

Dartmouth College

## Dartmouth Digital Commons

---

Computer Science Technical Reports

Computer Science

---

11-1-2008

# Toward Evaluating Lighting Design Interface Paradigms for Novice Users

William Brandon Kerr  
*Dartmouth College*

Fabio Pellacini  
*Dartmouth College*

Follow this and additional works at: [https://digitalcommons.dartmouth.edu/cs\\_tr](https://digitalcommons.dartmouth.edu/cs_tr)



Part of the [Computer Sciences Commons](#)

---

### Dartmouth Digital Commons Citation

Kerr, William Brandon and Pellacini, Fabio, "Toward Evaluating Lighting Design Interface Paradigms for Novice Users" (2008). Computer Science Technical Report TR2008-636.  
[https://digitalcommons.dartmouth.edu/cs\\_tr/318](https://digitalcommons.dartmouth.edu/cs_tr/318)

This Technical Report is brought to you for free and open access by the Computer Science at Dartmouth Digital Commons. It has been accepted for inclusion in Computer Science Technical Reports by an authorized administrator of Dartmouth Digital Commons. For more information, please contact [dartmouthdigitalcommons@groups.dartmouth.edu](mailto:dartmouthdigitalcommons@groups.dartmouth.edu).

# Toward Evaluating Lighting Design Interface Paradigms for Novice Users

William Brandon Kerr and Fabio Pellacini

Dartmouth Computer Science Technical Report TR2008-636

---

## Abstract

*Lighting design is a complex and fundamental task in computer cinematography, involving adjustment of light parameters to define final scene appearance. Many lighting interfaces have been proposed to improve lighting design work flow. These paradigms exist in three paradigm categories: direct light parameter manipulation, indirect light feature manipulation (e.g., shadow dragging), and goal-based optimization of light through painting. To this date, no formal evaluation of the relative effectiveness of these methods has been performed. In this paper, we present a first step toward evaluating the three paradigms in the form of a user study with novice users. We focus our evaluation on simple tasks that directly affect lighting features, such as highlights, shadows and intensity gradients, in scenes with up to 2 point lights and 5 objects under direct illumination. We perform quantitative experiments to measure relative efficiency between interfaces together with qualitative input to explore the intuitiveness of the paradigms. Our results indicate that paint-based goal specification is more cumbersome than either direct or indirect manipulation. Furthermore, our investigation suggests improvements to not only the implementation of the paradigms, but also overall paradigm structure for further exploration.*

---

## 1. Introduction

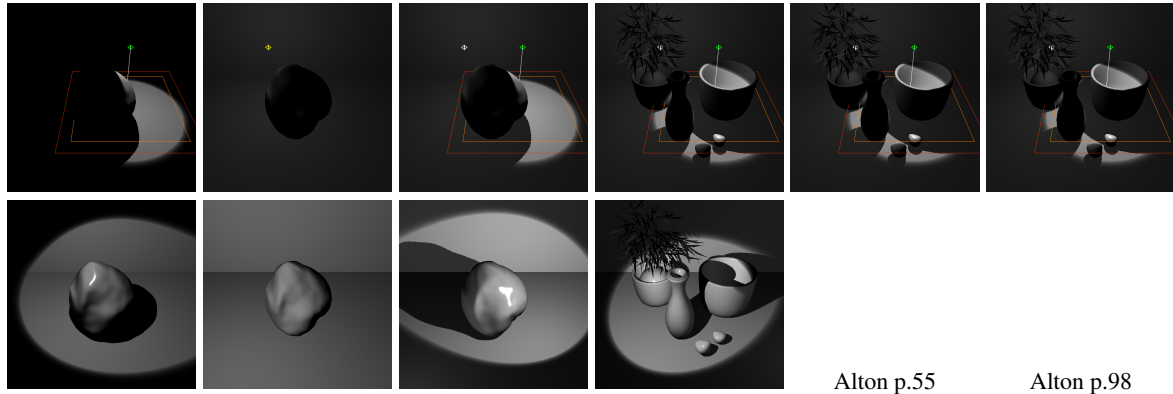
Lighting is a fundamental aspect of computer cinematography that establishes mood and enhances storytelling [Cal99]. Lighting design, the process by which artists place lights in the scene to achieve a final look, is a complex and labor intensive process. Expert lighters often take days to carefully light a shot in feature film animation. More importantly though, novice users are not capable of effectively lighting scenes since they lack the technical training required to effectively manipulate lights.

Various user interfaces have been presented to address the complexity of the lighting design task for novices. Focusing on the editing of point lights, we categorize these interface implementations into three main lighting design paradigms: *direct* light parameter manipulation, *indirect* light feature manipulation, and goal-based optimization of lighting through *painting*. Each of these interfaces represents a different metaphor for lighting design. Direct interfaces involve direct modification of the light source; these are common interfaces used in commercial software such as Maya [Aut07]. Indirect interfaces consist of click-and-drag modifications of lighting features such as position and size

of shadows, hotspots, and highlights [PF92, PTG02]. Painting interfaces use an optimization algorithm to adjust light parameters, minimizing the difference between the rendered image and a user-painted one [SDS\*93, PBMF07]. To this date, no formal evaluation of these paradigms has been presented.

