LightSketch
A Sketch-Modelling Program for Lighting Analysis

Daniel GLASER[1], Jan VOUNG[2], Ling XIAO[2], Bonnie TAI[2], M Susan UBBELOHDE[3], John CANNY[2], Ellen Yi-Luen DO [4]

[1] *Interdisciplinary Doctoral Program, U.C. Berkeley, USA*
[2] *Computer Science Division, U.C. Berkeley, USA*
[3] *Department of Architecture, U.C. Berkeley, USA*
[4] *Department of Architecture, University of Washington, USA*

Abstract: This paper presents a flexible, yet powerful lighting analysis tool, LightSketch. LightSketch is a sketch-based modelling program for lighting analysis. It allows the user to draw both architectural and lighting related symbols which are converted into a 3D model for lighting visualisation. It is motivated by examining the strengths and limitations of current lighting design practices. Its use is illustrated with design scenarios.

1   INTRODUCTION.

This paper presents LightSketch, an exploratory, sketch-based tool for lighting design. We address lighting design as an integrative problem consisting of both electric and daylight systems. Both systems contribute visual spectra to reveal architectural space and each has its strengths and weaknesses. We argue much utility out of tools for lighting analysis can come from the combined analysis of both types of lighting systems.

For both daylight and electric light, visual performance is judged by the amount of light available, its distribution (including glare), and colour rendering properties (Illuminating Engineering Society et al. 2000; Ne'eman 1998)). Energy efficiency relates to how well daylight is used in lieu of electric light and the technologies employed by the electric lighting system. This is a complex problem since good placement of electric lights and controls requires knowledge about spatial and temporal daylight distributions and the occupants themselves. The following example illustrates some of the difficulties for creating an effective lighting system.

As one occupant of the Richards Medical Research Laboratories reported, "the sunlight came in blindingly and melted the ice in our ice buckets" (Cohen 2000). In

this case, although the architects allowed for ample light through the windows, at times sunlight was a source of glare and heat. To counter the fenestration system the occupants constructed ad-hoc shades, negating any benefit of daylight. Hence, they relied on stable electric light for their primary lighting source during the daytime. Occupants also have been known to turn up electric lights to compensate for uneven daylighting (Illuminating Engineering Society et al. 2000)! Hence both electric and daylight must be considered simultaneously to achieve a good visual and energy efficient environment.

Currently, tools that support daylight and electric lighting have generally been separated to reflect the division of labour in building design. Typically the architect is responsible for the fenestration system while the lighting designer or engineer is responsible for the electric lighting system. Consistent with socially informed methods for technological development (Beyer and Holtzblatt 1998), the next sections describe how 11 lighting professionals and educators mainly in the Western United States work with light.

1.1 Daylighting Analysis

An ideal daylighting analysis tool should allow a building designer to consider a broad range of both context and performance variables for a complex architectural model. As a source of light, daylight varies by time of day, season, yearly variations, and sky conditions. Daylight changes both in its colour rendering properties and also its intensity due to sun position and atmospheric effects such as rain or cloud cover. There is a rich tradition of tools used for daylighting analysis (Hopkinson 1966).

Physical models are in widespread use today (all 7 daylighting experts used them) due to their remarkable construction affordances and relative simulation accuracy (Benton 1990). Simple models are very easy to construct since they are cheap and directly manipulable. For example, a designer can use different shades of paper and tin-foil to understand the lighting of a house by a lake. The easiest way for a designer to simulate daylight is to bring the model outside. This gives nearly accurate rendition of current sky conditions. To simulate for other times of day, a model can be tilted to adjust for sun position (Figure a), although this decreases accuracy. Both qualitative and quantitative readings can be ascertained from a model (Figures 1a and 1b). Nevertheless, the power and ease of use of physical models do have their shortcomings.

Physical models simplify sky conditions, have information retrieval limitations, and do not model electric lights. Only a handful of artificial sky chambers exist in the United States for designers to approximate overcast days (in addition to a few clear sky simulators elsewhere) and hence architects' access to calibrated skies are limited. Even when used, they represent a simplification of possible sky conditions. Photographs provide a familiar and easy to understand representation, but to be

useful, one expert claimed to take hundreds of photos on a single job![1] The physical dimensions of a camera may also make it difficult to get the desired view and can introduce an error term into the lighting simulation. Film, like any medium, also distorts the lighting representation. As a result, sensors are frequently used to provide for unambiguous quantitative values. Illuminance sensors are particularly useful since they can generate daylight factors that describe a space's performance at a general level. Although powerful, "the number of steps involved in measuring, recording, adjusting, and presenting the data in a useful format make the process time-consuming and labour intensive" (Lam 1986). A last limitation of using physical models is that there are no practical ways of simulating electric light and controls.

