EUROGRAPHICS 2019 / P. Alliez and F. Pellacini
(Guest Editors)

Volume 38 (2019), Number 2

Exploratory Stage Lighting Design using Visual Objectives

Evan Shimizu[1], Sylvain Paris[2], Matt Fisher[2], Ersin Yumer[2], and Kayvon Fatahalian[3]

[1]Carnegie Mellon University
[2]Adobe Research
[3]Stanford University

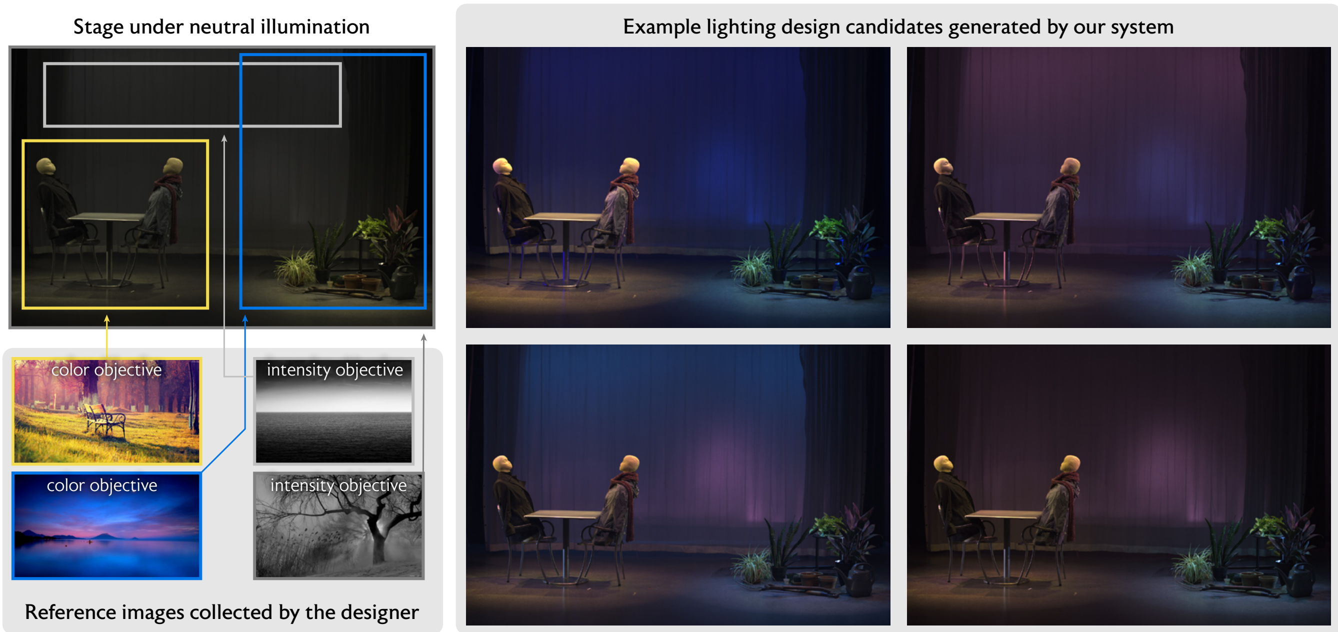


Figure 1: *We introduce a system that aids exploratory theatrical lighting design. Designers abstractly express intent in the form of reference images, and select aspects of the images (e.g., color or intensity) to apply to regions of the stage. From these visual objectives the system generates a gallery of design candidates. Designers can explore and refine lighting designs by adding or removing visual objectives.*

Abstract

Lighting is a critical element of theater. A lighting designer is responsible for drawing the audience's attention to a specific part of the stage, setting time of day, creating a mood, and conveying emotions. Designers often begin the lighting design process by collecting reference visual imagery that captures different aspects of their artistic intent. Then, they experiment with various lighting options to determine which ideas work best on stage. However, modern stages contain tens to hundreds of lights, and setting each light source's parameters individually to realize an idea is both tedious and requires expert skill. In this paper, we describe an exploratory lighting design tool based on feedback from professional designers. The system extracts abstract visual objectives from reference imagery and applies them to target regions of the stage. Our system can rapidly generate plausible design candidates that embody the visual objectives through a Gibbs sampling method, and present them as a design gallery for rapid exploration and iterative refinement. We demonstrate that the resulting system allows lighting designers of all skill levels to quickly create and communicate complex designs, even for scenes containing many color-changing lights.

CCS Concepts
• ***Computing methodologies*** → ***Graphics systems and interfaces;*** • ***Applied computing*** → *Performing arts;*

Keywords: lighting design, lighting, stage lighting, exploratory design, visual objectives

1. Introduction

Lighting design is a critical part of creating compelling imagery. Illumination provides cues about time of day, environment (e.g., the blue sheen of moonlight, the uniformly gray look of a rainy day), and mood or emotion (e.g., a mix of bright colors evokes a festive scene, purples and pinks can look romantic). In this paper, we investigate computational tools that support real-world, theatrical lighting design. Surprisingly, although lighting is a fundamental part of modern stage productions, tools available to theatrical lighting designers remain in a primitive state. Lighting designers create designs featuring tens to hundreds of colored light sources, on tight time budgets, using basic slider-per-light interfaces and limited pre-visualization support.

Although many efforts have explored computational techniques for crafting or altering illumination of virtual 3D scenes or digital images, in discussions with professional lighting designers it became apparent that theatrical scenarios present different challenges and workflows compared to virtual lighting. In theatrical lighting, designs must be physically realizable on a stage, preventing many powerful editing paradigms (e.g., compositing) that are possible when manipulating lighting in photos. In contrast to portrait or product photography, where positioning and choosing fixtures is a key part of the design process, stage lighting often is constrained to a fixed (or limited set) of light positions and angles dictated by the physical structures present in the theater space. Most notably, the early stage of theatrical lighting design is an abstract and exploratory process. As one designer told us, designers seek to establish creative illumination environments that often "are not like what you see in the real world." Theatrical lighting evokes a sense of place and mood with exaggerated colors and varied lighting angles. To create these effects, designers often do visual research: they collect images to reference (of abstract artwork, dramatic photographs, etc.) as they try to combine elements from these images on stage to achieve a desired look, tone, or feel.

Based on collaboration with professional lighting designers, we created a system that supports the exploratory aspect of theatrical lighting design. Exploratory design is the process of rapidly trying out different design possibilities and assessing their look on stage. The system allows a designer to specify *visual objectives* that capture the visual feel (color palette, intensity, contrast) of provided reference images. The system then employs a Gibbs-sampling [CG92] based approach to interactively generate design candidates that embody these objectives and adhere to common theatrical lighting design principles. Using an interface inspired by design galleries [MAB*97], the designer can mix-and-match aspects of their visual research to rapidly explore different design choices. The resulting designs are physically realizable, and can be transferred to a real stage.

