to mark position constraints on existing lines and curves. The tick marks define virtual (transparent) perspective sketch surfaces on the paper upon which subsequent strokes are situated in 3D. We provide an easy way of selecting sketch planes by applying tick marks on existing 3D lines (including the global axes) and curves. Horizontal (plan) and arbitrarily rotated vertical planes (elevation and section) are the most common sketch surfaces in architectural design [8] and arguably in general conceptual design. A horizontal sketch plane is quickly invoked using a single tick crossing a given 3D curve or the intersection of multiple curves. The plane passes through the 3D tick position on the curve(s) (Figure 8a) and is visualized using boundary lines and orthographic assisting lines passing through the tick. Two consecutive tick marks (without curve sketching in-between) crossing 3D curves define a vertical plane passing through the two 3D ticks (Figure 8b). In similar fashion, three consecutive tick marks crossing 3D curves define an arbitrary sketch plane (Figure 8c).

The automatic display of assisting lines on the selected sketch plane allows users to focus on their design without having to explicitly construct perspective lines. We also use simple visual indicators to guide the user’s spatial un-

derstanding of the current sketch plane and towards better viewpoints for 3D curve sketching. In Figure 8b-d, the gray triangles depict the angle of the sketch plane from the global zx -plane (symmetry plane), and from the xy -plane (floor plane), respectively, while the underground part ($z < 0$) of the sketch plane is shaded with a value reflecting sketchability [3] of the sketch plane in the current view.

In the design of most symmetric shapes including vehicles, slant symmetric curves crossing the center plane play an important role. Generating such curves using single view symmetric sketching is unstable [2]. In our system, however, they can be easily created by sketching on a slant sketch plane defined using three ticks on a pair of symmetric curves and a center curve (Figure 8d). The continuity of the symmetric 3D curve at the center point can be controlled by the user according to the proximity of the angles of sketched 2D strokes and horizontal assisting line.

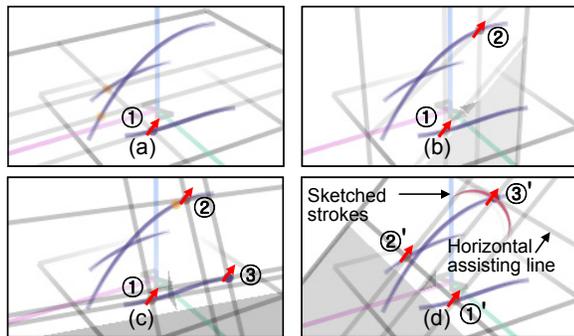


Figure 8: Tick-based sketch plane definition: (a) horizontal plane, (b) rotated vertical plane, (c) oblique plane, (d) sketching a continuous symmetric curve with a help of assisting lines.

Axis Widget

Sketch surfaces are defined in ILoveSketch using an axis widget [3]. The widget is invoked by a small lasso gesture that positions the widget on a 3D curve or on several almost intersecting 3D curves. A single subsequent stroke crossing the axis widget is used to define orthographic sketch planes (Figure 9a), orthographically extruded sketch surfaces (Figure 9b), and an obliquely extruded sketch surface [3]. Since good views for specifying the extrusion direction and sketching on the extruded surface are orthogonal to each other [3], oblique extrusions require an awkward immediate view change, prior to sketching on the extruded surface.

We suggest a new way to define the extrusion direction without view change (Figure 9c and d), while preserving the other axis widget interactions. Figure 9c shows a flick starting from the origin of the axis widget, which defines a horizontal vector for sketch surface extrusion. In ILoveSketch this defines a vector on the view plane, requiring a subsequent view change. Instead, we propose an angled flick gesture to define an arbitrary extrusion vector, where the first part of the angled flick defines the angle between the x -axis and the vector projected on the horizontal plane

of the axis widget, and the second part defines the angle between the vector and the z -axis of the axis widget. In fact, the two angles defined by an angled flick are the azimuth (ϕ) and zenith (θ) angles of spherical coordinates [30] (the inset of Figure 9d).

