APPEARANCE-DESIGN INTERFACES AND TOOLS FOR COMPUTER CINEMATOGRAPHY: EVALUATION AND APPLICATION

A Thesis
Submitted to the Faculty
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Computer Science
by
William B. Kerr

DARTMOUTH COLLEGE
Hanover, New Hampshire
March 7, 2011

Examinig Committee:

__________________________
Fabio Pellacini, Chair

__________________________
Andrew Campbell

__________________________
Lorenzo Torresani

__________________________
Kavita Bala

__________________________
Brian W. Pogue, Ph.D.
Dean of Graduate Studies
Abstract

We define appearance design as the creation and editing of scene content such as lighting and surface materials in computer graphics. The appearance design process takes a significant amount of time relative to other production tasks and poses difficult artistic challenges. Many user interfaces have been proposed to make appearance design faster, easier, and more expressive, but no formal validation of these interfaces had been published prior to our body of work. With a focus on novice users, we present a series of investigations into the strengths and weaknesses of various appearance design user interfaces. In particular, we develop an experimental methodology for the evaluation of representative user interface paradigms in the areas of lighting and material design. We conduct three user studies having subjects perform design tasks under controlled conditions. In these studies, we discover new insight into the effectiveness of each paradigm for novices measured by objective performance as well as subjective feedback. We also offer observations on common workflow and capabilities of novice users in these domains. We use the results of our lighting study to develop a new representation for artistic control of lighting, where light travels along nonlinear paths.
Acknowledgement

I would like to start by thanking my advisor, Fabio. I would say that Fabio has worked tirelessly to mentor myself and the other students in his lab, but that is not quite true. He will push himself beyond exhaustion to make sure that we get the guidance and support we need to produce top-tier research. The amount of advisor face time, discussion, and collaboration in our lab is unusual, and appreciated a hundred times over. Bottom line, Fabio, you have pushed me to become better than I would ever have allowed myself to be alone, and I can’t express enough how beneficial that has been and will be in the future. Also, thanks for teaching me all of those Italian swears and curses.

To my defense committee, thank you for taking the time to help shape this body of work into a presentable whole. Your feedback has been helpful, and I appreciate the interest you take in what I am doing.

I would like to thank my friends for making the graduate experience enjoyable and mostly sane, you know, about as sane as you can expect it to be! To my lab mates, it has been a pleasure working with you all. Thanks for a great environment, and the “co-suffering” during long SIGGRAPH deadlines! Finally, to my family, thanks for everything. Your support has always been incredible, and I don’t have to hope to make you guys proud, because you are so good at letting me know that you are. Plus, there was the whole me being born part, that has been pretty crucial.
It has been a long road. Cheers!
# Contents

Abstract ................................................. ii

Acknowledgement ...................................... iii

1 Introduction ........................................... 1

2 Human Study Related Work ............................ 3

3 Lighting Design Interface Study ...................... 5
   3.1 Overview of Project .............................. 5
   3.2 Related Work .................................. 8
   3.3 Methodology .................................. 9
      3.3.1 Study Overview .......................... 9
      3.3.2 Experiment ............................... 11
   3.4 Study Results ................................ 16
      3.4.1 Analysis ................................ 17
      3.4.2 Workflow Observations ................. 23
   3.5 Discussion of Results ......................... 26

4 Bendylights ............................................ 30
   4.1 Overview of Project ............................ 30
6.5 Interface Implementation .............................................. 94
6.6 Analysis ...................................................................... 101
6.7 Discussion of Results .................................................. 106

7 Conclusions ................................................................. 109

Appendix ................................................................... 111

References ................................................................. 116
Chapter 1

Introduction

In 3d computer graphics, a scene consists of many elements, including light sources, surface materials, and object geometry. We define appearance design as the process by which these elements are created and edited. The appearance design process takes a significant amount of time relative to other production tasks and poses difficult artistic challenges. For example, an artist may spend a week lighting a single scene in an animated film. The scene may contain hundreds of lights, and there may be complex goals such as establishing mood, creating contrast between objects, increasing visibility near details, and feature placement of things like shadows. Many user interfaces have been proposed to make appearance design faster, easier, and more expressive, but no formal validation of these interfaces had been published prior to our body of work. In most publications, a small case study is performed in which a few subjects, often including authors, create a small piece of content, e.g., [68] and [59]. Particularly detailed case studies can be found in [56] and [10]. Sections 3.2 and 5.2 review related work in the areas of lighting design and material design respectively.

With a focus on novice users, we present a series of investigations into the strengths and
Introduction

weaknesses of various appearance design user interfaces. We focus on novices because they make up the majority of potential users and stand to gain the most from the introduction of novel interfaces. The goal is to have as many people as possible capable of using graphics tools. We began by developing an experimental methodology for the evaluation of representative user interface paradigms for lighting design, and later adopted this methodology to study interface paradigms for material design. In these two studies, we discovered new insight into the effectiveness of each paradigm measured by objective performance as well as subjective feedback. We also offered observations on common workflow and capabilities of novice users in these domains. The intended outcome of evaluation studies is to guide development of future interfaces.

As a result of the lighting study, we discovered that both direct and featured-based indirect control of light are desireable depending on the user and the context. We observed that many subjects wanted to create local effects, but the global effects of light and the number of degrees of freedom hindered them. We began developing BendyLight, a lighting model in which light travels along nonlinear paths, allowing different parts of the scene to be lit locally. The BendyLight user interface employed both direct and indirect controls.

The rest of this document is organized as follows. In Chapter 2, we review related work that applies to the body of work in general, but may not appear in our description of individual projects. Chapter 3 gives an overview of the lighting design study. Chapter 5 gives an overview of the material design study. Chapter 4 gives an overview of the BendyLight project. Finally, in the remaining sections, we propose future directions for evaluation of interface paradigms for other design tasks.
Chapter 2

Human Study Related Work

This section motivates the need for human studies for interface evaluation, and indicates where our research fits compared to other human studies in computer graphics.

**Appearance Design Study.** Most publications in the area of appearance design do not perform comparative user studies, e.g., [68, 61, 40]. Those that do, usually have small scale, e.g., [64]. This lack of validation motivated our investigation into user interface performance. Talton et al. take a different approach by statistically mapping the distribution of desireable edits in a variety of appearance design domains [84]. They develop a large-scale collaborative editing system that suggests desirable configurations based on what previous users have produced. Material editing was part of this system, but a direct comparison of various interface paradigms was not explored. Additional studies of note include Plushie [56] and EverybodyLovesSketch [10]. The Plushie study was a successful validation of the user interface, as the domain was very specific and subjects created something meaningful in this domain. The EverybodyLovesSketch paper demonstrated a large scale validative case study with roughly 50 subjects working for two weeks. Neither study was comparative.
Human Study Related Work

**Perceptual Study.** There is a large body of work in the area of perception related to computer graphics. While we focus on studies in which subjects perform editing tasks themselves, many of these studies affect appearance design. In many cases, strong results can be achieved when subjects only look at images. For example, Ramanarayanan et al. explore perceptual tolerance to changes in lighting and surface materials [76], discovering that significant compression and warping can be achieved without compromising visual equivalence. Křivánek et al. extend the idea of visual equivalence to measure the effect of rendering approximations to global illumination on material perception relative to scene conditions. They recommend perceptually-based techniques to improve global illumination rendering, specifically rendering that uses virtual point light (VPL) techniques [47]. Vangorp et al. study the effect shape has on material perception, suggesting surface properties that are experimentally proven to enhance or obscure details in surface reflectance [87]. Ramanarayanan et al. investigate perceptual tolerance to changes in complex aggregates such as large cluttering of objects like plants, discovering that significant changes can be made while still maintaining overall appearance [75]. Other studies have subjects perform an interactive task to indicate how they perceive a scene or object. For example, Wanger et al. explore the effects of scene features and how viewers perceive spatial relationships by having them move objects in a scene [88]. Cole et al. measure how well line drawings depict 3d shapes by having subjects orient gauge discs across surfaces [21]. A better understanding of how viewers perceive scene content can aid in what type of content artists might want to generate, both in terms of production efficiency and artistic expressiveness.
Chapter 3

Lighting Design Interface Study

Here, I give an overview of a user study designed to compare the relative performance of user interface paradigms for 3d lighting design with a focus on novices. The results of this study were originally published at SIGGRAPH 2009 [44].

3.1 Overview of Project

Lighting is a fundamental aspect of computer cinematography in establishing mood and enhancing storytelling [19]. 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, novice users are often not capable of effectively lighting complex scenes since they lack the technical training required to efficiently manipulate lights.

Various user interfaces have been presented to address the complexity of the lighting design task. Examining the editing of point lights, we categorized these interfaces into three main lighting design paradigms: direct light parameter manipulation, indirect light
3.1 Overview of Project

Figure 3.1: Example workflow of a novice subject lighting a simple scene with two lights using four different lighting interfaces. From the left: direct light parameter manipulation, indirect light feature manipulation, and goal-based optimization of lighting through painting working with either all lights (paint-all) or a single light at a time (paint-one). Vertical comments indicate which adjustment the subject is performing and at what times. Note the sketching style of the subject’s paintings.

feature manipulation, and goal-based optimization of lighting through painting. Each of these interfaces represents a different metaphor for lighting design. The workflow of direct and indirect interfaces is fundamentally one-light-at-a-time, by selecting and adjusting individual lights. Direct interfaces involve direct modification of the light source; these are common interfaces used in commercial software such as Maya [7] and Softimage [9]. Indirect interfaces consist of click-and-drag modifications of lighting features such as position and size of shadows, hotspots, and highlights [72, 71]. Painting interfaces differ in that they use an optimization algorithm to adjust the parameters of potentially all lights, minimizing the difference between the resulting rendered image and a user-painted one [79, 68].

Our paper presented a first step toward quantitatively evaluating the relative effective-
3.1 Overview of Project

ness of these interface paradigms. We specifically focused on novice users with no prior experience in lighting, since they form the majority of potential users and receive the most benefit from the introduction of intuitive interfaces. Out of the broad scope of lighting design, we limited our attention on the task of manipulating a small number of light sources in relatively simple scenes, rather than attempting to evaluate how subjects manipulate the hundreds of light used in complex computer cinematography environments. We focused on this simplified lighting task to ensure that novices could manage to complete it without incurring too much fatigue.

We performed a user study with 20 novice subjects, testing four user interfaces: direct, indirect and two variations of painting. We chose to evaluate two variations of the latter to validate whether the claimed benefits of this paradigm come from the ability to specify the lighting goal via a natural painting metaphor or the ability to work with all lights concurrently. Our study consisted of three parts. First, we had subjects manipulate lighting configurations to match exact target images, allowing us to measure the effectiveness of each interface in performing precise lighting adjustments. Second, we had subjects design lighting configurations based on suggested appearance, evaluating how each paradigm supported artistic exploration. Third, we had users fill out a series of questionnaires, collecting ratings for usability, preferences, and comments regarding each interface. Subjects worked with relatively simple scenes, of different geometric complexity, that are lit by up to eight lights, spending at least one hour with each interface. As an example of our setup, Fig. 3.1 shows screenshots of a subject lighting with each interface.
3.2 Related Work

Here, I review the body of work in lighting design interfaces that make up each of the three interface paradigms. For a review of other user interface and human studies in computer graphics see Sec. 2.

**Direct Interfaces.** Interfaces based on a direct manipulation paradigm, widely used in commercial software such as Maya [7], 3ds Max [6], and Softimage [9], require users to select lights and directly modify individual properties. For example, a light can be moved and reoriented in the scene by clicking and dragging widgets [22], 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 [72] allow users to manipulate the shadow volume to place shadows, while specular highlights are specified by clicking points on surfaces. More recently, Pellacini et al. [71], inspired by [33], showed how users can directly move and scale shadows and hotspots on object surfaces using a simple click-and-drag interface.

**Painting Interfaces.** Interfaces based on a painting paradigm further abstract 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 closeness between the painted image and the image generated by the renderer. Schoeneman et al. [79] use painted input as a goal for setting intensities of lights of known position in a global illumination renderer. Anrys and Dutré [3] and Mohan et al. [55] use a similar approach to relight real objects.
3.3 Methodology

whose appearance is captured using image-based lighting techniques. Poulin et al. [74] use sketches of shadows and highlights to place point lights for ellipsoid geometry. More recently, Pellacini et al. [68] presented a general painting interface for direct illumination of arbitrary scenes where all light parameters are derived using an efficient non-linear optimization framework. An example of painting for other lighting models, Okabe et al. [64] use a painting interface to edit image-based environmental illumination.

Several researchers have investigated methods for optimizing lighting-related parameters in order to achieve a variety of other goals, as surveyed in [67]. In the context of lighting design, Kawai et al. [43] maximize the subjective impression of scene qualities (e.g., pleasantness or privateness), while Shacked and Lischinski [80], Gumhold [36], Lee et al. [50] and Shesh and Chen [81] maximize low-level perceptual qualities for visualization. Costa et al. [25] explore the use of even more complex constraints. Rather than asking users to directly specify a lighting goal, Marks et al. [52] supports parameter exploration by generating many possible goals and letting the user choose the best options.

3.3 Methodology

This section details the experimental design, implementation, and procedure of the user study.

3.3.1 Study Overview

Goal. We seek to evaluate the relative efficiency of different interface paradigms in the context of simple lighting setups with a focus on novice users. Specifically, (1) we want to measure how efficiently these users can perform specific lighting adjustments and (2) we
3.3 Methodology

want to understand which interface paradigms provide a more intuitive exploration of the lighting design space.

**Users.** We choose to focus on novices with little or no prior knowledge of lighting design since they make up the majority of users who can take advantage of intuitive interfaces. We ask 20 subjects to light for about 4-5 hours each to ensure good statistical significance of our tests, while reducing fatigue. We considered extending our study to experts, but were concerned that long-term professional training with a very specific interface would likely bias their opinions and measured performance.

**Interfaces.** We compare four user interfaces, each following a major paradigm: direct, indirect and two variations of painting. Direct and indirect interfaces are similar in that they fundamentally support a one-light-at-a-time workflow, where users explicitly select and adjust individual lights. This workflow scales linearly with lighting complexity, becoming cumbersome for complex setups. They differ in the effectiveness with which they support adjustments, where indirect interfaces are believed to be superior since they allow explicit manipulation of lighting features. Painting interfaces are believed to be effective for two reasons. First, since painted goals can refer to all lights at once, theoretically they avoid the need to understand complex lighting setups, e.g. contribution of individual lights or quantity and types of lights. This would make paint scale better with lighting complexity. Second, it is believed that users can more directly express lighting goals using a natural painting metaphor that requires no understanding of the behavior of lights. To validate the relative effectiveness of these two benefits, we consider two variations of painting: painting with all lights, and painting with a single light at a time only. While the former should have all the claimed benefits of painting, the latter is more similar to the workflow of direct and indirect interfaces, differing from them in the use of painted images to control
light adjustments. For the remainder of this paper, we will refer to these four interfaces as direct, indirect, paint-all, paint-one.

Tasks. We ask users to perform two types of lighting tasks to measure interface effectiveness in performing specific adjustments and in aiding artistic exploration. 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.

3.3.2 Experiment

Overview. We ask subjects to complete a number of lighting trials where we record all of their actions for further analysis. Each subject performs the trials using all interfaces. These trials vary in the complexity of lighting, type of geometry, and task goal. In addition to recording subjects’ actions, we ask them to communicate their experiences on a series of questionnaires to collect subjective evaluation of each interface’s strengths and weaknesses.

Reducing Complexity. Since lighting design is a long and tedious process, a study that involves lighting tasks requires a careful triage between completeness and length. On the one hand, we want to achieve complex-enough lighting to ensure meaningful measurements. On the other, we want to ensure that subjects can successfully complete the required
3.3 Methodology

![Figure 3.2: Light types used in our study.](image)

tasks without incurring too much fatigue, to avoid bias in data collection. Working with novices makes this triage even more necessary. We simplify lighting and scene complexity to ensure that lighting effects and their relationship with scene geometry are clear. With regard to interfaces, we simplify implementation to ensure that they are simple enough to be quickly learned while sufficiently complete to capture the main characteristics of each paradigm.

Preparatory Studies. To determine exactly how to perform this needed simplification, we conducted formal and informal preparatory user studies on roughly 20 additional subjects, the results of which are not included in this paper. We compared different versions of scene geometry and lighting, choosing the most complex configuration that ensured at least one out of two subjects could complete the lighting. We also compared several variations in the implementation of the mentioned interface paradigms, choosing the one that showed the best performance in these preparatory studies. In the remainder of this section we will discuss our choices, only referring explicitly to these preparatory studies when we differ substantially from published work.

Lights. We limit the possible configurations of single lights by presenting subjects with two different types of lights typical of computer cinematography [19]: key lights and fill lights, shown in Fig. 3.2. In our implementation, key lights are represented as spotlights
3.3 Methodology

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 have distance squared falloff.

We chose to light our scenes with a maximum of eight lights, including two key and six fill lights. We performed preparatory studies to select this maximum lighting complexity. While this number might seem small, we found that most novices cannot complete lighting tasks for significantly larger configurations in reasonable time for an experiment without incurring substantial fatigue. Furthermore, while it is known that production lighting uses hundreds of lights, the reader should be aware that such lights are selectively turned on only on a small number of objects [19], a practice also supported by commercial packages (e.g. known as “light linking” in Maya). This process essentially reduces production lighting to a large set of small and independent lighting tasks, each closer to our complexity.

![Figure 3.3](image)

**Figure 3.3:** Starting and target scene configuration for matching trials.
3.3 Methodology

Scenes. We include two versions of scene geometry in our experiments, shown in Fig. 3.3: a room-like environment containing two characters and the same scene strongly simplified to obtain an abstract environment where the characters are simple oval shapes. Subjects were given only one or the other geometry type to work with, allowing us to compare the effect of geometric complexity of lighting design. We also considered an abstract blob scene and a realistic tabletop scene, following [87] guidelines for geometry, and found that possible lighting configurations became too simplistic to effectively measure performance.

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, allowing us to focus our measurements precisely on the tasks of determining shape and intensity of lights.