CAAD has enabled architects and daylighting consultants to overcome many of the barriers with physical models (Figure 1c). Although quantitative and qualitative renderings of both daylight and electric light systems are possible, their interfaces are usually cumbersome to use. Interaction is limited to WIMP type interfaces[2] versus the design space afforded by physical materials. Hence, constructing a CAAD model can take the same time to build as a physical one[3].
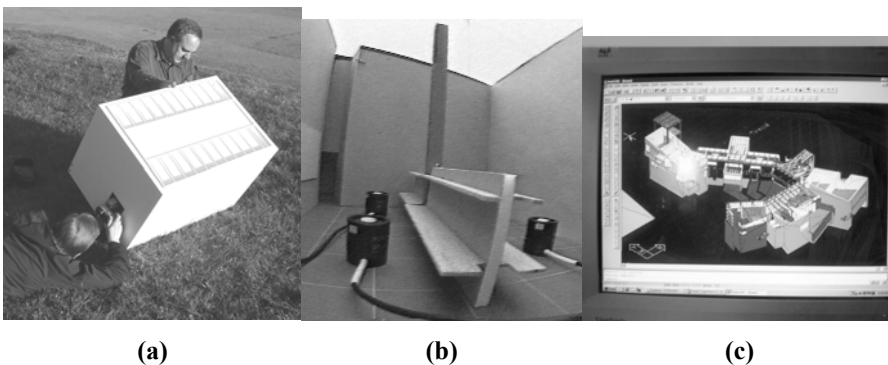

| (a) | (b) | (c) |

Figure 1 Daylighting analysis tools: a) a physical model outside; b) light Sensors placed in a physical model and c) a CAAD model.

1.2   Electric Lighting Analysis

An ideal electric analysis tool will use existing professional symbols for input, consider lamps and controls, manage both quantitative and qualitative data, and have the capacity for daylight analysis. Electrical lighting systems are usually designed by electrical engineers and/or lighting consultants. From interviews and examining

---

[1] Personal communication, Joel Loveland, Lighting Design Laboratory, Seattle

[2] WIMP is a term to describe standard Windows, Icons, Menus, and Pointing interfaces (see van Dam, A. (1997). "Post-WIMP user interfaces." *Communications of the ACM* **40**(2): 63-7. for more discussion)

[3] According to George Loisos, Loisos + Ubbelohde, Oakland, CA

their designed artefacts, they use sketch paper, simulation, and photorealistic renderings.

The process of electric lighting design typically begins with trace paper (Figure 2a). This allows lighting designers to quickly sketch their ideas with standardized symbols. Although this is a malleable process, it does not provide any information on lighting quality. Cumbersome simulation tools are employed when designers need predictive tools (Figure 2b). These programs also have WIMP interfaces that are cumbersome to use. They also have very little daylight integration capabilities. Only sparingly, will lighting designers resort to a high-quality lighting simulation package for rendering a concept. Of over 100 images looked at in lighting design offices, only 1 was a photorealistic rendering (Figure 2c)! This image was not used during the design stage, but only after to document a completed design since it was costly to build and simulate.
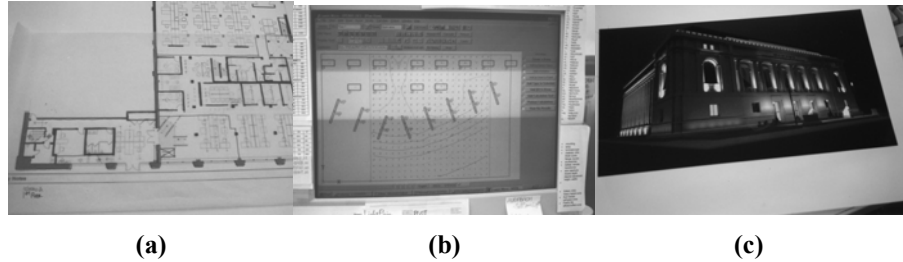

| (a) | (b) | (c) |

Figure 2 a) Sketching paper as a design tool; b) less frequent simulations are used, while c) rarely are photoreaslistic simulations employed.