This paper presents a first step toward quantitatively evaluating the relative effectiveness of these interface paradigms. We specifically focus on novice users with no prior experience in lighting, since they are the majority of potential users and since they receive the most benefits from the introduction of intuitive interfaces. Out of the broad scope of lighting design, we focus our attention on the manipulation of a small number of point light sources, rather than attempting to evaluate how subjects manipulate the hundreds of light used in computer cinematography. We choose to focus on this simplified lighting task since our focus is on users with no prior lighting experience, thus we want to ensure that the design task is manageable. Furthermore, we reduce the complexity of the user interface implementations tested to focus solely on the key features of each paradigm.

We perform a user study with 18 novice subjects, who



**Figure 1:** From left to right: trials 01, 02, 03, 04, 05, and 06; top: initial configuration; bottom: goal image.

are asked to perform a variety of lighting tasks using implementations of all three interface paradigms. The study consists three parts. First, we ask subjects to manipulate lighting configurations to match exact target images, allowing us to quantitatively measure the effectiveness of each interface. Second, we ask subjects to design lighting configurations based on suggested appearance, evaluating how each paradigm supports artistic exploration. Third, we ask user to fill in a series of questionnaires, collecting usability ratings, preferences, and comments regarding each interfaces.

From the data collected in the study, we conclude that painting interfaces can be slow and cumbersome compared to the parameter-space exploration found in direct or indirect interfaces. However, subjects perform just as well using direct interfaces than indirect interfaces. Finally, observations of painting paradigm suggest that interface implementation that semantically indicate lighting features, such as shadow or highlight brushes, are likely to be much more effective than pixel based painting.

## 2. Related Work on Lighting Design Paradigms

**Direct Interfaces.** Interfaces based on a direct manipulation paradigm, widely used in commercial software [Aut07, Avi06], require users to select lights and directly modify their individual properties. For example, a light can then be moved and reoriented in the scene by clicking and dragging it, or have properties such as intensity modified with a slider.

**Indirect Interfaces.** Interfaces based on an indirect manipulation paradigm allow users to directly interact with lighting features as they appear on object surfaces. Illumination hot spots, shadows, and specular and diffuse highlights can be adjusted by dragging and scaling them across surfaces, without the need to explicitly edit light parameters. Poulin and Fournier [PF92] allow users to manipulate the shadow volume to place shadows, while specular highlights are specified by clicking points on surfaces. More recently, Pel-

lacini et al. [PTG02] showed how users can directly move and scale shadows and hotspots on objects surface using a simple click-and-drag interface. Our implementation of an indirect paradigm is based on this latter work.

**Painting Interfaces.** Interfaces based on a painting paradigm further abstracts the idea of lighting by requiring users to paint a desired goal image that is then matched by optimizing light parameters to minimize the difference between the painted image and the rendered one. Subjects paint directly onto the scene to ensure that the painted image is close to one generated by the renderer. Schoeneman et al. [SDS\*93] used painted input as a goal for setting intensities of lights of known position in a global illumination renderer. Anrys and Dutré [AD04] and Mohan et al. [MTB\*05] use a similar approach to relight real objects whose appearance is captured using image-based lighting techniques. Poulin et al. [PRJ97] use sketches of shadows and highlights to place point lights for ellipsoid geometry. Recently, Pel-lacini et al. [PBMF07] presented a general painting interface for direct illumination of arbitrary scenes where all light parameters are derived using an efficient non-linear optimization framework. Our implementation of a painting interface is based primarily on this latter work, but with a simplified painting toolset.

Several researchers have investigated methods for optimizing lighting-related parameters in order to achieve a variety of other goals, as surveyed in [PP03]. In the context of lighting design, Kawai et al.'s method [KPC93] maximizes the subjective impression of scene qualities (e.g., pleasantness or privateness), while Shackled and Lischinski [SL01], Gumhold [Gum02], Lee et al. [LHV04] and Shesh and Chen [SC07] maximize low-level perceptual qualities for visualization. Costa et al. [CSF99] explores the use of even more complex constraints. Rather than asking users to directly specify a lighting goal, Marks et al. [MAP\*00] supports parameter exploration by generating many possible goals and letting the user choose the best options.

### 3. Study Overview

**Goal.** We seek to evaluate the relative efficiency of different interface paradigms in the context of simple lighting setups with a specific focus on novice users. Specifically, (1) we want to measure how quickly users can perform specific lighting adjustments and (2) we want to understand which interface paradigms provide a more intuitive interpretation of the lighting design task as a whole.

**Subjects.** We selected subjects with no prior knowledge of lighting design since they make up the majority of potential users and since they are the most likely candidates to take advantage of intuitive interfaces given their lack of conditioning on a single interface or work flow. Furthermore, we believe that studying novices allows us to understand how intuitive lighting design paradigms since such conditioning can bias opinions. Given novice subjects, we restrict the lighting and scene complexity to sufficient simplicity such that lighting effects and their relationship with the geometric composition of the scene are clear.