While many of the techniques employed in the system (image-based relighting, design galleries, sampling-based design generation) are inspired by prior work in computer graphics, we contribute a combination of these ideas that is specifically targeted at the requirements of theatrical lighting design workflows. In user studies, we observe that lighting designers find the system useful for rapidly creating good starting points for complex designs and for communicating these designs to other designers or directors.

2. Background

The difficulty of creating compelling scene illumination by directly manipulating low-level light parameters (e.g., position, color, intensity) has motivated many explorations of more intuitive interfaces for lighting design. Some systems give the designer direct control over key lighting features, such as the placement of highlights or shadows [PF92, PTG02]. Others offer painting-based interfaces where an artist directly specifies target pixel values [SDS*93, ADW04, PBMF07]. Both lighting feature manipulation and paint-based systems, as well as other goal-based systems (e.g., ensure a region of the scene is adequately lit [KPC93, SW14, KP09]) seek to optimize low-level lighting parameters to meet objectives that fully define the desired output at specific locations, e.g., "put a shadow here" or "add a highlight there". In comparison, we seek to help practitioners set the overall tone and feel of a scene via image examples.

We extract statistical properties of the images in order to reduce the solution space, but these statistics do not fully characterize the desired output. Some systems are able to estimate lighting direction from a single image [LMGH*13], however the designer's reference images are not guaranteed to contain realistic lighting. As a result, our aim is to help designers explore viable solutions, not attempt specific goal-based optimization.

Our approach is inspired by Design Galleries and similar work [MAB*97, SSCO09] and similarly replaces manual parameter tweaking by visual exploration. While Design Galleries demonstrates lighting design on simple 3D scenes illuminated by a few single-color light sources, we aim to work with real-world stages with more complex content and more full-color lights, on the order of tens or hundreds of sources. Sampling the entire solution space [MAB*97] is intractable, so our system allows designers to direct exploration and express intent in the form of visual objectives that illustrate properties of the lighting design they have in mind. Our system is also related to the gallery-based website creation tool by Lee et al. [LSK*10], however our system generates new designs based on input from a different domain (images), rather than examples of existing lighting designs.

By combining elements of different photographs of the same scene, compositing-based methods [ALK*03, ADA*04, BPB13] provide the freedom to create lighting conditions not possible in the real world. Unfortunately, this flexibility prevents their use in theatrical scenarios since the output must be realizable on a physical stage. Therefore, we use compositing for visualization and limit it to physically accurate linear blending [PVL*05]. The fast real-time rendering techniques in [DAG95] could be implemented in our system to provide better support for animated effects, however this is not the focus of the work.

Our system samples the space of lighting designs to synthesize suggestions that have similar statistics to example images provided by the user. Sampling-based content creation has seen recent success in many areas of computer graphics, with applications in color suggestion [LRFH13], 3D modeling [CKGF13, YAMK15], and generating 3D scene layouts [FRS*12]). While prior work focused on the task of offline learning from large data collections, interactive exploration requires immediate generation of plausible samples based on a single or small number of example images.

Since the early work of Dorsey et al. [DSG91] few computer graphics researchers have targeted work specifically at the needs of theatrical lighting. While growing complexity of stage lighting configurations (including the introduction of color changing LED lighting) has led to increased interest in software visualization and authoring tools, leading commercial systems remain based on slider-per-light control interfaces and visualization tools that lack photorealistic rendering support [Lig16, ML16, Vec16, CAS16]. These tools are best suited for addressing timing and light beam animation concerns of stage spectacular or concert lighting, not assisting a designer with subtle color and intensity choices needed to evoke tone or mood.

3. Observations and Design Principles

The first phase of our project involved extensive discussions with theatrical lighting design professionals about their creative process. We describe this process below, then identify the key principles that guide the design of our system.

At the beginning of a lighting design project, it is common for designers to collect images that represent potential design ideas. For instance, a painting may be used to specify a color palette and a movie still for light intensity and contrast. These *reference images* are used throughout the design process to communicate lighting design ideas to the rest of the design team (e.g., other designers or a director). Examples of reference images are shown in Figures 2-right, 3-left, and 10-bottom.

Using the gathered reference images, designers then try to visualize their ideas on the actual stage to evaluate the color, contrast, and mood created by the lighting configuration. However, due to the lack of high-quality visualization tools, most design iteration occurs only after access to the physical stage is available. This forces designers to rapidly iterate over their designs under tight time constraints, accepting minor inaccuracies in the design until the entire show has been roughed in. During this process, designers first work to establish the overall look of the scene, adjusting the global intensity distribution and color palette. Once the general look has been set, the designer may continue to adjust individual lights as needed to refine the design.

Lighting designers deal with tens or hundreds of light sources, and significant experience is required to organize these devices in a way that allows for them to realize their ideas on stage. The majority of light sources used in a theatrical setting are static and cannot change where they point, or their beam properties after being placed. The primary way of dealing with this complexity is to organize lights with the same "function," such as "front light" or "side light" into *light groups*, and use the same setting for all lights in a group. Designers often call light groups "systems" but we avoid this term in the paper for clarity. Although individual lights may be adjusted during later refinement, light group granularity manipulation is an effective way to achieve results quickly during the exploration phase. Light groups are created by the designer as they place lighting fixtures in the available locations around the stage.

In order to work with current lighting control interfaces, lighting designers must specify brightness in terms of light intensity, which is the percentage of the total luminous output of a lighting device. When specifying the intensity of a scene, designers select some light groups, called *key lights*, to highlight the main elements of the scene. Key lights are the brightest light groups in the scene. The remaining groups, called *fill lights*, are used for illuminating the rest of the stage and determine scene contrast. (Fill lights "fill in" the shadows created by key lights.) Designers consider the contrast of a design to be the intensity ratio between these two groups of lights (how much brighter the key light regions are compared to the rest of the stage).

From this understanding of the lighting design process, we determined that a system that addresses the exploratory lighting design needs of designers should embody the following principles:

Designer-Controlled Exploration. Designers do not wish to cede control of design decisions to the system. Instead, they wish to quickly generate, explore, and visualize multiple design possibilities in order to ultimately make good global design decisions.

Communicate Visual Intent via Images Designers should be able to rapidly communicate desired visual properties of lighting to the system using images, much like how they communicate ideas to other designers and directors.

Partial Design Specification In cases where a designer has specific properties in mind, generated designs should adhere to those constraints. For example, a designer may only wish the system to generate changes to a specific stage region, leaving the rest of the stage unchanged.