1.3 Summary

Good simulation tools address both daylight and electric light issues. They provide both qualitative and quantitative analyses, are easy to use, and take into account the many variables for daylight simulation. A gulf exists between flexible design tools and those with sophisticated analysis capabilities. As a result, it is exceedingly difficult for designers to go through many design-and-test cycles with current technologies. We propose LightSketch, a program that will allow for architects to quickly sketch a space and see multidimensional lighting performance. It will use flexible interaction methods as a means for obtaining powerful performance measures for integrated lighting design.

2 RELATED WORK

We describe sketch-input, lighting simulation and analysis, visualisation and energy analysis tools to facilitate comparisons with LightSketch. Building designers like to draw since it is a quick way of communicating ideas and it allows for rapid revisions at low cost (Herbert 1993) (Fraser and Henmi 1994). Recognizing this, there have

been a number of CAAD (Gross and Do 1996; Tolba et al. 1999; Schweikardt and Gross 2000) and other geometric design systems ((Igarashi et al. 1997; Igarashi et al. 1999)) that employ sketch-based interfaces. There are now toolkits to support programmers who are building such applications (e.g. (Hong and Landay 2000; Long 2001)). This work looks to extend drawing as interface to lighting design and analysis.

(Jung et al. this volume) provide a sketch-based interface to a goal-directed lighting design tool. In this system, the user paints where they would like light to be, and the system suggests light fixtures depending on function of the lighting source. Its current version only works with electric light, but it is foreseeable that it could be extended to manage some aspects of daylight design. It could be used in conjunction with LightSketch.

RADIANCE (Ward 1994) is a high-quality global illumination tool. It provides a robust simulation environment and rendering features. Since it uses either meticulously constructed CAAD models or text-based input, it only has limited usability by design professionals. Nevertheless, this type of sophisticated lighting simulation can be used to increase the utility of many architectural lighting analysis tools that only use simplified skies and daylight factors. As the next section describes, LightSketch uses RADIANCE as a simulation engine.

Most lighting analysis tools can also be useful for energy analysis. Comparisons between designs by experts with programs that articulate lighting performance are common forms of evaluation. Full blown energy simulation programs like DOE-2 allow for quantitative performance measures, but have user acceptance barriers and simplified models. There were no users of this software among the lighting designers or daylight consultants interviewed. The video LightScapes (Sumption et al. 1991) presented a vision of how to illustrate the complex dynamics between the daylight and electric lighting systems, but it is not an interactive computer program. The next section describes LightSketch, an easy to use program that provides for sketch-based lighting analysis.

3   LIGHTSKETCH

LightSketch allows architects to quickly sketch a space and see its lighting performance. A small, but highly relevant vocabulary for lighting design is currently being developed in conjunction with architects and lighting designers. It supports freehand sketching and automatically infers a three-dimensional model for lighting analysis. It has links to both simulation and visualisation modules to allow a designer to see lighting performance.

3.1   User-Interface

LightSketch starts with an empty graphical canvas. As a sketch-based system, there is almost no "user-interface" per se, but instead the user can draw symbols on the

main drawing canvas interpretation. Similar to the observations of lighting systems in architecture, the user draws in plan. Each stroke that the user makes runs through an interpreter described in the next subsection. Strokes are highlighted different colours when they are matched with a vocabulary word. The vocabulary consists of both architectural and electric lighting symbols such as walls, windows, skylights, fluorescent lights, down lights, and sconces.

Figure 3 illustrates some of the sketch vocabulary LightSketch is familiar with. In Figure 3a, a single wall sconce is recognized when the user draws, in any order, two lines and one circle. In Figure 3b, two rectangles are recognized as different objects. As part of its fenestration grammar, LightSketch recognizes interior rectangles, as skylights while the lines constituting the exterior rectangle are walls. Figure 3c is similar to Figure 3b, but the lighting system is very different. Instead of using a skylight, an interior rectangle with a diagonal is recognized as a fluorescent light and two lines on the walls are each understood as windows.
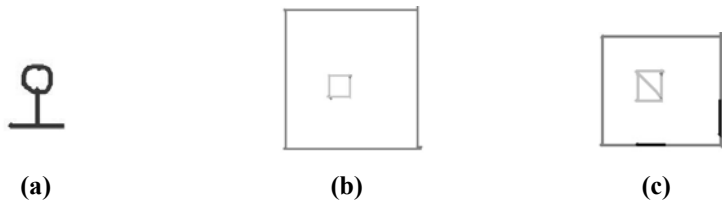
| (a) | (b) | (c) |

Figure 3 Some symbols used by LightSketch: a) a single wall sconce; b) 4 walls with a skylight; and c) 4 walls, 2 windows, and a fluorescent light.