**Interfaces.** We compare three user interfaces, each following a major paradigm: direct, indirect and painting. We simplify the implementation each interface, described in Sec. 4, to ensure that it is simple enough to be quickly understood by novices while sufficiently complete enough to capture the main characteristics of each paradigm. Furthermore, in choosing the details of each implementation we attempt to focus user judgments on the general characteristics of each paradigm, rather than the details of the particular implementation.

**Scenes.** We include two scenes in our experiments, shown in Fig. 1: a scene containing an abstract *blob* and a *still life* scene with various realistic objects (vase, basin, plant, and smooth stone). We chose the geometry of the surfaces following [VLD07] guidelines, making sure to include variations in complexity and curvature. In both cases the objects are placed on a floor plane with a back wall plane to establish spatial reference points. All materials in the scene are lit with the Phong illumination model. As in traditional introductory photography education on lighting, we limit all scene elements to grayscale, simulating black-and-white photography. This aids subjects’ understanding and recognition of lighting features, as they do not have to factor in hue blending.

**Lights.** We limit the possible configurations of single lights by presenting subjects with two different types of lights typical of computer cinematography [Cal99]: *key lights* and *fill lights*, shown in Fig. 2. In our implementation, key lights are represented as spotlights and cast shadows. They have seven degrees of freedom: position, orientation, intensity, and cone angle. Fill lights are represented as omni-directional point light sources that do not cast shadows or create specular highlights, following common practices in computer cinematography. They have four degrees of freedom: position and intensity. All lights are without distance falloff.

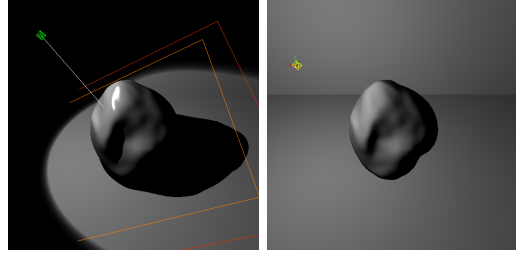


Figure 2: Left: key light. Right: fill light.

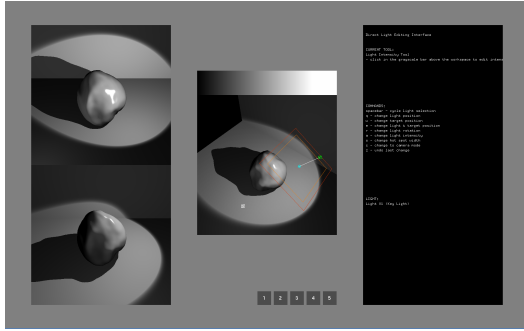
**Tasks.** To achieve our goals, we ask users to perform two types of lighting tasks. During *matching trials*, users are asked to match a given scene and set of lights to an image of the same scene under a target lighting configuration. Matching trials allow us to quantitatively measure users’ performance, while providing a clear goal for subjects who have never experienced lighting design before. This provides context for the more subjective *open trials*, where users are given an image from a movie set lit with a specific style and asked to light an unrelated still life scene with the same style. Given the differences in scene geometry, open trials require users to light the scene with a different lighting configuration than the one used for the goal. These trials allow us to observe how users explore the space of possible lighting configurations, a more natural but harder to measure task. We perform four matching trials of progressively more complex scenes, the first two designed to familiarize the subject with each type of light. We perform two open trials with different goal styles. Details of each trial are summarized in Tab. 1 and Fig. 1. By providing these two different contexts, subject can make a more educated and broader evaluation of the usability of each interface. After performing the trials, we ask subjects to fill out questionnaires that gather comments, ratings, and rankings in a variety of categories, as further discussed in Sec. 5.

Trial	Geometry	Lighting
Matching Trial 1	blob	1 key light
Matching Trial 2	blob	1 fill light
Matching Trial 3	blob	1 key and 1 fill light
Matching Trial 4	still life	1 key and 1 fill light
Open Trials 1 and 2	still life	1 key and 1 fill light

Table 1: Geometry and lighting configuration for each trial.

### 4. Lighting Interface Implementation

We balance complexity with functionality in the implementation of the three lighting design interfaces to ensure rapid learning, while retaining the core functionality that distinguishes each of the respective paradigms. This section serves as a quick overview of our implementations and their moti-



**Figure 3:** *Common interface layout*

variations. To finalize specific implementation details, we performed informal user studies of various implementation options to choose the most usable ones. One simplifying assumption we took in our implementations is to perform light selection explicitly for all paradigms. One of the claimed benefits of indirect and painting interfaces is that they do not require such explicit selection for complex setups when large quantities of lights make selection infeasible. However, in our preliminary experiments we found that due to the simplicity of the lighting setups, consisting of maximum of two lights, such benefit is negligible and outweighed by longer training. We thus leave the evaluation of light selection to future work.

**Interface Layout.** We use the same screen layout for all three interfaces divided into 4 windows (Fig. 3). All edits are carried out in the workspace window, centered on the screen. The text window on the right side displays a descriptive list of commands for the current interface to avoid the need for memorization, together with the name of the currently selected light. On the left side of the screen, there are two windows stacked directly on top of each other. The top-left displays the goal image for the current trial. The bottom-left displays an active view of the current lighting configuration in the scene from the default camera configuration, which matches the goal image camera viewpoint at all times.