Trials. We perform three matching trials of progressively more complex lighting, going from two to eight lights, and two open trials with different goal styles. Details of each trial are summarized in Fig. 3.3 and Fig. 3.4. We vary matching trials in lighting complexity to observe possible changes in workflow and trends in the effectiveness of each interface under these conditions. Specifically we include relatively simple 2 and 4 light cases, where

Figure 3.4: Target configuration for open trials. Starting configuration and scenes are identical to trial 3.
3.3 Methodology

subjects are expected to quickly converge on a good solution, as well as a more complex
8 light case, where we further increase the error of the initial configuration to observe
the ability to handle greatly differing initial and target images. For open trials, we select
target images that obviously differ from the given scene in content to force freeform artistic
exploration. We choose one high-key and one low-key lighting goal to force two different
interpretations of the scene. Subjects are given access to 8 lights for open trials.

The same initial and goal lighting configurations are used for all subjects and all inter-
faces. We limit interaction time to 5 minutes for 2 and 4 lights, and 10 minutes for 8 lights
to adjust for the increase in complexity. Subjects can opt to end the trial sooner if satisfied
with the current result. At the end of each matching trial, subjects rate the accuracy of the
matching with a scale 1 to 5. For open trials, subjects use the same scale to rate how closely
they have achieved their interpretation of the style.

Questionnaire. After performing all trials with all interfaces, subjects complete a ques-
tionnaire where they rate the interface on a scale of 1 to 5 in the following areas: (1) natural
way to think about lighting; (2) ease of determining what to modify (selection), (3) how it
should be modified (intention), and (4) in carrying out the adjustment; (5) preference on
few lights, and (6) many lights; (7) preference in matching trials, and (8) open trials; and
(9) overall preference. Subjects also strictly rank interfaces in each of the categories listed
above. Immediately after finishing trials for each single interface, subjects are asked to
leave free-form comments on each of these aspects. For reproducibility, we include copies
of all questionnaires as additional material.

Procedure. 20 subjects participated in the study, chosen from different age and educa-
3.4 Study Results

tional groups. All subjects had normal or corrected-to-normal vision. Of these 20, 12 subjects had no prior knowledge of 3d graphics, ranking their experience 1 on a scale 1 to 5. The lighting experience of the remaining subjects came from taking up to three classes in Maya, with no specific focus on lighting. All subjects possessed no significant real-world studio lighting experience, rating themselves 2 or lower on a scale of 1 to 5.

Subjects complete the study in four 60-minutes sessions, one for each interface. We randomize the order of the interfaces for each subject. Before each session, subjects complete a training phase to familiarize themselves with the specific interface, where an investigator explains the types of lights and how the interface is used to manipulate them. To ensure understanding, subjects are asked to use each interface command on the training scene shown in Fig. 3.2. Once trials begin, all user interface actions are recorded.

Trials were conducted in a controlled lighting environment with negligible ambient lighting, to simulate typical working conditions of lighting artists. We used a 24-inch Dell 2407WFPb LCD at 1920x1200 resolution at a distance of approximately 1 foot from the subject. All rendered images are 512x512 pixels on screen. We used an Intel 2.8 Ghz Core2 Quad Q9550 PC with 4 GB of RAM and an NVidia GeForce 9800 GT graphics card. All interactions were ensured to be realtime.

3.4 Study Results

This section details the results of the user study with analysis and observation.
3.4 Study Results

![Graph showing L₁ error over time for trial 1, corresponding to the screenshots in Fig. 3.1](image)

**Figure 3.5:** Example graph of $L_1$ error over time for trial 1, corresponding to the screenshots in Fig. 3.1

### 3.4.1 Analysis

We present our results in two 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. Unless stated otherwise, tests for statistical significance are computed with repeated measures analysis of variance (ANOVA) [83]. Repeated measures ANOVA is appropriate when a within-subject factor (in our case, each subject using all interfaces) creates correlations that violate the independence assumption of a traditional 1-way ANOVA. A $p$ value below 0.1 indicates a 90% confidence that the two population means differ given the measure of the sample. In all figures, error bars represent standard error.

**Time to Completion.** Generally, subjects are able to complete each trial within the allotted time limit with one or more interfaces. On average, time to completion is similar across all interfaces and lies relatively close to the time limit (290 s for trials 1 and 2, 520 s for trial 3 and 410 s for trials 4 and 5).

**Image Error.** To evaluate user performance in matching trials, we compute the $L_1$ error between the subject’s rendered image and the goal image. We chose $L_1$ since we found it to be more robust than other metrics, including $L_2$, measures of relative error, and metrics over lighting parameters. Fig. 3.5 shows the $L_1$ error over time for one subject performing
3.4 Study Results

![Average of the final $L_1$ error for matching trials.](image)

**Figure 3.6:** *Average of the final $L_1$ error for matching trials.*

![Illustration of the average $L_1$ error over time (in seconds) for matching trials.](image)

**Figure 3.7:** *Illustration of the average $L_1$ error over time (in seconds) for matching trials.*

the same trial with all interfaces. When subjects are successful, error decreases toward the
correct solution, showing convergence to the goal. This convergence is not monotonic, but
interrupted by moments where users explore local configurations, often with a new tool.
For reference, in the original submission we included the error graph for all subjects in
supplemental material together with screenshots of the lighting configuration over time.

Fig. 3.6 shows the $L_1$ error between the final image and the target image averaged across
all subjects. Direct and indirect outperform painting in final error in all trials ($p < 0.094$),
3.4 Study Results

with slightly less significance on indirect compared to paint-all on trial 3. On trial 1, there is a slight advantage of indirect over direct and paint-one over paint-all \((p > 0.141)\). On trials 2 and 3, direct and indirect produce similar error, as do paint-all and paint-one \((p > 0.592)\).

**Convergence.** To illustrate the convergence behavior of different interfaces, we average the image error across all subjects over time in Fig. 3.7. While we acknowledge that this average is not statistically valid, since edits are not synchronized across subjects, we find that this gives an intuitive visual indication of overall behavior. The starting error begins much higher on trial 3, since we designed this trial to be more challenging than others. Notice the clear difference of the two paint interfaces vs. direct and indirect. For trials 1 and 2, direct and indirect roughly converge on a solution while paint-all and paint-one do not. On trial 3, the painting interfaces are able to reduce image error at first, by performing gross image adjustment, but plateau well before direct and indirect. This suggests that while users are able to light, different interfaces support different convergence behaviors. Overall, we conclude that painting interfaces are the worst performing in terms of final error and convergence, while direct and indirect are relatively close in performance.

**Subjective Image Quality.** Subjects rate the closeness of their final output to the intended target on a scale of 1 to 5. Fig. 3.8 shows the average ratings for each trial. We find that this subjective image quality roughly correlates with the \(L_1\) error of the final image with a linear correlation coefficient of \(-0.55\). In all matching trials, indirect and direct interfaces surpass paint-one and paint-all \((p < 0.114)\). People perform slightly better with indirect than direct on trial 1 \((p = 0.118)\), but otherwise there is no significant difference between direct and indirect or between paint-all and paint-one on matching trials \((p > 0.359)\). In open trials, subjective image quality for the painting interfaces relative to the direct and indirect interfaces on open trials are substantially higher. In fact, there is no
3.4 Study Results

**Figure 3.8:** Average of subjective image quality ratings.

**Figure 3.9:** Interface rankings and ratings from questionnaire.

A major difference between direct, indirect, and paint-all ratings on trial 4 ($p > 0.167$). This suggests that while people perceive their performance with painting interfaces to be worse than the others in adjustments tasks (matching trials), in exploration tasks (open trials) subjects perceive their performance with painting to be greatly improved.

**Rankings and Ratings.** Subjects rate and rank each interface in 9 categories where ratings can have ties, but rankings are forced choice (see Sec. 3.3.2). Average ratings and stacked frequencies of rankings are shown in Fig. 3.9. For evaluating statistical significance of ranks we use the Friedman test [32], a nonparametric test that takes into account within-subject effects to derive the confidence of consistent distinction indicated by a low p-value.

In all categories except preference on open trials, the non-painting interfaces rank sig-
3.4 Study Results

significantly higher than painting ($p < 0.025$). As with the subjective image ratings, direct and indirect still rank higher than the painting interfaces on open trial preference, but by significantly less.

The majority of subjects rank indirect higher than direct in most categories. By analyzing the ratings across subjects using two-step cluster analysis that first sequentially clusters and then performs hierarchical clustering, we found that there are two separate groups that consistently prefer direct better than indirect and vice versa. This suggests that subjects distinguish between the two. Between these groups, more people prefer indirect over direct. With almost no exception, the differences between paint-one and paint-all are negligible ($p > 0.371$).

We find similar trends in the ratings. On all categories except open trial preference, direct and indirect rate much higher than paint-one and paint-all ($p < 0.005$). With almost no exception, there is no significant distinction between direct and indirect or between paint-all and paint-one in the ratings ($p > 0.249$).

To further understand which factors are perceived as important in judging interfaces, we correlate the overall preference ratings to ratings in all other categories using linear correlation. Across all interfaces, the ratings of adjustment and matching trial preference correlate most closely with overall preference (corr. coeff. 0.89 and 0.84 respectively). This implies that our subjects prefer interfaces that give them the control required to precisely control and adjust lighting features.

**Subject Prior Experience.** In Fig. 3.10 we compare subjects with previous lighting experience to those without by showing the average over matching trials of the $L_1$ error and subjective image quality ratings. For direct ($p < 0.005$) and indirect ($p < 0.2$) interfaces, subjects with prior experience perform better overall. This is to be expected given that they
3.4 Study Results

have some, albeit small, prior training using graphics software. It is interesting to note that their improvement is noticeable whether they use direct or indirect, even if their experience using commercial software only exposed them to direct interfaces. Performance with the painting interfaces is consistent across both groups ($p > 0.507$, paint-one subjective $p = 0.207$).

**Direct vs Indirect.** While we found that novices can light with direct and indirect interfaces, a closer comparison of the two is warranted. In most cases, these interfaces perform closely, with indirect affording slightly quicker convergence. When subjects are forced to choose they rank indirect higher in most categories, including overall preference where 13 out of 20 prefer indirect. We also observe that if we partition the subjects into those who did and did not rank indirect higher than direct in overall preference, these groups do not correlate with those with prior direct interface experience and those without (corr. coeff. 0.043). This implies that prior training in the use of direct does not imply its preference. This is quite unexpected since subjects should perform significantly better with a more familiar interface. Taking into account all these facts, we conclude that while there are some who like using direct more than indirect, indirect is an overall more efficient and preferable interface.

**Scene Complexity.** We have shown that varying interfaces clearly influence lighting performance. Unexpectedly, we found that scene complexity influences user performance to a significantly lesser degree. In particular, we found no significant differences, across all measurements, between subjects who worked on scenes with character geometry and those who worked on scenes with abstract geometry. Furthermore, while the matching trials become increasingly more complex with the number of lights, we found that given enough time to complete, the lighting complexity of the scene does not hinder performance.
3.4 Study Results

![Chart](image.png)

**Figure 3.10:** Comparison of subjects with prior lighting experience to those without in terms of final $L_1$ error (left) and subject image quality ratings (right). Values averaged over all matching trials.

significantly. As discussed above, subjective estimations do not suggest that working with more lights is significantly worse, and while final image error is slightly higher with 8 lights, the initial configuration of the trial also has higher error.

### 3.4.2 Workflow Observations

To better interpret our findings so far, it is helpful to discuss the typical workflow our users showed. We will use the example of one subject illustrated in Fig. 3.1 with corresponding error in Fig. 3.5. In supplemental material for the original publication, we included similar diagrams for all subjects. We also show recorded lighting trials for a few subjects in the paper video.

**Sketching Paint.** The most noticeable trend regarding the painting interfaces is that people consistently draw rough sketches to represent areas where light should or should not appear, but pay little attention to how light would actually affect the scene in terms of gradients. The applied paint is usually very bright or very dark, and does not respect objects boundaries (even though tools were provided to simplify this). Furthermore, when subjects attempt to paint local lighting features, such as shadows or highlights, they do not compensate for the overall effect of the light, essentially attempting to treat some features
3.4 Study Results

![Graphs showing error comparison between painted images and lighting configurations]

**Figure 3.11**: Error of the painted image versus error of resulting lighting configuration for trial 2 with paint-all. *Left: Example subject. Right: All subjects’ average.*

as independent from the others. We found this behavior in all subjects. Finally, 55% of subjects openly commented that they just can’t figure out what paint adjustments to make to get the lights to go where intended.

These observations are confirmed by visualizing the error of the painted image and the image generated by the optimized lighting configuration, to the goal image in matching trials, shown in Fig. 3.11. In these graphs, there is a clear trend of paint error exceeding the error of the lighting configuration. This behavior clearly indicates that subjects are attempting to steer the optimization of lights with rough sketches, rather than attempting to reproduce the target image by painting.

**Sketching One vs All Lights.** While we observed this sketching behavior in each of the painting interfaces, there are fundamental differences between the two. We found that sketching can only effectively control one light at a time since subjects tend to sketch individual lighting features. However, 40% of subjects make comments about the difficulty of controlling one light at a time, e.g. “because I was painting for individual lights it was sometimes hard to account for the effects of the other lights”. Essentially when lighting complexity increases, painting single lights becomes less intuitive and more difficult. At the same time, while painting all lights addressed this confusion, it remains unintuitive
3.4 Study Results

since subjects expect to be able to paint individual lighting features while the optimizer guesses which light to apply it to. We believe that this mismatch between sketching versus painting is the key to interpreting the low performance of painting interfaces in matching trials.

One-Light-At-A-Time Workflow. Direct, indirect and paint-one share the same fundamental workflow, where users work one light at a time. When using the direct and indirect interfaces, 95% of subjects work by first blocking the key lights, that show the most prominent lighting features, then adjusting fill lights, to eventually go back and refine iteratively. When using paint-one, subjects exhibit the same order of adjustments, but only 35% are actually able to refine due to time limits. While our matching trials progressively increase in lighting complexity with 2, 4, and 8 lights, we found this iterative workflow is common to all three trials. Furthermore, 45% of subjects exhibited the behavior of turning off all lights then adding them back one at a time, adjusting iteratively. We believe that the consistency of this behavior is remarkable, since many of our subjects had no knowledge of lighting and yet they converged to the same workflow. We believe this implies that alternating blocking and refinement, used in computer cinematography [19], is a fundamental aspect of lighting design.

One-Light-At-A-Time Adjustments. Sharing the same workflow, the difference between the three interfaces discussed is in how quickly adjustments can be performed. Looking at questionnaire comments helps in interpreting this difference. 30% of subjects comment that direct requires more trial and error than indirect, while 35% comment that moving light sources along 3d axes can be confusing. 35% of subjects commented that they had trouble moving fill lights with indirect, since they exhibit no sharp and clearly distinguishable feature, the control of which was not the design goals of indirect methods. This
3.5 Discussion of Results

explains why the benefit of the indirect interface versus direct decreases on more complex trials that have a higher ratio of fill versus key lights. We conclude that the indirect interface allows subjects to work faster in general, but the performance increase is less noticeable when controlling soft gradients, as expected with an interface designed to control “hard” lighting features.

**Exploration.** In open trials the performance of painting metaphors increases. This is unexpected since we verified that these methods provide poor control for lighting. We believe this is due to the fact that subjects are performing a purely exploratory task that requires no precise adjustment. 70% of subjects commented directly that they are conducting an exploration task, more akin to just blocking, rather then refining. E.g., “the open-ended trials I tended to experiment more with, as if I expected to serendipitously encounter a nice lighting configuration,” and “I could aim for what I know I could do and still have it be in the approximate style”. This does not mean that subjects are randomly moving lights around, but rather that they are using exploration to converge on something they find to be acceptable. If one takes into account how poorly people can control lighting with painting, we conclude that painting metaphors are better at pure exploration, but they lack the needed control for a full solution to lighting design.

3.5 Discussion of Results

In this section, I summarize the major findings of our study. Strictly speaking these observations only apply within the boundary of the tested cases in our study.

**Novices Can Light.** We have found that, in the modest complexity cases shown, novices can light. While this may seem obvious, we believe it is remarkable that subjects with little training can perform these lighting tasks reasonably quickly, provided the
3.5 Discussion of Results

interface supports them. This suggests that future work on lighting design for novices will have an impact for a large number of users.

**Direct and Indirect.** We found that while subjects can light well whether they use a direct or indirect interface, indirect manipulation performs overall better in our tests providing a more effective control of lighting adjustments. This suggests that future work on appearance design should consider indirect manipulation as its main control mechanism.

**Paint vs Non-paint.** Our most prominent result is the poor performance of the painting interfaces with respect to the non-painting ones. This is consistently observed in all of our data, including image error, convergence, and subjective evaluations. Note that this result may not generalize to other appearance design applications or users with extensive painting training.

**Paint Sketching.** The underlying assumption of a painting metaphor is that people are better at expressing a goal through paint than they are in configuring light parameters. This study shows that novices tend to be quite poor at painting accurate representations of lighting with the quality necessary to drive an optimizer. Users with more extensive painting training may fare better. However, our subjects tend to sketch rough illumination areas and expect the computer to interpret their input correctly. Our study suggests that interfaces designed specifically to support rough sketching might in fact perform better, as was found in material design [70, 2]. However, in the case of lighting design, it remains unclear how sketching can directly control the features of multiple lights.

**Paint Selection.** Besides control, painting is expected to be effective since it removes the need to understand how many individual lights contribute to the final image, especially troublesome for complex setups. However, the similarity between painting with all lights and painting with one light at a time in our study suggests that the inability to control
3.5 Discussion of Results

lighting details overshadows the perceived benefit from painting with all lights.

**Exploration.** The superiority of non-painting over painting interfaces is far less obvious for open trials. Taking into account the control problems of painting interfaces, this implies that painting is a better metaphor for tasks where it is desired to quickly block out rough scene lighting. This seems counterintuitive, since people should be better at painting when there is a clear reference. However, we found that painting interfaces afford to users who sketch rough illumination, the presentation of lighting configurations that might not have been encountered using a direct or indirect interface.

