Volume 0 (1981), Number 0 pp. 1–19 COMPUTER GRAPHICS *forum*

State of the Art in Artistic Editing of Appearance, Lighting, and Material

Thorsten-Walther Schmidt[1†] Fabio Pellacini[2] Derek Nowrouzezahrai[3] Wojciech Jarosz[4,5] Carsten Dachsbacher[1‡]

[1]Karlsruhe Institute of Technology [2]Sapienza University of Rome [3]Université de Montréal [4]Disney Research Zürich [5]Dartmouth College

[†]thorsten.schmidt@kit.edu [‡]dachsbacher@kit.edu

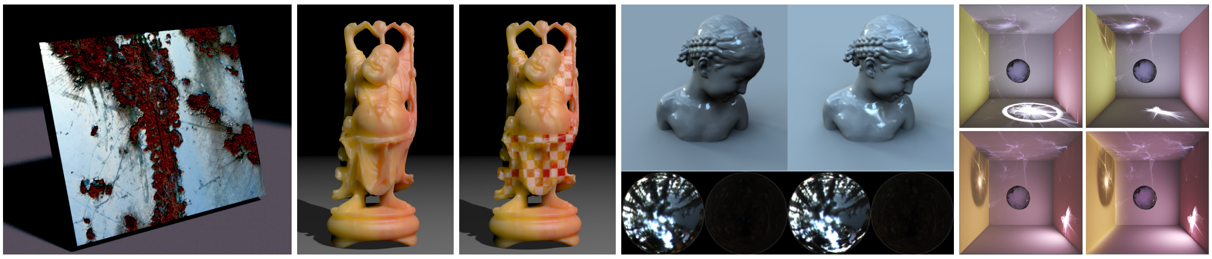


Figure 1: *Examples of artistic appearance editing (left to right): A rendering with spatially-varying reflectance and normals computed from a single image and a few input user strokes [DTPG11]; editing heterogeneous subsurface scattering acquired from a real-world material sample [STPP09]; adjusting natural environment lighting to obtain a desired shadowing and reflection appearance [Pel10]; direct manipulation of caustics and shadows with global illumination [SNM* 13]. (Images taken from [DTPG11, STPP09, Pel10, SNM* 13].)*

Abstract

Mimicking the appearance of the real world is a longstanding goal of computer graphics, with several important applications in the feature-film, architecture and medical industries. Images with well-designed shading are an important tool for conveying information about the world, be it the shape and function of a CAD model, or the mood of a movie sequence. However, authoring this content is often a tedious task, even if undertaken by groups of highly-trained and experienced artists. Unsurprisingly, numerous methods to facilitate and accelerate this appearance editing task have been proposed, enabling the editing of scene objects' appearances, lighting, and materials, as well as entailing the introduction of new interaction paradigms and specialized preview rendering techniques. In this review we provide a comprehensive survey of artistic appearance, lighting, and material editing approaches. We organize this complex and active research area in a structure tailored to academic researchers, graduate students, and industry professionals alike. In addition to editing approaches, we discuss how user interaction paradigms and rendering backends combine to form usable systems for appearance editing. We conclude with a discussion of open problems and challenges to motivate and guide future research.

Keywords: *Artistic editing, appearance editing, material design, lighting design*

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism— I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

---

1. Introduction

Synthesizing realistic images is among the longstanding goals of computer graphics, and its ambitious nature is evidenced by the advancements of our field towards realism with still a significant number of open problems. The acquisition and editing of detailed geometry, its animation, the careful modeling and reproduction of real-world material and lighting profiles, and the efficient simulation of physically accurate light transport are still in need of robust so-

Published by Blackwell Publishing, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main Street, Malden, MA 02148, USA.

lutions. But, as our field progresses, so do its goals: while realistic image synthesis remains an important challenge, so too does the ability to *design* a (potentially realistic) image that conveys an explicit mood or information to the viewer.

One of the aspects at the core of scene design is defining the appearance of objects, which comes from the interaction of surface materials and scene lighting. Appearance design is the process by which artists edit material and lighting properties in order to achieve a desired look. In general, this is a complex and laborious process, since artists are manually solving an underconstrained inverse problem: given a desired appearance, determine the material and light settings to achieve it. In fact, even for simple scenes and highly-trained digital artists, appearance design may take several hours. Furthermore, in cases where the design goals cannot be obtained in the confines of physically accurate simulation models, more flexible artistically motivated models need to be developed. Many different approaches, ranging from physically based to purely artistic, have been proposed to intuitively edit the appearance of individual objects as well as entire scenes.

In this report we present a summary of the state of the art in artistic editing of lighting and material that includes the following topics:

- *lighting design*: the editing of lighting parameters to define a final scene appearance, which is fundamental to computer cinematography;
- *material design*: the definition of the reflectance properties of a surface or the scattering properties of materials, ranging from whole surface changes to precise adjustment in textured regions;
- *whole appearance design*: the coupled editing of the interaction between surface materials and scene lighting, when it may be difficult to segment and treat separately;
- *preview rendering*: rendering systems tailored to adapt to the more flexible demands of an appearance editing framework.

We organize prior work along two axes, defining first *what* is edited or manipulated, and second *how* these elements are edited, including the interaction paradigms they rely on. We also provide an overview guide to the methods covered in this report, providing a quick way to assess their usefulness for different practical scenarios (see Tab. 1).

2. What is Appearance Design?

The *appearance* of an image depends on complex local and global interactions of light in a virtual scene. Light emitted from light sources travels in the scene, and is subsequently reflected, transmitted or absorbed locally at the surfaces of the objects, until it finally reaches an image sensor. When participating media are present, light can also be emitted, scattered, and absorbed in the volume surrounding surfaces. This combination of global transport and local interactions repeats indefinitely until light reaches a state of equilibrium.

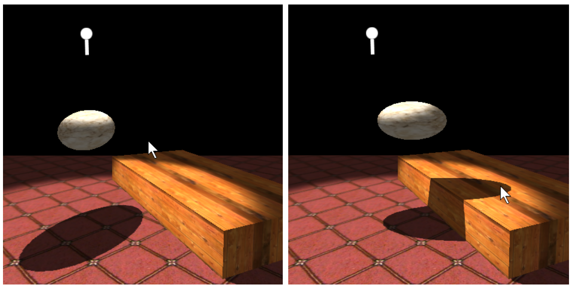


Figure 2: *Using Pellacini et al.'s interface for interactive cinematic shadow design [PTG02], users can indirectly and automatically reposition shadow-casting objects by dragging their shadow. This is an example how appearance design methods can assist the user in modifying the appearance parameters (e.g. the relative object positions) in order to achieve a desired final appearance (e.g. the shape and placement of the shadow). (Images taken from [PTG02].)*

Given this light transport process, it is clear that both the initial *lighting* emitted from sources, as well as the local *material* interactions, play a significant role in the final appearance of a scene. As such, modifying the initial *lighting* state and/or the local *material* reflectance behaviors is a simple way to affect both the local and global appearance of the final image.

Appearance design is a fundamental task at the tail end of digital content creation: given objects' surfaces and their relative placement in space and time, the goal of appearance design is to define the look of the final images that meets specific stylistic or artistic requirements. In general, the final image appearance relies on several controllable *appearance parameters*:

- the position, orientation, and emission profiles of light sources, ranging from simple point sources to realistic area and environment illumination;
- the camera parameters, including position, framing, aperture, lens model, shutter time, etc.;
- the materials that define the potentially spatially-varying shading response (e.g. via BRDFs, shaders, node-based networks, etc.) of each object;
- the light transport simulation algorithm and its settings.

Final images are computed by solving the *rendering equation* [Kaj86], which specifies the appearance of a point x by computing the radiance $L(c \leftarrow x)$ towards a viewer at point

c as:

$$L(c \leftarrow x) = \underbrace{L_e(c \leftarrow x)}_{\text{lights \& camera}} + \int_{\mathcal{S}} \underbrace{f_r(c \leftarrow x \leftarrow y)}_{\text{materials \& camera}} \underbrace{L(x \leftarrow y)}_{\text{transport}} G(x \leftrightarrow y)\, \mathrm{d}y, \tag{1}$$

where $L_e(c \leftarrow x)$ is the radiance emitted from light sources and $f_r(c \leftarrow x \leftarrow y)$ is the bidirectional reflectance distribution function (BRDF), that captures how the material at x reflects incident radiance from another point y towards c. $G(x \leftrightarrow y)$ is the so-called geometry term which accounts for the mutual visibility as well as the distance and orientation of x and y. Note that the integral is over all surfaces $\mathcal{S}$ in a scene from which light may arrive at x. We could equivalently express this integral over the space of incident unit directions about x or the multi-dimensional space of light paths in the scene [Vea98]. In the equation above we ignore volume scattering governed by the more complex *radiative transfer equation* [Cha60].

Note that the recursive definition of appearance (with L inside the integrand) means that the appearance of any point is not only the result of material-light interactions, but may also depend recursively on the appearance of all other points in a scene. Indeed, we note that the *appearance parameters* affect each term in this image formation equation. In scenes of even moderate complexity, predicting the behavior of the final appearance as any of these parameters are directly edited quickly becomes intractable for even the most skilled and experienced artists.

There have been efforts to catalog the appearance of highly diverse objects from photographs with the aid of crowdsourcing, for applications such as surface retexturing and material and image browsing [BUSB13]. From the point of view of appearance design, this can be seen as a useful database for retrieving appearances of already-existing real-world objects as a source of inspiration, but the key responsibility of actually selecting and editing (i.e. *designing*) the appearance of a specific scene remains on the artists.

In our discussion, an *appearance design approach* is a semi-automatic process for editing the final appearance of an image or animation sequence that abstracts the task of determining suitable settings of the lighting and/or material settings in a scene. Specifically, any such approach will take some higher-level input specification of the appearance edits desired by the user, and then automatically computes the lighting (Sect. 4) or material (Sect. 5) settings, or both, in order to best meet the user's requests.

2.1. Challenges & Complexity of Appearance Design

Appearance design tools inherently deal with different rendering challenges than standard rendering. In a typical renderer used for generating animations, mostly the camera, geometry, and, to some extent, lighting change, while the appearance of materials remains mostly static during a shot. Furthermore, though lighting and material may change, they have a predefined evolution during a shot. This is fundamentally different from the need to dynamically explore the entire parameter space during appearance design.

Historically, the strategy to bridge this gap has been to perform some precomputation which is then cached using more flexible intermediate representations. Typically, the system first enforces certain constraints, e.g. fixed camera, fixed lighting, or fixed materials, and caches the possible space of parameters for the remaining free variables. The choice of what is cached and its representation varies significantly across the proposed techniques, and is also highly dependent on the provided editing functionality.