Adhere to Theatrical Design Principles The system should model theatrical lighting design principles so that generated designs are plausible and visually attractive. For example, it should understand relationships between key and fill lights and respect the designer-provided light groups.

Speed To enable interactive exploration, speed is a paramount requirement in every aspect of the system (analyzing designer intent, suggesting designs, visualizing results). Techniques that fail to provide instantaneous feedback inhibit the exploratory process, and are not viable for use.

4. Method

In this section, we describe a system for stage lighting design that is based on the observations, principles, and conversations with experts described in Section 3. We assume that the position and angle of scene light sources has already been specified by a lighting designer (a common setup for theatrical lighting design scenes), so the lighting design task is to assign a color and intensity to each scene light source. We will refer to an assignment of colors and intensities to all lights as a *lighting configuration*. We first provide an overview of how a designer works with the interface to specify design ideas and refine system-generated design candidates that embody these concepts. We then describe the key algorithmic details of the system's implementation.

4.1. System Overview

Figure 2 shows a screenshot of out interface. At the beginning of a work session, the designer imports reference images that embody aspects of the tone, feel, and visual appearance they seek to

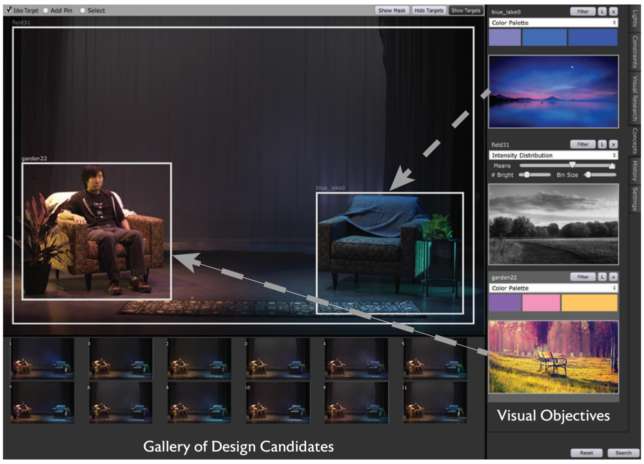


Figure 2: ***User interface.*** *The visual concepts interface takes the current light parameters, and a list of visual objectives (right panel), and generates design candidates that satisfy these objectives (bottom panel). The user can localize each objective to a different region of the stage (boxes on top left panel). In this case, the two color objectives apply to the marked boxes, while the intensity objective (greyscale image) applies to the entire stage. The process is iterative; the user can move a candidate to the stage and assign new objectives which will only modify the targeted regions' light color or intensity.*

recreate on stage. We refer to these images, along with a specification of which characteristics of the image the designer finds desirable (color palette, intensity distribution, or both), as *visual objectives*. Figure 2-right shows two visual objectives that suggest desired color palettes, and one that suggests overall scene intensity and contrast.

Given a set of lights on stage, an organization of those lights into groups, and a set of one or more visual objectives, the system generates a set of lighting configurations, which we call *design candidates*, that embody the objectives. When generating design candidates, the system respects standard theatrical lighting design principles (e.g. key and fill lights) to generate a diverse set of designs. Similar candidates are clustered and presented to the user in the form of a design gallery (Figure 2-bottom).

The interface encourages interactive exploration. The designer can select an appealing candidate, replace visual objectives, or add new constraints to refine the design, and iteratively repeat the process. Examples of constraints include specifying a stage region visual objective should be applied to, or specifying what light groups to use as key lights. In Figure 2, the designer targets the color palette from the blue lake reference image to the right side of the stage, and the warm color palette from the outdoor meadow image to the left side of the stage. The middle image (landscape scene) is used to set the lighting intensity of the scene.

The intent of the exploratory process is to establish the overall look and tone of the scene. Once the designer is satisfied with the base look, it is possible to continue to fine-tune individual light parameters with a traditional slider-per-light interface (not shown in Figure 2).

4.2. Intensity Visual Objective

Since the primary goal of the visual concepts system is to aid experimentation with the placement and intensity of color on stage, we provide designers with two types of visual objectives: *intensity* and *color*. Designers express their visual intent by providing a reference image, which may be a painting, abstract pattern, or photograph that inspires them. Since the content (and light sources) of reference images often bear little resemblance to the target stage, it is not reasonable to assume a reference image results from physically accurate light transport that, if inverted, would yield accurate stage lighting parameters. For the same reason, an image-based optimization approach is also not desirable. Instead, the system extracts a model of the visual objective from the image, then generates design candidates by sampling from the extracted model. For simplicity, we first describe our model for the intensity visual objective and how it is used determine light intensities. We delay description of the color visual objective to Section 4.3.

4.2.1. Defining the Intensity Visual Objective

When describing the lighting intensity of a scene, lighting designers consider two factors: the scene's overall brightness, as well as its contrast (brightness differences between key and fill lights). As shown in Figure 4-top, when sampling light source intensities based on image histograms, most design candidates poorly match the average intensity of the visual objective.

Echoing how lighting designers think about brightness in terms of key and fill light intensities, our model of lighting intensity is based on two parameters extracted from the reference image. Most reference images are low dynamic range images, so to approximate lighting intensities, we convert the pixels to the CIELAB color space and treat the L* component as the intensity of the pixel. The visual objective's average intensity, μ_a, is given by the average pixel intensity of the reference image. The average intensity of key lights, μ_k, is estimated as the average intensity of the top 15% of pixels ordered by L value in the image (these pixels are assumed to be illuminated by key lights). Figure 3 provides two examples of intensity parameters.

Although the system assumes that light sources are already positioned (both location and angle), stages contain many lights, so exploration of lighting angle is possible by turning lights from a particular position or direction on or off. Therefore, while design candidates should accurately reflect the average overall intensity and key light intensity of the visual objective, it is also desirable for a sampling scheme to produce high variance in individual light intensities across candidates in order to produce a diverse set of candidates.

We find that an approach that is similar to Gibbs sampling [CG92], where each light intensity is conditioned on prior assigned light intensities, is able to generate design candidates that meet both of these goals. The method naturally extends to incorporate additional design constraints that may be optionally supplied by the designer (see Section 4.4), and can produce large numbers of candidates at interactive rates.