**Factors Influencing Lighting.** Besides prior training, differences in interfaces are the only significant factors we found in lighting design performance. Within the boundary of our study, we notice no influence from geometric complexity. We found that lighting is mostly linear in the number of lights, suggesting that given enough time, people are not hindered by the increase in complexity. While we absolutely do not believe that our findings on scene complexity extend to the significantly more complex scenes used in feature film production, it is our opinion that for the simple scenes that novices are likely to manipulate complexity does not matter.

**Limitations.** First, as in all user studies, the main limitation of our work is the space of lighting tasks we have explored. Second, we did not include expert subjects, in lighting or painting, since they belong to a completely different user group. Third, we have only explored a fraction, albeit large, of all possible lighting models, of which natural illumination and color certainly deserve further consideration. We believe that these three extensions are the most important explorations to focus on in future work to eventually reach a comprehensive study of lighting design paradigms.

**Implications.** While this study is just the first step toward a comprehensive evaluation
3.5 Discussion of Results

of lighting design interfaces, it already provides clear information on general trends that will influence future lighting interface development. Furthermore, we believe that these same trends are general enough to be applied to other appearance design domains, such as material design. Finally, we expect that the principles employed in designing our studies can be adopted to evaluate other appearance design tasks, such as natural illumination and material design.
Chapter 4

Bendylights

This section describes a rendering technique and lighting system called BendyLights that allows artists to bend light along nonlinear paths. Artists can control these paths using splines. Inspired by the findings from our lighting interface study [44], we designed BendyLights to work using both direct and indirect control schemes. This work originally appeared at EGSR 2010 [46].

4.1 Overview of Project

In computer cinematography, lighting is crucial in supporting storytelling [19]. In order to convey certain moods and appearances, artists require a more flexible control of lighting than possible with physically-based models. In particular, non-physical tools like lights that do not cast shadows or have arbitrary spatial falloffs are often used when controlling the lighting on characters or to mimic indirect lighting effects. Cinematic lighting models have been introduced to provide artists with non-physical controls. For example, [11] controls light shape and falloff with a complex combination of shading parameters. While these
4.1 Overview of Project

We propose BendyLights, a spotlight-based lighting model for artistic control in which light travels non-linearly along spline paths from a point source. During rendering, we determine the light direction at a point by inverting a cubic equation and compute shadows by deforming geometry and using shadow mapping. Fig. 4.1 shows scenes lit with BendyLights. Compared to traditional cinematic lights and light linking, BendyLights have the advantage of being able to bend light rays in a more flexible manner, allowing for more creative and artistic lighting effects.

Models are powerful, artists have found that lighting different objects with different light sources, or "light linking," is necessary to achieve many desired effects. These techniques are common in production environments [19], and well-supported by most commercial packages such as Maya [7].

![Figure 4.1: Row 1: An example of basic BendyLights operations. (A) Non-deformed spotlight. (B) Setting a constant radius. (C) Bending light rays from B. (D) Bend from C with the radius of A. Row 2: An example of an animated BendyLight in an animated scene with motion blur. Row 3: Example of lighting using a BendyLight that is not possible using traditional linear lighting techniques. Shown is a spotlight before deformation, the resulting edited with BendyLights, and our best approximation of this edit with traditional linked linear lights.](image)
4.2 Related Work

following advantages.

- Lighting effects, including shadows and highlights, can be adjusted independently at different spatial locations using only a single light.
- Smooth spatial consistency of the non-physical lighting effects is maintained across surfaces.
- Light paths can be easily controlled with familiar spline editing tools; artists can also drag shadows, highlights and hotspots while our interface appropriately deforms the spline.
- \textit{BendyLight} lighting effects can be animated by keyframing the spline control points.
- \textit{BendyLights} can be rendered in real-time on GPUs and easily integrated in production renderers; they also work with global illumination.

In the paper, we present a perceptual study that investigates the believability of \textit{BendyLights} used in real scenes [46]. The study asks participants to compare a scene with a linear light and a scene with a \textit{BendyLight}, selecting which (or both) are more physically accurate in terms of lighting. We conclude from the study data that an artist can use \textit{BendyLights} to create not only non-photorealistic effects, but also effects that are perceived to be accurate compared to linear lights as well as nonlinear effects that are so convincing subjects think linear lights are less realistic.

4.2 Related Work

\textbf{Artistic Lighting Control.} Artists often need and use non-physical lighting in their compositions [19]. Brazel controls light shape and falloff at the light source with a complex combination of shading parameters [11]. Todo et al. control diffuse and specular features
4.2 Related Work

of lighting on surfaces in a non-photorealistic rendering context [86]. Obert et al. propose a method for controlling the transport of indirect illumination from surface to surface [63]. Ritschel et al. control the directions in which light is reflected off of objects’ surfaces [77]. Our method focuses on non-linear deformations for the light directions coming from a point source.

**Nonlinear Light.** The idea of nonlinear lighting has been explored in the past. Berger et al. [14] use ray marching through discretely placed refraction layers to bend view rays, and Musgrave [58] replaces refraction with total reflection. Gröller introduces a more common nonlinear raytracing technique that renders using linear ray segments, but does so at each step of a numerical solution to differential equations [34]. Similar techniques are used by Gutierrez et al. [38] to simulate atmospheric phenomena, Weiskopf [90] for gravitational fields, Falk et al. [30] for panorama maps, and Satoh [78] for generalized nonlinear dynamical systems. Gutierrez et al. extend this technique for photon mapping [37]. Stam et al. use a slightly different approach with ray marching by perturbing rays according to bounded force fields [82]. Rendering of nonlinear rays can also be accomplished by voxelizing over a vector field, used by Ihrke et al. [41] to render refractive objects and Zhao et al. [94] to render heat shimmering and mirage. Gröller also discusses a method for using rays curved according to parametric equations, storing them in hierarchical bounding volumes (with some assumptions on monotonicity of the ray) and computing intersections with linear segments [34].

Weiskoph et al. propose using GPU hardware to do nonlinear raytracing without shadows, but the speed is still relatively slow [91]. Zhao et al. update this algorithm from multiple passes to a faster single pass, but still without shadows [94]. Ihrke et al. render caustics and shadows at interactive rates, but modifying lighting directions requires a more
expensive and slow computation, as well as a voxelized representation of the scene [41].

To our knowledge, we are the first to propose a method for rendering nonlinear light for artistic control. We do so without voxelization, discrete linear steps, or precomputation. Because our light model can be represented linearly after deforming surface positions, we take advantage of GPUs to achieve real-time rendering.

### 4.3 Model

**Goals.** We seek to develop a light model that allows artists to control shadows and incoming light directions independently in different parts of the scene while smoothly interpolating at all scene locations to keep scene appearance consistent. Furthermore, lighting should remain consistent during object animation and the lighting model should be keyframe-able to allow for time-varying lighting effects.

**BendyLights.** To achieve these goals, we propose a spotlight-based lighting model where, conceptually, light travels along nonlinear splines from a point source. We use quadratic Bezier splines [57] to keep render time low, and represent the light as a radially-symmetric tube surrounding the spline, much like the cone of a spotlight. See row 1 of Fig. 4.1 for a basic example. Manipulating the shape of BendyLights is achieved by moving spline control points in space. To render with BendyLights, we need to determine the incident light direction and nonlinear visibility function at all scene points. We define the lighting direction as the tangent of the light tube at that location. We compute visibility by non-linearly transforming geometry according to the inverse of the nonlinear light path. Linear computation of shadows on this deformed geometry is equivalent to nonlinear computation of shadows on the original geometry.

**Parametric Representation.** We define BendyLight starting from an existing spotlight
4.3 Model

Figure 4.2: Parameterization for 3 segments of the quadratic spline.

$S$ defined by a coordinate system. The source position of $S$ is the first control point that of the quadratic Bezier spline $C$ defining the center of the light tube (Fig. 4.2). $C$ can have an arbitrary number of quadratic segments. Given a spline segment, a position on $C$ in world coordinates can be written as follows:

$$P_c(u) = c_{p0}(1 - u)^2 + c_{p1}2(1 - u)u + c_{p2}u^2$$ (4.1)

where $P_c$ is a point on $C$ at $u$ and $c_{p0}, c_{p1}, c_{p2}$ are control points for the spline segment (Fig. 4.2). Light directions within the tube are defined on the plane orthogonal to the tube spline at every $P_c$. We define the radius of the tube on these planes by another spline:

$$R_c(u) = c_{r0}(1 - u)^2 + c_{r1}2(1 - u)u + c_{r2}u^2$$ (4.2)

where $R_c$ is the scalar radius of the tube at $u$ and $c_{r0}, c_{r1}, c_{r2}$ are its control points (Fig. 4.2). We keep a consistent coordinate system along the tube with orthogonal basis vectors $(\hat{u}_u, \hat{v}_u, \hat{w}_u)$. The forward vector $\hat{u}_u$ is the normalized derivative of $C$ at $u$. $\hat{v}_u$ and $\hat{w}_u$ are oriented by an “up” vector. Using this basis, a world position $P_w$ can be written in the tube’s parametric space as

35
4.3 Model

Figure 4.3: Deformation of geometry. Red: spotlight $S$. Blue: BendyLight. Orange: scene geometry.

Deformation. Given a surface position $Q$ with normal $N$ in world space, we need to determine its location $Q'$ after deformation into $S$’s light space (Fig. 4.3). To do so, we want to find the coordinates $(u, v, w)$ of $Q$ with respect to the BendyLight tube (see Fig. 4.3). We find $u$ where $Q$ is orthogonal to the tube by solving the following equation:

$$(Q - P_c(u)) \cdot \hat{u}_u = 0 \quad (4.4)$$

See Sec. 4.4 for our approach to solving equation (4.4). Once $u$ is found, we can compute

$$v = \left(\frac{(Q - P_c(u)) \cdot \hat{v}_u}{R_c(u)}\right) \quad (4.5)$$

$$w = \left(\frac{(Q - P_c(u)) \cdot \hat{w}_u}{R_c(u)}\right) \quad (4.6)$$

Next we simply perform a change of basis from $(\hat{u}_u, \hat{v}_u, \hat{w}_u)$ of the tube to a set of basis vectors $(\hat{u}_l, \hat{v}_l, \hat{w}_l)$ for $S$’s local light space. The coordinates $(u, v, w)$ from the tube become $(u', v', w')$ in this basis. We can now define
4.3 Model

\[ Q' = P_l + u' \hat{u}_l + v' \hat{v}_l + w' \hat{w}_l \]  

(4.7)

where \( P_l \) is the world-space position of light source \( S \), \( u' = u \), \( v' = v \), and \( w' = w \).

Let \( L_i' \) be the non-deformed light direction from \( S \) incident at \( Q' \), and \( L_i \) be the deformed light direction incident at \( Q \). Since \( S \) is a spotlight, \( L_i' \) is simply the ray from \( Q' \) to \( S \)'s source.

\[ L_i' = (P_l - Q') \]  

(4.8)

\[ L_i = - \left( P_c'(u) + \frac{Q - P_c(u)}{|Q - P_c(u)|} \right) \]  

(4.9)

where \( P_c'(u) \) and \( R_c'(u) \) are the first derivatives of \( C \) and the radius spline at \( u \) respectively.

This takes into account both the tangent to the center curve \( C \) and the spread of light rays moving outward along the tube radius.

**Shading and Shadows.** Reflectance and shading can be computed using the deformed light direction, \( L_i \), incident at world-space surface location \( Q \) and normal \( N \). Because we have simply replaced the value of \( L_i \), traditional rendering techniques like perturbing normals or displacing surface positions still work. In Sec. 4.6 we demonstrate that *BendyLights* also work with techniques like motion blur and global illumination. Shadows can be drawn by computing visibility from deformed surface positions \( Q' \) along the linear light directions \( L_i' \). The resulting shadow lookups are equivalent to casting bent rays in the scene. Since \( Q' \) is non-linearly related to \( Q \), this might require tesselating the geometry to avoid shadow artifacts, the same procedure required by any non-linear mesh deformation. We find that practical applications require only modest tesselation. Fig. 4.5 shows an example of tesselated wireframes.

**Animation.** We can animate *BendyLights* by keyframing its control points for positions or radii along the tube. Additionally, following equations (4.1) and (4.2), we can control
4.4 Implementation

<table>
<thead>
<tr>
<th>Scene</th>
<th>Triangles</th>
<th>OpenGL fps</th>
<th>RenderMan Pass</th>
<th>Shadow s</th>
<th>Main s</th>
<th>Total s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tabletop Bunny</td>
<td>102k</td>
<td>77.8</td>
<td>8.3</td>
<td>4.0</td>
<td>12.3</td>
<td></td>
</tr>
<tr>
<td>Tabletop Vases</td>
<td>15k</td>
<td>137.0</td>
<td>5.7</td>
<td>3.5</td>
<td>9.2</td>
<td></td>
</tr>
<tr>
<td>Tabletop Bowl</td>
<td>16k</td>
<td>104.8</td>
<td>6.6</td>
<td>3.7</td>
<td>10.3</td>
<td></td>
</tr>
<tr>
<td>GI Bunny</td>
<td>118k</td>
<td>51.9</td>
<td>19.0</td>
<td>5.2</td>
<td>24.2</td>
<td></td>
</tr>
<tr>
<td>GI Dancer</td>
<td>69k</td>
<td>49.0</td>
<td>20.1</td>
<td>5.4</td>
<td>25.5</td>
<td></td>
</tr>
<tr>
<td>Bedroom</td>
<td>143k</td>
<td>34.3</td>
<td>15.2</td>
<td>13.7</td>
<td>28.9</td>
<td></td>
</tr>
<tr>
<td>Outdoor</td>
<td>348k</td>
<td>25.0</td>
<td>14.1</td>
<td>9.4</td>
<td>23.5</td>
<td></td>
</tr>
<tr>
<td>Rocks</td>
<td>88k</td>
<td>55.5</td>
<td>12.3</td>
<td>2.2</td>
<td>14.5</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1: For each scene in this paper: number of triangles, render time for our OpenGL renderer (in frames per second), and render time using RenderMan via Pixie (seconds per frame). Times for GI Bunny and GI Dancer for direct illumination from BendyLight only.

the global placement of the light tube through spotlight $S$’s local transform. This allows us to animate a BendyLight using traditional keyframing toolsets. See Sec. 4.5 for more information.

**Extensions.** This deformation model lets us “cheat” any parameter of a standard lighting model in a spatially coherent way, since all deformations occur smoothly along world-space splines. Additional lighting parameters like intensity, radial falloff, or tube cross section shape can be expressed by mapping one parametric space to another (e.g., a non-linear intensity spline’s parametric space → the deformation tube’s parametric space → world space).

4.4 Implementation

In this section we discuss details on how to render with BendyLights in real-time as well as in a production rendering system.

**Solving the Parametric Equation.** To solve equation (4.4), we must find the roots of
4.4 Implementation

a polynomial over $u$, the tube spline’s parametric variable (Sec. 4.3). This method must be fast, as it will be applied to every surface position in the scene. It must also be numerically stable, as the spline can bend arbitrarily and imprecise roots would cause strong artifacts. Since we use quadratic splines for $C$, equation (4.4) requires inverting a cubic in $u$. While a closed-form solution exists, we found that its direct use is often numerically unstable as intermediate values vary by several orders of magnitude. We instead use Cardano’s method [85] that produces a relatively fast and stable solution to this equation. Assume we have a cubic as in equation (4.4) written as $x^3 + ax^2 + bx + c = 0$. By substitution we can show that $x = t - a/3$ where $t$ can take the values

\[
\begin{align*}
t_1 &= \sqrt[3]{-\frac{q}{2} + \lambda} + \sqrt[3]{-\frac{q}{2} - \lambda} \\
t_2 &= \left(-\frac{1}{2} + \frac{\sqrt{3}}{2}i\right)\sqrt[3]{-\frac{q}{2} + \lambda} + \left(-\frac{1}{2} - \frac{\sqrt{3}}{2}i\right)\sqrt[3]{-\frac{q}{2} - \lambda} \\
t_3 &= \left(-\frac{1}{2} - \frac{\sqrt{3}}{2}i\right)\sqrt[3]{-\frac{q}{2} + \lambda} + \left(-\frac{1}{2} + \frac{\sqrt{3}}{2}i\right)\sqrt[3]{-\frac{q}{2} - \lambda}
\end{align*}
\]

(4.10)

with $\lambda = \sqrt{\frac{p^2}{4} + \frac{p^3}{27}}$, $p = b - \frac{a^2}{3}$ and $q = c + \frac{2a^3 - 9ab}{27}$.

To determine which of the roots correctly represent $u$, we assume that the light tube should not intersect itself, as a single light source should not hit a surface position at more than one direction under direct illumination. For any surface position $Q$, we iterate over each quadratic segment of tube spline $C$. If a $u$ exists such that $Q$ is within the tube at $u$, it is immediately accepted. Otherwise, we choose a default $u$ for which $Q$ will not interfere with shadow lookups. We implement another minor optimization by storing bounding boxes for tube segments so segments can be pruned for some surface positions.

Real-Time on the GPU. We implement BendyLights entirely in GLSL shaders. Our system renders in two passes. First, we generate a shadow map using the deformed geometry by solving for the light tube’s parametric variables and deforming vertex positions $Q$ to
4.5 Applying Light Study Findings

$Q'$ in a vertex shader. Depth tests are then carried out with standard shadow maps. Since the vertex transformation is non-linear, geometry must be tesselated to a degree. Note though, that this is not as problematic for shadow maps since lower precision can often be used, as is shown by the common practice of using approximated shadow meshes. Fig. 4.5 shows an example of tesselated wireframes. Second, we make a shading pass where vertices are passed through a vertex shader unchanged, but we solve equation (4.4) to get $L_i$. The deformed surface position $Q'$ is also calculated in a pixel shader to compute a hotspot with falloff. We do all shading per-pixel to ensure no shading artifacts appear.