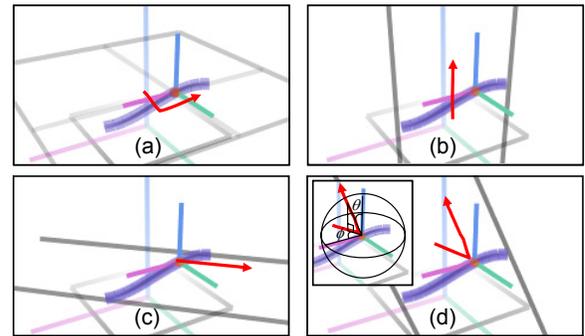


Figure 9: Sketch surface definition by using an axis widget: (a) orthographic plane by applying a span gesture, (b) orthographically extruded surface by applying an axis-aligned flick gesture, (c) horizontally extruded surface by applying a flick gesture starting from the origin of the axis widget, (d) obliquely extruded surface by applying an angled flick gesture.

Multi Selection, Copy-and-Project

EverybodyLovesSketch allows multi selection, unlike ILoveSketch. The user can select more than one 3D curve by applying small lasso gestures on them (Figure 10a), and then extrude all of them along any direction by using new axis widget interaction techniques (Figure 10b).

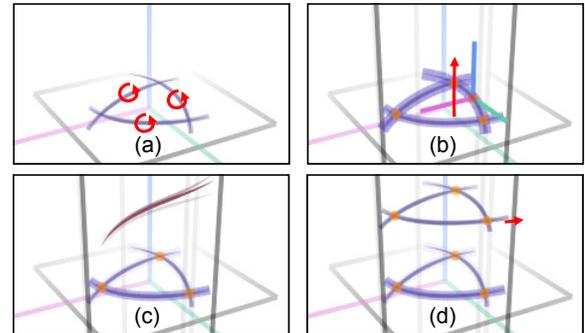


Figure 10: Multiple extruded surfaces: (a) selecting multiple curves, (b) extruding the curves by a flick gesture crossing the axis widget, (c) sketching curves on the active sketch surface among the multiple extruded surfaces, (d) copy-and-projecting the selected curves by applying a tick mark on one of the boundary lines of the active sketch surface.

The active sketch surface is the one with maximum sketchability and is visually distinguished from other extruded sketch surfaces by darker boundary lines. Multiple sketch surfaces streamlines the workflow allowing a user to rapidly switch the active surface by changing viewpoints, and sketch 3D curves on all of them continuously (Figure 10c). The intersection points of a sketched 3D curve on the current sketch surface become important anchors for subsequent sketching on the neighboring surfaces.

The user can copy-and-project selected 3D curves by applying a tick on one of the boundary lines of the active sketch surface (Figure 10d). In some extrusion-based 3D modeling systems, the user creates an extruded volume by selecting a closed area and dragging it along the vertical direction [14]. In our system the user instead projects the selected 3D curves orthogonally onto the plane that is defined with the 3D tick position (a point on the plane) and the extrusion direction (the normal vector of the plane).

Freeform NURBS Sketch Surface

Line drawing emphasizes the basic structure and composition of a design while leaving room for exploration of the space between the lines [18]. Involving surfaces early in the design process often biases the viewer visually and can shift user focus from ideation to surface quality and continuity [21, 33]. It is however common, even in traditional perspective drawing, to sketch in the interior of 3D design curves, as if they formed a virtual 3D surface. We thus introduce a *temporary* freeform sketch surface defined by two to four 3D curves that make a closed loop(s) (Figure 11a) by selecting them and using an area-fill (same as scratch-out) gesture (Figure 11b). The user then sketches 3D curves on the NURBS surface created using trimmed boundary curves [28] (Figure 11c), and finally throws away the sketch surface by scratching it out (Figure 11d).