After the user completes their model, LightSketch creates a visualisation from the drawing. Figures 4 and 5 illustrate two drawings and a corresponding visualisation. In Figure a, the user has sketched an electric lighting system for a room. This design includes two wall sconces and three down lights. Also noticeable is a wiring diagram connecting the luminaries with light switches. After the user presses a button, the sketch is converted into a 3D model for visualisation. In Figure b, this fully constructed model is viewed through the RADIANCE visualisation package. This sketch and visualise cycle illustrates the simplicity of providing a qualitative analysis for a simple electric lighting design.

Figure 5 illustrates another qualitative aspect of lighting performance. In Figure a, the user draws another sketch of a room with windows, skylights, and an electric light. The visualisation illustrated in Figure b shows the combined performance of daylight and electric lighting systems for a single moment in time. It also makes apparent contrasting colour rendering properties of the two light sources. In particular, one wall is rendered a warm colour by the skylight while another cool with the electric lamp.
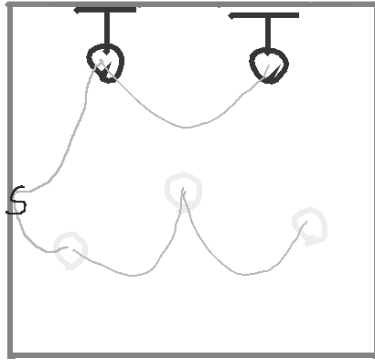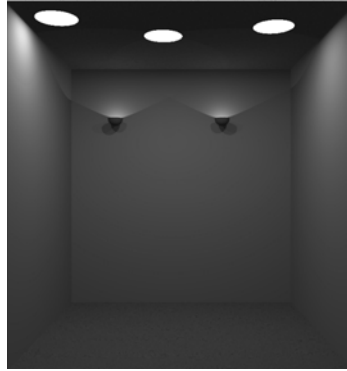
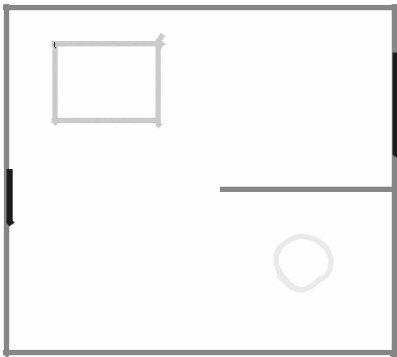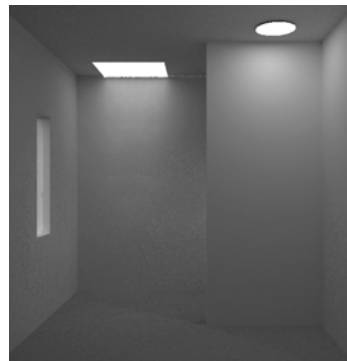|	
(a)	**(b)**

Figure 4 a) A plan of a square room with electric lights and controls drawn with LightSketch and b) a north-facing perspective generated with radiance.

(a)	**(b)**

Figure 5 a) A plan of a room with both daylight (square skylights and windows) and electric light (circular down light) and b) corresponding perspective.