Relighting systems' primary function is to provide interactive editing of the lighting (while typically keeping the scene and materials static). Early examples include parameterized ray tracing [NSD94], ray trees [SS89, BP96], and the G-Buffer approach [ST90, GH00]. The Lpics [PVL*05] and Lightspeed [RKKS*07] systems also fall within this category. Direct-to-Indirect transfer techniques [HPB06, LZT*08] exploit our ability to compute direct lighting efficiently and leverage a possible precomputation to extend this to indirect illumination. Most of these methods gain efficiency by exploiting the linearity of light transport and they often capitalize on the assumption that camera movement occurs much less frequently than shading changes.

Although it may initially seem conceptually similar, material editing is inherently different than relighting. In contrast to relighting, BRDF editing is fundamentally nonlinear when global illumination is considered. In particular, editing n BRDFs in a scene with d light bounces leads to an n-variable polynomial of degree d [BAERD08]. Unfortunately, representing this explicitly is only practical for a small number of bounces. Several researchers have investigated this problem for both surface BRDFs [BAOR06, SZC*07, BAERD08], and more recently for editing participating media parameters [STPP09, HR13, ZHRB13].

Relighting, and to some extent material editing, systems have exploited a vast set of techniques developed in the precomputed radiance transfer literature [SKS02, KSS02, NRH04, NRH03, WTL04, KAMJ05, SM06, AUW07, CPWAP08, Ram09]. These techniques typically exploit the linearity of light transport and the fact that light (transport) is often sparse in a suitably chosen basis space (e.g. frequency or wavelet domain). In return for the efficiency gained through precomputation, these methods typically restrict the lighting (e.g. environment only), or material properties (e.g. diffuse only).

Although PRT techniques can provide interactive feedback when editing a specific set of parameters, once the parameter set changes, a new, expensive precomputation must be performed. For interactive design, this can lead to slow

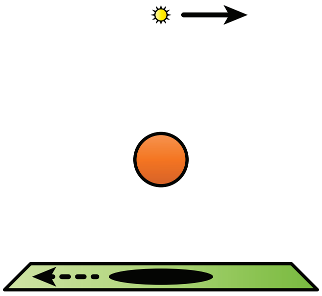


Figure 3: *Direct vs. indirect manipulation of a point light source. With direct manipulation, the user moves the light source (solid arrow) and the shadow follows accordingly (dashed arrow).* Direct interfaces *are trivial to implement but oftentimes unintuitive. With indirect/goal-based manipulation, the user moves the shadow (dashed arrow) and the system solves for the new light source position (solid arrow).*

interaction times, for instance, a level designer for a game must wait for an overnight simulation to see interactive lighting changes when the scene geometry is modified. The recent Modular Radiance Transfer [LAM*11] approach addresses this challenge by trying to decouple the precomputation from the scene.

3. Interaction Paradigms

Artistic editing should never be isolated from user interaction, and so we categorize according to three interaction paradigms [KP09, KPD10] (cf. Figs. 3 and 4):

- With *direct* interfaces, artists directly edit light/material parameters, such as positions and surface colors. This is the most commonly available interface in commercial software. While easy to implement, direct interfaces are neither efficient nor intuitive, since final appearance often depends unpredictably on these parameters.
- *Indirect* interfaces let users specify appearance qualifiers, e.g. shadow positions or material contrasts, and the system computes the necessary rendering parameters.
- *Goal-based* interfaces allow artists to define the rendered colors directly, for example by painting, while the system solves a complex and typically non-linear optimization to determine the rendering parameters.

The *effectiveness* of user interaction with these paradigms was investigated by Kerr et al. [KP09, KPD10], as well as the *selective applications* of edits for complex materials and lighting [PL07, AP08, Pel10].

4. Lighting Design

Lighting design focuses on modifying the parameters of lighting models under fixed geometry and material conditions. These models can be categorized by the complexity of effects they support, namely direct illumination and shadows from point and directional lights [PF92, PTG02], area- and image-based (direct) illumination [Pel10, OPP10], and (full) global illumination including diffuse interreflections and caustics [RTD*10, SNM*13]. Finally, some systems [NJS*11, HR13, KISE13] allow manipulation of volumetric effects. Due to high computational demands, most works have focused on direct illumination. Previous lighting design works leverage sketch-, click-and-drag, and paint-based editing concepts. Kerr and Pellacini's studies [KP09] stress that, although painting interfaces are useful in some scenarios, typical editing operations can be better achieved using *direct* and *indirect* manipulation. We follow this distinction in our discussion below.

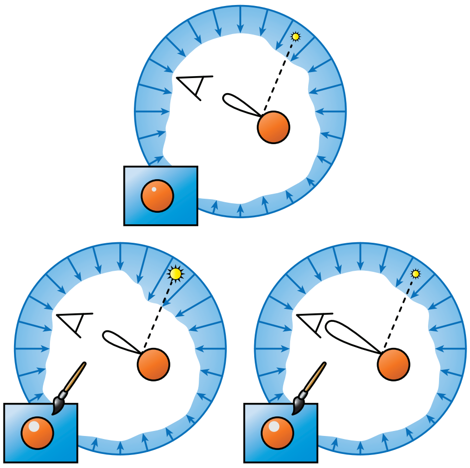


Figure 4: *Goal-based interaction. Top: the original scene illuminated by an environment map; the inset in the bottom left depicts a rendering which shows a highlight. A goal-based interface allows the user to paint a new highlight. Bottom left: After painting the desired appearance, the system solves for new light parameters (e.g. a brighter area around the sun in the environment map). Bottom right: After painting the desired appearance, the system solves for new material parameters (e.g. modifying the BRDF lobe).*

4.1. Directly Controlled Lighting

We first focus on methods to directly control *lighting features* (not to be mistaken with direct lighting parameter control). While indirect interfaces allow artists to roughly sketch the desired appearance of lighting features and let the underlying system solve for the model parameters, sometimes more direct control over the illumination, e.g. to exactly (dis)place features, is beneficial.

A straightforward technique to directly manipulate lighting is what is commonly called "light linking" [Bar97]; here, users can select which light sources affect which objects in

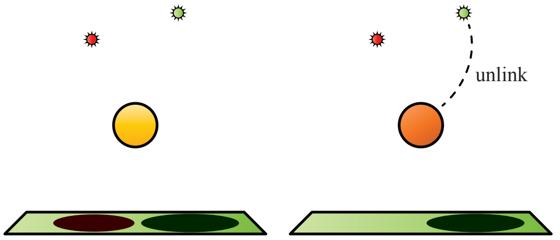


Figure 5: *Light linking. Left: Original configuration with two point light sources and two objects. Right: The green point light is unlinked from the sphere object, hence also casting no shadow onto the bottom plane anymore.*

the scene, allowing to explicitly set shadow caster/receiver relations among them (see Fig. 5).

Apart from directly activating and deactivating light sources, the simplest and arguably most intuitive kind of direct interaction with the scene illumination normally arises from click-and-drag interfaces. For example, Ritschel et al.'s [RTD*10] *Interactive On-Surface Signal Deformation* is an object-space appearance-guided editing tool for manipulating shadows, caustics and indirect light with a custom interface that couples space warping effects for reflection and shadowing with inter-object markups for indirect light exaggeration (Fig. 6).

In designing direct user interfaces and interaction paradigms for lighting design, one important aspect is that—in contrast to materials and scene geometry—the illumination (and thus appearance) is only a by-product of the rendering process and usually not explicitly hand-authored by artists. Therefore, lighting design methods for non-trivial scenarios have to introduce abstractions and visualizations of the underlying light field, which is a five-dimensional, complex function and an effective visualization thereof is difficult [RKRD12]. That said, if the transport is limited to, e.g., a fixed viewing direction (as in cinematic lighting preview systems [PVL*05, HPB06, RKKS*07, SZC*07]) or direct lighting from a finite set of directions, then good visual mappings can be found. For example, Kerr et al. [KPD10] control spot or directional light sources using guided visualizations of the underlying user-deformable lighting volume. Another editing approach is lattice-based deformations, as in Obert et al.'s work [OPP10]. Here, a factored representation of visibility is efficiently stored in compressed matrices, enabling interactive shadow editing under environment illumination.

As industry-leading studios adopt physically based rendering (PBR) in their art generation pipelines, the usefulness of simple manipulation approaches that address effects without considering underlying PBR concepts and constraints decreases (see Section 7.1 for more details). Schmidt et al.'s path space manipulation (PSM) [SNM*13] (see Fig. 7) includes direct manipulation approaches for global illumination effects such as (multi-refracted) caustics, diffuse and glossy indirect bounces, and direct/indirect shadows. Their object-space selection interface respects the UI and interaction paradigms of the underlying digital content creation (DCC) toolchain, and is built atop a parameterized regular expression engine in the spirit of Heckbert [Hec90]. This work is a very general approach which subsumes previous methods, e.g. BendyLights [KPD10] or reflection editing [ROTS09], as special cases. Tabellion and Lamorlette [TL04] use shader falloff-function editing on the hue of indirect color bleeding effects, which can also achieved with PSM. Similarly, Nowrouzezahrai et al. [NJS*11] edit the underlying physical processes of volume rendering.

Lastly, goal-based approaches have also been developed using painting methods in high dynamic range [CRH07, Col08] to sketch both highlights and directly paint and modify environment illumination.

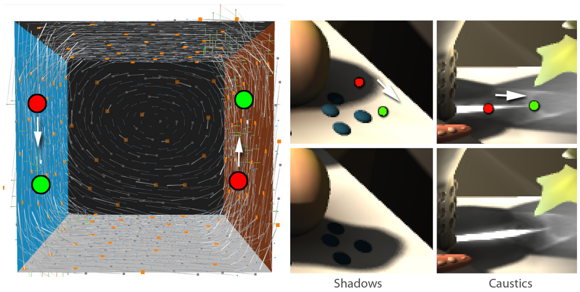


Figure 6: *Interactive On-Surface Signal Deformation [RTD*10] provides a direct interface for lighting manipulation. Users specify constraints (red and green dots) and the underlying system solves for a smooth deformation field, which can be used to accurately displace features such as shadows and caustics. (Images taken from [RTD*10].)*

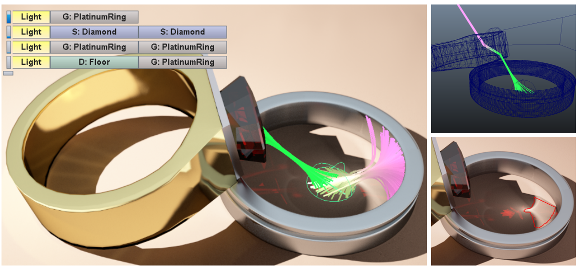


Figure 7: *Schmidt et al. [SNM*13] visualize light transport paths using edge bundling techniques. Selection of lighting features is done in a semi-automatic fashion by analyzing and ranking virtual illumination inside a user-specified region of interest. (Image taken from [SNM*13].)*

4.2. Indirectly Controlled Lighting

Another class of artist-driven lighting design tools offer an *indirect* approach to manipulating lighting parameters in order to reach a desired appearance. Such approaches allow users to indirectly affect the final appearance $L(x \to c)$ of a shot by abstracting the underlying image formation process and exposing interaction paradigms to act on these abstractions. After user manipulation of the abstracted parameters, an underlying processing system automatically determines the settings of *appearance parameters* using inverse image formation models. We overview examples of such *indirect shading and lighting* editing tools below.