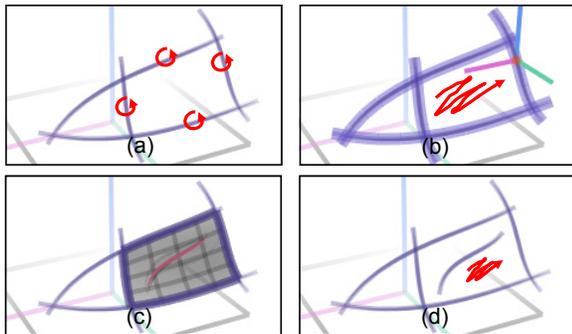


Figure 11: Sketching on a temporary freeform sketch surface: (a) selecting 3D curves making a closed loop, (b) depicting a surface area by applying an area-fill gesture, (c) sketching 3D curves on the surface, (d) erasing the surface.

Interactive Grid

In any plane sketch mode, two taps in the plane creates an interactive grid on the plane (Figure 12a), which is a dynamic version of perspective grid sheets [7, 8] used in traditional perspective drawing. The grid size is set to half the distance between the two tap positions on the plane, simplifying the tedium of measurement subdivision and extension using diagonals in traditional drawing (Figure 4). Once an interactive grid is activated, the second tap is treated as the new first tap. Thus, the user can redefine a new grid with a single subsequent tap. If the new tap is applied on the grid point between the previous two tap positions, the new interactive grid has half the grid size of the previous one (Figure 12b). Turning off the interactive grid is handled like everything else, by simply scratching it out.

A linear stroke drawn along a grid line is approximated to a straight line while an elliptic stroke passing through four grid points that make a diamond shape is approximated to a circle or ellipse (Figure 12c and d). This special consideration is due to the fact that drawing precise lines and circles is an important skill in traditional perspective drawing [7, 8, 10, 18, 35].

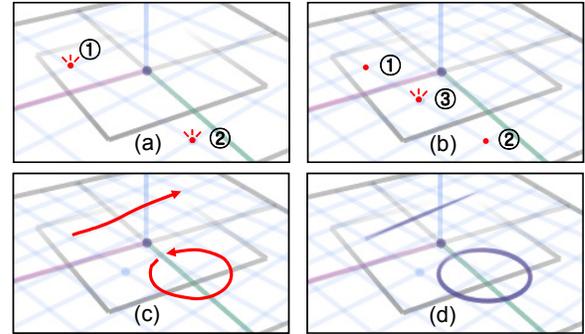


Figure 12: Interactive grid: (a) activating an interactive grid by applying two taps on a plane, (b) refining the grid by applying a tap on the middle grid point, (c) linear stroke along a grid line and elliptic stroke within a grid rectangle, (d) resulting line and circle.

USER EVALUATION

EverybodyLovesSketch was implemented in Java and was formally evaluated by a group of non-professional users.

Participants

49 high school students from three classes learned and used our system in their design class (75 min/day) for different projects over 11 days each. A standard school not specialized in art, design, or engineering was chosen for the study. 69% of the students had a prior experience of using 3D modeling systems including SketchUp (56%), AliasStudio (38%), and AutoCAD (36%).

- Class A - freeform architectural design: 18 students (18 males), Grades 11-12 (age 16-19)
- Class B - car design: 20 students (14 males, 6 females), Grades 10-12 (age 16-19)
- Class C - freeform topic including cartoon characters: 11 students (5 males, 6 females), Grades 11-12 (age 15-19)

Hardware System

- Desktop computers (2.4 GHz CPU, 1.0 GB RAM)
- Wacom Intuous3 4×6 graphics tablets and grip pens

Schedule

- Before the first class: The students were asked to draw a cube, airplane, car and house in perspective on paper with a pen. This served as a baseline of their skills.
- Instruction session 1: On the first day, an author of this paper explained the design concepts of our system to students and gave them a general lesson on its use.
- Instruction session 2: On the second day, the author gave a 3D model creation demo. The students ran the system for the first time, and calibrated pen gestures.

- Usage days 1-5: Led by their design teacher, the students performed design projects with a crib sheet (one letter size page) which served as the system manual. They were asked to save their work every day (3D curves and command execution log were stored). The author visited the classes again on Day 5 to answer student questions.
- Usage days 6-9: The students continued the projects while saving their work daily. On Day 9, students completed a questionnaire, and again sketched a perspective cube, airplane, car, and house on paper with a pen.