**Common Interface Features.** Keyboard shortcuts are used to select different tools in each interface. To speed up learning, operations common to all interfaces have identical controls. We allow the camera to be freely modified in the workspace window and support the undo of the most recent edit, restoring light parameters for direct and indirect light editing, and clearing the last paint stroke in the painting interface. Light selection is performed by cycling over available lights with a key press.

**Direct Light Editing.** The direct light editing interface is implemented in a manner similar to Xsi [Avi06] and supports the following operations: (1) light translation; (2) rotation of spotlight’s cone by translation of a target point; (3) translation of light and target simultaneously; (4) rotation of

spotlight’s cone by axis-aligned rotation; (5) scaling of spotlight’s cone width; (6) intensity selection using a grayscale palette. All manipulations are performed in world coordinates since we found other choices, such as local or camera coordinate spaces, to be confusing for novices.

**Indirect Light Editing.** The indirect lighting interface is implemented in a manner similar to [PTG02] and supports the following operations: (1) shadow translation and scaling; (3) spotlight hotspot translation and scaling; (5) diffuse and (6) specular highlight location and scale; (7) light intensity selection as in the direct interface.

**Light Painting Implementation.** Our painting interface uses the non-linear optimization method of [PBMF07], but with a simplified set of brushes. In particular, we provide a circular brush similar to the ones in Adobe Photoshop [Ado07] that allows only brightening or darkening, requiring only one brush type. We found such a semantic particularly easy since no tool selection is needed. The user indicates whether the painted image refers to all lights or only the selected one. All active paint can be cleared to black, preserving contributions from unselected lights. The current rendered image can be copied to the paint layer to facilitate refinement. Due to the locality of non-linear optimization, we allow users to reset the optimizer when stuck in a local minimum. This resets the simplex to a starting configuration centered around an estimated configuration. Subjects are told this is like “giving the computer a kick in the right direction when it gets stuck.”

We differ from published work in that we run the optimizer continuously while the user is painting rather than waiting for the full optimizer convergence. This gives users interactive feedback while editing and turned out to be significantly superior to a paint-then-optimize workflow. We noticed that subjects use this feedback to “steer” the optimization toward their desired goal similarly to user assisted optimization methods. We also differ from [PBMF07] in that we do not support advanced brushes, such as gradient and shadow painting. We found these to be confusing given the short user training, but acknowledge that they could be very useful in more complex scenarios.

## 5. Experimental Methodology

**Subjects.** 18 novice subjects participate in the study chosen from different age and educational groups. All subjects were over the age of 18 and had normal or corrected-to-normal vision. On a scale 1 to 5, subjects ranked on average their previous experience with digital lighting design as 1.3 and their experience with real-world studio lighting as 1.1, with 1 being least experience.

**Study Environment.** All trials were conducted in a controlled lighting environment with fluorescent lighting and no outside illumination, to simulate typical working conditions of lighting artists. We used a widescreen Dell 2407WFPb

LCD at 1920x1200 resolution with a view size of 24 inches across the diagonal at a distance of roughly 1 foot from the subject. All rendered images are 512x512 pixels on screen and approximately 7.7 inches across the diagonal.

**Trials.** All subjects complete the entire study in 3 sessions of approximately 60 minutes each. Each session features only one interface, and we randomize the order of the interfaces for each subject ensuring all permutations were explored exactly three times. We record all user activities in-interface, allowing us to analyze user activities. Before carrying out the experiment, subjects complete a training phase to familiarize themselves with the various tools for the lighting design interface. An investigator explains the types of light in the experiment, how the interface is used to manipulate them and answers subjects’ questions. To ensure understanding, we ask subjects to experiment with each interface feature before the trial begins.

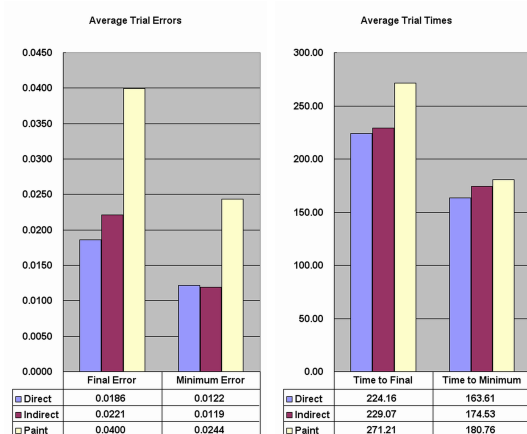
**Matching Trials.** Subjects complete 4 matching trials (Fig. 1). The same goal configuration and initial lighting setups are used for all subjects and all interfaces. Subjects are given 5 minutes to light the target scene. When either time has run out, or the subject decides his task is complete, he chooses a rating between 1 and 5 indicating his satisfaction with the accuracy of the matching.