**Offline Production Renderer.** Our GPU implementation shows that *BendyLights* can be implemented well using standard shadow mapping and shaders. This implies that our model can be integrated into production renderers for high-quality cinematic lighting. We use Pixie [4], an open source version of RenderMan, to render all of the figures in this paper that do not show editing gizmos. Our RenderMan implementation uses a light shader to compute the shading from the pixel shader in our GPU implementation. Aside from syntax, the two shaders are virtually identical. To compute the shadow maps, we first make a shadow pass with pre-deformed geometry using the vertex shader code from our GPU implementation. We could also have used raytracing on the deformed geometry for shadow computation, but chose shadowmaps for speed. Performance for each rendering implementation is summarized in Tab. 4.1.

4.5 Applying Light Study Findings

In the lighting interface study, we found that both direct and indirect control schemes are desirable when designing a lighting interface [44]. For this reason, we modeled *Bendy-Lights* so that they could be modified either with a direct or indirect interface using familiar
4.5 Applying Light Study Findings

**Figure 4.4:** An example of dragging shadows with a BendyLight using an indirect interface.

methods.

**Direct Control.** Using a direct control scheme, BendyLights can be controlled both rigidly and non-rigidly. Since control points of spline $C$ are defined with respect to a standard spotlight $S$, we can move, rotate and scale the entire tube the exact same way we would edit $S$. To deform the tube non-rigidly, a user drags each of the control points that define spline $C$. Each control point has an associated radius and RGB value that can be directly edited. See Fig. 4.1, Row 1 for some basic edits and the supplemental video from the original publication for a demonstration of all controls.

**Indirect Control.** We adapt the techniques in [71] to drag shadows, hotspots, and highlights by either transforming the tube rigidly or deforming it non-rigidly, demonstrated in Fig. 4.4. When we refer to deforming surface positions, we mean the process by which $Q$ is transformed to $Q'$ in Sec. 4.3. We start by explaining how to accomplish indirect edits that transform the BendyLight tube rigidly.

For **shadow dragging**, let $M$ be the surface position under the mouse pointer upon click after deformation. Let $P_i$ be the position of light source $S$ from Sec. 4.3. Let $P$ be a pivot point, which is the first intersection of the ray from $P_i$ to $M$ with the deformed geometry used for shadowmaps. Let $M'$ be the current surface position under the mouse pointer as
4.5 Applying Light Study Findings

the user drags the mouse, after deformation. We move $P_l$ to new position $P_l'$. 

$$P_l' = P + (P - M') \times |P_l - P| / |P - M|$$  \hspace{1cm} (4.11)

For hotspot dragging, $M$ and $M'$ define a rotation for $S$’s coordinate frame, which moves the hotspot with the mouse pointer. We define two vectors

$$v = M - P_l / |M - P_l|, \quad v' = M' - P_l / |M' - P_l|$$  \hspace{1cm} (4.12)

Using quaternions, we can rotate $S$’s coordinate frame around the axis $v \times v'$ by $cos^{-1}(v \cdot v')$ radians. This moves the hotspot with the mouse pointer no matter which point of the hotspot is clicked.

For highlight dragging, let $X$ and $N$ be the surface position and normal currently under the mouse pointer, and $V_i$ be the incoming view direction, none deformed. Let $L_i$ be the incoming light direction as in Sec. 4.3. We compute two vectors for a rotation as in the case with hotspots.

$$v = L_i, \quad v' = -V_i + 2(N \cdot V_i)N$$  \hspace{1cm} (4.13)

Here, $v'$ represents the view direction reflected over $N$. Using quaternions, we can rotate $P_l$ around $X$ along the axis $v \times v'$ by $cos^{-1}(v \cdot v')$ in radians. This aligns $L_i$ with the view reflection angle, assuring a specular highlight at $X$.

To drag shadows, hotspots, and highlights in a way that deforms the BendyLight tube, we have the user select which control points of the tube are of interest. We apply the same rigid edits as before, but instead of changing $S$, which affects all of $C$’s control points, we modify only the selected control points.
4.6 Results

In this section, we present the application of BendyLights. First we discuss the performance of BendyLights in our rendering implementations. Second, we show many examples of how BendyLights can be used to create useful artistic effects. Third, we illustrate key differences between BendyLights and traditional linear point lights.

4.6.1 Rendering

Scenes. We test BendyLights on a variety of scenes to show robustness. The first set illustrates simple edits and features for clarity. This set includes a bunny on a plane (Table-top Bunny: Fig. 4.1), two vases on a plane (Tabletop Vases: Fig. 4.5), a vase with a bowl (Tabletop Bowl: Fig. 4.1), a bunny next to colored walls (GI Bunny: Fig. 4.11), and an animated dancer next to colored walls (GI Dancer: Fig. 4.11). The second set is used to
4.6 Results

![Fig. 4.6: An example of using a static BendyLight to modify shadow placement in a dynamic scene. The dancer’s shadow is bent while maintaining the original lighting on the dancer herself. Shadows are spatially consistent across all objects.](image)

show BendyLights applied to scenes that might occur in production animation. This set includes an animated dancer in a bedroom (Bedroom: Fig. 4.7), an outdoor scene with an animated horse (Outdoor: Fig. 4.8), and a rocky hill with vases to show bumpy terrain (Rocks: Fig. 4.10).

**Performance.** In Tab. 4.1, we list each scene and a timing for each of our rendering systems. These timings are for a single BendyLight with three segments (same as Figure 4.7). We render images at $686 \times 382$ and shadowmaps at $2048 \times 2048$ resolution. The machine is an Intel Core2 Quad 2.83GHz with 4GB of RAM and an NVIDIA GeForce 9800GT GPU. Our editing system works in real-time.

4.6.2 Example Edits

**Light Shaping.** Because a BendyLight follows a spline, artists can shape the direction light travels in a way that is much like modeling a tube. Fig. 4.1, Row 1 shows what this looks like.

**Shadows.** When designing lighting, an artist might want to change the position or shape of shadows. In Fig. 4.5 we show how a single BendyLight can be used to modify the
4.6 Results

Figure 4.7: An example of an animated deformation of light directions in an animated scene with motion blur. We deform the light so that the shadow of the dancer always falls on the bed. BendyLights allow spline-based manipulation of light as it travels in world space, modifying incident lighting angles and shadows in a spatially consistent way.

shadow of the blue vase so that it no longer obscures the orange vase, without changing the way the blue vase itself is lit. Similarly, we can dynamically edit the shadows in a scene with animated geometry such as Fig. 4.6. Notice that the first object along a nonlinear light path occludes that light, casting a shadow on any other object the curved path intersects. The deformation of light directions is defined at all surface locations, meaning all objects added to the scene, including moving objects, will be lit similarly to other objects in close spatial proximity. Sometimes this effect can be accomplished with many linked lights, but other times it cannot (comparison in Section 4.6.3).

Lighting Direction and Intensity. Artists might also want to modify the direction at which light is hitting some objects. In Fig. 4.5, a BendyLight is used to light the blue vase more from the front without modifying the lighting hitting the orange vase. Similarly, we change the incident lighting on the bunny in Fig. 4.8 without changing the incident lighting on the woman standing on the rock. An animated horse runs through this scene to show
4.6 Results

Figure 4.8: An example of using a static BendyLight to modify the incoming light direction in a dynamic scene. The bunny is lit from a different direction while maintaining the light hitting the woman. As the horse moves across, its shadow remains consistent with those in close proximity.

how the light deformation is defined at every position in space. To the discerning eye, it may not appear that all light is coming from one source in this example, but the deformation gives the artist the power to modify these lighting directions in a way that guarantees that objects with close proximity to one another will have similar lighting. Light intensity can be spatially controlled in a similar way.

**Animated Light Deformation.** BendyLight deformations can be animated smoothly by keyframing and interpolating control point parameters. In Fig. 4.7 we show an animated BendyLight illuminating an animated scene with motion blur. The lighting bends so that as the dancer moves, her shadow always falls on the bed.

**Global Illumination.** Once the direct illumination is bent with BendyLights, other rendering techniques are executed linearly as usual. In Fig. 4.11 we show two examples of BendyLight with single-bounce global illumination computed by a gather loop. For example, we bend the light to get a more silhouette-like shadow shape of the animated dancer on the wall.
4.6 Results

Figure 4.9: The example from Fig. 4.8 with a comparison to linked lights. As the horse moves across the view, there is a point using linked lights where its shadow agrees with the dancer’s shadow, but crosses the bunny’s shadow. With BendyLights, the three shadows appear to agree with each other, even though they are nonphysical. Additional nonlineairties, such as the converging shadow rays from the tree on the right, causing a smaller shadow, are also not possible with traditional linked lights.

Figure 4.10: Using a BendyLight, the shadow of the blue vase is bent to look appealing against the rocks and orange vases. Casting this shadow on a plane reveals that the shadow has a nonlinear shape that would not be possible using a traditional linear light. This nonlinearity is masked by the natural terrain.

4.6.3 Comparison With Traditional Lighting

We present a comparison of BendyLights to standard linear lights and light linking, the process by which a light source can be “linked” to some objects and be active only on them.

Linear Lights Comparison. By exploiting the nonlinearity of BendyLights, artists can
4.6 Results

produce effects that would be impossible with a traditional linear light. Our first example is nonlinear changes in the radius of a *BendyLight*. In Row 3 of Fig. 4.5, the shadows of the vases are bulged and compressed by bending light rays. A more practical example is shown in Fig. 4.1, Row 2. With *BendyLights*, the shadow is bent so that it hits the outside of the bowl in a pleasing way. We show that the closest possible linear light in the far right image still distorts on the bowl. Finally, in Fig. 4.10 we show another example where lighting and shadows are bent, this time over a surface like natural terrain. We replace the rocky terrain with a planar surface to make the nonlinearity of the shadow more visible. Non-planar surfaces like the bowl and terrain make it easier achieve believable with such edits. In order to achieve these nonlinearities with linear lights in general, every pixel sample in the image would have to be lit by a different light.

**Linked Lights Comparison.** When *BendyLight* edits are more subtle, the resulting light features may appear locally linear (e.g., Rows 1 and 2 of Figure 4.5). A skilled artist might achieve the same with complex linear light linking setups. *BendyLights* offer two primary advantages over light linking. First, *BendyLights* require fewer light sources to achieve similar effects. Second, the lighting directions under *BendyLights* map uniquely to every position in world space, interpolating smoothly. Positions close to each other in space will be lit similarly. A comparison can be seen in Fig. 4.9. We take the *Outdoor* scene from Fig. 4.8 and attempt to reproduce the same lighting effects with three linked lights using traditional linear lights. The advantage of using *BendyLights* can be seen when the horse moves to the middle of the shot between the woman and the bunny. At this point, the shadows being cast must simultaneously be consistent with the dancer and the bunny. Using linked lights, if the horse’s shadow is consistent with the woman’s, it crosses the bunny’s shadow in an awkward way. With *BendyLights*, all shadows agree with each
4.6 Results

Figure 4.11: Examples of using a BendyLight with single-bounce global illumination. Left: angle at which light is hitting red wall and bunny changed to decrease redness of bunny. Right: animation of dancing woman, shadows bent to better silhouette her shape.

other at all times. We render the scene from an alternate camera angle to accentuate this difference (Fig. 4.9).

**Human Study.** We conducted a study to determine how tolerant viewers are of the non-physical lighting behavior of *BendyLights*. Subjects view 11 pairs of side-by-side images, one of a scene lit with a traditional spotlight, and the other of the same scene lit with a *BendyLight*. They then see the same 11 pairs of images where each image is mirrored along the horizontal axis, for a total of 22 pairs of images. The location (right or left) of the *BendyLight* image is randomized. Images used in the study can be found below. Subjects were given these instructions:

(a) Examine each pair of images.

(b) Either one or both of the two images is physically correct in terms of lighting. It may be the case that one image is not physically correct in terms of lighting.

(c) If one image is correct and the other is not, click on the button to indicate the correct image.

(d) If both images are correct, click on the button labeled “both.”

We run the study twice, once locally with 22 subjects and once on Amazon’s Mechanical Turk with 500 subjects. With Mechanical Turk we cannot control the physical conditions.
4.6 Results

of the experiment like room lighting, screen resolution, or whether the subject is actually trying. We show the second set of 11 pairs of mirrored images to measure how consistent subjects are with their answers. Through the local study, we estimate how consistent subjects should be, and apply this measure to eliminate Mechanical Turk subjects who might have been randomly clicking to get paid faster. Only 1 subject out of our local 22 chose with < 60% consistency, i.e., that subject chose the same answer for < 60% of the mirrored pairs compared to the original pairs. Since three local subjects were between 60% and 65% consistent, we set our threshold for Mechanical Turk subjects at 60%. This resulted in 355 subjects whom we allow to change their mind sometimes, but require to be more consistent than chance.

![Choice Distribution](image)

**Figure 4.12:** Result of study on 355 subjects. Each row represents the sum of choices for a pair of images and the same pair of images mirrored horizontally.

Figure 4.12 shows the resulting choices of the 355 Mechanical Turk subjects. Keep in mind that accuracy of the *BendyLight* image was determined directly compared to a linear light, allowing less tolerance than viewing the *BendyLight* alone. We selected scenes that would demonstrate an artist’s control when using *BendyLights*. Image pairs P5, P6, P9 and P11 were designed to be noticeably unrealistic lighting effects. Subjects thought the *BendyLight* images in these pairs were accurate < 20% of the time. P1, P2, P7 and P10
were designed to be appealing whether the light was bent or not. P2 was less convincing than we intended with only 31% of subjects thinking the *BendyLight* image was accurate, and P7 was more convincing with 79%. P3, P4, and P8 were designed to improve the unbent lighting with a *BendyLight*, and at least half of the subjects thought the *BendyLight* images in these pairs were realistic. The results show that *BendyLights* can be unrealistic when edits are extreme, equally realistic when compared to linear lights in many situations, and “better” than realistic in some situations where viewers claim a traditional linear light is not accurate by comparison. We conclude that *BendyLights* give artists sufficient power to make edits expressing a wide range of realism compared to traditional linear spot lights, as required by their application.

Images used in the study can be found in the appendix at the end of this document.

### 4.7 Discussion of Results

We have presented *BendyLights*, the first lighting model for artistic control in which, conceptually, light travels along nonlinear spline paths. This allows artists to deform shadows and set light directions independently at different scene locations, while still using only one light, which is consistently defined at all surface locations and times (during animation). *BendyLights* can be controlled using familiar editing interfaces, and work in scenes containing a variety of geometric complexities and with animation. They can be easily deployed both in GPU-based real-time editing systems as well as offline production renderers such as RenderMan, integrating with effects like motion blur and global illumination.

**Limitations.** *BendyLights* come with a few limitations. First, if the *BendyLight* tube self-intersects, shading artifacts can occur. This can be avoided by simply constraining the tube from self-intersecting during editing. Second, in order for shadows to appear smooth
4.7 Discussion of Results

when bent, geometry must be tesselated to an appropriate degree, because *BendyLights* deform geometry for visibility. We find that the required tesselation is reasonable, especially for production scenes where many surfaces are already non-linearly deformed for animation purposes.

**Future Work.** In the future, we are interested in extending the ability to nonlinearly bend light to other light types, in particular area lights and environmental illumination. We would also like to explore global illumination methods where all bounces are bent.
Chapter 5

Material Design Interface Study

Following the success of the lighting interface study [44], we began investigating how novices use interfaces for material design. We designed and conducted a user study similar to our previous work with the task of editing analytic BRDFs. The work originally appeared at SIGGRAPH 2010 [45].

5.1 Overview of Project

Real-world materials exhibit a wide variety of reflectance behaviors, from matte surfaces to highly glossy finishes. For opaque materials, the bidirectional reflectance distribution function (BRDF) [62] captures the directionally-varying appearance of real-world surfaces. Material design is the process by which artists define properties of surface materials, such as their color, specular roughness, etc. This can be a difficult and time consuming task, since the variety of real-world materials is large. Many user interfaces have been proposed to simplify the process.

The paper represents a first step toward quantitatively evaluating the effectiveness of
5.1 Overview of Project

Figure 5.1: Example workflow of two novice subjects editing materials using three different interfaces for material design.

user interfaces for material design. We focus on novice users without previous experience in material design since they stand to gain the most from intuitive interfaces and represent the majority of potential users of computer graphics design applications. We are specifically interested in the task of editing the parameters of realistic materials represented as analytic BRDFs, since they are the simplest and most commonly used models. We conduct a user study investigating the relative effectiveness of three interactive material design interfaces: physical sliders, by which users set the parameters of analytic BRDF models, such as diffuse albedo and specular roughness; perceptual sliders by which users set perceptually-inspired parameters, such as diffuse luminance and gloss contrast; and image navigation by which material variations are displayed in arrays of image thumbnails and users make edits by selecting them. Fig. 5.1 shows examples of output from our subjects using each of these interfaces.

We simplify material design tasks so that they can be accomplished by novices and effectively measured, focusing on physically-based isotropic BRDFs of real-world materials.
5.1 Overview of Project

respected by the Cook-Torrance and Ward BRDF models [24, 89]. We choose the Cook-Torrance model since Ngan et al. find it fits well to measured material data, with Ward fitting well in many cases [60]. We specifically include Ward BRDFs since a perceptual parameterization of this model has been published [69]. Much previous work in material design focuses on the modeling and editing of achromatic reflectance, so we investigate the use of material design interfaces with and without color. We also investigate the editing of analytic BRDF parameters in the presence of spatial variation across a surface, but not the editing of the spatially-varying patterns themselves.

We follow the methodology from our lighting study to investigate how novices use these interfaces to make precise adjustments and artistically explore broad material variations. Our study consists of three parts. First, in six matching trials, subjects are asked to match a target BRDF as closely as possible. Second, in three open trials, subjects use their own creativity to design a BRDF based on a suggestive image. Finally, in questionnaires, subjects give both quantitative and qualitative feedback about the interfaces. Twenty subjects spend roughly one hour with each of the three interfaces. We design the study so that editing is carried out in real time.