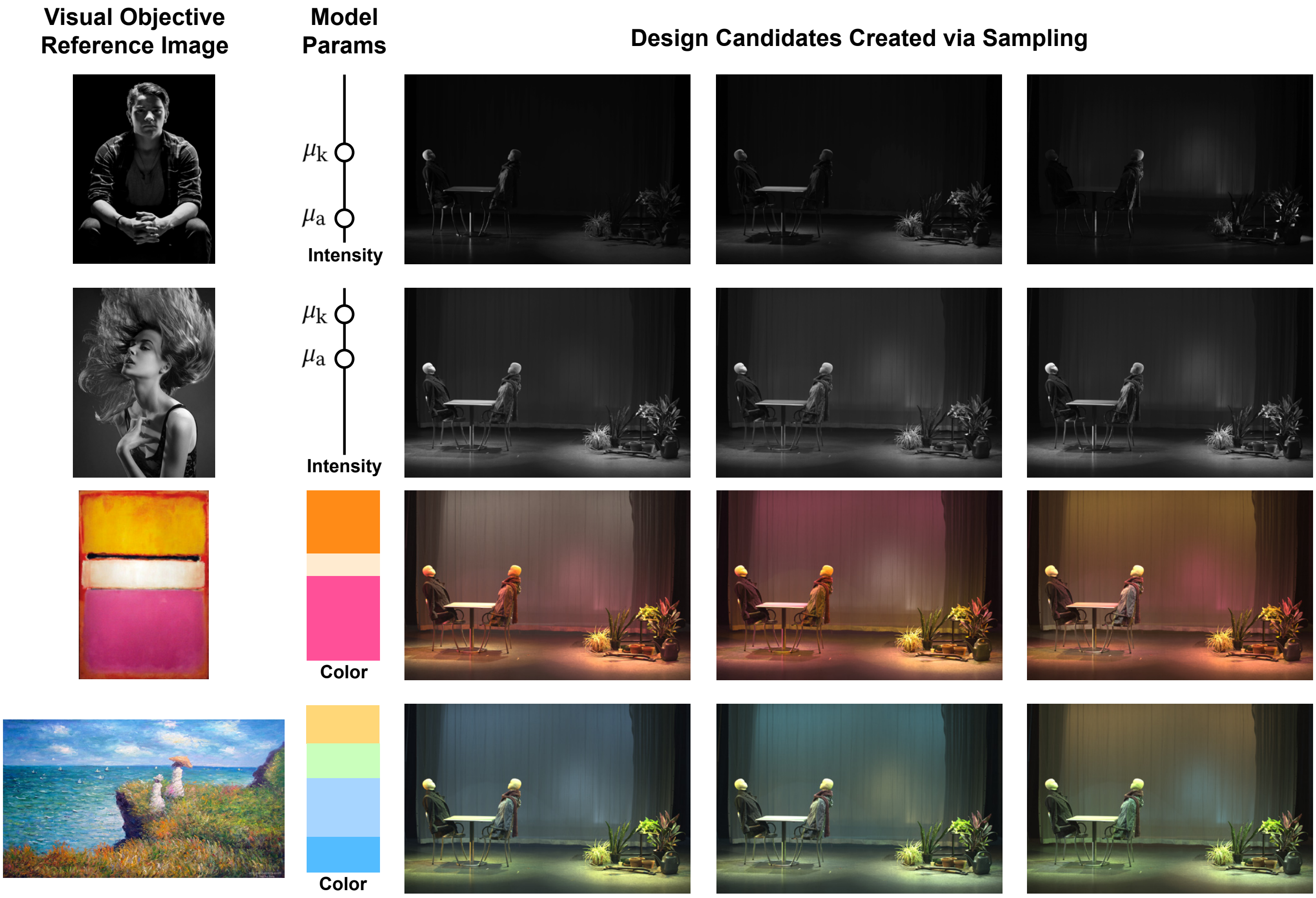


Figure 3: ***Intensity and color visual objectives.*** *In the* Intensity *model, we sample light intensities using a model based on the image's per-pixel intensities. In the* Color *model, we sample light color to match the image's color palette. In these examples, the visual objective is targeted to the entire stage.*

4.2.2. Generating Design Candidates with Gibbs Sampling

For simplicity, we introduce our approach to sampling light intensities assuming that one fourth of a stage's lights are key lights and that all lights on stage have the same maximum intensity. It is common for lighting designs to modify the key to fill light ratio, and we subsequently relax both of these assumptions in Section 4.2.4.

Given a stage with L lights, the sampler randomly selects $L/4$ lights to serve as key lights (uniform distribution). Starting with the key lights in random order, the intensity for the key light is drawn from $\mathcal{N}(\mu_k, \sigma_k)$, where σ_k is a tunable parameter (5% of the available intensity range by default, a threshold used by lighting designers as the just noticeable difference). To match the average light intensity specified by the visual objective (μ_a), the algorithm sets the intensity of the fill lights, in random order, one-by-one, adjusting the distribution for the remaining lights as each step. Specifically, after the j^{th} light intensity has been set, the goal is to shift μ_{j+1} for the $(j+1)^{\text{th}}$ fill light so that, if all remaining sources were sampled from this new distribution, the expected average across all lights is μ_a. That is, we define μ_{j+1} so that:

$$\frac{1}{L}\left(\underbrace{\sum_{i=1}^{L/4+j} x_i}_{\text{lights set so far}} + \underbrace{\left(\frac{3L}{4} - j\right)\mu_{j+1}}_{\text{expected intensity of remaining lights}} \right) = \mu_a \qquad (1)$$

where x_i's represent previously sampled intensities, including the key lights. Rearranging terms yields:

$$\mu_{j+1} = \frac{1}{3/4 - j/L}\left(\mu_a - \frac{1}{L}\sum_{i=1}^{L/4+j} x_i\right) \qquad (2)$$

Sample x_{j+1} is then sampled from $\mathcal{N}(\mu_{j+1}, \sigma_a)$, where σ_a is a tunable parameter (10% of the intensity range by default, fill lights typically vary more than key lights) and determine μ_{j+2} according to the value of x_{j+1}. Intuitively, this formula progressively adjusts the intensity of the light sources until the desired average intensity is reached. Formally, it amounts to sampling the distribution:

$$P(x) = \frac{1}{4}\mathcal{N}(\mu_k, \sigma_k) + \frac{1}{L}\sum_{j=1+L/4}^{L} \mathcal{N}(\mu_j, \sigma_a) \qquad (3)$$

where the μ_j parameters are computed as previously described. This procedure ensures that on average, key light sources have intensity μ_k and all the lights considered together have average intensity μ_a. Unlike a standard Gibbs sampler, our approach is not dependent on prior design candidates, and thus does not require a burn-in period. We provide pseudocode in the supplemental material.