3D Sketching Results

78% of the students in Class A and 90% of students in Class B created their first meaningful 3D models in the first two usage days (Figure 13). The learning curve of Class C is relatively slower than Class A and B because some Class C students first designed 2D cartoon characters and later translated them into 3D. 47 students out of 49 (96 %) succeeded in creating meaningful 3D shapes. Based on the definition of (initial) learnability by Nielsen [32], this is strong evidence that EverybodyLovesSketch is a highly learnable system that allows users to reach a reasonable level of usage proficiency within a short time.

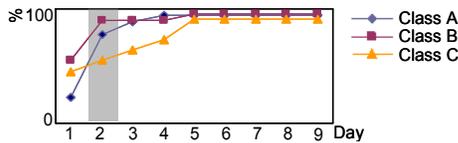


Figure 13: First completed 3D model (accumulated).

After completing their first 3D shape, most students created more complicated ones without major problems. Example designs from a student from each class are shown in Table 2. The majority of students tried to make many models one after another, whereas a few students (especially in Class C) focused on perfecting one model. There were some students who showed a special talent in 3D sketching. One student created an appealing design on Day 1 (Figure 14a) despite having no exemplary sketching skills (Figure 14b). This leads us to believe that 3D sketching is a good way to express ideas for people who have good 3D design ideas regardless of their drawing skills.

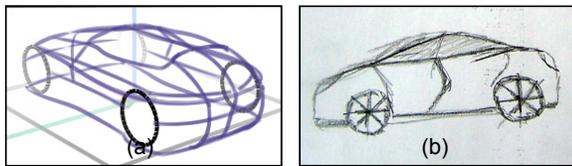


Figure 14: Immediate learning example of one student: (a) 3D sketch result from Day 1, (b) pen-and-paper sketch drawn before the first class.

Figure 15 shows examples of the transition from 2D sketching to 3D sketching. One student from Class C wanted to create a 3D dinosaur cartoon character. She started from a 2D design idea (Figure 15a), and later converted it into a 3D model (Figure 15b). While sketching approaches based on shape inflation [19, 31, 39] are especially tailored to such shapes, she was effectively able to

Table 2: Daily results of one student from each class.

	Of Class A	Of Class B	Of Class C
1			
2			Absent
3			
4			
5			
6			Absent
7			
8			
9			

express her ideas with our system. Another student from Class C showed a dramatic learning curve in understanding 3D. She stayed with drawing curves on planes until Day 5 (Figure 15c), but cried eureka on Day 5 and immediately created a decorative 3D model (Figure 15d).

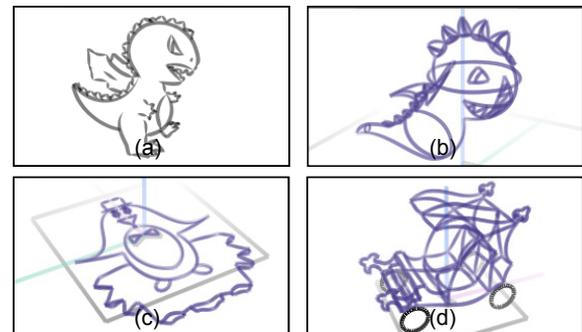


Figure 15: 2D to 3D sketching transition: (a) dinosaur character drawn in 2D, (b) its 3D version, (c) cartoon character drawn on the floor plane, (d) 3D coach drawn by the same student as in (c).

To get better insight into how novices with no perspective training learn a new system and develop 3D sketching skills, we monitored changes of usage patterns over the days based on command execution logs. The reader who has an interest in how each feature of our system was gradually learnt and eventually incorporated by the students in the overall workflow is advised to see the Appendix which provides detailed analysis of the log data.