As with any user study, our observations only strictly apply within the boundary of the tested cases. However, we believe that these trends are general enough to apply to other closely-related material domains, affecting the development of future material design interfaces. Additionally, we believe that the principles used to design our study can be employed to evaluate additional material design tasks in moving toward a comprehensive evaluation of material design interfaces.
5.2 Related Work

Here, I review the body of work in material design interfaces for BRDFs that make up each of the three interface paradigms tested in our study. I also discuss some of the choices we made in limiting the scope of material design. For a review of other user interface and human studies in computer graphics see Sec. 2.

**Analytic BRDF Models.** Of the various BRDF representations, we are specifically interested in the editing of analytic BRDF models since they give users the ability to define a BRDF using only a small number of parameters. We choose to study the editing of Cook-Torrance BRDFs [24], since they fit measured data well [60], and the isotropic version of the Ward BRDFs [89], since they still fit measured data reasonably well [60] and since a perceptual parameterization has been investigated by Pellacini et al. [69]. To avoid confusing novices with a large number of models, we do not include other BRDF models that are commonly used to represent realistic materials, such as Blinn-Phong, Lafortune, He, and Ashikhmin-Shirley [18, 48, 39, 5].

**Physical Sliders.** A common practice in material editing is to edit a BRDF by modifying directly the parameters of the analytic model. For example, a user might input values for diffuse albedo and specular roughness. Off-the-shelf modeling and animation software such as Maya [7] use this type of interface.

**Perceptual Sliders.** Since physical parameters are only indirectly related to the perceived appearance of a material, researchers have investigated perceptual parameterizations where each parameter represents a perceptually-meaningful dimension of surface appearance, such as diffuse luminance or gloss contrast. These parameterizations are also scaled such that linear changes in the parameters yield linear changes in the perceived appearance of the surface material. Pellacini et al. develop a perceptual parameterization of the
5.2 Related Work

Ward BRDF model through psychophysical experiments [69]. Westlund et al. investigate correspondences between traditional appearance standards such as gloss, sheen, and haze and analytic material models [92]. Wills et al. develop a method for finding perceptual embeddings of measured material data and a method for traversing that embedding [93]. In this work, we investigate whether manipulating perceptual parameters has benefits over manipulating physical ones in editing tasks. Since no perceptual parameterizations exist for all BRDFs used in this study, we develop perceptually-inspired parameterizations based on [69] and [61].

**Image Navigation.** With image navigation, a user can view variations of materials by browsing arrays of thumbnail images. Material edits are made by selecting a desired image. Marks et al. explore this idea for computer graphics and suggest arrangement schemes for displaying sets of images with variations of interest [52]. Adobe Photoshop [1] uses an interface called “variations” to show multiple image configurations resulting from photographic adjustments like hue and saturation changes. More recently, Ngan et al. propose a user interface specific to BRDF editing that uses a perceptually-inspired image difference metric to arrange possible BRDF configurations with perceptually-uniform spacing, including varying between BRDF models [61]. Some argue that since novices do not have a deep understanding of material appearance, allowing them to choose rendered images directly might be beneficial. In this study, we compare this interface to direct parameter setting.

**Other Interfaces.** [20] suggest a painting interface for editing BRDF highlights with brush tools. Similarly, [66] develop a painting interface for non-photorealistic highlights. We do not include such interface types in this study because they utilize custom material models specific to the control scheme. [73] introduce the idea of optimizing material pa-
5.3 Study Overview

rameters to match painted color points on surface geometry. We exclude this interface type because it is unclear how to extend it to support texture variations robustly. BRDFs can also be defined by curves over an angular parameterization. Lawrence et al. use such a control scheme for editing factored components of nonparametric BRDFs [49]. We do not include curve editing because its full space of possible BRDFs cannot be created using the interfaces we study. Finally, material appearance can be specified by writing a script in a shading language that can be interpreted by the program that renders the image [23]. We don’t investigate the use of shader programming as a material editing interface, because a great deal of training would be required to teach programming to novices.

Spatial Variation Editing. Editing spatially-varying BRDFs is considerably harder than editing single BRDFs. Three editing tasks are normally performed on spatially-varying BRDFs: changing the spatial patterns (by texture painting or synthesis), selecting regions of similar appearance (e.g. using [70]) and altering the BRDFs of the selected regions. In this paper, we focus on interfaces for editing the parameters of BRDFs, leaving the study of interfaces for editing and selection of spatial patterns to future work. To test whether the presence of spatial variation affects the editing of BRDFs, we investigate the editing of spatially-varying BRDFs represented as linear combinations of basis BRDFs with spatially-varying weights [49], where users edit the parameters of the basis BRDFs. We choose this model since it fits measured data well.

5.3 Study Overview

Goal. We seek to evaluate the relative effectiveness of different interface paradigms for material design in the context of designing realistic materials with a focus on novice users. Specifically, (1) we want to measure how efficiently these users can perform specific ma-
5.3 Study Overview

terial *adjustments* and (2) we want to understand which interface paradigms provide better artistic *exploration* of possible material variations.

**Novice Users.** We focus on *novices* with little or no prior knowledge of material design since they make up the majority of users who can take advantage of intuitive interfaces. We would like to have as many people as possible capable of using graphics tools. All subjects rated their experience level with material design as either 1 or 2 on a scale from 1 to 5, and can be considered novices.

**Reducing Complexity.** Since material design is a non-trivial process, we require a careful triage between completeness and length. On the one hand, we want to achieve complex-enough material editing tasks to ensure meaningful measurements. On the other, we want to avoid bias in the data by ensuring subjects can successfully complete the required tasks without incurring too much fatigue. Working with novices makes this triage even more necessary. We simplify the material editing task by focusing on editing the parameters of analytic BRDFs, and while we include different BRDF models, we do not ask subjects to select between different models during trials. We simplify implementation of interfaces to ensure that they can be quickly learned while sufficiently complete to capture the main characteristics of each paradigm.

**Materials.** In our design tasks, subjects edit materials represented as isotropic Ward [89] and Cook-Torrance [24] BRDFs (Sec. 5.2). We investigate three variations of these models. First, we use achromatic materials for half of the trials and color for the other half. Much previous work in material design focuses on the modeling and editing of achromatic reflectance, and we would like to discover how important chromaticity is in the design process. Second, we include two trials where BRDFs have two specular lobes, since, for some materials, such BRDFs fit measured materials better than single lobe ones [60]. Third,
5.3 Study Overview

to determine whether the presence of spatial variation affects the design tasks, we investigate the editing of spatially-varying BRDFs represented as linear combinations of two basis BRDFs with spatially-varying weights, where users edit the parameters of the basis BRDFs. We choose this model since it fits measured data well [49]. Examples of each material type can be found in Fig. 5.2.

**Lighting.** In our study, materials are lit by direct illumination from a real-world environment map. Natural illumination is considered ideal for material perception when only a single image is available [28, 31]. We use the Grace Cathedral environment map [27] since [61] suggest that the choice of illumination environment has little effect on material distinction as long as it is natural, and recommend the Grace Cathedral map. We choose to use direct illumination, rather than global illumination, since we want to preserve interactivity and high image fidelity during the design task. We use the tone mapping equation $Image = (Intensity \cdot 2^{exposure})^{1/\text{gamma}}$ with a gamma of 2.2. Exposure is fixed so that the goal for a trial is clearly visible. Subjects have no control over exposure or gamma.

**Geometry.** The geometry in our images consists of a sphere floating in space. We use a sphere to avoid occlusion artifacts in glossy reflections caused by computing direct illumination only. Vangorp et al. [87] suggest there may be shapes better than spheres for material discrimination, but that spheres possess many desirable properties. We determine that a sphere shape is the best for our purposes given our rendering limitations.

**Interfaces.** We compare three user interfaces: physical sliders, perceptual sliders, and image navigation. Implementation details of these interfaces are presented in Sec. 5.5. Physical and perceptual slider interfaces are similar in that they are designed to manipulate one parameter at a time. Physical sliders alter the parameters of analytic BRDF models directly; these parameters are related to physical reflectance properties such as diffuse en-
5.4 Experiment

ergy and specular roughness. Such parameters, though, are not directly related to material appearance. Perceptual sliders alter perceptually-meaningful material parameters, such as diffuse luminance and gloss contrast. These parameters are scaled to be linearly related to perceptual distances. Perceptual parameterizations are believed to be more natural for editing purposes since they are directly related to how humans perceive materials. With image navigation, a user can view variations of materials by browsing arrays of thumbnail images. Material edits are made by selecting a desired image. It is believed that image navigation is a useful editing metaphor since novices can directly select rendered images, rather than specifying parameter values, and since they can preview several variations and combinations of multiple parameters simultaneously.

**Tasks.** We ask subjects to perform two types of material design tasks. During *matching trials*, they are asked to match a material of an object under fixed environmental lighting to an image of the same object and lighting with a target material configuration. Matching trials allow us to quantitatively measure users’ performance, while providing a clear goal for subjects who have never experienced material design before.

This provides context for the more subjective *open trials*, where users are given a photograph of several real-world objects with a round target area removed and asked to creatively design a material that would look good if assigned to an object placed in the target area. These trials allow us to observe how users artistically explore the space of possible material configurations, a more natural but harder to measure task.

5.4 Experiment

We ask subjects to complete a number of trials, during which all actions are recorded for further analysis. Each subject performs all trials using all interfaces. These trials vary in the
5.4 Experiment

number of material parameters, material model type, number of lobes, presence of color, presence of spatial variation, and task goal.

Preparatory Studies. We conducted formal and informal preparatory user studies on 15 additional subjects, the results of which are not included in this paper. Different implementations of the various interfaces were tested to determine a locally optimal set of controls and to remove any implementation errors. The open and matching goals used in the final study were tested to ensure that time limits were appropriate and that the tasks could be completed.

![Starting and goal configurations for training and matching trials. Material models are listed above. Time limits are listed below (see Sec. 5.4).](image)

Trials. We perform six matching trials and three open trials with a progressively increasing number of degrees of freedom in the material model. Starting configuration, goal configuration, and time limit for each trial are summarized in Fig. 5.2 and Fig. 5.3.

For matching trials, goal configurations were taken from parametric fits presented in [60] to measured materials in [54]. For grayscale trials 1 and 2, the diffuse and specular coefficients of “metallic blue” and “white bball” were desaturated. For grayscale trial 3, the goal was modeled by hand after a rendering of “acrylic violet”, since a fit was unavailable.
5.4 Experiment

[Image: 2-lobe Ward (gray), Cook-Torrance (color), Textured Ward (color)]

**Figure 5.3:** Goals for open trials. Starting configurations are identical to matching trials 3, 5, and 6.

Color trials 4 and 5 use fits for “blue bball” and “ch-ball-green-metallic” respectively. Trial 6 uses fits for “white-bball” and “metallic gold” weighted by a texture. We vary matching trials in material complexity to observe possible changes in users’ workflow and interfaces’ effectiveness under these conditions.

For open trials, we select target images that differ from the workspace lighting environment and vary in content to encourage free-form artistic exploration. We choose one grayscale and two colored material goals with objects of varying material properties. Objects in the same goal image share some material properties to keep the objective from being completely unspecified.

The same initial and goal material configurations are used for all subjects and all interfaces. Each trial has a fixed time limit, and subjects can end the trial sooner if satisfied with the current result. At the end of each matching trial, subjects rate the accuracy of the matching on a scale of 1 to 5. For open trials, subjects use the same scale to rate how satisfied they are with their result.

**Questionnaire.** After performing all trials with all interfaces, subjects complete a questionnaire where they rate each interface on a scale of 1 to 5 in the following categories: (1) natural way to think about material editing, (2) preference in matching trials, (3) prefer-
5.4 Experiment

ence in open trials, and (4) overall preference. Subjects also strictly rank interfaces in each of these categories. Immediately after finishing trials for each single interface, subjects are asked to leave free-form comments on aspects of each interface. For reproducibility, we include copies of the questionnaires as additional material.

**Procedure.** Twenty subjects participated in the study, chosen from different age and educational groups. All subjects had normal or corrected-to-normal vision. Subjects edit materials for about three hours each to ensure good statistical significance of our tests, while keeping fatigue low.

Subjects complete the study in three 60-minutes sessions, one for each interface. We randomize the order of the interfaces for each subject. Before each session, subjects complete a training phase to become familiar with the specific interface. We train each subject individually to allow questions, accommodating each subject’s learning needs. The instructor verifies that the subject uses each part of the interface, and answers the subjects’ questions. Before proceeding to the experiment, the subject uses the interface until he or she feels comfortable. During both the guided and free portions of the training, a single sample goal was shown (Fig. 5.2). Once trials begin, all user interface actions are recorded.

The study is conducted in a controlled lighting environment with negligible ambient lighting, to simulate typical working conditions of artists and maximize visibility of the screen. We use a 24-inch Dell 2407WFPb LCD display at $1280 \times 800$ resolution at a distance of approximately 1 foot from the subject (monitor native resolution: $1900 \times 1200$). All rendered images are 256x256 pixels on screen covering an area of 4 square inches. We used an Intel 2.8 Ghz Core2 Quad Q9550 PC with 4 GB of RAM and an NVidia GeForce 9800 GT graphics card.
5.5 Interface Implementation

In this section, we discuss our implementations of the user interfaces included in the study. For reproducibility, we included a video as supplemental material with the original publication that showed each interface in detail.

**Rendering.** We use the real-time rendering algorithm of [13] to preview BRDF edits under direct natural illumination. Our implementation renders 45 frames per second (fps) on the 256 x 256 pixel images of a sphere used in the study. We considered adding global illumination as in [12], but decided that the potential artifacts resulting from approximating the BRDF at a lower frequency might affect our measurements. The algorithm we use doesn’t allow the roughness and Fresnel terms of the Cook-Torrance BRDF model to be simultaneously modified. It takes approximately 0.6 seconds to switch between these parameters in our implementation.

**BRDF models.** In our implementation, we parameterize the isotropic Ward BRDF $\rho_w$ as

$$
\rho_w = \frac{\rho_d}{\pi} + \rho_s \frac{e^{-\tan^2 \theta_h/\alpha^2}}{4\pi \alpha^2 \sqrt{\cos \theta_i \cos \theta_o}}
$$

(5.1)

where $\rho_d$ is the diffuse albedo, $\rho_s$ is the energy of the specular component, $\alpha$ is the surface roughness and $\theta_i, \theta_o, \theta_h$ are the angles between the surface normal and the incoming, outgoing and half-angle respectively.

We parameterize the Cook-Torrance BRDF $\rho_{ct}$ following [60] as

$$
\rho_{ct} = \frac{\rho_d}{\pi} + \frac{\rho_s}{\pi} \frac{DGF}{\cos \theta_i \cos \theta_o}
$$

(5.2)

with

$$
F = F_0 + (1 - F_0)(1 - \cos \theta_h)^5,
$$
5.5 Interface Implementation

![Study interface layout.](image)

Figure 5.4: Study interface layout.

\[ D = e^{-\left(\tan \theta / m\right)^2}, \]

\[ G = \min \left(1, \frac{2 \cos \theta_h \cos \theta_i}{\cos \theta_b}, \frac{2 \cos \theta_h \cos \theta_o}{\cos \theta_b}, \frac{2 \cos \theta_h \cos \theta_i}{\cos \theta_b}, \frac{2 \cos \theta_h \cos \theta_o}{\cos \theta_b} \right), \]

where \( \rho_d \) is the diffuse albedo, \( \rho_s \) is the energy of the specular component, \( \alpha \) is the surface roughness, \( F_0 \) is the Fresnel reflectance for a direction orthogonal to the surface, and \( \theta_b \) is the angle between the outgoing and half-vector direction.

Some trials use BRDFs \( \rho_{ww} \) with 2 Ward lobes defined by

\[ \rho_{ww} = \frac{\rho_d}{\pi} + \lambda \left( \rho_{s1} e^{\left(-\tan^2 \theta / \alpha_1^2\right)} + \rho_{s2} e^{\left(-\tan^2 \theta / \alpha_2^2\right)} \right) \]

(5.3)

where \( \lambda = 1 / \sqrt{\cos \theta_i \cos \theta_o} \). In other selected trials, we use a spatially-varying material. This spatial variation is modeled as a weighted sum of two Ward BRDFs \( \rho_{\text{sum}} = w_1 \rho_{w1} + w_2 \rho_{w2} \), where the weights \( w_1 \) and \( w_2 \) are spatially-varying and sum to one at all surface points [49].

**Universal Interface Features.** All interfaces use the same screen layout consisting of a workspace window, a goal window, and rating buttons (Fig. 5.4). An undo key allows the user to walk back through an unlimited number of edits. To compensate for the fact that
5.5 Interface Implementation

materials created using this system may not conserve energy, a warning indicator appears in the upper right corner of the user’s image when the BRDF is not energy conserving.

**Physical Sliders.** We use a slider interface as the means by which a user sets the parameters of the BRDF model, e.g. the diffuse albedo $\rho_d$, specular energy $\rho_s$, roughness $\alpha$, $m$ and Fresnel term $F_0$. Each user controlled parameter is listed with a slider bar next to it. The parameter can be changed by clicking anywhere on the bar, and gradual changes can be seen by dragging the slider continuously across the bar.

Setting model parameters directly would require specifying the red, green and blue coefficients of $\rho_d$ and $\rho_s$. This would ignore common color editing practices, artificially handicapping the interface. We use CIELAB luminance (L) for achromatic intensity, and saturation and hue for chromaticity [29]. We use hue and saturation since they are the default in Maya [7].

**Perceptual Sliders.** Perceptual parameterizations differ from physical ones in both effect and scale. In this paper, we choose perceptually-inspired parameters based on [69, 92]. Furthermore, since all perceptual parameterizations are derived from achromatic data, we follow [93, 69] and derive parameters for grayscale diffuse and specular components, and then add hue and saturation to them. We use the same saturation and hue controls as with physical sliders. Slider controls work the same way as with physical sliders, but modify the perceptually-inspired parameters.