Figure 6 shows a time varying visualisation of a model designed with an early version of LightSketch (Glaser and Ubbelohde 2002). It shows multiple snapshots of the room's quantitative performance at different times of day and seasons. Figure 6a shows a square model with a single south window. It illustrates how the available light changes by time of day and season. The top row of six snapshots represents a sequence spanning a morning hour across the year. This shows seasonal variations in lighting performance. A column represents daily variance. Figure 6b illustrates the consequences of adding a second window to the west side of the room.
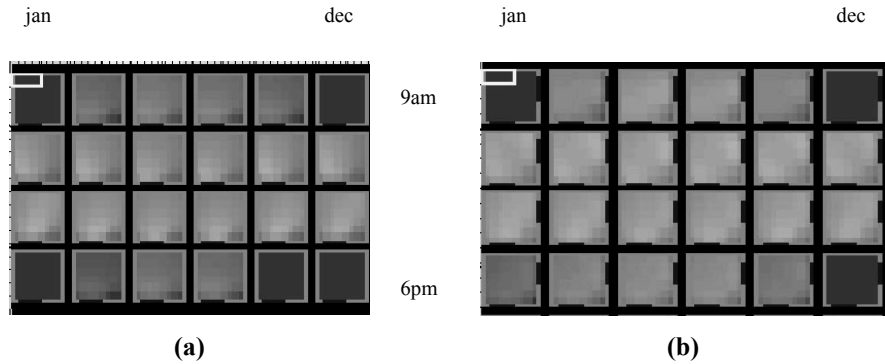
jan   dec   jan   dec

     (a)   (b)

Figure 6 Lightsketch interfacing with a daylighting analysis visualisation tool illustrating temporal trends in the data.

3.2   Software Architecture

LightSketch works in conjunction with a simulation engine and visualisation module (Figure 7). LightSketch itself is composed of four distinct modules: SATIN, the LightSketch Interpreter, a finishing module, and scene converter. These modules perform three main tasks - recognition of sketch-based input, modelling, and visualisation. Each in turn will be discussed.
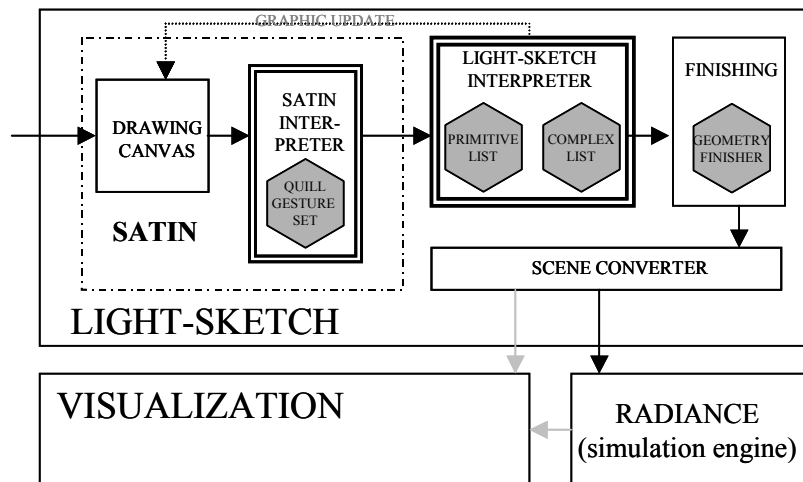

Figure 7 The main modules of LightSketch and its relationship to the simulation engine and visualisation routines. Grey arrows are under development.

SATIN (Hong and Landay 2000) is employed as a drawing canvas and stroke

378

recogniser. The strokes the user enters onto the canvas are parsed as *primitives* by SATIN through programmer defined interpreters. A primitive expresses a simple geometric or iconic form. For example, lines, circles, and "S"'s (for a light switch) are all SATIN interpreters that were built to recognize lighting vocabulary. Building a SATIN interpreter is a non-trivial task and QUILL (Long 2001) was employed to specify unique and unambiguous primitives.

After the primitives are recognized, they are passed to the LightSketch Interpreter to create *complex* objects. Complex objects consist of primitive or other complex objects associated with each other through one or more *constraints*. A constraint is a relationship such as touching, inside of, and 'T' connects. There also is an identity constraint in the case of a wall that is formed by a single line. In the case of Figure 3a the sconce is formed by a line 'T' connect another line touching a circle. The relationship between primitive and complex geometry is bi-directional—namely geometry can be re-converted into primitives. In the wall sconce example, if the two lines were first recognized as walls and the circle is recognized as touching, then the wall association is removed and only a sconce object remains.

A finishing module converts the symbols created by sketching into 3D-form suitable for lighting simulation. It determines a suitable roof and floor slab through a convex hull algorithm. It then assigns default reflectance values for all surfaces.

A scene converter allows for LightSketch to interface both with a lighting simulation and visualisation package. Currently, there are links to RADIANCE and its standard visualisation package. LightSketch communicates with RADIANCE through file I/O and invokes its scene viewer on demand. RADIANCE was chosen due to its accuracy and power for simulating light. Currently, LightSketch is being reintegrated into the visualisation routines shown in Figure 6.