Subjective Data

On the last day of the study, we asked students to complete a questionnaire on each feature (with 5-point Likert scale) and provide a general opinion of our system (Table 3). Overall, they liked the system, but gave relatively low scores to a few features. Their scores about the interactive grid (Q16) were a bit low because it was rarely used (most students focused on freeform shapes rather than prismatic shapes), and inadvertently turned on from time to time when they applied the tap (period) gestures with an intention of multi-stroke curve settlement (immediate and explicit settlement) [3] as their 2D strokes had already settled automatically (implicit settlement based on time-out) [3]. A

simple solution to prevent this false activation will be to provide users with the implicit method only for curve settlement. This monotony [34] will help users form a habit of using taps only when activating the interactive grid. The students in Class A evaluated automatic 3D rotation [3] (Q18) relatively low. When selecting a rotated vertical sketch plane, which students in Class A used often, automatic 3D rotation based on sketchability of the horizontal sketch surface defined by the first tick, sometimes moves the region where they intended to apply the second tick off-screen. Automatic 3D rotation should be rectified in the future reflecting this observation. In general, students answered that EverybodyLovesSketch was a simple and easy 3D sketching system that helped them develop the 3D shapes they envisioned (Q24-31). We quote some of their comments on the questionnaire sheets as follows.

I am able to draw the 3D curves I normally could not draw in this program that helps me get my ideas out of my head.

Amazing concept, super easy to draw and fix my ideas.

It becomes addicting. I strongly like the program.

Time goes by too fast when using the program. Very fun.

Table 3: Evaluation of high school students.

1	Multi-stroke curve creation	strongly dislike			strongly like
2	Automatic curve connection	strongly dislike			strongly like
3	Tangential curve connection with repair strokes	strongly dislike			strongly like
4	Gesture-based curve editing	strongly dislike			strongly like
5	Paper (2D) navigation	strongly dislike			strongly like
6	Space (3D) navigation	strongly dislike			strongly like
7	Tick marks to define a sketch plane	strongly dislike			strongly like
8	Span gesture to define an orthographic sketch plane	strongly dislike			strongly like
9	Flick to define an orthographically extruded sketch surface	strongly dislike			strongly like
10	Flick to define an horizontally extruded sketch surface	strongly dislike			strongly like
11	Angled flick to define obliquely extruded sketch surface	strongly dislike			strongly like
12	Way of defining 2-,3-,4-boundary sketch surface	strongly dislike			strongly like
13	Assisting lines on a sketch plane	strongly dislike			strongly like
14	Triangular indicator depicting the slope of a 3D plane	strongly dislike			strongly like
15	Underground ($z < 0$) indicator (shaded part) of a 3D plane	strongly dislike			strongly like
16	Interactive grid for measurement and line/circle drawing	strongly dislike			strongly like
17	Visualization of {sketch surface \cap existing 3D curves}	strongly dislike			strongly like
18	Automatic 3D rotation	strongly dislike			strongly like
19	Single-view symmetric curve sketching	strongly dislike			strongly like
20	Two-view curve sketching	strongly dislike			strongly like
21	Curve sketching on multiple extruded surfaces	strongly dislike			strongly like
22	Tick mark to copy-and-project selected curves	strongly dislike			strongly like
23	Toggling symmetry	strongly dislike			strongly like
24	The program was _ to learn.	very difficult			very easy
25	The program was _ to use.	very difficult			very easy
26	The user interface of the program was _.	very complex			very simple
27	How to use the program was _ to remember.	very difficult			very easy
28	Sketching with the program feels _ to pen-and-paper sketch.	very dissimilar			very similar
29	The 3D curves from 2D strokes were _ what I had imagined.	very far from			very close to
30	3D sketching with the program was _.	very difficult			very easy
31	3D curve sketching _ design evolution from an initial idea.	greatly hindered			greatly aided
32	My ability to sense 3D space has _ while using the program.	significantly declined			significantly improved
33	My perspective drawing skill has _ while using the program.	significantly declined			significantly improved
34	I am _ to continue to use this program.”	strongly disinclined			strongly inclined

Class A: —+— ; Class B: —+— ; Class C: —+—

Many students felt their 3D spatial understanding (Q32) and perspective drawing skills (Q33) improved during the course of using our system. Further observation for a longer period is required to verify this, but we do note an improvement in students' pen-and-paper sketches in terms of better convergence of parallel forms and more confident stroke styles (Figure 16).