To determine the correct scaling of each parameter axis in BRDF model’s configuration space, we use the image-based BRDF difference metric from [61] since psychophysical data has not been published for the range of BRDF parameters we investigate. Letting $I(\rho)$ be the image corresponding to BRDFs $\rho$, we can approximately compute the perceptual distance, $d$, between BRDFs $\rho_1$ and $\rho_2$ as
5.5 Interface Implementation

\[
d^2(\rho_1, \rho_2) = \sum_{p \in \text{pixels}} \sum_{c \in r,g,b} \left( 3\sqrt{I_{p,c}(\rho_1)} - 3\sqrt{I_{p,c}(\rho_2)} \right)^2
\] 

We scale our perceptually-inspired parameters such that equal steps of the parameter yield steps according in the distance metric. We included a comparison of this metric to our parameterizations as a supplemental document with the original publication. For all parameterizations that follow, \( \rho_d \) and \( \rho_s \) are represented achromatically according to CIELAB luminance in the range \([0, 1]\).

For Ward BRDFs, we use the parameterization from [69] with a modified \(d\) parameter:

\[
L = \rho_d
\]

\[
c = \sqrt[3]{\rho_s + \rho_d/2} - \sqrt[3]{\rho_d/2}
\]

\[
d = 1 - \alpha^{1/4}
\]

where \(L\) is the diffuse luminance, \(c\) is the gloss contrast, and \(d\) is the gloss distinctness. We raise \(\alpha\) to a power of \(1/4\) because it more closely matches scaling according to equation (5.4) which is valid for a larger range of \(\alpha\) than the original experiment covered in [69]. The trials using textured Ward simply have two instances of the perceptually inspired Ward parameters.

For Cook-Torrance BRDFs, we use the following parameterization:

\[
L = \rho_d
\]

\[
c = \sqrt[3]{\rho_s F_0 + \rho_d/2} - \sqrt[3]{\rho_d/2}
\]

\[
d = 1 - m^{1/4}
\]

\[
s = \sqrt[3]{\left[\left(1 - F_0\right)\epsilon\right]/\left[\left(1 - \epsilon\right) F_0\right]}
\]

where \(L\) is the diffuse luminance, \(c\) is the gloss contrast, \(d\) is the gloss distinctness, \(s\) is the gloss sheen and \(\epsilon = 0.02\) is the minimum allowed value of \(F_0\) when \(s \in [0, 1]\). Cook-
Torrance parameters $L$, $c$, and $d$ are similar to their Ward counterparts; with added $s$ to set the contrast of the specular component at grazing angles while preserving its contrast at non-grazing angles.

For Ward BRDFs with two lobes, we use the following parameterization:

\[
\begin{align*}
L &= \rho_d \\
\frac{c}{3} &= \sqrt[3]{\rho_{s1}} + \rho_{s2} + \rho_d/2 - \sqrt[3]{\rho_d/2} \\
\frac{b}{\rho_{s1}} &= \rho_{s1}/(\rho_{s1} + \rho_{s2}) \\
\frac{d}{4} &= 1 - \frac{\alpha_1^{1/4}}{\rho_{s1}} \\
\frac{h}{\alpha_2^{1/4}} &= (\alpha_2^{1/4} - \alpha_1^{1/4})/(\alpha_{\text{max}}^{1/4} - \alpha_1^{1/4})
\end{align*}
\]

where $L$ is the diffuse luminance, $c$ is the gloss contrast, $b$ is a lobe blending parameter, $d$ is the overall gloss distinctness, $h$ is a haze parameter and $\alpha_{\text{max}}$ is the maximum possible $\alpha$ value for normalization.

**Image Navigation.** We base our implementation of image navigation on [61]. Their interface consists of a series of tabs that reveal different image arrays. Some tabs show variations of material model parameters along two axes, while others serve as color pickers for the diffuse and specular coefficient parameters. Images are spaced according to the image
5.5 Interface Implementation

difference metric in equation (5.4), and the spacing size is determined by a user-controlled slider. As in [61], we limit the interface to display only two parameters simultaneously to ensure that thumbnails are large enough to perform accurate selection.

Fig. 5.5 shows what our two-parameter layout looks like using image navigation. We implement a system by which all model parameters can be assigned to either a horizontal or vertical axis. From the current configuration, two steps in either direction for either parameter axes are shown. This results in a five by five image array of 25 images representing different combinations of two parameters. We also give the user preset configurations that are helpful combinations of parameters to reduce confusion (e.g. diffuse versus specular brightness or a diffuse color picker). Since our perceptually-inspired parameterizations scale similarly to the difference metric in [61], we space images by equal steps in that parameter space. This may cause the space displayed in a 2D image to be scaled differently on the horizontal than the vertical in error space even though they are uniform in parameter space, but we don’t find it to be a problem. We do not allow a slider to determine the size of these steps, because with real-time feedback we feel this would be like taking the perceptual sliders interface and simply giving multiple previews at a time. By giving buttons that increase and decrease the step sizes on a log scale, we keep image navigation and perceptual sliders implementations to their respective interface metaphors, while giving image navigation the power to make small and large edits. Rendering time for the thumbnails depends on the material configuration, but is normally 0.25 seconds with the exception of arrays where both gloss distinctness and sheen vary simultaneously where it is 2.5 seconds (see previous section). We account for this in our analysis.
5.6 Analysis

We present an analysis of our data in two parts. First, we analyse the output of the rendering system as subjects proceed through each trial. Second, we analyse the feedback from users at the end of each trial and in the questionnaires. Unless stated otherwise, tests for statistical significance are computed with repeated measures analysis of variance (ANOVA) [83]. This handles within-subject factors that create correlations which invalidate the assumption of independence in standard one-way ANOVA. A $p$ value below 0.1 indicates a 90% confidence that the two population means differ given the measure of the sample. In all figures, error bars represent standard error.

**Time to Completion.** We investigate the work speed of users with each interface. Generally, subjects are able to complete each trial within the allotted time limit with one or more interfaces. In Fig. 5.6, we show the mean time to completion for each matching trial over all subjects.

Time to completion for image navigation is almost always significantly higher than either physical or perceptual sliders on matching trials ($p \leq 0.051$), excepting trial 6. We
5.6 Analysis

believe trial 6 differs because many subjects ran out of time or gave up early with image navigation, reducing its mean time and resulting in matches of lower quality. We conclude that image navigation must be slower to work with on trial 6, and that we are reaching the limit of subjects’ patience. The time to completion for physical and perceptual sliders shows no significant difference on trials 2-6, but physical sliders average 20 seconds faster than perceptual sliders on trial 1 ($p = 0.053$).

In open trials, the meaning of time to completion is less defined since the standard of judgement used by the subject can vary from trial to trial or even interface to interface. The only statistically significant differences ($p < 0.1$) were between perceptual sliders (69.0s) and image navigation (107.5s) on trial 7 ($p = 0.039$), and physical sliders (113.5s,201.9s) and image navigation (150.9s,179.3s) on trials 8 ($p = 0.080$) and 9 ($p = 0.048$) respectively.

Trial 1 in grayscale and trial 4 in color use the same BRDF model, as do trials 2 and 5. The average factor of time to completion between grayscale trials and color trials is 1.886.

**Matching Error.** To evaluate user performance in matching trials, we compute the error between the subject’s BRDF and the goal BRDF using the image-based difference metric in equation (5.4) [61]. This metric has been shown to capture perceived differences in BRDFs.

Fig. 5.7 shows the error over time for one subject performing the same trial with all interfaces. When subjects are successful, error decreases toward the correct solution, converging on some low error value. This convergence is not monotonic, because users explore the configuration space in order to reach the desired goal. In the accompanying supplemental material for the original publication we included error graphs for all subjects on all trials together with rendered images of their material configurations at fixed time intervals.
5.6 Analysis

![Graphs of error over time for individual and averaged trials.]

**Figure 5.7:** Left: example graphs of error over time (in seconds) for the individual trials shown in Fig. 5.1. Right: illustration of the error over time (in seconds) for matching trials averaged over all subjects. Error values are from equation (5.4), no normalization.

To summarize the overall performance of each interface we analyse the final image error for each matching trial averaged over all subjects (Fig. 5.6). Both physical and perceptual sliders outperform image navigation on all trials \( (p < 0.064) \) except for trial 4, where image navigation has roughly the same error as perceptual sliders. However, it took a longer amount of time to complete this trial with image navigation \( (p \leq 0.026) \). The error on trial 2 is especially high for image navigation. This could in part be due to the rendering limitation specific to image navigation on Cook-Torrance BRDFs (Sec. 5.5). However, this anomaly cannot be seen in trial 5, which also uses the Cook-Torrance model. The goal in trial 2 happens to be particularly bright, and this error discrepancy is not as pronounced when error is computed with clamped intensity values. We conclude that these failure cases are reasonably in alignment with the rest of the data when taking this into account.

Surprisingly, there is no significant difference in errors between perceptual and physical sliders except on trial 2 where physical sliders outperform perceptual sliders \( (p = 0.064) \).
5.6 Analysis

This trial again exhibits a difference that we cannot identify conclusively. The other Cook-Torrance trial does not show such a difference between physical and perceptual sliders, nor do the other grayscale trials.

As with the time to completion, we compare grayscale trials 1 and 2 to color trials 4 and 5. The average factor in error between grayscale trials and color trials is 2.167.

Convergence. To illustrate the convergence behavior of different interfaces we average the image error across all subjects over time in Fig. 5.7. This average is not statistically valid, but we find that it gives a revealing visual summary of overall behavior. As can be seen in the graphs, physical and perceptual sliders tend to converge more quickly than image navigation, and with lower error. We also see that convergence behavior of physical and perceptual sliders are similar, though trial 2 seems to show better convergence with physical sliders. Finally, we note that trials 2 and 6 show particularly poor convergence for image navigation. In trial 6 with a spatially-varying BRDF, many subjects give up or run out of time using image navigation. We also see slower and poorer convergence with the slider interfaces on this trial. Again, we cannot identify conclusively what causes the differences in trial 2.
5.6 Analysis

**Subjective Image Quality.** At the end of each trial, subjects rate their work on a scale of 1 to 5, with 1 being the worst and 5 being the best. Matching trials are rated in terms of how close the workspace and goal images match. Open trials are rated in terms of how satisfied the subject is with their result. Fig. 5.8 shows the average ratings for each trial. This subjective image quality correlates with the computed error of the final image with a linear correlation coefficient of $-0.5895$.

In matching trials, subjects on average rate their work better when using sliders than with image navigation on all trials ($p \leq 0.058$). Not only do subjects perform objectively better using slider interfaces compared to image navigation as measured by error, they perceive themselves as doing better as well. Ratings for the slider interfaces compared to one another contain no significant differences, except on trial 2 ($p = 0.042$), as with the computed error.

In open trials, there is no significant difference in the image ratings between any of the three interfaces, except for physical sliders having a slightly higher average rating than image navigation on trial 7 ($p = 0.015$).

**Interface Rankings and Ratings.** Subjects rate and rank each interface in 4 categories where ratings can have ties, but rankings are forced choice (see Sec. 5.4). Average ratings and stacked frequencies of rankings are shown in Fig. 5.9. For evaluating statistical significance of ranks we use the Friedman test [32], a nonparametric test that takes into account within-subject effects. A low $p$-value indicates high confidence that subjects have made a significant distinction between two interfaces.

In all categories except preference on open trials, slider interfaces outrank image navigation ($p = 0.074$ on perceptual vs. image navigation in the natural category, $p \leq 0.002$ otherwise). We find no statistical difference between ranks for the two slider interfaces.
5.6 Analysis

![Interface Ratings and Rankings](image.png)

**Figure 5.9:** *Left: Average interface ratings from questionnaire over all subjects. Rating 5 implies best. Right: Sum of interface rankings over all subjects. Rank 1 implies best.*

Roughly half of subjects rank physical sliders higher than perceptual in overall preference, and vice versa.

We find similar trends in the interface ratings. The slider interfaces average to roughly equivalent in all categories. When comparing image navigation to slider interfaces, except for open trial preference, image navigation is rated much lower in all categories ($p \leq 0.002$).

**Complexity.** We have shown that the interface used to perform material editing influences performance. The complexity of the material being edited has an effect on how difficult the task is to perform, but the relative performance of the interfaces remains unchanged. Difficulty seems to scale linearly on average with the number of user controlled parameters in the material model. Regression on average time to completion suggests a linear relationship ($r^2 \geq 0.919$). Error has a similar trend with physical ($r^2 = 0.996$) and perceptual ($r^2 = 0.880$) sliders, but not as much with image navigation ($r^2 = 0.499$). It is unclear if we increased complexity further, that novices would still be able to accomplish the task. Trial 6 appears to indicate that there is a point at which many subjects will give up. Our data does not indicate any significant trends in material editing between using the
5.7 Workflow Observations

Ward or Cook-Torrance BRDF model. The Cook-Torrance BRDF in trial 2 appears to be more challenging than the Ward BRDF in trial 1, but the Cook-Torrance BRDF in trial 5 appears to be less challenging than the Ward BRDF in trial 4. We make no claims as to the usefulness of one model over another, as our study is not designed to give subjects a choice between the two.

5.7 Workflow Observations

We now discuss common trends in the way our subjects use the different interfaces to edit materials. In Fig. 5.1 we show work done by two different subjects. Corresponding error graphs can be found in Fig. 5.7. Images and error graphs from all subjects and all trials can be found in the supplemental material of the original publication, as well as selected videos of workflow.

**Blocking and Refinement.** Subjects do not fix each parameter value independently and permanently. They make rough adjustments to move the configuration into a good local space and hierarchically refine into smaller and smaller spaces until the precise configuration is reached. This means that parameters are revisited and changed many times during the course of an editing session. Such behavior is universal across all subjects.

**Inability to Configure Image Navigation.** We notice that the majority of the time spent when using image navigation is not spent changing the configuration of the material. Subjects appear to have trouble setting up the 2D navigation array of images. Not only do they have to figure out which axes to look at, they must also determine the scale and granularity of those axes. We observed that most of our subjects were confused by this, despite having preset configurations. Subjects comment “I felt limited by the layout because I could not find the combination I needed to find a match. I was a bit confined by the tools
5.7 Workflow Observations

and felt like I could not control my work as much;” and “[with image navigation] in a way, you know what to change, but not clear how exactly to get there. In the slider approach, that part was a little easier.”

When using either physical or perceptual sliders, subjects made changes far more often. As can be seen in Sec. 5.6, this led to faster and better convergence on a goal. Additionally, not only did image navigation yield changes less often, those changes were undone more often. Undo is used roughly twice as often with image navigation than with the slider interfaces ($p \leq 0.051$). Physical and perceptual sliders share roughly the same undo usage.

**Image Navigation as Sliders.** When using image navigation, almost all subjects displayed behavior of using only one axis at a time, effectively reducing it to a slider interface with 5 discrete configurations visible at a time. While most occurrences of this behavior were interleaved with use of the 2D array or the color pickers, some subjects would go entire trials using only this technique. This leads us to believe that there are many situations where users think in independent parameter space.

**Sliders Equalized by Interactivity.** Universally, subjects rarely snap sliders to a particular value. They almost always drag them to see the material in their workspace change gradually. This suggests that the optimal workflow for novices is to smoothly vary appearance until the image looks like what they are looking for. Doing this seems to be less confusing than seeing several images side-by-side. It also suggests that subjects are not anticipating precisely what value a parameter should be, making many of the perceptual scalings in perceptual parameterizations irrelevant when interactivity is available.

We investigate this behavior by disabling the ability to drag in the slider interfaces, leaving only the option to click a specific location on the slider, and running five additional subjects through the otherwise unchanged study. In this situation, all subjects essentially
5.7 Workflow Observations

mimicked a dragging action by repeatedly clicking at small intervals along the bar. The average number of clicks for physical and perceptual sliders was roughly equal, except on trial 3 where perceptual sliders averaged roughly 1/3 more clicks than physical sliders ($p = 0.034$). We believe that interactivity nullifies the differences between these two interfaces and that novices prefer nudging controls until an image looks right, rather than purposefully setting values.

**Material Properties.** After using each interface, subjects were asked what they thought the most and least difficult aspect of the design process was. This question was open for interpretation, but we did get several comments about specific parameters and properties of the material models. We categorize these comments into the adjustment of color, relative diffuse and specular intensities, and highlight shaping (specular roughness and fresnel effects). The number of times each of these categories were mentioned (sum of all three questionnaires per subject) are listed below:

(a) Color: 23 most difficult, 8 least difficult

(b) Relative intensities: 2 most difficult, 7 least difficult

(c) Highlight shaping: 5 most difficult, 13 least difficult

We draw two pieces of information from this data. First, because color is mentioned most often, users must feel it is an important factor in the overall material appearance. Second, a majority of subjects felt that color was the most difficult part of the design process. This is surprising given that most work in developing perceptual parameterizations of materials has been done in grayscale.

**Exploration.** In open trials subjects perform an exploratory task that requires less fine tuning. Many subjects commented directly that they were exploring in a wide space rather than refining. For example, one subject commented “the open trials had me looking all over the place for cool options, where the matching I tended to make smaller changes.”
5.8 Discussion of Results

Another commented “my workflow was completely random and experimental when doing open trials.”

We observe that the performance of the image navigation interface compared to sliders improves greatly from matching to open trials. Users explain in comments, “I used [image navigation] much like the other ones for matching, but for open trials it was a lot easier to see something good here;” “[with image navigation] matching was very difficult. I had to try many different things. The open trials were enjoyable. I could pick from the options the [preset] buttons brought up;” and “[with image navigation] the open ones were easier because I got a better view of what I wanted.”

We conclude that given its problems with precise adjustment, image navigation must be better at pure exploration, but lacks the needed control for a complete solution to material design. Otherwise, it would not be able to compete so closely with sliders in these open trials.

5.8 Discussion of Results

I now discuss the results of the material design interface study. Strictly speaking, our observations only apply within the boundary of the tested cases, as with all user studies. At the same time, it is our belief that the trends observed in this study should apply to other similar appearance design tasks for novices.

Novices Can Edit Materials. We have found that novices are capable of designing and editing realistic materials. When an interface supports them, novices can perform relatively complex tasks in an efficient way. This suggests that future work on material design interfaces and tools for novices would be fruitful.