Poulin and Fournier [PF92] and Pellacini et al. [PTG02] infer light positions in the context of a simple direct and local illumination model, allowing users to sketch shadow and highlight boundaries atop a final rendered shot. Similarly, light emission direction and distribution characteristics can be optimized given painted [SDS*93] or higher-level [KPC93] user annotations of initial renderings, exploring the physically feasible solution space for lighting design. Mattausch et al. [MIW13] begin with physically based shadows and provide a tool to artistically edit the shadow boundaries in a scene akin to freeform curve editing. Consistency within the scene is achieved by computing and using shadow volumes to render the edited shadows. More complex lighting and reflectance parameters, such as light cone angles, colors and specular surface coefficients can also be automatically computed using similar image-space editing interfaces. Pellacini et al. [PBMF07], for example, let artists paint color, light shape, shadows, highlights, and reflections using a suite of tools tailored for painting lighting features. The inverse image formation models in these works are often based on (potentially non-linear) optimization backends that search the space of *appearance parameter* settings for an appropriate solution [Bar97, CdSF99]. Recent appearance-based interfaces expose image markup techniques to allow users to identify and isolate shadow features, after which the underlying processing system infers a coupled relationship between complex all-frequency shadows and distant environment lighting (still exclusively in the context of direct illumination) [Pel10,OPP10]. Ritschel et al. [ROTS09] also expose an image-space constraint-identification interface to users, focusing on editing the reflection behavior from mirror surfaces. Their underlying system infers spatially-varying reflection directions after user input.

Several sketching-based approaches have been proposed to design complex material, reflectance, and scattering profiles. Pellacini and Lawrence [PL07] present a system where users sketch appearance constraints on the image canvas in order to infer, warp, and transfer appearance from different spatially- and temporally-varying reflectance datasets. Nowrouzezahrai et al. [NJS*11] generalize photon beam primitives [JNSJ11] to non-physical effects, allowing artist-driven sketching of heterogeneous volumetric media densities with art-parameterized shading models (see Fig. 9, left). Their system can additionally infer scattering and absorption coefficients in participating media with single-scattering from user-sketched color constraints. Dong et al. [DTPG11] present a data-driven system that automatically infers higher-dimensional appearance manifolds from a single image, given user-sketched constraints. As with An et al.'s approach [ATDP11], their system allows users to automatically warp the spatially-varying appearance of selected objects in a rendered image. Note that, as with all indirect manipulation approaches, the user is not directly exposed to the underlying behavior or *appearance parameters* of the image formation process. Bousseau et al. [BCRA11] optimize illumination from environment maps to improve the perception of material characteristics without the need for user intervention, which can be interesting e.g. for product design, where isolated key objects are usually presented in an otherwise simplified background context.

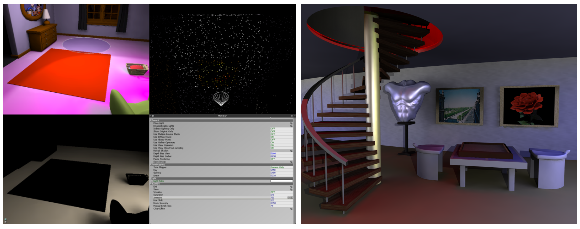


Figure 8: *Left: The user interface of iCheat [OKP*08], a method that enables editing the intensity and color of indirect, global illumination effects. Right: Example result with user-adjusted global illumination. Note that the indirect illumination on the floor matches the modified illumination inside the stairwell. (Images taken from [OKP*08].)*

In the context of indirect appearance manipulation for global illumination editing, Obert et al. [OKP*08, Obe10] expose a painting interface for artists to edit the intensity and color of indirect light. Their system additionally exposes a labeling interface, allowing users to identify indirect sender/receiver relationships (the first example of an *object-space* editing markup). With these relationships in place, indirect illumination can be exaggerated while maintaining important visual constraints necessary when generating plausible rendering results (see Fig. 8).

5. Material Design

We refer to *material* interactions as any local interaction that manipulates the distribution of light at a surface or in a volumetric medium. Examples of materials include spatially-varying BRDFs and BSDFs that model the local reflection and transmittance profiles of a surface, the BSSRDF properties of subsurface scattering effects, the scattering properties and phase function profiles in participating media, or any combination of these properties. While these examples relate primarily to physically accurate light transport models and simulations, non-physical or *artistic* models of local light interaction are also valid material descriptors in the context of our survey.

As with lighting design tools, material design tools and approaches enable artists to edit the final appearance of a scene. These edits, however, modify the material properties of the scene, typically assuming fixed geometry and lighting conditions. More analogies between material and lighting design can be drawn, as material behaviors can be edited using direct, indirect and goal-based editing paradigms. However, unlike lighting design, the literature in material design is much more recent and less established. This only recent development of more sophisticated material design approaches can be attributed to a handful of factors. Firstly, appearance modeling via material manipulation is fundamentally more difficult to formulate mathematically: while appearance changes *linearly* with respect to illumination, whether with direct- or global-illumination effects, according to well-understood mathematical models, the relationship between the final appearance and the materials in a scene is *non-linear* in general. Secondly, typical digital content creation workflows first associate materials based on the underlying “physical composition” (e.g., metal, dielectric, etc.) of an object, and only then begin light placement and manipulation in the scene; as such, lighting is more closely associated with final scene setup for rendering than the materials are, and so it is also more likely to be edited in order to attain a desired appearance. A final factor that contributed to the delayed development of material design approaches relates to the relative evolution of mathematical models and simulation techniques for lighting and materials: the number of different light transport models, as well as their evolution, is much smaller than that of materials. New surface, subsurface, and volumetric material models arise more often than new methods for simulating light transport. Moreover, these material models cover a wide breadth of approaches, including basis-space reflectance models better suited to interactive shading [WRG*09], new microfacet distributions to more accurately model real-world BRDF data [WMLT07], and new volumetric distributions for accurate subsurface reflectance behavior [JAM*10]. In short, the number of material representations has increased steadily over time [DRS08, Bur12], making it difficult for any concrete material design approach to be proposed and adopted.

5.1. Material Editing Approaches

Directly editing the parameters of an underlying material model is the simplest form of material design, for example, editing the reflectance of a diffuse BRDF. Unfortunately, this approach is unintuitive as these parameters often expose too many degrees of freedom to a user and, in many cases, changes to the final appearance caused by direct parameter edits can be difficult to predict, and may also depend on scene geometry [VLD07]. Moreover, inconsistencies in the scale and physical interpretation of parameters between different material models further complicates intuitively controlled edits using this approach. These limitations are even more pronounced in the case of volumetric appearance editing where many physical parameters are decoupled from the final observed appearance. For example, the absorption coefficient quantifies (in a complex, scene-scale dependent fashion) the amount of light at different wavelengths that is *removed* (absorbed) during volumetric transport, instead of the amount and light that is *affected* (scattered) during transport (such as with the diffuse reflectance of a surface BRDF model) [NJS*11]. Similarly, predicting appearance from phase functions is difficult, requiring a mapping of parameters to perceptually uniform appearance spaces for effective translucent material design [GXZ*13]. As with lighting design, more sophisticated and intuitive material design approaches allow users to specify constraints on the final targeted *appearance* before applying an underlying system to automatically determine the *material property settings* necessary to best match the requested edits.

Several model-driven approaches to material design have recently been proposed. Song et al. [STPP09] edit heterogeneous subsurface scattering on and inside surfaces, starting from the simplified diffusion model of subsurface scattering [JMLH01]. They approximate this BSSRDF model as a product of two parameterized blurring kernels, leading to a representation that is amenable to various simple parametric and image-based editing operations. Sadeghi et al. [SPJT10] present an artist-driven and controllable model for hair rendering, exposing intuitive artistic controls for generating images of hair under complex light transport scenarios (see Fig. 9, right). Their approach is built atop a high-performance rendering model that allows for rapid artist iteration. Furthermore, Sadeghi et al. explicitly alter the energy-conservation of the underlying mathematical model as many art-driven edits *require* non-physical lighting interactions (e.g., the capability of creating energy during transport) in order to obtain a desired look, even if this results in an invalidation of the underlying physical plausibility of the final edited reflectance behavior. Obert et al.’s [OKP*08] painting interface for indirect illumination exaggeration, discussed earlier in Sect. 4.2, can also be interpreted as a material design tool as the editable *transfer functions* they expose to users (indirectly) encode emitted lighting distribution or, equivalently, the reflection of the (indirect) incident lighting distributions at surfaces. Colbert’s thesis [Col08] covers several approaches to material editing based on the concept of BRDF lobe (i.e. highlight) sketching under environment illumination, allowing for spatially varying BRDF editing as well as appearance matching. Colbert et al.’s BRDF-Shop [CPK06] interface allows users to sketch spatially-varying BRDF distributions for commonly used phenomenological reflectance models and, coupled with an interactive rendering tool, allows artists to very quickly visu-

Figure 9: *Nonphysical, art-directed lighting and material, as used in production. Left: The flexible volumetric lighting technique of Nowrouzezahrai et al. [NJS*11] enables animated, curved light beams to visually enhance storytelling. Right: Sadeghi et al. [SPJT10] present an artist-friendly hair shading system which enables rapid modeling of desired material looks for the characters' different types of hair. (Images taken from [NJS*11, SPJT10] and ©Disney Enterprises, Inc.)*

alize and manipulate simple reflectance behaviors in scenes lit by direct illumination from environment maps. Khan et al. [KRFB06] use an image-based approach to estimate surrounding incident illumination given a single image, allowing the material properties of objects in an image to be edited with phenomenologically plausible results. Muñoz et al. [MnES*11] present a method for capturing approximate BSSRDFs from single images, using optimization to fit input data to smooth diffusion profiles, which can then be used as a basis for further editing.

Data-driven and optimization-based techniques have also successfully been applied to the problem of material design. An and Pellacini [AP08] formulate image- and material-editing as a local search and energy minimization problem, allowing users to sketch rough appearance editing constraints and then automatically searching for patterns in the unedited dataset to warp and blend into the edited regions. Of note, they apply their approach to editing higher-dimensional spatially-varying reflectance datasets, as well as simple images and HDR map datasets. Dong et al. [DTPG11] deduce spatially-varying properties for a simple parametric reflectance model (e.g., glossiness, normal variation, and diffuse albedo), using an image patch of a planar surface lit by a directional light source and targeted user markups in the scene. This allows them to decompose an image into a product of shading and reflectance maps that are then exposed to users in order to permit a variety of different appearance editing post-processes. An et al. [ATDP11] retarget measured material appearance by warping the reflectance behavior of a captured material according to a set of template reflectance behaviors. This approach allows users to more quickly design complex reflectance behaviors, combining the spatial variation of captured materials with the reflectance behaviors of the template materials. Kautz et al. [KBD07] edit large-scale bidirectional texture functions (BTFs) using an out-of-core system to devise new BTF data from user-specified spatially-varying microgeometry and reflectance profiles, leveraging a data-driven analysis of existing BTF datasets during the interaction process. In a similar spirit, Iwasaki et al. [IDN12] provide efficient editing of so-called bi-scale BRDFs, where the effective BRDF is the result of integrating an analytical BRDF model over the visible normal distribution of small-scale microgeometry. Because they represent both normal distributions and small-scale BRDFs as a sum of spherical Gaussians, the effective BRDF can be synthesized at high rates, allowing interactive editing of microgeometry and highly-glossy small-scale BRDFs, and real-time image synthesis using effective BRDFs.