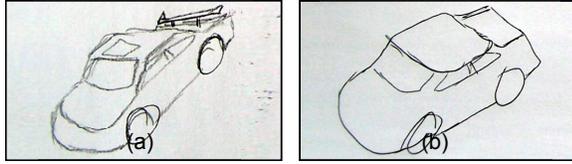


Figure 16: One student's pen-and-paper sketches drawn before and after using EverybodyLovesSketch.

The teacher for the three classes also provided feedback:

As a teacher of design, for the past 18 years I have integrated extensively design software in the program. Without question, EverybodyLovesSketch is not only extremely easy to use, but also very accessible to the student learner. EverybodyLovesSketch allows the students to transfer their ideas digitally, thus sophisticating their original designs. The ability to sketch in a 3D environment, not available in any current design software, motivates and excites students. Its unique feature to create both the initial design as well as begin the 3D modeling refines the process into a far more efficient one. Perhaps its best feature – a clean interface – creates a natural drawing space where the digital pen becomes the pencil. EverybodyLovesSketch is an extremely powerful, innovative and creative design tool. EverybodyLovesSketch is the most exciting design software I have used in my 23 years of working with and teaching Computer Aided Design.

CONCLUSIONS

We designed and implemented EverybodyLovesSketch, a broadly applicable 3D curve sketching system, based on lessons learned from ILoveSketch, and informed by an analysis of traditional perspective drawing. A user study with a large group of high school students showed that EverybodyLovesSketch is a highly learnable and usable system that enables anyone who wants to draw to transform mental design concepts to 3D shapes by sketching.

ACKNOWLEDGMENTS

We thank John Paul II Catholic Secondary School (London, Ontario) Design class students and their teacher Mike Santolupo for participating in our user study and providing valuable feedback, Eun-Jin Lee for processing command execution log data and video production, Noah Lockwood for video narration, and the anonymous reviewers for helpful comments.

APPENDIX: COMMAND EXECUTION LOG ANALYSIS

We analyzed the command execution logs of students in Class B. Because the teacher gave Class B specific tasks over the user test period (Day 1: mouse, Day 2: mouse, Day 3-5: car, Day 6-9: car), it was possible to observe the usage pattern of students and their extended learning [17] (change in performance over time [4]).

Figure 17a shows that relatively many 3D curves were created on the days when a new design starts (Days 1, 2, 3, and 6). Not surprisingly the most lively and exploratory design activity happens on the first day of each task period.

Figure 17b depicts the increase in frequency of navigation over time. In particular, frequent 3D tumble actions and its dramatic rising curve show that the students got gradually familiar with spending time and effort to explore 3D space to sketch 3D curves and evaluate them from various view-points. We see this as a positive departure from traditional perspective drawing, which is philosophically [13] a *visual field*, as opposed to a *visual world* view of 3D sketching.

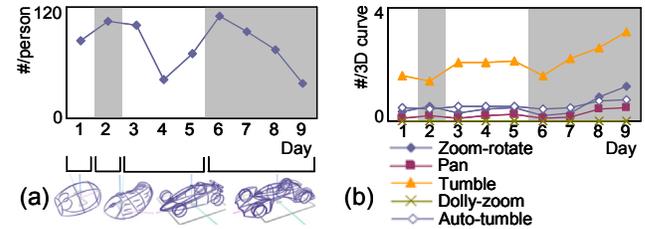


Figure 17: (a) Average number of 3D curves per student, (b) average number of navigation actions per 3D curve.

We calculated how many times each 3D curve sketching method was used per student. Shown in Figure 18a are the number of sketch planes that each student selected on average. The orthographic planes selected by using span gestures appeared most on the first day of each design. The orthographic planes based on ticks were selected less than those based on span gestures, but were used steadily. We reasoned that the students used span gestures to select orthographic planes, including the center and floor planes, for sketching major profile curves on them on the first design days, whereas they used tick-based orthographic planes for sketching details. Over days, the rotated vertical plane and oblique planes defined by tick marks got used more. So did extruded surfaces (Figure 18b) because they are used frequently to modify existing 3D curves (the proportions of orthographic, horizontal, and oblique extrusions are 74%, 9%, 17%, respectively). The use of freeform sketch surfaces increased dramatically too. The students used freeform sketch surfaces on Day 2 more than on Day 1 for mouse design, and during Day 6-9 more than during Day 3-5 for car design. Naturally, the number of freeform sketch surfaces increased as the design progressed because they are created on existing 3D curves.