Physical and Perceptual Sliders. We found that subjects can perform material edit-
5.8 Discussion of Results

ing equivalently well whether they use physical parameters or the perceptually-inspired parameters provided by our implementation. Additionally, the subject pool is split in half as to which is preferred. We conclude that interactivity is more important than whatever advantages the perceptually-inspired parameters we gave our subjects yield.

**Image Navigation vs. Sliders.** Our most prominent result is the poor performance of the image navigation interface compared to individual parameter adjustment via sliders. This is because image navigation cannot show enough parameter combinations simultaneously due to the limited screen real estate. Perhaps the parameter-based organization used in [61] is not optimal, but if so, the optimal layout remains undiscovered.

**Material Complexity.** We find color to be a significant challenge in material editing. It takes almost twice as long for subjects to match colored materials than grayscale, and the error is significantly higher. Subjects also tell us that color is often the most difficult part of the design process. We believe that there is need for an investigation of methods for perceptual color manipulation of materials under colored lighting. Note that editing color in material editing is very different from setting color in image editing.

We find the difficulty of trials, measured by time to completion and error, to be linear in the number of material parameters given to the users for the slider interfaces. When editing materials with more than one lobe, subjects could accomplish the task given enough time. We found spatially-varying materials to be more challenging than the other types of materials studied, resulting in higher final error. Finally, we discovered no significant difference between editing Ward or Cook-Torrance BRDFs.

**Common Workflow.** Our subjects exhibit common workflow patterns. We notice that subjects generally employ a block-and-refine workflow, moving from large edits to small edits. In slider interfaces, subjects do not set parameters directly, but prefer to smoothly
5.8 Discussion of Results

change them until they look right. This interactivity is important, and reduces the effect of
the parameterization type.

**Exploration.** The advantage of slider interfaces over image navigation is less obvious
in open trials. Taking into account the control problems of image navigation, this implies
that navigation is a better metaphor to support exploration of broad material variations.

**Limitations.** The main limitation of this work is the *scope* of material editing tasks we
investigate. First, we have only studied a subset of possible BRDF models. Second, we did
not explicitly investigate whether novices can effectively pick a material model from a list
of available options. Third, we did not investigate the creation of spatial patterns, although
we believe that this task is well beyond the capability of novice users. Fourth, we forego
the study of interfaces such as painting because of material representation restrictions.

**Future Work.** There are many opportunities for future work in this area. We only
study a small subset of interfaces and models for material design, and the development of
a method to compare interfaces that operate on different material models would be useful.
Long term studies of expert users interacting with material design interfaces on complex
scenes with long rendering times would also be of interest.
Chapter 6

Spatially Varying Material Design

Interface Study

Following the success of the material interface study [45], we began investigating material design that was more complex than parameter tuning of homogeneous reflectance properties. Our next study examined the way that spatially varying materials could be created, meaning that every position on the surface of an object could have differing material properties. We designed and conducted a user study similar to our previous work with the task of editing spatially varying analytic BRDFs.

6.1 Introduction

Material design is the process by which artists define properties of surface materials, such as their color, specular roughness, etc. For opaque materials, the bidirectional reflectance distribution function (BRDF) [62] captures the directionally-varying appearance of real-world surfaces. In our previous study, we examined how parameters of analytic BRDF
models could be adjusted to design reflectance properties of surface materials [45]. In that study, we focused on the adjustment of model parameters on surfaces containing a single BRDF applied over the whole object (with one technical exception). In other words, the materials were homogeneous.

Most materials we encounter in the real world, however, are not homogeneous. Assuming only opaque surfaces, every position on the surface of an object can have a different BRDF. We call such materials spatially varying. While much work has gone into acquiring spatially varying materials from the real world [35, 49, 51, 53], it is still common practice in computer cinematography to define spatially varying materials with photographically-edited or painted parametric texture maps, e.g., those used in the Blender Foundation open movie projects [15, 16, 17]. Designing materials in this way is often an elaborate and time-consuming process.

We believe that the traditional methods of creating parametric maps, unwrapped and
flattened in a program like Photoshop [1] or on-surface in a program like Mudbox [8], forces the artist to solve two very different problems simultaneously. For one problem, the artist must evaluate the reflectance response of the material under various values of the given parameter set. An example of this might be answering the question, how does one create a BRDF that looks like iron and a BRDF that looks like rust? For the other problem, the artist must apply these parameters according to spatial context on the given shape of the geometry. For example, the artist must think about how rust and iron interact with each other and assign these two material types to areas of a model depending on how likely those areas are to be exposed to weather. As such, we believe that we can make the design of spatially varying materials more principled by separating the source (materials) and the assignment (spatial relationships).

To answer this problem, we propose the development of tools for material design that accomplish the same goals as tools like Photoshop do for photo manipulation. In this paper, we provide a user study that demonstrates the utility of an interface that takes spatially varying BRDFs (SVBRDFs) that represent semantically homogeneous substances (rust, metal, wood, concrete) as input, and produces SVBRDFs that represent combinations of these source materials mapped to geometry. In other words, Photoshop allows source images to be arranged and composited in interesting ways to create new and expressive compositions, and we need a similar tool for materials.

In our study, we develop a prototype interface of the proposed editing system (Material Painting), which inherently operates on substances as materials, and compare it to an implementation of traditional painting of parametric maps (Parameter Painting), which operates on images as parameters. From a rendering perspective, both of these interfaces are editing in the same parametric domain, but with different modality and presentation.
6.2 Related Work

Using novice subjects, we show that the interface using the metaphor and mental model of painting with materials outperforms and is preferred to the interface using the parametric mapping metaphor. Examples of our subjects’ work can be found in Fig. 6.1.

Our results indicate that there is a need to develop robust toolsets for the editing of reflectance functions using a palette of reflectance functions, in the spirit of a Photoshop for materials. Even our naive implementation can present the task in a way that novice subjects find more desireable and effective than traditional methods. While our results strictly apply to novices, we believe that this interface metaphor shows promise, and could be extended to produce advanced tools helpful to expert artists.

6.2 Related Work

Here, I review the body of work in material design related to our study. For a review of other user interface and related human studies in computer graphics see Sec. 2.

**Analytic BRDF Models.** Of the various BRDF representations, we are specifically interested in the editing of analytic BRDF models since they give users the ability to define a BRDF using only a small number of parametric maps. We choose to study the editing of BRDFs that consist of a sum of the Oren-Nayar model [65] for diffuse response and a Cook-Torrance model [24] for specular response. We use Cook-Torrance since it has been shown to fit measured data well [60], and Oren-Nayar since it is known to handle a range of diffuse surfaces well, from rough stone to near-Lambertian plastic. To avoid confusing novices with a large number of models, we do not include other BRDF models that are commonly used to represent realistic materials, such as Blinn-Phong, Lafortune, He, Ward and Ashikhmin-Shirley [18, 48, 39, 89, 5].

**Spatial Variation Design.** Designing and editing spatially-varying BRDFs is consider-
6.2 Related Work

ably more difficult than editing single BRDFs. Pellacini et al. proposed a selection system that allows users to select and modify regions of similar appearance in existing spatially varying materials. While these adjustments are useful, we are interested in investigating how spatial variations are constructed in the first place. Kautz et al. developed a system called BTFShop [42], which is built to handle editing of Bidirectional Texture Functions (BTFs), in the same spirit as our proposed interface direction towards painting with materials. While our work is not interested in BTFs, first introduced by Dana et al. in 1999 [26], BTFShop handles many of the operations one might expect from a material painting toolset, such as a form of material blending, masking, contrast adjustments, etc. Our study shows that development in the direction of BTFShop with a focus on light-weight material representation and compositing operations would be useful. We have excluded here literature on texture synthesis, as we are focusing on comparisons between parametric-based and material-based interface paradigms rather than between mapping techniques. However, the synthesis work on BTFs by Zhou et al. should be mentioned here as it includes ways to composite in material features smoothly [95].

Other Interfaces. Much work has gone into designing reflectance properties of BRDFs. Colbert et al. suggest a painting interface for editing BRDF highlights with brush tools [20]. Similarly, Pancowski et al. develop a painting interface for non-photorealistic highlights [66]. Poulin and Fournier introduce the idea of optimizing material parameters to match painted color points on surface geometry [66]. Lawrence et al. use a curve-editing control scheme for editing factored components of nonparametric BRDFs [49]. We exclude these interface types from the study because it is unclear how to extend them to robustly support design of texture variations.
6.3 Study Overview

**Goal.** We seek to evaluate the relative effectiveness of two interface paradigms for material design in the context of designing realistic, spatially varying materials with a focus on novice users. Specifically, (1) we want to measure how efficiently these users can perform specific material *adjustments* and (2) we want to understand which interface paradigms provide better artistic *exploration* of possible material variations.

**Novice Users.** We focus on *novices* with little or no prior knowledge of material design since they make up the majority of users who can take advantage of intuitive interfaces. We would like to have as many people as possible capable of using graphics tools. Some of our subjects are more experienced than others, three subjects rating their experience level 3–4 and five rating 2 or lower on a scale from 1 to 5. However, all subjects are at least no further than the student level, and none have worked professionally as experts.

**Reducing Complexity.** Since material design is a non-trivial process, we require a careful triage between completeness and length. On the one hand, we want to achieve complex-enough material editing tasks to ensure meaningful measurements. On the other, we want to avoid bias in the data by ensuring subjects can successfully complete the required tasks without incurring too much fatigue. Working with novices makes this triage even more necessary. We simplify the material editing task by focusing on applying materials to simple geometry with only a single parametric material model. We simplify implementation of interfaces to ensure that they can be quickly learned while sufficiently complete to capture the main characteristics of each paradigm.

**Materials.** In our design tasks, subjects edit materials represented as a sum of Oren-Nayar [65] and Cook-Torrance [24] BRDFs (Sec. 6.5). We limit the application of this material model to a single diffuse and specular lobe at each pixel of the material maps,
6.3 Study Overview

meaning a uniform number of material parameters at each surface location. We also allow bump-mapping through the application of a height field. Examples of the materials can be found in Fig. 6.2.

**Lighting.** In our study, materials are lit by simple directional lights to keep render costs low and focus on keeping the interfaces interactive with real-time material compositing. We found that controls for lighting added complexity to the interface, and decided to fix lighting for all tasks.

**Geometry and Camera.** The geometry in our images consist of a single plane with a height map. We view this geometry through a fixed orthographic camera positioned directly above the geometry. This geometry and camera setup allows us to avoid the complexity of having subjects rotate and apply materials in three dimensions. Our subjects see the appearance of a three-dimensional object with the simplicity of assigning material as one would apply paint to a canvas. Examples of the geometry can be found in Figures 6.2 and 6.3.

**Interfaces.** We compare two user interfaces: **Parameter Painting** and **Material Painting**. Implementation details of these interfaces are presented in Sec. 6.5. In organizing the material model to one set of parameters per surface location, we limit material design to the application of texture maps representing each of the parameters in the model. **Parameter Painting** uses image-based painting and compositing techniques, similar to Photoshop, to define the material applied to surface geometry. This interface represents the common practice of designing materials parametrically regardless of whether the mapping is done in three or two dimensions, e.g., Mudbox or Photoshop. **Material Painting** uses material-based painting and compositing operations that have no near commercial equivalent. The **Material Painting** interface uses a mental model where the palette is made of materials
rather than pixels, and operations are performed on materials rather than images. In effect, the two interfaces can achieve the identical results (given the right palette), but require the subject to think about the problem differently.

Tasks. We ask subjects to perform two types of material design tasks. During matching trials, they are asked to match the materials on an object under fixed lighting to an image of the same object and lighting with a target material configuration. Matching trials allow us to quantitatively measure users’ performance, while providing a clear goal for subjects who have never experienced material design before. Target goals for matching trials can be seen in Fig. 6.2.

This provides context for the more subjective open trials, where subjects are given a text description that vaguely describes the materials and the subjects are asked to creatively design a material that would look good on a given object and follow the text guidelines. The text given to subjects as goals can be seen in Fig. 6.3. Open trials allow us to observe how users artistically explore the space of possible material configurations, a more natural but harder to measure task.

6.4 Experiment

We ask subjects to complete a number of trials, during which all actions are recorded for further analysis. Each subject performs all trials using both interfaces. These trials vary in the types of materials or texture maps available.

Preparatory Studies. We conducted formal and informal preparatory user studies on 16 additional subjects, the results of which are not included in this paper. Different implementations of the various interfaces were tested to determine a locally optimal set of controls and to remove any implementation errors. The open and matching goals used in
6.4 Experiment

the final study were tested to ensure that time limits were appropriate and that the tasks could be completed.

Figure 6.2: Starting and goal configurations for all matching trials. Time limits are listed below each trial (see Sec. 6.4).

Trials. We perform three matching trials and two open trials. Starting configuration, goal configuration, and time limit for each trial are summarized in Figures 6.2 and 6.3.

For matching trials, we design each one to focus on particular types of edits. The first trial is made of mostly diffuse materials, so the parametric complexity is lower than in other trials. The second trial requires more complicated compositing, with less obvious masking and blending requirements. The third trial involves metal and rust, so both diffuse and specular components are important.
**6.4 Experiment**

![Image of Armor and Stone Wall](image)

**Figure 6.3:** Starting and goal configurations for all open trials. Time limits are listed below each trial (see Sec. 6.4).

For open trials, we choose objects that imply different types of materials, but leave room for variations. The first open trial looks like a suit of armor, so it makes sense that the text asks for metal and leather textures. The second open trial looks like a stone wall, so it makes sense that one would use stone and moss.

For all trials, the subject is given 11 materials or equivalent parameter maps (see Sec. 6.5). The materials and maps given correspond to the particular trial the subject is working on. For example, subjects get five metal materials, 5 rust materials, and a constant material for...
6.4 Experiment

the metal plate in trial 3.

The same initial and goal material configurations are used for all subjects and all interfaces. Each trial has a fixed time limit, and subjects can end the trial sooner if satisfied with the result at that time. At the end of each matching trial, subjects rate the accuracy of the matching on a scale of 1 to 5. For open trials, subjects use the same scale to rate how satisfied they are with their result.

**Questionnaire.** After performing all trials with both interfaces, subjects complete a questionnaire where they rate each interface on a scale of 1 to 5 in the following categories: (1) natural way to think about material editing, (2) preference in matching trials, (3) preference in open trials, and (4) overall preference. In case of equal ratings, subjects also strictly rank both interfaces in each of these categories. Immediately after finishing trials for each single interface, subjects are asked to leave free-form comments on aspects of each interface.

**Procedure and Setup.** Eight subjects participate in the study. All subjects have normal or corrected-to-normal vision. Each subject edits materials for about three hours to give significance of our tests, while keeping fatigue low.

Subjects complete the study in two sessions lasting around 100 minutes each, one session for each interface. We randomize the order in which the interfaces are presented for each subject. Before each session, subjects complete a training phase to become familiar with the specific interface. We train each subject individually to allow questions, accommodating each subject’s learning needs. The instructor verifies that the subject uses each part of the interface, and answers the subjects’ questions. Once trials begin, all user interface actions are recorded.

The study is conducted in a controlled lighting environment with negligible ambient
6.5 Interface Implementation

lighting, to simulate typical working conditions of artists and maximize visibility of the screen. We use a 24-inch Dell 2407WFPb LCD display at 1900 × 1200 resolution at a distance of approximately 1 foot from the subject. All rendered images are 512×512 pixels on screen covering an area 5.3×5.3 inches. We used an Intel 3.0 GHz Core2 Extreme X9650 PC with 4 GB of RAM, four processor cores, and an NVidia GeForce GTX 580 graphics card. The interfaces can be controlled fully with either a mouse and keyboard or a pen and tablet. We use a Wacom Intuos 3 graphics tablet with an 8 × 6 inch drawing surface.

6.5 Interface Implementation

In this section we discuss our implementations of the user interfaces included in the study. We intend to publish video detailing all interface features.

**Parametric BRDF Model.** We parameterize the Oren-Nayar BRDF $\rho_{on}$ as

$$
\rho_{on} = \frac{\rho_d}{\pi} (A + B \cdot \max(0, \cos(\phi_i - \phi_o)) \cdot \sin \alpha \tan \beta)
$$

(6.1)

with

$$
A = 1 - \frac{0.5\sigma^2}{\sigma^2 + 0.33}
$$

$$
B = \frac{0.45\sigma^2}{\sigma^2 + 0.09}
$$

$$
\alpha = \max(\theta_i, \theta_o)
$$

$$
\beta = \min(\theta_i, \theta_o)
$$

where $\rho_d$ is the diffuse albedo, $\sigma$ is the surface roughness, $\theta_i$ and $\theta_o$ are the angles between the surface normal and the incoming and outgoing direction respectively, and $\phi_i$ and $\phi_o$ are
the azimuth spherical coordinates for the incoming and outgoing direction respectively in the local frame centered at the surface normal.

We parameterize the Cook-Torrance BRDF $\rho_{ct}$ as

$$\rho_{ct} = \frac{\rho_s}{\pi} \frac{DGF}{\cos \theta_i \cos \theta_o}$$  \hspace{1cm} (6.2)

with

$$F = F_0 + (1 - F_0)(1 - \cos \theta_b)^5$$

$$D = \frac{e^{-(\tan \theta_h/\sigma)^2}}{\sigma^2 \cos^4 \theta_h}$$

$$G = \min \left( 1, \frac{2 \cos \theta_h \cos \theta_i}{\cos \theta_b}, \frac{2 \cos \theta_h \cos \theta_o}{\cos \theta_b} \right)$$

where $\rho_s$ is the energy of the specular component, $\sigma$ is the surface roughness, $F_0$ is the Fresnel reflectance for a direction orthogonal to the surface, and $\theta_b$ is the angle between the outgoing and half-vector direction.

The entire model is expressed as the sum $\rho_{on} + \rho_{ct}$. We allow the user to perturb surface normals with a bump map represented by a grayscale height field. Thus, we provide four parameters that can be mapped to surfaces in our system: diffuse $\rho_d$ (RGB), specular $\rho_s$ (RGB), roughness $\sigma$ (float), and height (float). The linear floating point parameters are represented visually in grayscale. For simplicity, we use the same roughness parameter for both Oren-Nayar and Cook-Torrance. The material assigned to any surface in our system can have only a single value for each parameter at each surface location.