Ben-Artzi et al. [BAOR06, BAERD08] express the per-pixel outgoing radiance in the reflection and rendering equations, in a fixed-view image relighting context, as an expansion of basis elements defined according to the materials in a scene. After precomputing this extended radiance formulation, users can interactively edit the materials by reweighting the radiance expression, all while shadows and indirect illumination remain up-to-date. Sun et al. [SZC*07] similarly express outgoing radiance according to basis elements based on a library of potential materials. They instead perform the decomposition in object space, and separately across each individual light bounce (supporting up to two bounces of indirect illumination), in order to additionally support interactive view manipulation. More recently, Wu et al. [WDR11] combines micro- and macro-scale editing in a relighting framework in order to model editing across levels of detail, including the proper antialiasing of reflectance across scales. The same authors recently showed how to efficiently handle the problem of inverse bi-scale design [WDR13], e.g. deriving micro-scale parameters from user-supplied macro-scale appearance edits, using an efficient search in precomputed libraries of normal distributions and small-scale BRDFs.

Kerr and Pellacini [KP10] evaluate several material design paradigms to determine which, if any, provides a superior editing workflow for novice users. They found that manipulating BRDF parameters, either directly (e.g., varying the Phong exponent) or across perceptually mapped input spaces, outperformed image-based material design paradigms, although it is unclear whether this was due to the limitations of the underlying image-space manipulation tool they employed. Nonetheless, they clearly conclude that the type of user interface exposed to an artist for material editing can play a significant role on the utility of the underlying material design approach.

6. Measuring Interface Effectiveness

The previous sections have introduced a variety of interfaces for editing surface appearance and environment lighting. To measure their effectiveness, Kerr and Pellacini [KP09] established a user study methodology used to determine the effectiveness of the various interfaces relative to one another. This methodology have been used in many subsequent studies in appearance editing [KP09, KP10, OKKP12, KRG*14, JMB*14]. We have summarized briefly the results of these findings before. In this section we summarize the experimental methodology.

All the studies focus on comparing interface paradigms, rather than attempting to tease out differences between implementations of the same paradigm. Within a study, interfaces are kept as consistent as possible between each other in terms of window layout, keyboard shortcuts, etc., and features specific to one paradigm are not available in the others, even if this might be useful in practice. Furthermore, it is believed that real-time feedback is necessary for effective appearance design, so in all studies faithful real-time rendering is available.

The main concern in measuring relative interface effectiveness is that one wants to record precise objective observations while leaving enough artistic freedom for users to explore the design space. To achieve both goals, experiments are run by performing two separate kinds of tasks. In *matching trials*, users are given target images that they need to match precisely using each interface. Matching tasks are meant to measure effectiveness of each interface to precisely control appearance. The matching images and starting configurations are chosen as representative of simple design tasks users routinely perform. Subsequently, in *open trials*, users are given a starting configuration and a non-precise verbal description of a desired look as guidance for exploring the design space. These trials record users' workflows during free form exploration.

During the experiments, objective and subjective data is collected. Objective measurements such as time to completion and image differences are used as measures of interface effectiveness. While software instrumentation collects a variety of other data, such as mouse clicks, button presses, etc., these are mostly used to cross validate the findings of the former measures and to qualify users' workflows. Together with objective measurements, users are required to fill in questionnaires, where they are asked to rank and rate each interface in a variety of categories, from specific design tasks to overall effectiveness.

The collected data is analyzed to determine whether statistically significant trends in interface preference emerge. The analysis methods differ depending on the collected data, with ANOVA [Fis25] been the most-used for time to completion, image error, and subjective ratings. From this analysis one can determine which interface users prefer and for which operations. Analysis of different datasets are correlated to strengthen each conclusion. For example, in most studies, time to completion and image error correlate well with subjective preference. A final analysis is performed by reporting a set of observations on users workflows derived by manually summarizing the recorded editing histories. For example, whether exploration is monotonically converging or whether users work in a block-and-refine workflow. These observations are mostly informal since this experimental design does not provide enough data to produce statistically-significant workflow observations.

In most of the published literature, authors focused on running the experiments on novice users, since novices are the vast majority of users and the most likely to benefit from improvements in interfaces. While we believe that the general methodology is valid also for experts, this has not been tested so far.

The choice of novices in turn puts constraints on tasks' complexity. On the one hand, one wants complex tasks that can reflect real-world editing scenarios well. On the other, one needs to reduce user fatigue and ensure that the trials are doable. In most studies, users work for a few hours and they are generally able to complete all tasks. Scenes are relatively simple in terms of geometric complexity, but remain varied in terms of appearance. This also puts less pressure on the rendering system.

While the described methodology worked well for measuring current interface paradigms, future efforts in this domain should focus on statistically characterizing overall design workflows. Covering such a large space of operations while still ensuring statistically-significant observation would require drastically larger datasets that likely require a different experimental design. In a way, the work published so far focused on answering precise questions about appearance design, while in the future the goal should be to understand the design problem as a whole.

7. Rendering Challenges and Production Concerns

While solving for the light transport inside a scene at interactive rates has been tackled by many researchers (see [DK09, RDGK12, DKH*13] for recent surveys on interactive and scalable global illumination methods), their techniques are typically concerned with slightly different requirements than appearance design: dramatic changes in lighting and material are not that common and dynamism is mainly focused on geometry and camera change. Unfortunately these are exactly those components of the image formation (Eq. 1) which are usually held fixed during appearance editing.

From a rendering standpoint, two main concerns have been explored. First, appearance design should be interactive. But as scene complexity increases, it remains difficult to provide interactive feedback during light and material manipulation. This has lead to the development of various re-

Figure 10: *Reiner et al. [RKRD12] developed tools for light transport inspection and evaluated their effectiveness in comparison to straightforward approaches. From left to right, false-color rendering, spherical plots of radiance at one point in space, detecting important incident light paths, volumetric inspection, and particle flow tools. (Image taken from [RKRD12].)*

lighting engines [PVL*05, HPB06, RKKS*07]. The second main consideration has been to derive non-physical appearance models that would be inherently more controllable for artists [KPD10, NJS*11, SNM*13].

Interactive appearance design is increasingly of major interest for the special effects and animated film industries. Most major studios have adopted relighting techniques from academia, or performed their own targeted investigations. In a production environment, there are several constraints that limit the applicability of available techniques. These constraints are leading the industry in a certain direction and can be used by researchers as a guide for finding tangible future research problems.

The primary production requirement is that an appearance design tool *must* produce a preview that matches the final appearance of a shot. This preview can be noisy, consider a subset of the scene, or be lower quality in some fashion, but, given enough render time it should produce a result matching the final render. An interactive preview is of no benefit if it does not provide feedback about the final shot. This poses a significant engineering challenge, since the entire material and lighting systems (lights, textures, arbitrary shaders, etc.) available in the final renderer need to be replicated in the design tool. Some prior systems have tackled this challenge with varying degrees of automation [PVL*05, RKKS*07].

Due to this duplicate engineering burden, however, the current trend has been to incorporate interactive appearance design directly into the final renderer. The shift has been towards enabling progressive updates in the final renderer (typically using ray or path tracing) with minimal need for precomputation or caches. Both the commercially available Arnold and Renderman renderers currently provide such functionality. This complexity has also inspired many studios to move towards more principled, physically based lighting and material models [Bur12]. In addition to providing vastly simplified and more intuitive models for artists—in comparison to the arbitrary shaders of the past—these systems tend to be easier to port or abstract between rendering systems due to their more rigid interfaces.

Nonetheless, these production rendering systems typically evolve slower than the pace of cutting-edge research, so studios must often make the difficult decision to either build their own separate design tools (at considerable development and maintenance effort) or to rely on updates to their core renderers. In spite of the trend for progressive final renderers, a similar challenge now exists for leveraging the full power of the GPU for interactive appearance design.

7.1. Physically-based Rendering Adoption in Industry

The recent industrial adoption of PBR, especially in feature-film and video game production, has grown from the need to author and edit photorealistic content in as predictive and standardized a manner as possible. Advances in physically-accurate global illumination simulation techniques have simplified the incorporation of advanced shading effects into content generation pipelines, rendering hand-crafted ad-hoc approximations of such effects an unnecessary artifact of the limitations of simpler rendering approaches. Indeed, prior to the incorporation of end-to-end PBR workflows, lighting and shading artists were often forced to resort to “hacks” in order to synthesize a shading effect that was not natively supported by the underlying non-GI-capable renderer.

Fundamentally, PBR adoption in industry consists of two principal components [MHM*13]: physically-correct and consistent specifications for virtual luminaires in a scene, including emission profiles and geometric shape; and, physically-correct definitions of the reflectance profiles used to describe the scattering of light at surfaces in a scene. The standard radiometric units and mathematical models used in realistic image synthesis form the foundation for this segmentation and all necessary definitions.

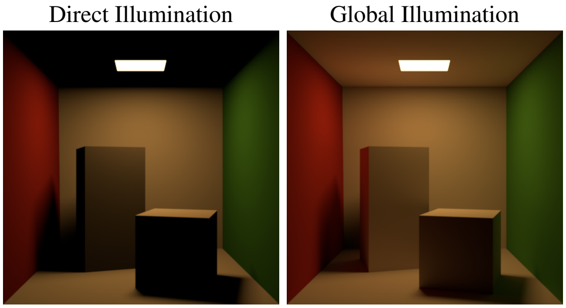
Previously, the explicit and correct separation of these two components, *lighting* and *material*, was left to the implementer and need not follow any prescribed pattern or standard. In the context of industrial PBR workflows, and especially in the design and implementation of shaders, we will discuss how enforcing a strict and standardized separation of lighting and reflectance behaviors, and their parameters, leads to a more intuitive, predictable, consistent, and maintainable content generation system.