4.2.3. Properties of Gibbs Sampling

Given our decision to represent the intensity objectives in terms of key light and overall intensity, we also considered directly sampling light intensities from a bi-modal distribution featuring a mode for the key lights and a mode for the fill lights. Specifically:

$$P_{\text{BM}}(x) = \frac{1}{4}\mathcal{N}(\mu_k, \sigma_k) + \frac{3}{4}\mathcal{N}(\mu_x, \sigma_a)$$

where μ_x is computed as in Equation 2, but only once for the first $(j+1)^{\text{th}}$ fill light. We found the Gibbs-sampling approach to be preferable for two reasons: it generates design candidates that are both more consistent with the visual objective (matches average intensity and contrast) and exhibit more variation across candidates (encourages exploration).

Consistent Average Intensity Gibbs sampling consistently generates candidates whose average intensity matches the intensity visual objective because the sampled distribution is sequentially adapted so that the expected average μ_a is conditioned on the previous samples' values. If the average intensity of the previous intensity samples is "off-target", the distribution is adjusted so that the expected average intensity of the entire lighting configuration is back "on-target". Formally, since $P_{\text{BM}}(x)$ is an average of independent random variables, the variance of the output is σ_a^2/L. In contrast, the variance in Gibbs sampling comes from only the last sample, and is σ_a^2/L^2.

Greater Per-Light Intensity Variance. While Gibbs sampling reduces variation in overall intensity of design candidates, it increases variance in the intensity values assigned to individual lights. As shown in the middle and bottom rows of Figure 4, this leads to candidates with higher contrast between light intensities and greater diversity in lighting direction. Ignoring the key light sources (since they are treated similarly in Gibbs and bi-modal distribution sampling), the variance in intensity values of any specific fill light source ℓ across design candidates is σ_a^2 for direct bi-modal distribution sampling and $\sigma_a^2 + \text{Var}_c[\mu_{r_c(\ell)}]$ for Gibbs sampling, where $\text{Var}_j[\cdot]$ is the variance operator across candidates, $r_c(\ell)$ the random index assigned to the light ℓ when generating the candidate c, and $\mu_{r_c(\ell)}$ is computed via Equation 2.

4.2.4. Generalizing the Sampling Procedure

The proceeding sections assumed that $1/4$ of the lights in a lighting configuration were key lights. In practice, this results in candidates that are too dark for bright reference images (and vice versa for low intensity images). We address this issue by determining the number of key lights in a lighting configuration from the intensity of the reference image. We compute the proportion b of pixels in the reference image brighter than the mean image intensity, and then perform Gibbs sampling using $L_k = bL/2$ light sources as key lights.

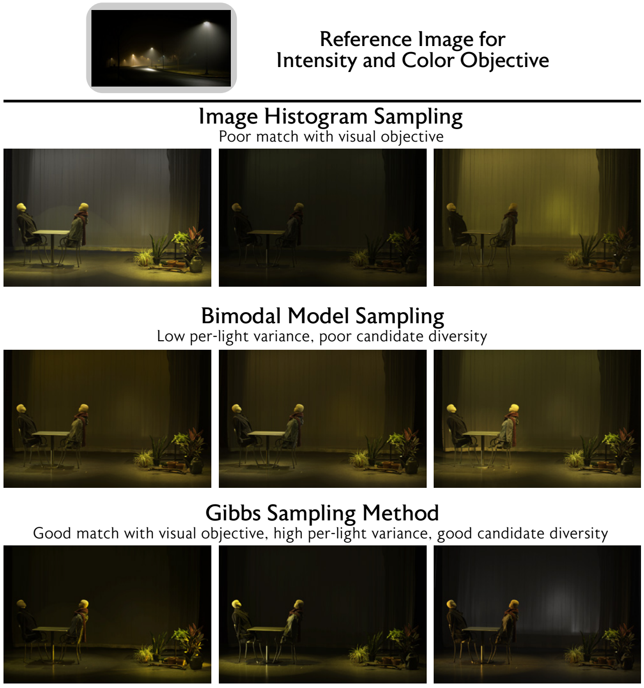


Figure 4: ***Design candidates generated from three sampling techniques.*** *Sampling light group intensities according to the reference image's pixel intensity histogram yields widely varying results that often do not make the visual appearance of the reference (top). Directly sampling from a bi-modal distribution representing key and fill light intensities (middle) results in candidates that lack the diversity and contrast of those produced by the Gibbs sampling technique (bottom).*

This choice ensures that when 50% of the pixels in the reference image are brighter than its mean intensity, one fourth of the light sources are selected as key lights. With this change, Equation 2 becomes:

$$\mu_{j+1} = (L\mu_a - \sum_{i=1}^{L_k+j} x_i)/(L - L_k - j) \qquad (4)$$

and Equation 3:

$$P(x) = \frac{b}{2}\mathcal{N}(\mu_k, \sigma_k) + \frac{1}{L}\sum_{j=1+L_k/L}^{L}\mathcal{N}(\mu_j, \sigma_a) \qquad (5)$$

4.2.5. Normalizing Light Brightness

It is common for stage lights to have different maximum brightnesses (different lumen outputs) and vary significantly in the area of the stage they illuminate. We account for this variation by performing a normalization step at the end of the Gibbs sampling procedure. In practice, we normalize the values produced by Gibbs sampling by the total brightness of each light, estimated from HDR images captured as described in Section 5. This ensures that each light has a similar global effect on the scene.

4.3. Color Visual Objective

We now describe the color visual objective and how it is used to set light colors during sampling. Given a reference image, the system extracts a palette of C RGB colors, as well as per-color weights. The images are typically JPGs and PNGs encoded in the standard sRGB color space, and clustering occurs within this space. Color weights sum to 1, so they represent the fraction of lights that should take on each color in the palette. To obtain a color palette from an image, our implementation performs K-means clustering on RGB image pixels, then extracts cluster means ($K = C$). The fraction of image pixels belonging to each cluster determines cluster weights. Use of more advanced palette extraction techniques, such as that of Lin et al. [LRFH13] is possible, but their runtimes (minutes per image) are not suitable for interactive use and would require preprocessing of reference images. Users also have the option to manually tweak colors if they are unsatisfied with palettes automatically extracted by the system.

During design candidate generation, color assignment occurs after intensity sampling is complete. For each color c, the sampler uniformly selects a light source whose color has not been assigned, and sets its color to c. The process of assigning c to lights continues until the total normalized intensity of all lights with color c has reached the color's weight. Then the process proceeds to the next color in the palette until a color has been assigned to all lights.

Given the algorithm above, all colors but the last are over-represented in the lighting configuration, so the system randomizes color order for each design candidate. This ensures that the under-represented color varies for all candidates, increasing candidate diversity. For real-world control, we rely on existing lighting control consoles to appropriately handle the conversion from sRGB colors to the proper colors for each lighting device.