4   CONCLUSIONS AND FUTURE RESEARCH

In this paper, we have examined the practices of lighting design and analysis. Highlighted, we have proposed an easy to use, yet powerful lighting analysis tool LightSketch. It allows a designer to quickly see a range of lighting performance variables that would otherwise be tedious to accomplish using other methods.

There are number of areas under current development. First, the sketch vocabulary is being broadened and refined in conjunction with lighting designers to handle other symbols such as sensors, light shelves, and window shades. Also, specification of more complex 3D shapes using simple sketch-based operators is also being examined since both the simulation and visualisation modules can manage practically any set of geometry or lighting source. Performance details related to simulating light across multiple times/days/sky conditions are also being resolved. Further refinements to the sketcher related to specifying occupant views and schedules are being planned. The visualisation routines are being upgraded to better assess energy consequences. Lastly, specifying surface materiality is also a current area of investigation.

ACKNOWLEDGEMENTS

REFERENCES

Benton, C. 1990. Diminutive Design. *Lighting Design + Application,* 20(5): 10.

Beyer, H. and K. Holtzblatt. 1998. *Contextual Design: Defining Customer-cantered Systems*. San Francisco, Calif., Morgan Kaufmann.

Cohen, J. 2000. Architecture Discovers Science. *Science,* 287(5451): 210-214.

Fraser, I. and R. Henmi. 1994. *Envisioning Architecture: an Analysis of Drawing*. New York: Van Nostrand Reinhold.

Glaser, D. and S. Ubbelohde. 2002. Techniques for Managing Planar Daylight Data. *Building and Environment,* 37(8-9): 825-831.

Gross, M. and E. Do. 1996. Ambiguous Intentions: A Paper-like Interface for Creative Design. *User Interface Software Technology UIST 96, Seattle*, 183-194. New York: ACM Press.183-194

Herbert, D. M. 1993. *Architectural Study Drawings*. New York: Van Nostrand Reinhold.

Hong, J.I. and J.A. Landay. 2000. SATIN: A Toolkit for Informal Ink-based Applications. *User Interfaces and Software Technology, San Diego*, 63-72. New York: ACM Press.

Hopkinson, R. 1966. *Daylighting*. London: Heinemann.

Igarashi, T., S. Matsuoka, S. Kawachiya and H. Tanaka. 1997. Interactive Beautification: A Technique for Rapid Geometric Design. *ACM Symposium on User Interface Software and Technology (UIST) 1997*, 105-14. ACM Press.

Igarashi, T., S. Matsuoka and H. Tanaka. 1999. Teddy: A Sketching Interface for 3D Freeform Design. *SIGGRAPH 99*, 409-16.

Illuminating Engineering Society, M. S. Rea and Illuminating Engineering Society of North America. 2000. *The IESNA Lighting Handbook: Reference & Application*. New York, N.Y.: Illuminating Engineering Society of North America.

LightSketch

Jung, T., M. Gross and E. Do. 2003. Light Pen, Sketching Light in 3D. *Proceedings of CAADFutures 2003: Digital Design: Research and Practice,* eds. M-L. Chiu, J-Y. Tsou, T. Kvan, M. Morozumi and T-S. Jeng, 495-505. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Lam, W.M.C. 1986. *Sunlighting as Formgiver for Architecture*. New York: Van Nostrand Reinhold.

Long, A., Quill. 2001. *A Gesture Design Tool for Pen-based User Interfaces*, Ph.D. Thesis. U.C. Berkeley.

Ne'eman, E. 1998. Quality of Integrated Daylight and Electric Light. *Proceedings of the First CIE Symposium on Lighting Quality, Ottawa, Canada*, 206-210.

Schweikardt, E. and M. Gross. 2000. Digital Clay: Deriving Digital Models from Freehand Sketches. *Automation in Construction,* 9(1): 107-115.

Sumption, B., B. Haglund and A. Zabrodsky. 1991. Imagining Light: A Visualisation Of Daylighting Data. *Proceedings of CAADFutures 1991,* ed. G.N. Schmitt. 97-104. Zurich: Braunschweig/ Wiesbaden: Vieweg Verlag.

Tolba, O., J. Dorsey and L. McMillan. 1999. *Sketching with Projective 2D Strokes*, 149-57.

Ward, G. 1994. The Radiance Lighting System. *Computer Graphics,* 28(7): 459-72.