Industrial lighting and shading experts have presented several similar anecdotal examples of how PBR workflows have resulted in more predictable and robust content generation experiences, both for small- and large-scale productions and teams. Many of these examples similarly discuss the shortcomings of previous non-PBR ad-hoc shading workflows, especially in the presence of secondary lighting effects, and we draw upon many such examples presented by McAuley et al. [MHM*13] in Fig. 11: as alluded to earlier, the lack of explicit segmentation between lighting and material properties can lead to unexpected behavior during content generation, especially in the presence of secondary illumination; for example, in the context of simple direct-illumination simulations, an artist is afforded an additional degree of freedom when specifying lighting and material parameters, as the same image can be generated using any (infinite) number of potential lighting and material parameter settings (Fig. 11, left column), however once secondary lighting effects (such as physically-accurate global illumination) are included, this additional degree of freedom disappears and an artist must properly account for the different roles that lighting and material play in realistic image synthesis.

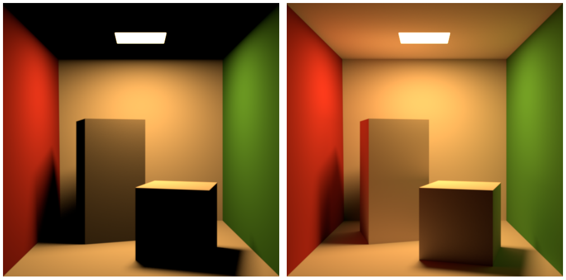
By adhering to well-defined rules and models, industry-wide standardized PBR methodologies seem to converge to several important consequences:

- content creators and technical implementors can more often link technical and artist-facing discussions with a consistent "language" based on radiometry,
- the impact parameter changes will have across a sequence remains both predictable and consistent (see below),
- content for large-scale projects, spanning across potentially many different teams and over large timelines, can be developed, merged and integrated independently with less fear of causing visual conflicts in the final result,
- digital assets can more easily be maintained over longer periods and, in some cases, across different (PBR-compliant) rendering platforms,
- photo-realistic appearance results automatically and implicitly from the application of PBR methodologies, and design "recipes" based on these approaches (see below), and;
- previous ad-hoc solutions are no longer necessary.

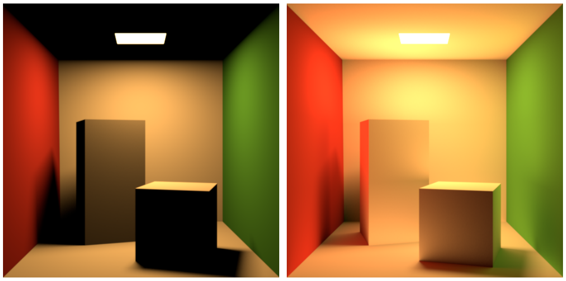
Perhaps the most important benefit of PBR workflows is the flexibility they afford during material/reflectance design: a wide range of interesting, realistic reflectance behaviors can be rapidly developed and iterated upon, all in a predictive manner that is easy to maintain within and across groups/studios. We discuss the simple, but important, properties and models that lead to this flexibility below.



Original Lighting (L_e) and Material (f_r) Parameters



$L_e \to 4 \times L_e$ and $f_r \to 1 \times f_r$



$L_e \to 1 \times L_e$ and $f_r \to 4 \times f_r$

Figure 11: *Without global illumination (left column), the same image can be obtained by either manipulating the emission parameters (left, center) or the material properties of the scene (left, bottom); however, with even a single bounce of indirect illumination enabled (right column), these two edits yield significantly different results.*

Please note that while PBR workflows have been increasingly adopted in the industry, non-photorealistic rendering (NPR [GG01]) is another important design space. While NPR techniques and appearance editing inherently seem to be more tightly coupled – they oftentimes simulate and improve upon traditional artistic workflows [KCWI13] – NPR still warrants its own specialized appearance editing methods, as exemplified in the work on stylized shading by Todo et al. [TABI07].

Material Properties and Design in PBR. A PBR material is typically encapsulated as a BRDF f_r and, in order to maintain physical correctness, all BRDFs should respect

- non-negativity, such that $f_r \geq 0$, and
- energy conservation, such that $\int f_r(x' \leftarrow x \leftarrow x'')\, G(x \leftrightarrow x'')\, \mathrm{d}x'' \leq 1$, for all x'' visible to x .

Of these two properties, energy conservation is arguably the most important in the context of PBR workflows since, if respected, it ensures that images will only over-saturate (even in the presence of indirect illumination) if the lighting is purposefully adjusted to attain such an effect (unlike e.g. Fig. 11 bottom, right). This simple but reassuring property allows lighting and material artists to control the scale of tones and colors in a final shot in a predictable and consistent manner (see below for more details). In the case of bi-directional techniques, it is also important for BRDFs to respect the *reciprocity* property, $f_r(x' \leftarrow x \leftarrow x'') = f_r(x'' \leftarrow x \leftarrow x')$.

Moreover, a consistent terminology based on radiometric parameterizations of existing BRDF models should be used as often as possible throughout a PBR authoring workflow; for example, the diffuse *reflectivity* or *albedo* ρ_d can be concretely and intuitively (and correctly) described as the ratio of light directly reflected by a perfectly diffuse surface lit from a uniform distant white lighting environment. Such definitions override existing ad-hoc and ambiguous definitions (such as, in the case of diffuse BRDFs, describing the albedo of the surface as its "color"). These radiometrically-motivated parameterizations/definitions can additionally lead to useful secondary intuitions about the behavior of shading and reflectance models; again using the example of diffuse BRDFs, the definition of energy conservation *and* the definition of ambient occlusion appear by expanding the definition of the albedo into its equivalent mathematical form using the reflection equation with direct, uniform unit illumination:

$$\rho_x = \int_\Omega 1 \times \frac{\rho_x}{\pi} \cos\theta_i \, d\omega_i \quad (\leq 1) \,. \tag{2}$$

Another benefit of radiometrically parameterizing individual BRDF models is that such parameterizations generalize, in a decoupled fashion, to convex weighted combinations of BRDFs: if each BRDF is guaranteed to conserve energy, and if the impact on energy conservation of each of the BRDFs' parameters is understood, then artists can easily author new materials that will conserve energy (with their own set of radiometrically-motivated parameters) by combining existing base BRDF models. This process duplicates a common, pre-existing workflow in non-PBR systems.

For example, the diffuse albedo ρ_d must take on values between 0 and 1, by definition (and by energy conservation), and a perfect mirror BRDF reflection is similarly parameterized by the ratio of perfectly reflected light along the mirror direction, using a single reflectivity parameter ρ_s that also takes on values between 0 and 1 (see Fig. 12). Since each model is independently energy conserving, then a simple composite BRDF that sums the two models, $f_r = f_d + f_s$, will also respect energy conservation now if $\rho_d + \rho_s \leq 1$.

This simple example (Fig. 13) generalizes to arbitrary convex weighted combinations of energy-conserving BRDFs, including materials with spatially-varying parameters.

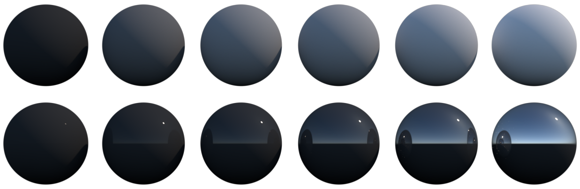


Figure 12: *An artist can predictably control the amount of diffusely (top) and specularly (bottom) reflected light, across different lighting scenarios, by increasing albedo ρ_d and specular reflectivity ρ_s if the underlying BRDFs are energy conserving.*

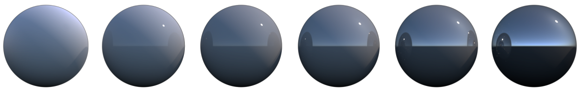


Figure 13: *Combining diffuse and specular reflection while maintaining energy conservation; from left to right: perfect diffuse reflection ($\rho_d = 1, \rho_s = 0$) to perfect mirror reflection ($\rho_d = 0, \rho_s = 1$).*

The reflectance *behavior* of a material can also be parameterized in a fashion that guarantees energy conservation and, as such, predictable rendering variations. A common example is the parameterization of *glossiness* in metallic BRDF models, which allows artists to modify the glossiness of a material without having to worry about whether their modification will result in oversaturation or "intensity explosions" in the context of global illumination simulations (Fig. 14).

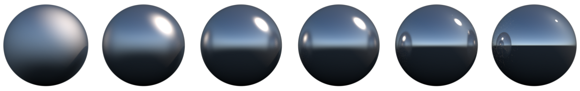


Figure 14: *Predictable appearance of highlights and blur as one modifies the glossiness in a simple metallic BRDF model.*

Radiometrically-relevant parameterizations can be derived for a wide range of existing material models, including microfacet models where equivalences can be established between e.g. microfacet distribution parameters and phenomenological glossiness factors [Jak10]; note, for example, how the range of appearance of the microfacet model's renderings in Fig. 15 is similar to that of Fig. 14.

Maintaining predictable, radiometrically-based parameterizations of material models can also facilitate the design of reflectance behaviors that span a broad class of materials, including metals/conductors and dielectrics/insulators, all using a single set of parameters. Also, inverse analysis can be performed, allowing an artist to classify a material

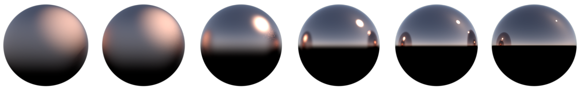


Figure 15: *The effects on glossiness when increasing the microfacet roughness.*

in one of these categories by simply referring to the values/ranges of its parameters. For example, guidelines based on e.g. meteorological radiometric studies can be used to provide artists with specific valid ranges of parameters (such as glossiness, specular color spectra, etc.) in order to design reflectance behaviors based on – or at least inspired by – real materials and objects.

From a software engineering perspective, these parameterizations are also amenable for implementation in node-based, layered/hierarchical material graphs, such as the system used in the “The Order 1886” production by Ready at Dawn Studios [MHM*13] where a base material object is defined with default material parameters, and all other materials are derived from the base material and encoded using a sparser, differential parameterization. One benefit of such a layered approach is that global appearance changes can be propagated by simply modifying interior/non-leaf material nodes.

Decoupling Light Transport in PBR. Separating material models from lighting models allows for the underlying *light transport* simulation to be completely decoupled from the *data* it operates on. Here, different rendering frameworks are free to implement any physically-based light transport simulation algorithm and, as long as it correctly estimates the rendering (or radiative transport) equation, the aforementioned benefits yielded from the division of material and lighting behavior will be maintained. Of course, “correctness” can depend on the goals of the renderer, e.g. if it aims to provide an unbiased Monte Carlo integral estimate, but is meant here moreso to qualify that the radiometric quantities of interest are computed in a statistically sound manner.

Indeed, several existing rendering-specific domain specific languages facilitate this decoupling. The *Open Shading Language* (OSL) specification [MHM*13] explicitly decouples transport integration from e.g. material models using a *radiance closure* paradigms; any shading engine that implements the OSL specification can incorporate any underlying light transport algorithm that samples lighting according to the closures specified by the shaders. In contrast, OpenGL and DirectX provide shader programming languages that leave any such decoupling in the hands of the shader developer.