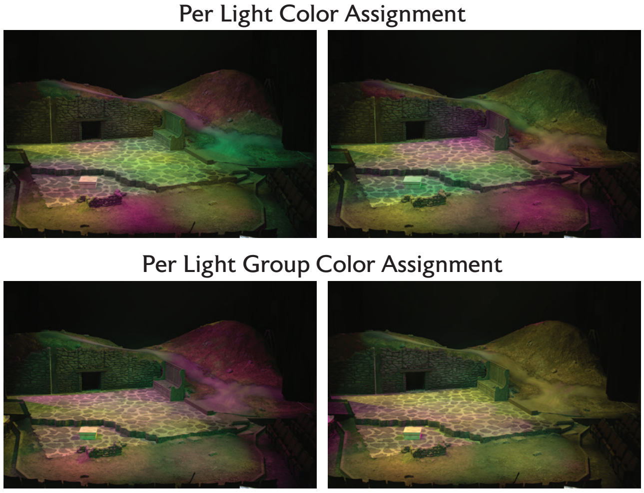


Figure 5: *Assigning colors at light-group granularity (bottom) yields more pleasing designs (note higher spatial coherence) than when unique colors are assigned to individual lights (top). In both cases design candidates are generated from the same color visual objectives.*

4.4. Refining the Candidate Generation

While the proposed system is fundamentally an exploratory design tool, we quickly observed that designers benefited from increased control over the visual appearance of design candidates. In this section, we discuss several extensions to the Gibbs sampling process that enhance designer control.

4.4.1. Targeting Regions of the Stage

Designers not only wish to explore different distributions of light colors and intensities, but also different placements of these visual attributes on stage. Therefore, the system allows designers to target a visual objective to a specific region of the stage simply by drawing bounding boxes on the visualizer output (see boxes in Figure 2). Given this 2D screen region, the system automatically infers the light sources that affect the selected region, and design candidate generation is constrained to only manipulate the intensity and color values for selected lights.

Light selection from a 2D screen region is performed using a collection of heuristics computed on rendered pixel values. For each light ℓ, we classify rendered pixels as ℓ's *bright pixels* (15% of pixels that receive the most light from ℓ) and *highlight pixels* (top 5%). We set these thresholds empirically and use the same values for all experiments. While these values may not be optimal, our user studies indicated that the system performed as expected in the majority of cases with these values. A light source is added to the selection if:

- The light affects a significant part of the selected area. ($>$ 25% of the selection is covered by bright pixels)
- The light's influence is mostly contained within the selected area. ($>$ 50% of the light's bright pixels are contained by the selection)
- The selected area contains highlights caused by the light. ($>$ 5% of the light's highlight pixels are contained by the selection)

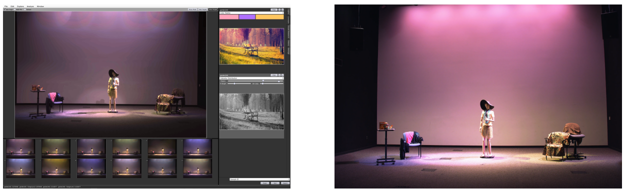
Our approach to inverse light selection is similar in spirit to that of EnvyLight [Pel10], however we select light sources based on 2D screen regions, rather than use 3D surface manipulation to select environment-map pixels. Our experiences indicate that 2D screen-space selection is sufficient to quickly select foreground or background regions of the stage, even without a 3D model of the scene.

4.4.2. Respecting Light Groups

Designers organize lights into light groups to reflect desired spatial and angular coherence of lighting on stage. Although the sampling procedures Sections 4.2 and 4.3 were described terms of individual light sources, it is most common to apply the sampling procedure at the granularity of entire light groups where assignments of intensity and color are made per group, not per light. Figure 5 illustrates the aesthetic benefits of manipulating light groups. Notice that per-light color and intensity assignment (top) results in spatially incoherent illumination across the stage. When a designer targets visual objectives to a stage region, the sampling process is run over the selected subsets of the light groups.

4.4.3. Partial Design Specification

We have found that the Gibbs sampling based approach easily extends to accommodate scenarios where a designer has a clear view



Design Tool Interface   Photograph of Real-World Stage

Figure 6: *Left: Design interface featuring visual objectives and a visualization generated by compositing single-light basis photographs of a stage. Right: a real-life photograph of the same lighting configuration realized on the same stage. In addition to visualizing lighting configurations, the system is also able to directly control the stage lights during the exploratory design process for real-world preview. Note that the right image is missing a spotlight visible in the interface rendering. This is due to a light malfunction that occured between capture and real-world use. See the companion video for additional demonstrations.*

of certain elements of a final design. For example, a designer might choose a particular light group serve as key lights of the scene. This is handled by setting the intensity of these lights first as key lights, and sampling the remaining lights as normal. Because the sampling method is designed to condition future intensity distributions on lighting parameter values that have been previously set, other intensity constraints, such as fixing lights to maintain a given intensity, or lights whose intensity is defined in terms of that of other lights, are also supported by the system.

4.5. Presenting Design Candidates

Given a set of visual objectives, the system uses the algorithms described previously to generate a set of N design candidates. The candidates are grouped into c clusters and presented to the designer as a design gallery ($N = 100$ and $c = 12$ in our system). To ensure instantaneous feedback, the system clusters design candidates in a streaming manner. As each new design candidate is generated, the system renders a visualization of the stage corresponding to the new candidate, and measures the average per-pixel L_2 distance (CIELAB color space) to that of existing design candidates. If the distance to all existing design candidates is sufficiently large, a new cluster containing the candidate is made (until c clusters exist). If the distance to an existing candidate is sufficiently small, the new candidate is rejected and the threshold decreased for all future samples. The decreasing threshold encourages diverse clusters to be found quickly. Once c clusters have been generated, all new candidates that are sufficiently diverse are added to existing clusters. Full details about the clustering process can be found in the supplemental material.

5. Implementation

To provide a viable solution for theatrical design workflows, our system must generate realistic visualizations of a stage at real-time rates. This section describes our image-based visualization solution that meets these requirements.

5.1. Rendering

Our system provides interactive, photorealistic previews using a custom renderer that is based on HDR image composition. The system accepts as input L HDR basis images that each depict the stage illuminated by a single light at full intensity (all images are from the same viewpoint). Similar to prior systems for cinematic relighting [PVL*05], our tool generates visualizations of different lighting configurations using linear combination of these images. The blend weights for each image are determined by the color and intensity of the corresponding light source. A linear tone map, without gamma correction, is applied at the end of the rendering process.

Compositing single-light images into final lighting visualizations is efficient, even for high light count scenes. We perform simple linear tone mapping to display the resulting image. On a quad-core CPU, the renderer generates hundreds of 480×245 thumbnails per second for a scene with 44 lights (more than could feasibly be displayed at once in an on-screen gallery).