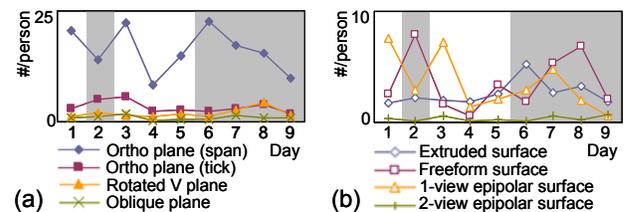


Figure 18: Avg. number of sketch surfaces per student.

To visualize usage over time, we compared the proportion of 3D sketching methods used during Days 3-5 and Days 6-9 (Figure 19). The role of the orthographic plane sketch-

ing and single-view symmetric curve sketching [2, 3], which served as an entry point to 3D sketching due to simplicity, was decreased, whereas that of rotated vertical plane, extruded surface, freeform surface, and two-view sketching [3, 23] was increased. This supports the hypothesis that our system has good extended learnability.

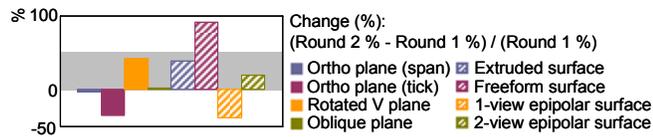


Figure 19: Change of use of 3D sketching methods.