**Open Trials.** After matching trials, subjects complete 2 open trials of maximum 10 minutes each. When either the user is satisfied or time runs out, he assigns a rating from 1 to 5 indicating how closely he has achieved his interpretation of the style. We choose one high key and one low key lighting goal from [Alt95] (pp. 55 and 98 respectively) to force two different interpretations of the scene.

**Questionnaire.** After completing each session, the subject is given a questionnaire where he is asked to rate the interface on a scale of 1 to 5, 1 been worst, 5 being best, in the following areas: easy to learn, natural way to think about light editing, easy to use, and work speed. Finally, after completing all sessions, the subject is asked to strictly rank each of the three interfaces in order of each of the categories on the interface rating questionnaire as well as in the order of his general usage preference. For each interface, Subjects are also asked to leave comments about various aspects of the interface, under the following guidelines: “What did you like/dislike about this interface?”. “Which tools helped the most/least in this interface?”, “Did you find this interface easier to use on the matching trials or on the open-ended trials?” To ensure reproducibility copies of the questionnaires are included in the additional materials.

## 6. Analysis

We present our results in 2 parts. First, we analyze the output of the rendering system as subjects proceed through each trial. Second, we compile the input provided by users in the questionnaires. All tests for statistical significance



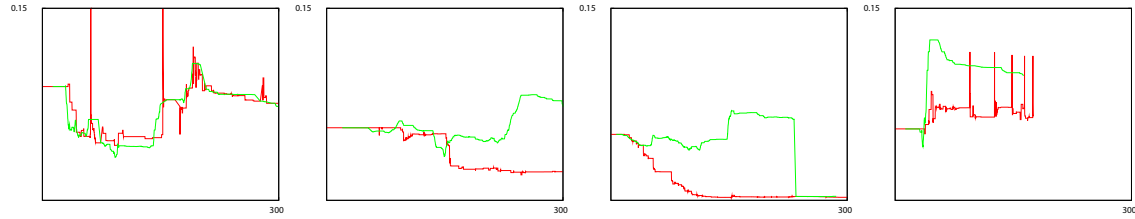
**Figure 5:** Data averaged over all subjects and trials. Time refers to the time in seconds to the respective final or minimum error. Direct and indirect finish faster and with lower error than painting, but do not stand out from one another.

are computed with Kruskal-Wallis analysis of variance. The Kruskal-Wallis method does not assume normality and tests if the mean ranks between sets of measurements are significantly different. A  $p$  value below 0.05 indicates a 95% chance that the two sets differ.

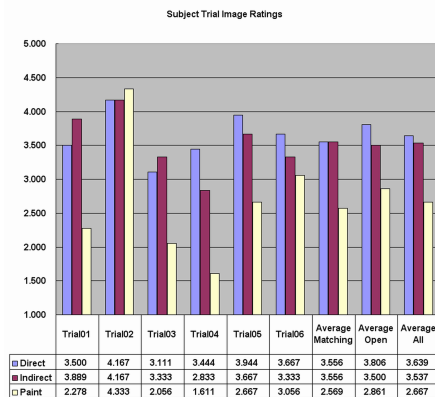
In matching trials, the goal image provides a definite reference for error calculation. We compute the average  $L^2$  error between pixels in the subject’s rendered image and the goal image during each trial for each subject. This error data is plotted over time, and examples can be seen in Fig. 9. Generally, graphs tend to decrease in error over time, showing that users are converging near the correct solution. Convergence is interrupted by moments where users explore local configurations, often with a new tool.

In Fig. 5 we show a comparison of the average trial errors and times over all subjects. We can observe that direct and indirect outperform painting in minimum error ( $p < 0.0001$ ), final error ( $p < 0.0001$ ), and time to final ( $p < 0.012$ ). The direct and indirect interfaces are virtually equals in terms of minimum error ( $p = 0.849$ ). Final error is greater than minimum error in general. This is because subjects have tendencies to experiment with other lighting options after finding a good solution. While indirect may have a higher average final error than direct ( $p = 0.100$ ), we attribute the difference in final errors between direct and indirect to time limits cutting exploration short before the user can return back to a previously acquired acceptable configuration.

In Fig. 4 we present some typical examples of error in painting interface trials. There exists a trend of user paint error increasing well above the error output by the optimizer. We believe this is an attempt by the user to steer the optimizer by painting extreme features that do not resemble ac-



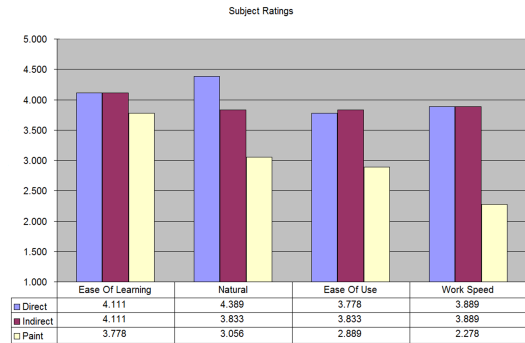
**Figure 4:** Typical error graphs for the light painting interface comparing the error of the subject's rendered image (red) with the error of the subject's paint to the goal image (green). From left to right: subject 02 - trial 03, subject 03 - trial 01, subject 04 - trial 02, and subject 05 - trial 04.