Shading Consistency in PBR. Maintaining a consistent appearance across various lighting scenarios can be difficult without an explicit decoupling of material and lighting models, particularly one that uses standardized radiometric units. However, since PBR systems adhere to these constraints, a single material model can be applied seamlessly while maintaining consistent shading behavior when e.g. moving between local- and distant-lighting models, which is common in the light rigs and (ir)radiance probes used in interactive games [MHM*13].

Moreover in the context of offline rendering, by enforcing energy conservation and reciprocity in material models (see above), specular reflections and specular highlights remain consistent without the need for ad-hoc heuristics or specialized shading models. This is also useful in the context of seamless transitions between localized lighting models and tracing secondary lighting rays against the actual scene geometry.

A similar consistency issue arises when enforcing physically-accurate fall-off models for geometric/area lighting, as well as for traditional point lighting models. By avoiding ad-hoc distance fall-off terms in lighting models, area lighting results can readily converge to point-lit results as the size of the geometric light’s size is reduced (and its power adjusted according to its physical properties). As such, if a desired look requires changes to the lighting properties, these changes are guaranteed to both interact correctly with the reflectance behavior encoded in the (separate) material models, as well as providing the correct indirect and secondary lighting effects.

While one can argue that these constraints may limit an artist’s ability to attain a desired look, the long-term benefits of enforcing the PBR guidelines discussed above are becoming increasingly relevant. This will continue to hold true, particularly as the complexity of virtual scenes increases, mandating growth in the size of teams working on authoring this content, as well as the time they spend developing and maintaining these virtual assets. Initial work on adapting light transport editing tools to PBR workflows has already begun (e.g., [SNM*13]) and is likely to be an important area of future work.

7.2. Progressive Rendering

Even with a PBR pipeline in place, the aggregate effect of lighting and material changes on the final image can still be complex and difficult to predict. Due to this, some form of interactive feedback during look development can greatly reduce iteration time by enabling an artist to quickly explore the complex space of appearance.

While there has been significant effort in creating dedicated relighting engines [PVL*05,HPB06,RKKS*07] to enable this interactive exploration, developing a separate tool comes with the drawback of increased pipeline complexity and maintenance costs. The recent trend—partially facilitated by the recent transition to PBR workflows and ray tracing approaches—has been instead to enable progressive updates directly within the core renderer. Most commercially available production renderers, such as Arnold and RenderMan, provide such functionality.

While enabling progressive updates in pure unbiased Monte Carlo rendering algorithms like path tracing (PT) [Kaj86] or bidirectional path tracing (BPT) [LW93] is straightforward, most algorithms require more careful consideration. These methods differ not only in the technical changes required to make them progressive, but also how amenable they are in providing predictive, progressive updates.

Markov Chain Monte Carlo (MCMC) techniques like Metropolis Light Transport (MLT) [VG97] can be more efficient at finding extremely difficult light paths, but their uneven convergence behavior can outweigh this benefit when used in a production setting where fast, progressive iterations are required [KGKC13]. Specifically, while PT and BPT might have slower convergence in difficult scenes, they tend to have predictable convergence and relatively uniform noise across the image. The convergence behavior of MCMC techniques, in comparison, can be much less predictable. During rendering the image may seem deceivingly smooth and converged, only to suddenly change due to an entirely new lighting feature appearing when the Markov Chain discovers an new "island" of important paths. This unpredictability can dramatically affect appearance, making appearance editing more difficult.

Introducing bias to the Monte Carlo method can also result in more efficient rendering. Examples of this include photon mapping [Jen01, JC98, JZJ08, JNSJ11] and virtual point light (VPL) [Kel97, WFA*05, WABG06, HKWB09, WKB12, NNDJ12b] methods, both of which can provide sub-linear cost in the number of simulated light paths by employing hierarchical data structures when looking up photons or virtual lights. Biased approaches can be trivially made progressive by simply averaging multiple statistically independent passes. There are two downsides to this however. Firstly, this straightforward approach will reduce variance as the number of passes increases, but does nothing to eliminate bias. Secondly, while splitting the simulation across many independent passes eliminates the need to store as many photons or virtual lights at once, it limits the methods' abilities to scale sub-linearly, and, in the limit, degrades to linear scaling. The first of these issues can be addressed by judiciously modifying the algorithms parameters to ensure that both variance and bias diminish to zero in the limit.

Hachisuka et al. showed how surface photon mapping could be formulated progressively to eliminate both variance and bias in the limit [HOJ08, HJ09]. The approach involves progressively shrinking the density estimation kernel across passes, based on a few extra statistics stored at each camera ray hit point. Knaus and Zwicker [KZ11] showed that if the passes are independent and identically distributed, the same asymptotic behavior can be achieved without the need to store any additional statistics and that the kernel shrinkage rate depends on the dimensionality of the blur. This allows using photon mapping as a block box with minimal changes, and provides for trivial parallelism across passes, each of which could be schedule on a different machine on a large render farm. Jarosz et al. showed how the same probabilistic approach could be used to enable a progressive variant [JNT*11] of the photon beams algorithm [JNSJ11]. Later, Novak et al. [NNDJ12a] showed that the same progressive shrinking can ensure a consistent algorithm when the photon beams are used as virtual beam lights in a many-light rendering approach. The radius of virtual spherical lights [HKWB09] could likewise be reduced across passes to ensure bias and variance are simultaneously reduced. A similar approach of radius shrinking can be applied to arbitrary path vertex connections of more general bidirectional methods, as simultaneously proposed by Hachisuka et al. [HPJ12] and Georgiev et al. [GKDS12].

8. Open Problems & Challenges

There are two main open issues that the community should be investigating further. First, we strongly believe that fast rendering is possibly the most significant bottleneck in designing complex enough scenes. Most user studies have shown that in the presence of interactive feedback, artists are significantly more productive. Considering that appearance design is ultimately a search problem for the right parameters, this should come as no surprise. While many rendering algorithms and systems have been used in the past, we are still far away from having interactive production-quality feedback, which is what is needed to make final artistic decisions. As a compromise, progressive preview rendering is increasingly replacing previous custom re-lighting engines as a standard.

From an interface perspective, we believe that it still remains unclear how to control all lighting-material interactions with a unified interface that supports all lighting types, including environment maps and local area lights, together with spatially varying opaque and translucent materials. On top of this, while being more promising to handle different types of material and lighting interactions in a unified manner, the additional implementation and precomputation complexity of the more advanced indirect methods seems to be a risk that so far has hindered their adoption in the industry. It is also challenging, and remains for future work, to convey additional information (exceeding preview renderings) about light transport and material interaction [RKRD12] (see Fig. 10). Throughout this report, we have discussed a variety of methods tailored to specific aspects of appearance design, but none encompass all aspects concurrently. Furthermore, it remains unclear whether there is such a unified approach for appearance designs, since it often seems that different appearance parameters are manipulated more effectively with different interfaces.

Finally, in our opinion, large-scale user testing should be pursued both to validate better current interaction methods and to drive further research in the field more formally.

References

[AP08] AN X., PELLACINI F.: AppProp: All-pairs appearance-space edit propagation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 27*, 3 (2008), 40:1–40:9. 4, 8, 19

[ATDP11] AN X., TONG X., DENNING J. D., PELLACINI F.: AppWarp: Retargeting measured materials by appearance-space warping. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 30*, 6 (2011), 147:1–147:10. 6, 8, 19

[AUW07] AKERLUND O., UNGER M., WANG R.: Precomputed visibility cuts for interactive relighting with dynamic brdfs. In *Proceedings of Pacific Conference on Computer Graphics and Applications* (2007), IEEE Computer Society, pp. 161–170. 3

[BAERD08] BEN-ARTZI A., EGAN K., RAMAMOORTHI R., DURAND F.: A precomputed polynomial representation for interactive BRDF editing with global illumination. *ACM Transactions on Graphics 27*, 2 (2008), 13:1–13:13. 3, 8, 19

[BAOR06] BEN-ARTZI A., OVERBECK R., RAMAMOORTHI R.: Real-time BRDF editing in complex lighting. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 25*, 3 (2006), 945–954. 3, 8, 19

[Bar97] BARZEL R.: Lighting controls for computer cinematography. *Journal of Graphics Tools 2*, 1 (1997), 1–20. 4, 6, 19

[BCRA11] BOUSSEAU A., CHAPOULIE E., RAMAMOORTHI R., AGRAWALA M.: Optimizing environment maps for material depiction. vol. 30, Eurographics Association, pp. 1171–1180. 6

[BP96] BRIÈRE N., POULIN P.: Hierarchical view-dependent structures for interactive scene manipulation. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 1996), ACM, pp. 83–90. 3

[Bur12] BURLEY B.: Physically-based shading at Disney. In *Practical physically-based shading in film and game production* (2012), pp. 10:1–10:7. 7, 10

[BUSB13] BELL S., UPCHURCH P., SNAVELY N., BALA K.: OpenSurfaces: a richly annotated catalog of surface appearance. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 32*, 4 (2013), 111:1–111:17. 3

[CdSF99] COSTA A., DE SOUSA A., FERREIRA F.: Lighting design: A goal based approach using optimization. In *Proceedings of Eurographics Workshop on Rendering* (1999), Eurographics Association, pp. 317–328. 6, 19

[Cha60] CHANDRASEKHAR S.: *Radiative Transfer*. Dover Publications, New York, NY, USA, 1960. 3

[Col08] COLBERT M. C.: *Appearance-driven Material Design*. PhD thesis, School of Electrical Engineering and Computer Science, University of Central Florida, 2008. 5, 7

[CPK06] COLBERT M., PATTANAIK S., KŘIVÁNEK J.: BRDF-Shop: Creating physically correct bidirectional reflectance distribution functions. *IEEE Computer Graphics and Applications 26*, 1 (2006), 30–36. 7, 19

[CPWAP08] CHESLACK-POSTAVA E., WANG R., AKERLUND O., PELLACINI F.: Fast, realistic lighting and material design using nonlinear cut approximation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 27*, 5 (2008), 128:1–128:10. 3

[CRH07] COLBERT M., REINHARD E., HUGHES C. P.: Painting in high dynamic range. *Journal of Visual Communication and Image Representation 18*, 5 (2007), 387–396. 5, 19

[DK09] DACHSBACHER C., KAUTZ J.: Real-time global illumination for dynamic scenes. In *ACM SIGGRAPH Courses* (2009). 9

[DKH*13] DACHSBACHER C., KŘIVÁNEK J., HAŠAN M., ARBREE A., WALTER B., NOVÁK J.: Scalable realistic rendering with many-light methods. *Computer Graphics Forum 31*, 1 (2013), 88–104. 9

[DRS08] DORSEY J., RUSHMEIER H., SILLION F.: *Digital Modeling of Material Appearance*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2008. 7

[DTPG11] DONG Y., TONG X., PELLACINI F., GUO B.: AppGen: Interactive material modeling from a single image. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 30*, 6 (2011), 146:1–146:10. 1, 6, 8, 19

[Fis25] FISHER R.: *Statistical methods for research workers*. Oliver & Boyd, Edinburgh, Scotland, UK, 1925. 9