REFERENCES

1. Apitz, G. and Guimbretière, F. 2004. CrossY: a crossing-based drawing application. *UIST*, 3-12.
2. Bae, S., Kijima, R., and Kim, W. 2003. Digital styling for designers: 3D plane-symmetric freeform curve creation using sketch interface. *ICCSA*, 701-710.
3. Bae, S., Balakrishnan, R., and Singh, K. 2008. ILoveSketch: as-natural-as-possible sketching system for creating 3D curve models. *UIST*, 151-160.
4. Bevan, N. and Macleod, M. 1994. Usability measurement in context. *Behavior and Information Technology* 13, 132-145.
5. Bourguignon, D., Cani, M. P., and Drettakis, G. 2001. Drawing for illustration and annotation in 3D. *Computer Graphics Forum* 20, 3, 114-122.
6. Buxton, W. 1986. Chunking and phrasing and the design of human-computer dialogues. *IFIP World Computer Congress*, 475-480.
7. Chelsea, D. 1997. *Perspective! For Comic Book Artists: How to Achieve a Professional Look in Your Artwork*, Watson-Guptill.
8. Ching, F. D. K. 1998. *Design Drawing*, John Wiley & Sons.
9. Cohen, J. M., Markosian, L., Zeleznik, R. C., Hughes, J. F., and Barzel, R. 1999. An interface for sketching 3D curves. *ISD*, 17-21.
10. Curtis, B. 2002. *Drawing from Observation: An Introduction to Perceptual Drawing*, McGraw-Hill.
11. Dorsey, J., Xu, S., Smedresman, G., Rushmeier, H., and McMillan, L. 2007. The Mental Canvas: a tool for conceptual architectural design and analysis. *Pacific Conference on Computer Graphics and Applications*, 201-210.
12. Edwards, B. 1999. *The New Drawing on the Right Side of the Brain*, Tarcher/Putnam.
13. Gibson, J. J. 1950. *The Perception of the Visual World*, Houghton Mifflin.
14. Google SketchUp. 2009. <http://sketchup.google.com/>.
15. Grossman, T., Balakrishnan, R., Kurtenbach, G., Fitzmaurice, G., Khan, A., and Buxton, B. 2001. Interaction techniques for 3D modeling on large displays. *ISD*, 17-23.
16. Grossman, T., Balakrishnan, R., Kurtenbach, G., Fitzmaurice, G., Khan, A., and Buxton, B. 2002. Creating principal 3D curves with digital tape drawing. *CHI*, 121-128.
17. Grossman, T., Fitzmaurice, G., and Attar, R. 2009. A survey of software learnability: metrics, methodologies and guidelines. *CHI*, 649-658.
18. Hanks, K. and Belliston, L. 2006. *Rapid Viz, Third Edition: A New Method for the Rapid Visualization of Ideas*, Thomson.
19. Igarashi, T., Matsuoka, S., and Tanaka, H. 1999. Teddy: a sketching interface for 3D freeform design. *SIGGRAPH*, 409-416.
20. Igarashi, T. and Hughes, J. F. 2001. A suggestive interface for 3D drawing. *UIST*, 173-181.
21. Kallio, K. 2005. 3D6B editor: projective 3D sketching with line-based rendering. *SBIM*, 73-79.
22. Kara, L. B. and Shimada, K. 2006. Construction and modification of 3D geometry using a sketch-based interface. *SBIM*, 59-66.
23. Karpenko, O., Hughes, J. F., and Raskar, R. 2004. Epipolar methods for multi-view sketching. *SBIM*, 167-173.
24. Kurtenbach, G. and Buxton, W. 1991. Issues in combining marking and direct manipulation techniques. *UIST* 137-144.
25. Lapidès, P., Sharlin, E., Sousa, M. C., and Streit, L. 2006. The 3D Tractus: a three-dimensional drawing board. *TABLETOP*, 169-176.
26. Laseau, P. 2001. *Graphic Thinking for Architects & Designers, Third Edition*, John Wiley & Sons.
27. Li, Y., Hinckley, K., Guan, Z., and Landay, J. A. 2005. Experimental analysis of mode switching techniques in pen-based user interfaces. *CHI*, 461-470.
28. Lin, F. and Hewitt, W. T. 1994. Expressing Coons-Gordon surfaces as NURBS. *Computer-Aided Design* 26, 2, 145-155.
29. Masry M., Kang D., Lipson H. 2005 A freehand sketching interface for progressive construction of 3D objects, *Computers & Graphics* 29, 563-575.
30. Matthews, P. C. 1998. *Vector Calculus*, Springer.
31. Nealen, A., Igarashi, T., Sorkine, O., and Alexa, M. 2007. FiberMesh: designing freeform surfaces with 3D curves. *SIGGRAPH*.
32. Nielsen, J. 1994. *Usability Engineering*, Morgan Kaufmann.
33. Piccolotto, M. A. 1998. *Sketchpad+ Architectural Modeling through Perspective Sketching on a Pen-Based Display*, Master's thesis, Cornell University.
34. Raskin, J. 2000. *The Humane Interface: New Directions for Designing Interactive Systems*, Addison-Wesley.
35. Robertson, S. 2004. *How to Draw Cars the Hot Wheels Way*, MBI.
36. Robertson, S. 2004. *Techniques of Scott Robertson Volume 2: How to Draw Cars*, DVD, Design Studio Press.
37. Robertson, S. 2004. *Techniques of Scott Robertson Volume 4: How to Draw Hovercraft and Spacecraft*, DVD, Design Studio Press.
38. Sachs, E., Roberts, A., and Stoops, D. 1991. 3-Draw: a tool for designing 3D shapes. *IEEE Computer Graphics and Applications* 11, 6, 18-26.
39. Schmidt, R., Wyvill, B., Sousa, M., and Jorge J. 2005. ShapeShop: sketch-based solid modeling with BlobTrees. *SBIM*, 53-62.
40. Schön, D. A. 1983. *The Reflective Practitioner*, Basic Books.
41. Tsang, M., Fitzmaurice, G. W., Kurtenbach, G., Khan, A., and Buxton, B. 2002. Boom chameleon: simultaneous capture of 3D viewpoint, voice and gesture annotations on a spatially-aware display. *UIST*, 111-120.
42. Tsang, S., Balakrishnan, R., Singh, K., and Ranjan, A. 2004. A suggestive interface for image guided 3D sketching. *CHI*, 591-598.
43. Zeleznik, R. C., Bragdon, A., Liu, C., and Forsberg, A. 2008. Lineogrammer: creating diagrams by drawing. *UIST*, 161-170.