**Shared Interface Features.** This interface is designed to present a simplified toolset capable of producing experimental data, while providing tasks that are representative of real material design tasks. These implementations are not designed to allow every feature an expert designer might want, nor the quality of content (material libraries, masks, pho-
6.5 Interface Implementation

![Diagram of Parameter Painting and Material Painting interfaces](image)

**Figure 6.4:** Layout for both interfaces. On the left for the Parameter Painting interface, A: work space, B: goal, C: map library, D: stencil mask library, E: selection mask library, F: diffuse layers, G: specular layers, H: roughness layers, I: height layers, J: scale slider. On the right for the Material Painting interface, A: work space, B: goal, C: material library, D: stencil mask library, E: selection mask library, F: material layers, G: scale sliders.

An expert designer might have available in a production environment. We want to design interfaces that control very similarly except in the key areas that differentiate the two interface paradigms. If these paradigms were fully realized in implementation, the type of work produced might diverge greatly. In other words, our experiment operates in a controlled domain.

Both **Material Painting** and **Parameter Painting** interfaces use the same controls. We use a layering system similar to Photoshop [1], in which each layer has alpha transparency and a blending mode for compositing. Each layer has one or more scale values that can be adjusted (explained below). The user can view a layer by itself at any time, where unpainted parts of the layer appear black. Layers can be copied and pasted for reordering or as foundations for new layers. We fix the number of layers available for simplicity.

To paint in a layer we use a stenciling system somewhat similar to Mudbox [8] in which the user controls a stencil object that can be oriented over the surface. In our system, pixels or materials can only be applied to the surface if they are copied from the stencil, as if it
6.5 Interface Implementation

![Image](image.png)

**Figure 6.5:** Example of using the stencil for both interfaces. *Top:* painting a rust diffuse map. *Bottom:* painting a rust material.

were a stamp. We also provide stencil masks that allow the user to change the shape of the stencil. In other words, the stencil holds material data (whether it be image or full material data), which can be applied to geometry when painting over the stencil with a brush. The shape of the stencil can be changed by adding a stencil mask, which makes areas of the stencil inactive. The position, orientation, and scale of the stencil can be modified by the user. When the stencil is modified, both the mask and stencil content (material/map) are transformed. We couple shape and content for simplicity. Figure 6.5 shows an example of painting with the stencil for both interfaces.

The paint brush is always a circular shape with fixed gaussian falloff. The user can paint, erase, or resize the brush. Both painting and erasing can only occur over an area where the stencil is active. This allows the painting of specific masking patterns, as well as the removal of layer content with specific masking patterns. The user can undo brush strokes, but only the most recent, as to enable correction of mistakes but avoid back tracking.
6.5 Interface Implementation

All interfaces use the same screen layout. The center area consists of a workspace window, a goal window, and buttons for ending the trial (Fig. 6.4). The workspace is where painting is carried out. The left area consists of a palette/library of materials or maps that can be painted with, a library of stencil masks that can change the shape of the stencil, and a library of selection masks for the particular geometry given. Selection masks make regions of the surface active or inactive so that features of the geometry can be easily isolated for painting. Above the three libraries of thumbnail images are larger previews of the selected material/map, stencil mask, and selection mask. The right area consists of layer thumbnails, sliders to scale layer contributions, and drop down boxes to change blending modes (and viewing modes for Parameter Painting interface). Layers are represented by thumbnails that update to represent what has been painted, similar to Photoshop.

**Parameter Painting Interface.** In the Parameter Painting interface, the user paints down image data through the stencil, which can be applied to only one layer of one parameter at a time. There are four sets of layers, one for each parameter in our material model. Each parameter is composited from four layers. Layers can be copied from one parameter and pasted into a differing parameter, but colored layers will automatically be converted to grayscale if being copied into a floating point parameter.

Each layer for each parameter has an independent scaling factor that multiplies the contribution of that layer. In the resulting material this scales energy for diffuse and specular, sharpness of highlights for roughness, and bumpiness and thickness for height.

Each layer can have one of two blending modes. Replace mode replaces the values of pixels painted in the composite of layers below it with pixels from the current layer according to its alpha transparency. Add mode adds the value of pixels painted in the current layer to the pixels in the composite of the layers below it.
6.5 Interface Implementation

At any time the user can switch between seeing the material resulting from the composites of all parameter layers (material result view) and seeing the flat image composite of the currently selected parameter (map view). When a layer is viewed by itself, it is always viewed as a flat image map.

**Material Painting Interface.** In the *Material Painting* interface, the user paints down full material data through the stencil, which includes data for all parameters of the material model. Where the *Parameter Painting* interface has four sets of layers, *Material Painting* has only one set consisting of four layers that get composited together to form the final material. This interface has the user think about the materials as actual substances that get applied one on top of the other.

Each layer has four scaling factors, one for each parameter of the material model. This allows the layer to be adjusted in a global way. The scaling operation is identical to what is applied in the *Parameter Painting* interface. The only difference between the two interfaces’ scaling adjustments is that the local spatial properties of the material parameters are predefined.

Each layer can have one of four blending modes. When layers are composited or materials are blended between layers, we simply linearly interpolate each of the four parameters. This is a naive stand-in approach designed as a proof of concept for the interface paradigm. *Replace* mode replaces the values of materials painted in the composite of layers below it with pixels from the current layer according to its alpha transparency. *Over and raise* mode adds the height parameter of materials painted in the current layer to the height of materials in the composite of layers below it, and operates the same as *Replace* mode for the remaining three parameters. This is used to represent adding or growing a substance over another. *Over and smooth* mode averages the height parameter of materials painted
in the current layer with the height of materials in the composite of layers below it, and
operates the same as Replace mode for the remaining three parameters. This is used to
represent aging, erosion, or the application of a goopy substance over another that might
reduce the bumpiness of the base surface. Add mode adds the value of the diffuse, specu-
lar, and height parameters of the materials in the current layer to the diffuse, specular, and
height parameters respectively of the materials in the composite of the layers below it, and
operates the same as Replace mode for the roughness parameter. Add mode is designed to
combine or replace parameters from one material to another. For example, you could give
stone a metallic sheen or wood a clear coat of gloss from a constant flat material. We do
not add roughness because it would not allow the same effect. A general approach would
require more blending modes, which we believe would add unnecessary complexity.

Interface Balance. The Parameter Painting interface is intrinsically more powerful
than the Material Painting interface, since any pixel value can be assigned to any param-
eter at any location, but using this power at its full potential would require us to provide
tools outside the scope of this study, e.g., implement and teach an equal to Photoshop and
Mudbox combined. Instead, we design the Parameter Painting interface to be capable of
producing anything the Material Painting interface can produce.

We strictly use the Parameter Painting interface to construct all materials available in
the Material Painting interface. Then for each material provided in the Material Painting
interface for a particular trial, we provide all component maps to our subjects in the Pa-
rameter Painting interface. This way, only search, alignment, and scaling are necessary
to make the Parameter Painting interface equivalent to Material Painting. To make things
even easier, we provide only eleven materials in each trial, one of which is a flat material
with a constant gray map assigned to each parameter. In the Parameter Painting interface,
6.6 Analysis

we provide an additional map to the stencil library that is all black, the equivalent of zero, in case the user would like to completely eliminate a parameter.

Note that when we compare the two interface paradigms, we have greatly simplified the Parameter Painting interface by providing all necessary maps. We have essentially limited the task to having to understand placement and parameter relationships. In a real design situation, all of these maps would have to be created. While this is true for the Material Painting interface’s materials as well, the idea behind Material Painting is that some other artist has done this work at a different stage of the pipeline. Thus, we are simply testing the difference between the mental model used in each interface.

Performance. We implement both rendering and compositing on the GPU. The system runs in real-time for both interfaces.

6.6 Analysis

We present an analysis of our data based on the output of the rendering system as subjects proceed through each trial as well as feedback from the subjects in the form of ratings and comments. Unless stated otherwise, tests for statistical significance are computed with repeated measures analysis of variance (ANOVA) [83]. This handles within-subject factors that create correlations which invalidate the assumption of independence in standard one-way ANOVA. A \( p \) value below 0.1 indicates a 90\% confidence that the two population means differ given the measure of the sample. In all figures, error bars represent standard error.

Time to Completion. We investigate the work speed of users with each interface. Generally, subjects are able to complete each trial within the allotted time limit with one or more interfaces. In Fig. 6.6, we show the mean time to completion for each matching and
6.6 Analysis

![Bar chart of Time to Completion and Final Error](image)

**Figure 6.6:** Left: average time to completion for all trials over all subjects (in seconds). Right: average final error for matching trials over all subjects.

open trial over all subjects.

Time to completion for both interfaces is fairly close on average, with the exception of trials 1 ($p = 0.115$) and 4 ($p = 0.229$), the first matching and open trial respectively. In those two trials, subjects are faster with the Material Painting interface, though with relatively poor significance. We believe that it takes longer for subjects to adjust to using the Parameter Painting interface than Material Painting, but that our simplification of Parameter Painting allows subjects to finish in a comparable amount of time to the Material Painting interface.

**Matching Error.** To evaluate user performance in matching trials, we compute the error between the subject’s materials and the goal materials using an image-based difference metric. Since camera and lighting are fixed, it makes more sense to define error based on what the subject could see. We use the $L_1$ error between the subject’s image and the goal image.

Fig. 6.7 shows the error over time for one subject performing the same trial with all
Figure 6.7: Top: *example graphs of error over time (in seconds) for the individual trials shown in Fig. 6.1. Bottom: illustration of the error over time (in seconds) for matching trials averaged over all subjects.*

interfaces. When subjects are successful, error decreases toward the correct solution, converging on some low error value. This convergence is not monotonic, because users explore the configuration space in order to reach the desired goal. In the supplemental material for the paper submission we will include error graphs for all subjects on all trials together with rendered images of their material configurations at fixed time intervals.

To summarize the overall performance of each interface we analyse the final image error for each matching trial averaged over all subjects (Fig. 6.6). On average, the *Material Painting* interface outperforms *Parameter Painting* in terms of error for trials 1 \((p = 0.114)\) and 3 \((p = 0.012)\) (stone pillar and metal plate), but both interfaces perform at the same error for trial 2 (wood panels). We believe that trial 2 differs because subjects had trouble getting the white paint to look correct using either interface. When using the *Parameter*
6.6 Analysis

![Rating Chart]

**Figure 6.8:** Average of subjective image quality ratings over all subjects.

Painting interface, we find that subjects typically match diffuse features well, but do not match the other parameters as easily. Because the height and specular properties of trial 2 are subtle where wood is shown and strong where white paint is present, we believe the subjects are able to get away with mostly diffuse using the Parameter Painting interface, since both interfaces produced poor output for the areas with white paint.

Convergence. To illustrate the convergence behavior of different interfaces we average the image error across all subjects over time in Fig. 6.7. This average is not statistically valid, but we find that it gives a revealing visual summary of overall behavior. As can be seen in the graphs, Material Painting tends to converge faster and with less variation than Parameter Painting in trials 1 and 3. We also see that both interfaces perform similarly on trial 2.

Subjective Image Quality. At the end of each trial, subjects rate their work on a scale of 1 to 5, with 1 being the worst and 5 being the best. Matching trials are rated in terms of how close the workspace and goal images match. Open trials are rated in terms of how satisfied the subject is with their result. Fig. 6.8 shows the average ratings for each trial. This subjective image quality correlates, though somewhat weakly, with the computed error of the final image with a linear correlation coefficient of $-0.35$. 

104
6.6 Analysis

Figure 6.9: Left: Average interface ratings from questionnaire over all subjects. Rating 5 implies best. Right: Sum of interface rankings over all subjects. Rank 1 implies best.

Subjects on average rate their work better when using Material Painting than with Parameter Painting on all trials, but this difference is only reasonably significant for trials 1 ($p = 0.038$) and 4 ($p = 0.104$). Not only do subjects perform objectively better using slider interfaces compared to image navigation as measured by error, they perceive themselves as doing better as well.

**Interface Rankings and Ratings.** Subjects rate and rank both interfaces in 4 categories where ratings can have ties, but rankings are forced choice (see Sec. 6.4). Average ratings and stacked frequencies of rankings are shown in Fig. 6.9. For evaluating statistical significance of ranks we use the Friedman test [32], a nonparametric test that takes into account within-subject effects. A low $p$-value indicates high confidence that subjects have made a significant distinction between two interfaces.

In all categories, Material Painting outranks Parameter Painting ($p \leq 0.034$). In fact, all subjects rank Material Painting higher in the categories of natural, preference for matching trials, and overall preference. Only one subject ranked Material Painting lower than Parameter Painting for the remaining category of preference for open trials.

We find similar trends in the interface ratings. In all categories except preference for
open trials, subjects rate *Material Painting* much higher than *Parameter Painting* \( (p \leq 0.000) \), with *Material Painting* still rated higher than *Parameter Painting* in preference for open trials by a lesser amount \( (p = 0.049) \).

When asked why they chose *Material Painting* for their overall preference over *Parameter Painting*, it seems that *Material Painting* matches their conception of the problem better. One subject states “material painting seemed like how I think about objects in the physical world.” Another commented “the fact that all parameters come with the material is easy to visualize. It is easier to think of the material as an object and vary its parameters on one layer rather than having the parameters scattered across columns.” Another writes “I liked the simplicity and intuitiveness of material painting.”

**Open Trials.** Subjects are given open trials to expose them to the two interface paradigms under a different context than matching. We believe that the subjects feel more free to experiment with the parametric maps in the case of open trials, and do not require the same level of control. In open trials, they still prefer the *Material Painting* interface, but don’t rate and rank the *Parameter Painting* interface as poorly as in the matching context.

### 6.7 Discussion of Results

I now discuss the results of the spatially varying material design interface study. Strictly speaking, our observations only apply within the boundary of the tested cases, as with all user studies. At the same time, it is our belief that the trends observed in this study should apply to other similar appearance design tasks for novices.

**Novices and Spatially Varying Materials.** We have found that novices find the task of creating spatially varying materials difficult. To have them produce quality output, we had to greatly limit the power of the given interface implementations. However, when an
6.7 Discussion of Results

interface supports them, novices can perform these relatively complex tasks in an efficient way. This suggests that future work on material design interfaces and tools for novices would be fruitful.

**Painting with Materials.** We found that subjects prefer using our *Material Painting* interface even in the prototype phase and being compared to a vastly simplified *Parameter Painting* interface. We conclude that an interface for designing spatially varying materials that presents itself as operating on materials rather than on parameters can be preferable to traditional parametric approaches.

**Creating Spatially Varying Materials.** We believe that the traditional methods of creating parametric maps, unwrapped and flattened in a program like Photoshop forces the artist to solve problems of material properties and spatial relationships simultaneously, and that these problems may be better approached separately. For the problem of material properties, the artist must evaluate the reflectance response of the material under various values of the given parameter set. For the problem of spatial relationships, the artist must apply these parameters according to spatial context on the given shape of the geometry. We believe that our study shows that we can make the design of spatially varying materials more principled by separating the source (materials) and the assignment (spatial relationships).

**Limitations.** The main limitation of this work is the scope of material editing tasks we investigate. First, we have only studied a subset of possible BRDF models. Second, we did not allow subjects to design these materials in the presence of different lighting and view configurations, as might be the case for animation applications. Third, we did not investigate the actual construction of base maps, which is a long and labor-intensive task.

**Future Work.** There are many opportunities for future work in this area. We only study a small subset of contexts and models for material design. Long term studies of expert
users interacting with material design interfaces on complex scenes with long rendering times would also be of interest. Now that we have shown that Material Painting interfaces are desireable with a simple naive implementation, it follows to develop Material Painting tools that solve difficult problems like material-based compositing, masking, and blending.
Chapter 7

Conclusions

Our lighting study from Chapter 3 demonstrated that painting interfaces might be the wrong way of approaching the problem of lighting design. This was a major result given the amount of research that focused on painting interfaces without validation. Our study was the first time interface paradigms for appearance design had been examined in the form of comparative user study in the context of design tasks. We were able to extract useful observations on how novice users approach the task. The results of this work were then used to determine a control scheme for our work in lighting interfaces that allow light to travel nonlinearly (Ch. 4).

The material study from Chapter 5 built upon the methodology we established for the lighting study. We showed that adjusting parameters of analytic BRDF models was very similar whether the models were parameterized based on physical surface properties or perceptual visual properties. We also showed that image navigation interfaces built to search the space of possible BRDF parameters more effectively perform poorly compared to simple, slider-based interfaces. We also drew several conclusions about the nature of the complexity of the task, and suggested further investigation into perceptual parameteriza-
Conclusions

tions of the colored components of BRDF models.

Finally, we conducted a study to investigate how users approach the problem of designing spatially-varying materials, a problem far more complex than the problem investigated in our previous material study. We were able to show that tools designed to paint with materials could outperform traditional parametric methods, which necessitates further exploration into the pursuit of a “Photoshop for materials.” Our study was a first step, using only a naive prototype implementation of such an interface, but demonstrated that it approached the problem in a preferrable way.
Appendix

*BendyLights* Study

Pairs of Images Used in *BendyLights* Study of Viewer Tolerance

Overview

- There are 11 pairs of images total (P1-P11).

- Subjects are shown these 11 pairs, then the same 11 pairs mirrored horizontally.

- Subjects are asked if the first, the second, or both images contain physically correct lighting.

- P1-P6 contain two camera position variants. P1 is a smaller version of the bend in P2, P3 of P4, and P5 of P6. For this reason, subjects do not view these pairs back to back with the same camera position. When a subject views an image sequence, the camera angle for the smaller bend is determined randomly, and the larger bend is shown from the differing camera angle.
Conclusions

Linear Light

BendyLight

P1 (camera 1)

P1 (camera 2)

P2 (camera 1)

P2 (camera 2)
Conclusions

Linear Light

BendyLight

P5 (camera 1)

P5 (camera 2)

P6 (camera 1)

P6 (camera 2)
Conclusions

Linear Light          BendyLight

P7

P8

P9

P10

P11
Bibliography


[27] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography.