[GG01] GOOCH B., GOOCH A.: *Non-Photorealistic Rendering*. AK Peters/CRC Press, Natick, MA, USA, 2001. 11

[GH00] GERSHBEIN R., HANRAHAN P. M.: A fast relighting engine for interactive cinematic lighting design. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 2000), ACM, pp. 353–358. 3

[GKDS12] GEORGIEV I., KŘIVÁNEK J., DAVIDOVIČ T., SLUSALLEK P.: Light transport simulation with vertex connection and merging. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 31*, 6 (2012), 192:1–192:10. 14

[GXZ*13] GKIOULEKAS I., XIAO B., ZHAO S., ADELSON E. H., ZICKLER T., BALA K.: Understanding the role of phase function in translucent appearance. *ACM Transactions on Graphics 32*, 5 (2013), 147:1–147:19. 7

[Hec90] HECKBERT P. S.: Adaptive radiosity textures for bidirectional ray tracing. *Computer Graphics (Proc. SIGGRAPH) 24*, 4 (1990), 145–154. 5

[HJ09] HACHISUKA T., JENSEN H. W.: Stochastic progressive photon mapping. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 28*, 5 (2009), 141:1–141:8. 14

[HKWB09] HAŠAN M., KŘIVÁNEK J., WALTER B., BALA K.: Virtual spherical lights for many-light rendering of glossy scenes. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 28*, 5 (2009), 143:1–143:6. 14

[HOJ08] HACHISUKA T., OGAKI S., JENSEN H. W.: Progressive photon mapping. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 27*, 5 (2008), 130:1–130:8. 14

[HPB06] HAŠAN M., PELLACINI F., BALA K.: Direct-to-indirect transfer for cinematic relighting. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 25*, 3 (2006), 1089–1097. 3, 5, 10, 13, 19

[HPJ12] HACHISUKA T., PANTALEONI J., JENSEN H. W.: A path space extension for robust light transport simulation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 31*, 6 (2012), 191:1–191:10. 14

[HR13] HAŠAN M., RAMAMOORTHI R.: Interactive albedo editing in path-traced volumetric materials. *ACM Transactions on Graphics 32*, 2 (2013), 11:1–11:11. 3, 4, 19

[IDN12] IWASAKI K., DOBASHI Y., NISHITA T.: Interactive bi-scale editing of highly glossy materials. *ACM Transactions on Graphics 31*, 6 (2012), 144:1–144:7. 8

[Jak10] JAKOB W.: Mitsuba renderer, 2010. http://www.mitsuba-renderer.org. 12

[JAM*10] JAKOB W., ARBREE A., MOON J. T., BALA K., MARSCHNER S.: A radiative transfer framework for rendering materials with anisotropic structure. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 29*, 4 (2010), 53:1–53:13. 7

[JC98] JENSEN H. W., CHRISTENSEN P. H.: Efficient simulation of light transport in scenes with participating media using photon maps. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 1998), ACM, pp. 311–320. 14

[Jen01] JENSEN H. W.: *Realistic Image Synthesis Using Photon Mapping*. A. K. Peters, Ltd, Natick, MA, USA, 2001. 14

[JMB*14] JARABO A., MASIA B., BOUSSEAU A., PELLACINI F., GUTIERREZ D.: How do people edit light fields? *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 33*, 4 (2014). 9

[JMLH01] JENSEN H. W., MARSCHNER S. R., LEVOY M., HANRAHAN P.: A practical model for subsurface light transport. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 2001), ACM, pp. 511–518. 7

[JNSJ11] JAROSZ W., NOWROUZEZAHRAI D., SADEGHI I., JENSEN H. W.: A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 30*, 1 (2011), 5:1–5:19. 6, 14

[JNT*11] JAROSZ W., NOWROUZEZAHRAI D., THOMAS R., SLOAN P.-P., ZWICKER M.: Progressive photon beams. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 30*, 6 (2011), 181:1–181:12. 14

[JZJ08] JAROSZ W., ZWICKER M., JENSEN H. W.: The beam radiance estimate for volumetric photon mapping. *Computer Graphics Forum (Proceedings of Eurographics) 27*, 2 (2008), 557–566. 14

[Kaj86] KAJIYA J. T.: The rendering equation. *Computer Graphics (Proc. SIGGRAPH)* (1986), 143–150. 2, 14

[KAMJ05] KRISTENSEN A. W., AKENINE-MÖLLER T., JENSEN H. W.: Precomputed local radiance transfer for real-time lighting design. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 24*, 3 (2005), 1208–1215. 3

[KBD07] KAUTZ J., BOULOS S., DURAND F.: Interactive editing and modeling of bidirectional texture functions. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 26*, 3 (2007), 53:1–53:10. 8, 19

[KCWI13] KYPRIANIDIS J. E., COLLOMOSSE J., WANG T., ISENBERG T.: State of the 'art': A taxonomy of artistic stylization techniques for images and video. *IEEE Transactions on Visualization and Computer Graphics 19* (2013), 866–885. 11

[Kel97] KELLER A.: Instant radiosity. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 1997), ACM, pp. 49–56. 14

[KGKC13] KŘIVÁNEK J., GEORGIEV I., KAPLANYAN A., CANADA J.: Recent advances in light transport simulation: Theory and practice. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 2013), ACM. 14

[KISE13] KLEHM O., IHRKE I., SEIDEL H.-P., EISEMANN E.:. In *Proceedings of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 2013), ACM, pp. 161–168. 4, 19

[KP09] KERR W. B., PELLACINI F.: Toward evaluating lighting design interface paradigms for novice users. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 28*, 3 (2009), 26:1–26:9. 4, 9

[KP10] KERR W. B., PELLACINI F.: Toward evaluating material design interface paradigms for novice users. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 29*, 4 (2010), 35:1–35:10. 8, 9

[KPC93] KAWAI J. K., PAINTER J. S., COHEN M. F.: Radioptimization: Goal based rendering. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 1993), ACM, pp. 147–154. 6

[KPD10] KERR W. B., PELLACINI F., DENNING J.: Bendy-Lights: Artistic control of direct illumination by curving light rays. *Computer Graphics Forum (Proceedings of Eurographics Symposium on Rendering) 29*, 4 (2010), 1467–1459. 4, 5, 10, 19

[KRFB06] KHAN E. A., REINHARD E., FLEMING R. W., BULTHOFF H. H.: Image-based material editing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 25*, 3 (2006), 654–663. 8, 19

[KRG*14] KARLÍK O., RŮŽIČKA M., GASSENBAUER V., PELLACINI F., KŘIVÁNEK J.:. *IEEE Transactions on Visualization and Computer Graphics 20* (2014). 9

[KSS02] KAUTZ J., SLOAN P.-P., SNYDER J.: Fast, arbitrary BRDF shading for low-frequency lighting using spherical harmonics. In *Proceedings of Eurographics Workshop on Rendering* (2002), Eurographics Association, pp. 291–296. 3

[KZ11] KNAUS C., ZWICKER M.: Progressive photon mapping: A probabilistic approach. *ACM Transactions on Graphics 30*, 3 (2011), 25:1–25:13. 14

[LAM*11] LOOS B. J., ANTANI L., MITCHELL K., NOWROUZEZAHRAI D., JAROSZ W., SLOAN P.-P.: Modular radiance transfer. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 30*, 6 (2011), 178:1–178:10. 4

[LW93] LAFORTUNE E. P., WILLEMS Y. D.: Bi-directional path tracing. In *Compugraphics '93* (1993), pp. 145–153. 14

[LZT*08] LEHTINEN J., ZWICKER M., TURQUIN E., KONTKANEN J., DURAND F., SILLION F. X., AILA T.: A meshless hierarchical representation for light transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 27*, 3 (2008), 37:1–37:9. 3

[MHM*13] MCAULEY S., HILL S., MARTINEZ A., VILLEMIN R., PETTINEO M., LAZAROV D., NEUBELT D., KARIS B., HERY C., HOFFMAN N., ZAP ANDERSSON H.: Physically based shading in theory and practice. In *ACM SIGGRAPH Courses* (New York, NY, USA, 2013), ACM, pp. 22:1–22:8. 10, 11, 13

[MIW13] MATTAUSCH O., IGARASHI T., WIMMER M.: Freeform shadow boundary editing. *Computer Graphics Forum (Proceedings of Eurographics) 32*, 2 (2013), 175–184. 6

[MnES*11] MUÑOZ A., ECHEVARRIA J. I., SERON F. J., LOPEZ-MORENO J., GLENCROSS M., GUTIERREZ D.: Bssrdf estimation from single images. *Computer Graphics Forum (Proceedings of Eurographics) 30*, 2 (2011), 445–464. 8

[NJS*11] NOWROUZEZAHRAI D., JOHNSON J., SELLE A., LACEWELL D., KASCHALK M., JAROSZ W.: A programmable system for artistic volumetric lighting. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 30*, 4 (2011), 29:1–29:8. 4, 5, 6, 7, 8, 10, 19

[NNDJ12a] NOVÁK J., NOWROUZEZAHRAI D., DACHSBACHER C., JAROSZ W.: Progressive virtual beam lights. *Computer Graphics Forum (Proceedings of Eurographics Symposium on Rendering) 31*, 4 (2012). 14

[NNDJ12b] NOVÁK J., NOWROUZEZAHRAI D., DACHSBACHER C., JAROSZ W.: Virtual ray lights for rendering scenes with participating media. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 31*, 4 (2012). 14

[NRH03] NG R., RAMAMOORTHI R., HANRAHAN P.: All-frequency shadows using non-linear wavelet lighting approximation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 22*, 3 (2003), 376–381. 3

[NRH04] NG R., RAMAMOORTHI R., HANRAHAN P.: Triple product wavelet integrals for all-frequency relighting. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 23*, 3 (2004), 477–487. 3

[NSD94] NIMEROFF J. S., SIMONCELLI E., DORSEY J.: Efficient re-rendering of naturally illuminated environments. In *Proceedings of Eurographics Workshop on Rendering* (1994), Eurographics Association, pp. 359–373. 3

[Obe10] OBERT J.: *Real-time cinematic design of visual aspects in computer generated images*. PhD thesis, School of Electrical Engineering and Computer Science, University of Central Florida, 2010. 6

[OKKP12] OU J., KARLÍK O., KŘIVÁNEK J., PELLACINI F.: Toward evaluating progressive rendering methods in appearance design tasks. *IEEE Computer Graphics and Applications PP*, 99 (2012), 1–1. 9

[OKP*08] OBERT J., KŘIVÁNEK J., PELLACINI F., SÝKORA D., PATTANAIK S.: iCheat, a representation for artistic control of indirect cinematic lighting. *Computer Graphics Forum (Proceedings of Eurographics Symposium on Rendering) 27*, 4 (2008), 1217–1223. 6, 7, 19

[OPP10] OBERT J., PELLACINI F., PATTANAIK S.: Visual editing for all frequency shadow design. *Computer Graphics Forum (Proceedings of Eurographics Symposium on Rendering) 29*, 4 (2010), 1441–1449. 4, 5, 6, 19