**Figure 6:** Average user rating given to the image at the end of each trial. Direct and indirect rate similarly except on trial 4. Paint rates lower than other interfaces except on trials 2 and 6.

tual gradients or gray levels in the target scene. Sometimes this approach yields positive results, as with subject 03 on trial 01 and subject 04 on trial 02. However, it most often yields negative results as with subject 02 on trial 03 and subject 05 on trial 04. Typically, these extreme edits over small local pixel areas do not have the intended impact on the painting optimizer, which operates on a pixel by pixel basis.

In addition to the actual error between a subject's lighting configuration and the goal image, we measure the perceived error by having the subject rate his or her own outcome at the end of each trial on a scale of 1 to 5. In figure 6 we present the average rating given to each trial and interface over all subjects. For trial 1, 3, 4, and 5 subjects tend to rate images produced by the paint interface lower in relation to the other interfaces ( $p \leq 0.019$ ). Trial 2 ratings are roughly the same for all interfaces ( $p \geq 0.501$ ). Indirect and paint interfaces rate closely for trial 6 ( $p = 0.433$ ), but direct rates higher than paint ( $p = 0.104$ ). Direct may rate higher than indirect on trial04 ( $p = 0.151$ ), but they rate similarly on all other trials ( $p \geq 0.331$ ).



**Figure 7:** Average rating given in each category on the interface rating questionnaires. Direct and indirect rate higher than painting except in ease of learning. Direct and indirect rate evenly to one another except in the natural category.

Fig. 7 shows the average rating given to each interface in the various categories from the questionnaire. Subjects favored the direct and indirect interfaces over the painting interface in the natural, ease of use, and speed categories ( $p \leq 0.061$ ), but did not favor any interface in the ease of learning category ( $p \geq 0.335$ ). No clear distinction was indicated between direct and indirect ( $p \geq 0.73$ ), except in the category indicating whether the interface was a natural way to think about light editing ( $p \leq 0.065$ ), for which the direct interface scored highly.

Subjects were also asked to rank each interface relative to each other using the numbers 1, 2, and 3, where 1 indicated strongest and 3 indicated weakest preference. Fig. 8 shows the resulting tallies for each interface in each category. Even in this case, direct and indirect interfaces are closely matched compared to painting, but the direct interface receives many more 1st rankings than indirect.

## 7. Discussion

We see that direct and indirect outperform painting, but not one another. This implies that the editing task for low complexity lighting is generally the same whether you are drag-

Rank	Ease Of Learning			Natural			Ease Of Use			Work Speed			General Preference		
	<i>i</i>	<i>ii</i>	<i>iii</i>	<i>i</i>	<i>ii</i>	<i>iii</i>	<i>i</i>	<i>ii</i>	<i>iii</i>	<i>i</i>	<i>ii</i>	<i>iii</i>	<i>i</i>	<i>ii</i>	<i>iii</i>
Direct	10	3	5	10	6	2	11	4	3	9	6	3	10	5	3
Indirect	4	10	4	6	9	3	4	11	3	7	7	4	5	10	3
Paint	4	5	9	2	3	13	3	3	12	2	5	11	3	3	12

**Figure 8:** Each row represents an interface. Each column is a tally of the number of times an interface received the indicated rank. 1 implies best, 3 implies worst. Direct is generally preferred to indirect, and both are generally preferred to painting.

ging lights at the source or indirectly by their features in the scene. Because the painting paradigm makes the space of possible edits difficult to explore, it falls short of the other two. While it is apparent that most subjects do not favor using the painting interface, it is unclear as to whether they ultimately prefer direct or indirect lighting to a significant degree. However, they tend to prefer the direct interface when forced to choose. Two subjects commented that they found the direct lighting interface more familiar at first, because they recognized a direct physical analogue to moving light objects around in real life. One of those subjects went on to comment that he felt he got better results with the indirect interface even though it felt less natural. While there may be a difference in usability and comfort that affects subjects' ratings, the error graphs still show no clear advantage of indirect over direct in terms of performing the trials.

Many of our subjects complained about the "jumpiness" of the indirect interface. Edits using the indirect interface depend on being able to move light features across the surfaces of objects. When a feature must be placed in an awkward place or dragged between surfaces with complex geometry, edits become more difficult. The behavior of our subjects indicates an exploratory approach to the lighting design task. This follows from the fact that their inexperience tells them little about what an acceptable configuration might be. They behave much like an optimizer, in that they move some knob one way until the image looks good, past the good configuration, and then back again to a stable state. It makes sense that an interface that provides more smooth edits would be more desirable to an artist who works in this way. The smoothness of indirect edits depend both on interface implementation and the complexity of scene geometry.

The study shows the painting interface to be the slowest and most error-prone of the three interfaces. It takes time to paint an accurate image of the desired scene, and even more time when there is not a perfect goal image to use as a reference. Several subjects commented that the matching trials were much easier than the open-ended trials with the painting interface. One commented that in the open-ended trials, she knew how she wanted the scene to look, but didn't know if the lights could achieve that effect. The error graphs and subject behaviors indicate that painting accurate lighting features is a more difficult task than that posed by the direct and indirect interfaces. Moreover, subjects expect to be able to roughly sketch lighting features with little intensity

matching and have those features appear. We note that most subjects were able to converge quickly on the correct lighting configuration in trial 02 using any of the interfaces. The painting interface performs well because that trial contains only soft lighting features, which are easier to paint rough representations of without needing to place a sharp shadow or hot spot.