Image composition-based visualization has the added benefit that the visualization is agnostic to the source of the HDR basis images. Basis images may be photographs of the actual target stage, a demo scene environment, or high-quality off-line renderings of a virtual stage environment (if physical access to a stage is not possible, or if lights are not positioned prior to the start of lighting design).

5.2. Image Capture

The stage visualizations we display in this paper were created from basis photographs acquired from real-world theater stages. The capture process is automated with a script that turns one light on to full intensity, captures an image stack for HDR image creation, then repeats the process for all lights. We capture RAW images of each light at different exposures (from -3 to +3 EV in 1 EV increments). HDR images saved in the OpenEXR format [ILM14] were created using Adobe Photoshop CC. After HDR processing, light group information was provided by the designer. The capture process took about 20 seconds per light, and HDR processing using 16 megapixel images took approximately 3 minutes per light on a laptop. More powerful computing hardware and higher quality cameras could notably reduce the length of the stage capture process.

We have captured basis images for four full-scale stage scenes. The first is a 44-light theatrical lighting design laboratory maintained by our university (present in all figures unless otherwise noted). The laboratory is used for lighting design courses and by designers to prototype designs prior to deploying them on a large production stage. Mannequins and props are placed in the stage environment to aid with the design process, which is a standard practice in prototyping lab environments. We captured this stage in four different scenic configurations, each time with the same light sources. The second stage is a 190-light stage used for feature theatrical productions (Figure 5). The third stage is a 37-light theatrical lighting design lab (Figure 6) that features all color changing fixtures (LEDs, scrollers, and other color mixing systems). The fourth stage (Figure 8) is a different configuration of the stage shown in

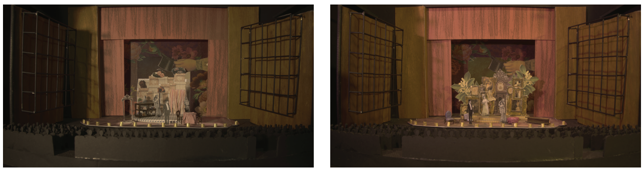


Figure 7: *Two previz lighting configurations for The Matchmaker, visualized on a small-scale (1 m^2) model created by a set designer prior to full-scale sets being built. The model is illuminated by eight lighting directions chosen by the lighting designer.*

Figure 5 with 267 lights. We do not capture multiple scenery configurations of the production stages due to time, however since light positions are fixed, a single set is sufficient for most designers to work with the visualizer.

In addition to full size stages, we have also used our system to capture basis images for a miniature (1 m^2) stage mockup (Figure 7), provided by a scenic designer to visualize a future stage design before building full-scale scenic elements (see Section 6.1.1). Despite its small scale, we captured basis images for the mockup using the same photo capture method as for the full-scale stages, with the position and direction of the lights chosen by the show's designer. We have also used the interface to perform design exercises on virtual lighting stages. In this scenario, we rendered HDR basis images using a photorealistic ray tracer.

5.3. Real-World Lighting Control

The speed at which our system generates design candidates makes it possible to use the interface to directly control the lighting configurations on a real stage. In collaboration with Electronic Theatre Controls (ETC), a major lighting control hardware company, we integrated our system with their ETC Eos lighting control system [Con17]. This integration allows our system to directly control actual stage lights in real time in response to user interaction (Figure 6). Integration with the Eos control system provides a comprehensive interface that supports multiple phases of lighting design, from early design explorations, enabled by the interface described in this paper, to subsequent fine-tuning with industry standard low-level controls. We invite the readers to view the capabilities of this integration in the companion video.

6. Evaluation

Evaluating a creative tool is difficult, as the goal of the tool is to enable users to iteratively work towards a satisfactory design of their own choosing. Ultimately, the true test of our design is whether the visual concepts interface is adopted by practitioners as part of daily lighting design workflows. Since it is infeasible to engineer a robust, fully-featured system for broad community use, we instead conducted three studies to evaluate the effectiveness of the interface at the professional and novice levels.

At the professional level, we conducted a case study, giving our interface to a professional designer working on a production at our university's School of Drama, simulating an industry-level process as closely as possible. We also provided the interface to a panel of lighting design and control experts for a period of two weeks. Professionals are able to analyze their own design process more easily than novices can, and we conducted interviews with them to determine how well our interface integrates into their workflows and supports their design process. For novices, we conduct a study with the Creativity Support Index [CL14], providing a quantitative measure of how satisfied the participants were with our interface compared to a baseline interface. These three studies cover all of the intended users of the visual concepts system, and the results demonstrate the utility of the interface among all participants.

6.1. Professional Evaluation

6.1.1. The Matchmaker: A Case Study

The visual objectives interface was used by a lighting design graduate student to design and program *The Matchmaker*, a main stage show at our university's School of Drama. This student worked in a professional capacity for several years before returning to graduate school. The designer used the interface for both pre-visualization and programming.

Pre-Visualization The lighting designer took a small-scale stage model created by the scenic designer for their show (Figure 7) and used the visual objectives interface to resolve a practical design issue in their production. The designer was having trouble convincing the director to allow him to proceed with his desired lighting design due to different mental pictures of what the design would look like on stage. To resolve this, the designer generated a number of lighting design candidates from his reference images and presented a few of the generated designs to the director. The renderings reduced confusion and made the director "much more comfortable", and also helped the designer to "figure out what I [the designer] like about the research and find new research as well".

Stage Capture The full-size stage for *The Matchmaker* consisted of 267 lights. After the lights were placed on the full stage, we ran an automated capture process to acquire basis HDR imagery in five hours. The interface was integrated with the theater's ETC Eos system, allowing the designer to view the generated designs by controlling lights on the real stage, as well as in the visualizer.

Programming The designer was allocated one work day (eight hours) to program the show according to the production schedule, with the expectation that most of the programming would be complete before on-stage rehearsals began the next day. They were able to explore designs based on their visual research for all four acts of the play within an hour. We only had time to capture scenery of one act, however the designer noted that this was not a problem for him when programming the lights. While exploring, the designer used the interface to create presets for each act, which were saved in the Eos console for later use. They then used the remainder of the day to fine-tune the presets with the Eos console (an instance of a sliders-based interface). The show was programmed within the allocated time, which "would not have been possible without [the visual objectives] interface." An example of a scene created with the visual objectives system is shown in Figure 8.

Figure 8: ***Example Matchmaker Design.*** *An example of a rendering for The Matchmaker and the reference image for the scene. This is the lighting for Act IV shown on the set for Act II (we did not have time to capture every act's scenery). The designs were mainly previewed by controlling lights on the main stage instead of using the visualizer.*

6.1.2. Expert Panel Feedback

To further evaluate the visual objectives interface, we conducted a series of interviews with lighting experts regarding how the exploratory interface affects their lighting design process.