[PBMF07] PELLACINI F., BATTAGLIA F., MORLEY R. K., FINKELSTEIN A.: Lighting with paint. *ACM Transactions on Graphics 26*, 2 (2007), 9:1–9:14. 6, 19

[Pel10] PELLACINI F.: envyLight, an interface for editing natural illumination. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 29*, 4 (2010), 34:1–34:8. 1, 4, 6, 19

[PF92] POULIN P., FOURNIER A.: Lights from highlights and shadows. In *Proceedings of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 1992), ACM, pp. 31–38. 4, 6, 19

[PL07] PELLACINI F., LAWRENCE J.: AppWand: Editing measured materials using appearance-driven optimization. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 26*, 3 (2007), 54:1–54:9. 4, 6, 19

[PTG02] PELLACINI F., TOLE P., GREENBERG D. P.: A user interface for interactive cinematic shadow design. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 21*, 3 (2002), 563–566. 2, 4, 6, 19

[PVL*05] PELLACINI F., VIDIMČE K., LEFOHN A., MOHR A., LEONE M., WARREN J.: Lpics: a hybrid hardware-accelerated relighting engine for computer cinematography. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 24*, 3 (2005), 464–470. 3, 5, 10, 13, 19

[Ram09] RAMAMOORTHI R.: Precomputation-based rendering. *Foundations and Trends in Computer Graphics and Vision 3*, 4 (2009), 281–369. 3

[RDGK12] RITSCHEL T., DACHSBACHER C., GROSCH T., KAUTZ J.: The state of the art in interactive global illumination. *Computer Graphics Forum 31*, 1 (2012), 160–188. 9

[RKKS*07] RAGAN-KELLEY J., KILPATRICK C., SMITH B. W., EPPS D., GREEN P., HERY C., DURAND F.: The lightspeed automatic interactive lighting preview system. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 26*, 3 (2007), 25:1–25:11. 3, 5, 10, 13, 19

[RKRD12] REINER T., KAPLANYAN A., REINHARD M., DACHSBACHER C.: Selective inspection and interactive visualization of light transport in virtual scenes. *Computer Graphics Forum (Proceedings of Eurographics) 31*, 2 (2012), 711–718. 5, 10, 14

[ROTS09] RITSCHEL T., OKABE M., THORMÄHLEN T., SEIDEL H.-P.: Interactive reflection editing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 28*, 5 (2009), 129:1–129:7. 5, 6, 19

[RTD*10] RITSCHEL T., THORMÄHLEN T., DACHSBACHER C., KAUTZ J., SEIDEL H.-P.: Interactive on-surface signal deformation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 29*, 4 (2010), 36:1–36:8. 4, 5, 19

[SDS*93] SCHOENEMAN C., DORSEY J., SMITS B., ARVO J., GREENBERG D.: Painting with light. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 1993), ACM, pp. 143–146. 6

[SKS02] SLOAN P.-P., KAUTZ J., SNYDER J.: Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 21*, 3 (2002), 527–536. 3

[SM06] SUN W., MUKHERJEE A.: Generalized wavelet product integral for rendering dynamic glossy objects. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 25*, 3 (2006), 955–966. 3

[SNM*13] SCHMIDT T.-W., NOVÁK J., MENG J., KAPLANYAN A. S., REINER T., NOWROUZEZAHRAI D., DACHSBACHER C.: Path-space manipulation of physically-based light transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 32*, 4 (2013), 129:1–129:8. 1, 4, 5, 10, 13, 19

[SPJT10] SADEGHI I., PRITCHETT H., JENSEN H. W., TAMSTORF R.: An artist friendly hair shading system. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 29*, 4 (2010), 56:1–56:10. 7, 8, 19

[SS89] SÉQUIN C. H., SMYRL E. K.: Parameterized ray tracing. In *Computer Graphics (Proc. SIGGRAPH)* (New York, NY, USA, 1989), ACM, pp. 307–314. 3

[ST90] SAITO T., TAKAHASHI T.: Comprehensible rendering of 3-d shapes. In *Proceedings of ACM SIGGRAPH* (New York, NY, USA, 1990), ACM, pp. 197–206. 3

[STPP09] SONG Y., TONG X., PELLACINI F., PEERS P.: SubEdit: A representation for editing measured heterogenous subsurface scattering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 28*, 3 (2009), 31:1–31:10. 1, 3, 7, 19

[SZC*07] SUN X., ZHOU K., CHEN Y., LIN S., SHI J., GUO B.: Interactive relighting with dynamic BRDFs. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 26*, 3 (2007), 27:1–27:10. 3, 5, 8, 19

[TABI07] TODO H., ANJYO K.-I., BAXTER W., IGARASHI T.: Locally controllable stylized shading. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 26*, 3 (2007), 17:1–17:7. 11

[TL04] TABELLION E., LAMORLETTE A.: An approximate global illumination system for computer generated films. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 23*, 3 (2004), 469–476. 5

[Vea98] VEACH E.: *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University, 1998. 3

[VG97] VEACH E., GUIBAS L. J.: Metropolis light transport. In *Proceedings of ACM SIGGRAPH* (1997), pp. 65–76. 14

[VLD07] VANGORP P., LAURIJSSEN J., DUTRÉ P.: The influence of shape on the perception of material reflectance. *ACM Transactions on Graphics 26*, 3 (2007). 7

[WABG06] WALTER B., ARBREE A., BALA K., GREENBERG D. P.: Multidimensional lightcuts. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 25*, 3 (2006), 1081–1088. 14

[WDR11] WU H., DORSEY J., RUSHMEIER H.: Physically-based interactive bi-scale material design. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 30*, 6 (2011), 145:1–145:10. 8, 19

[WDR13] WU H., DORSEY J., RUSHMEIER H.: Inverse bi-scale material design. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 32*, 6 (2013), 163:1–163:10. 8

[WFA*05] WALTER B., FERNANDEZ S., ARBREE A., BALA K., DONIKIAN M., GREENBERG D. P.: Lightcuts: a scalable approach to illumination. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 24*, 3 (2005), 1098–1107. 14

[WKB12] WALTER B., KHUNGURN P., BALA K.: Bidirectional lightcuts. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 31*, 4 (2012), 59:1–59:11. 14

[WMLT07] WALTER B., MARSCHNER S. R., LI H., TORRANCE K. E.: Microfacet models for refraction through rough surfaces. In *Proceedings of Eurographics Symposium on Rendering* (2007), Eurographics Association, pp. 195–206. 7

[WRG*09] WANG J., REN P., GONG M., SNYDER J., GUO B.: All-frequency rendering of dynamic, spatially-varying reflectance. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 28*, 5 (2009), 133:1–133:10. 7

[WTL04] WANG R., TRAN J., LUEBKE D.: All-frequency relighting of non-diffuse objects using separable BRDF approximation. In *Proceedings of Eurographics Workshop on Rendering* (2004), Eurographics Association, pp. 345–354. 3

[ZHRB13] ZHAO S., HAŠAN M., RAMAMOORTHI R., BALA K.: Modular flux transfer: efficient rendering of high-resolution volumes with repeated structures. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 32*, 4 (2013), 131:1–131:12. 3

Table 1: *Overview of different techniques, grouped by primary goal of edits (lighting or material or combined appearance), as well as complexity of scene description (surface graphics only or participating media). For the interaction columns:* paradigm *refers to the kind of interface provided, according to Sect. 3;* scope *states whether the editing/interaction has only local effects or also includes subsequent global effects;* UI *describes what type of user interface is provided (parameters refers to traditional parameter tweaking). The* manipulation *column states which part of the scene description is being modified. Abbreviations used: Global Illumination (GI); Non-Photorealistic Rendering (NPR); Image-Based Lighting (IBL); Bidirectional Reflectance Distribution Function (BRDF); Spatially Varying BRDF (SVBRDF) = Bidirectional Texture Function (BTF); Temporally & Spatially Varying BRDF (TSVBRDF); Bidirectional Scattering Surface Reflectance Distribution Function (BSSRDF).*

		Interaction			Manipulation
Class/Method		Paradigm	Scope	UI	
Lighting					
Lights from Highlights	[PF92]	indirect	local	click & drag	direct lighting
Lighting Controls	[Bar97]	direct	local	parameters	direct lighting
Interactive Shadow Design	[PTG02]	indirect	local	click & drag; constraints	direct lighting
Lpics	[PVL*05]	direct	global	parameters	cinematic relighting
Direct-to-indirect Transfer	[HPB06]	direct	global	parameters	cinematic relighting
Lightspeed	[RKKS*07]	direct	global	parameters	cinematic relighting
Dynamic BRDF Relighting	[SZC*07]	direct	global	parameters	cinematic relighting
Goal Based Lighting Design	[CdSF99]	goal-based	global	place light sources	surface GI
Lighting with Paint	[PBMF07]	goal-based	global	painting	surface (NPR; GI)
BendyLights	[KPD10]	direct	global	manipulators	surface GI
HDR Painting	[CRH07]	direct	local	painting	IBL
All-Frequency Shadow Design	[OPP10]	indirect	local	click & drag deformation	IBL shadows
envyLight	[Pel10]	indirect	global	painting; parameters	IBL
Material					
Real-time BRDF editing	[BAOR06]	direct	local	parameters; curve editing	BRDF
BRDF-Shop	[CPK06]	goal-based	local	painting	BRDF
Interactive BTF Editing	[KBD07]	direct	local	drill-down; curve editing	SVBRDF
AppWand	[PL07]	goal-based	global	constraint painting	TSVBRDF
Polynomial BRDF	[BAERD08]	direct	global	parameters	BRDF
AppGen	[DTPG11]	goal-based	global	component sketching	SVBRDF
Bi-scale Material Design	[WDR11]	direct	global	parameters; visualization	BRDF
SubEdit	[STPP09]	direct	global	masking; selection	SVBSSRDF
AppWarp	[ATDP11]	goal-based	global	template sketching	B(SS)RDF
Interactive Albedo Editing	[HR13]	direct	global	painting	heterogeneous media
Appearance (Lighting & Material)					
Image-Based Material Editing	[KRFB06]	direct	local	specify matte & material	image (photograph)
AppProp	[AP08]	goal-based	global	painting	image (photograph)
iCheat	[OKP*08]	indirect	global	painting; labeling	surface light transport
Path-Space Manipulation	[SNM*13]	direct	global	filters; manipulators	surface light transport
Reflection Editing	[ROTS09]	indirect	local	click & drag	reflected light
On-Surface Signal Deformation	[RTD*10]	direct	local	constraints; click & drag	on-surface signal
Artist-Friendly Hair Shading	[SPJT10]	direct	local	falloff curves; parameters	hair scattering
Artistic Beams	[NJS*11]	direct	local	spatial curves; shaders	heterogeneous media
Volume Stylizer	[KISE13]	indirect	global	painting	heterogeneous media