## 8. Conclusions

We have presented a first step toward the evaluation of lighting design interface paradigms as part of an applied workflow. The low-complexity lighting task is generally the same whether you are dragging lights at the source or indirectly by their features in the scene, but both perform better than paint-based optimization. Our observations suggest that a semantic-based goal specification for optimization might be more appropriate for local light feature shaping than painting pixel intensities. Novices tend to prefer interfaces that allow the space of possible edits to be explored easily.

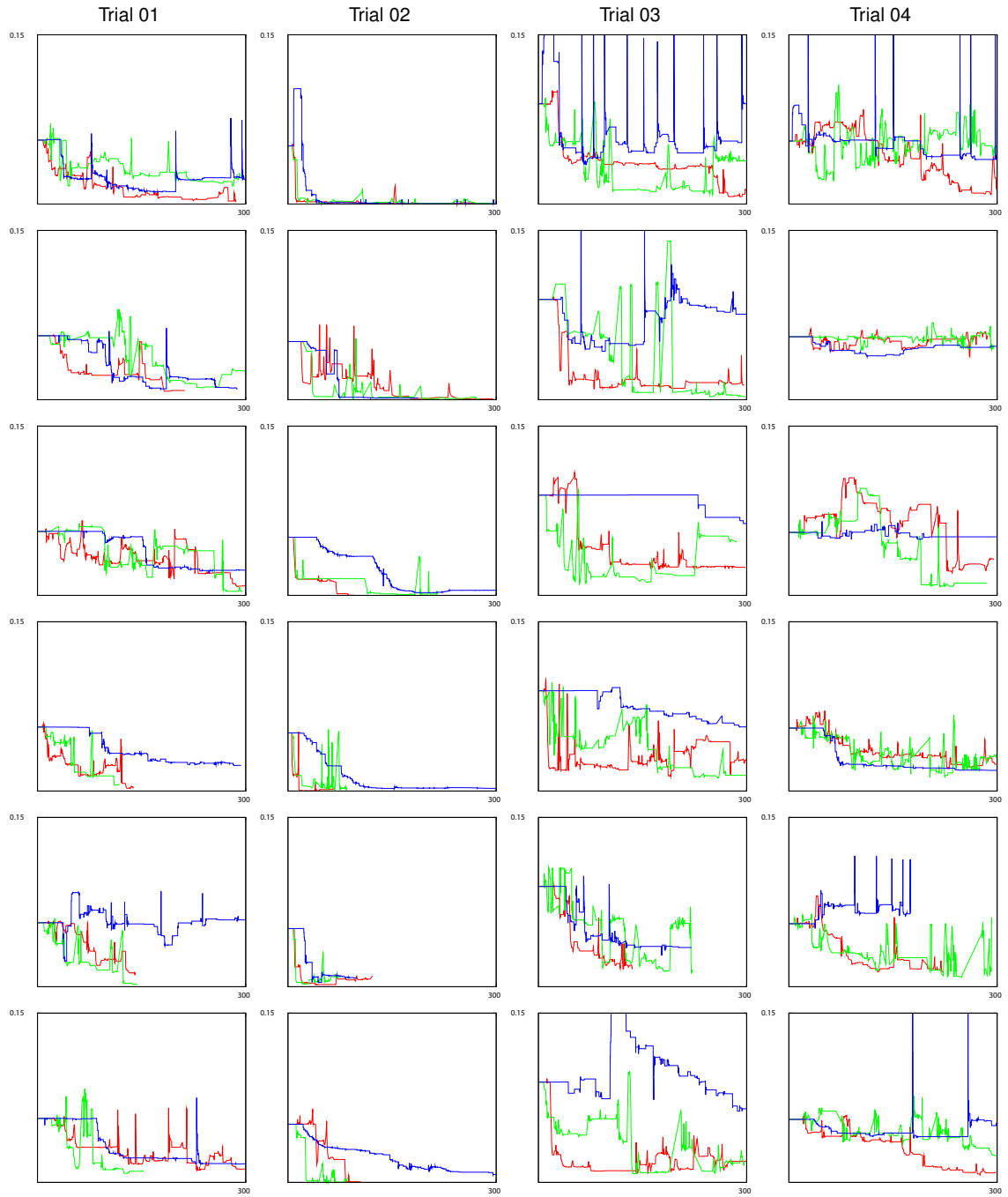
**Limitations.** By using novice subjects we avoid bias, but we also sacrifice experience. Professional lighting designers may perform differently relative to each interface paradigm based on a deeper understanding of the lighting task itself. Additionally, alternative implementations of the interface paradigms may improve or hinder performance. For example, dragging across convex hulls might smooth out indirect interface edits, but at the cost of imprecision. Additional brush types might increase the ability to paint accurately, but also increase user confusion. Finally, performance of the interface paradigms may vary under different lighting tasks. The local light feature shaping of this study may be too simple to take advantage of some interface strengths.

**Future Work.** Future work in this area could focus on the work flow involved in lighting scenes of increased lighting and geometric complexity. It is unclear as to whether novices can complete such a task in a timely manner, or whether meaningful data can be gathered from experienced lighting artists. It would also be of interest to investigate the effect of interface implementation enhancements on overall performance. It would be beneficial to identify a minimum set of features necessary to maximize effectiveness.



## References

- [AD04] ANRYS F., DUTRÉ P.: Image based lighting design. In *In the 4th IASTED International Conference on Visualization, Imaging, and Image Processing* (2004).
- [Ado07] ADOBE SYSTEMS INC: Photoshop CS 3, 2007.
- [Alt95] ALTON J.: *Painting With Light*. University of California Press, Berkeley, 1995.
- [Aut07] AUTODESK INC: Maya 2008, 2007.
- [Avi06] AVID TECHNOLOGY INC: Softimage xsi 6.5, 2006.
- [Cal99] CALAHAN S.: Storytelling through lighting, a computer graphics perspective. In *Advanced RenderMan: Creating CGI for Motion Pictures*, Apodaca A. A., Gritz L., Barzel R., (Eds.). Morgan Kaufmann, 1999, pp. 223–233.
- [CSF99] COSTA A. C., SOUSA A. A., FERREIRA F. N.: Lighting design: A goal based approach using optimization. In *Rendering Techniques '99* (New York, NY, 1999), Springer Wien, pp. 317–328.
- [Gum02] GUMHOLD S.: Maximum entropy light source placement. In *VIS '02: Proceedings of the conference on Visualization '02* (2002), IEEE Computer Society, pp. 275–282.
- [KPC93] KAWAI J. K., PAINTER J. S., COHEN M. F.: Radioptimization - goal based rendering. In *Proc. of SIGGRAPH-93: Computer Graphics* (1993), pp. 147–154.
- [LHV04] LEE C. H., HAO X., VARSHNEY A.: Light col- lages: Lighting design for effective visualization. In *VIS '04: Proceedings of the conference on Visualization '04* (Washington, DC, USA, 2004), IEEE Computer Society, pp. 281–288.
- [MAP\*00] MARKS J., ANDALMAN B., P.BEARDSLEY, FREEMAN W., GIBSON S., HODGINS J., KANG T., MIRTICH B., PSTER H., RUMMLER W., RYALL K., SEIMS J., SHIEBER S.: Design galleries: A general approach to setting parameters for computer graphics and animation. In *Proc. of SIGGRAPH-00: Computer Graphics* (2000), pp. 389–400.
- [MTB\*05] MOHAN A., TUMBLIN J., BODENHEIMER B., GRIMM C., BAILEY R.: Table-top computed lighting for practical digital photography. In *Eurographics Rendering Workshop* (2005), pp. 165–172.
- [PBMF07] PELLACINI F., BATTAGLIA F., MORLEY R. K., FINKELSTEIN A.: Lighting with paint. *ACM Trans. Graph.* 26, 2 (2007), 9.
- [PF92] POULIN P., FOURNIER A.: Lights from highlights and shadows. In *SI3D '92: Proceedings of the 1992 symposium on Interactive 3D graphics* (1992), ACM, pp. 31–38.
- [PP03] PATOW G., PUEYO X.: A survey of inverse ren- dering problems. In *Computer graphics forum* (2003), vol. 22, pp. 663–687.
- [PRJ97] POULIN P., RATIB K., JACQUES M.: Sketching shadows and highlights to position lights. In *Proceedings of Computer Graphics International 97* (1997), IEEE Computer Society, pp. 56–63.
- [PTG02] PELLACINI F., TOLE P., GREENBERG D. P.: A user interface for interactive cinematic shadow design. *ACM Trans. Graph.* 21, 3 (2002), 563–566.
- [SC07] SHESH A., CHEN B.: Crayon lighting: sketch- guided illumination of models. In *GRAPHITE '07: Pro- ceedings of the 5th international conference on Com- puter graphics and interactive techniques in Australia and Southeast Asia* (New York, NY, USA, 2007), ACM, pp. 95–102.
- [SDS\*93] SCHOENEMAN C., DORSEY J., SMITS B., ARVO J., GREENBERG D.: Painting with light. *ACM Transactions on Graphics* (1993), 143–146.
- [SL01] SHACKED R., LISCHINSKI D.: Automatic light- ing design using a perceptual quality metric. In *Computer graphics forum* (2001), vol. 20, pp. 215–227.
- [VLD07] VANGORP P., LAURIJSSSEN J., DUTRÉ P.: The influence of shape on the perception of material re- flectance. *ACM Trans. Graph.* 26, 3 (2007), 77.



**Figure 9:** Error graphs for all matching trials, with time in seconds on the horizontal axis. Error is the average  $L^2$  pixel error between the subject's image and the goal image. Each row of graphs represents a different subject and each column represents a different trial. red: direct interface, green: indirect interface, blue: paint interface