Experimental Setup We interviewed four lighting industry professionals: the graduate student from Section 6.1.1, a community theater designer and lighting control expert, a professional lighting designer, and a lighting control expert with Broadway production experience. Participants were first interviewed about the current state of lighting design and how designers and directors communicate ideas. Participants had knowledge of how our interface worked, but were not able to use it themselves.

After the interview, participants were given the full interface to experiment with for at least two hours at their discretion, after which they were interviewed a second time. The second interview asked participants to consider how the new interface would change the design process and the communication process between director and designer. Note that the interviews were conducted before the designer from Section 6.1.1 designed *The Matchmaker*.

Interviews were transcribed and coded by the authors following standard qualitative evaluation practices [SC97]. The full list of codes is presented in the supplemental material.

Overall Evaluation Overall, the expert users were excited by the possibilities presented by the interface. All participants indicated that our interface makes it easier to communicate abstract lighting design ideas to the director and the rest of the design team, and that the interface would make it easy to experiment with new ideas and set up base looks very quickly. The experts also validated some of the design goals used to build this system, specifically noting that lack of time for iterating on a design is one of the primary limiting factors for creating the ideal design.

Communicating Abstract Ideas In every interview, experts mentioned that the primary challenge with pitching a lighting design to a director is that it is difficult to picture what the show will look like until the artistic team gets into the theater space. As one expert put it "[A lighting designer's] art is how to translate an image into gel (filter) colors and onto a stage. They use very artistic words like vibrant, and muted, or dark; things that try to convey what they're doing" (Expert #2). Failing to accurately convey the design idea to the design team leads to delays and miscommunications similar to those encountered in Section 6.1.1.

Our interface bridges this communication gap by providing a high quality visualization, and a method to quickly turn images into lighting design ideas. The panel agreed that the lighting designs produced by the system "absolutely" (Expert #3) captured the feel of the reference images. One participant strongly preferred the real-world results, saying that the visualization "[did] not do this interface justice" (Expert #1).

The system's speed allowed designers to "take the visual references and translate those relatively quickly into something" (Expert #1). The quality of the generated designs was also good enough to create to create a "really good starting point" (Expert #2) during the tech rehearsal process, when a designer is the most time constrained.

Accelerating Choices Full-color LED fixtures are becoming increasingly common, and designers are now putting off color decisions to the point where "decisions are being made during the technical rehearsal process[, and] there is not enough time during tech rehearsal to actually make all those decisions" (Expert #1). Before LEDs, designers typically had a constrained set of colors available in their gel book (a swatch book for color filters). With the visual objectives interface, we can replace the gel book with a collection of reference images for their favorite color palettes. This "would allow someone to use those reference images and go straight to the stage with those colors" (Expert #2). Color palettes have the added benefit of automatically generating an entire color scheme for the designer, instead of selecting colors one by one.

Helping Novice Designers Participants in the expert study noted that the proposed interface, though motivated by expert design principles, would also "have a huge benefit for novice users who don't understand how you translate a thought in your head onto stage" (Expert #2). Novice users, including high school and community designers, make up a large portion of the lighting controls market. For these users, the ability to quickly and intuitively create compelling lighting designs with little to no tweaking would be very useful.

6.2. Novice and Intermediate User Evaluation

We also evaluated the ability of the visual objectives interface to assist novice and intermediate users. Our preliminary studies demonstrated that obtaining an objective measure of lighting design quality is very difficult, so instead we use the Creativity Support Index (CSI) [CL14], to measure the extent to which our interface supports a user performing a creative lighting design task. The CSI is a weighted average of six self-reported scores measuring the ability

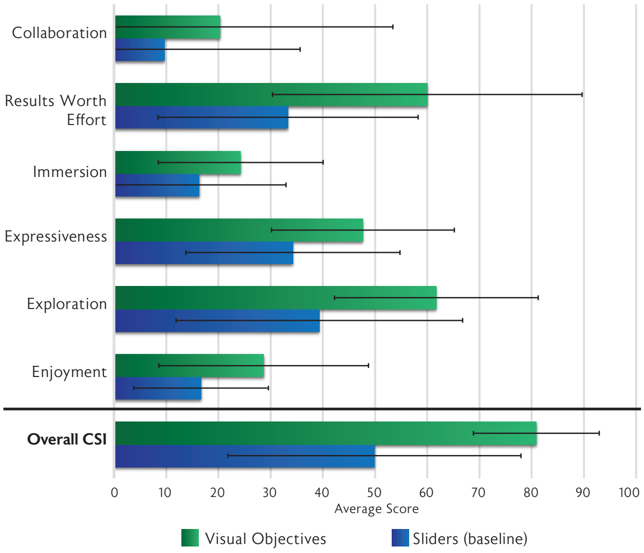


Figure 9: ***Average CSI Scores.*** *CSI scores plotted by factor and overall. Error bars represent one standard deviation. The maximum score is 100. The visual objectives interface outperforms the sliders interface at a significance level of $p < 0.05$ in all factors except collaboration.*

of an interface to support enjoyment, exploration, expressiveness, immersion, collaboration, and to minimize effort. Weights are computed from pairwise comparisons indicating which of the six factors were most important to each individual user. CSI scores have a maximum value of 100.

We compared the CSI for the standard sliders-per-light interface with the visual objectives interface added to the sliders-per-light interface. We noticed that expert users liked using the visual objectives interface alongside the sliders instead of completely replacing them, so the visual objectives interface configuration for this study allowed access to the sliders. This setup measured the added benefit of the visual objectives interface to existing interfaces for non-professional use. Both interface configurations used the same renderer, and the task was performed in a virtual environment.

6.2.1. Experimental Setup

We recruited 15 participants: 10 with little to no experience with lighting design interfaces, and 5 with moderate experience with lighting design interfaces. 10 participants had prior experience with other visual art.

Each participant was given a 20-30 minute tutorial explaining how to use both interfaces in the study. Following the tutorial, the participant was given a brief description (listed in Figure 10) of a scene and instructed to create three lighting designs that match the description. The scene descriptions were taken from an in-class exercise run in our university's introduction to lighting design course. This scenario simulates the situation where a designer must create design variations for a director. Users performed two tasks (one with each interface) and were allowed to take as much time as needed for each task. The CSI survey was administered at the end of each task. The order of the two scene descriptions and the order of the interfaces were randomized.

Since users had limited exposure to lighting design, we preselected research images for the visual concepts interface. Participants were allowed to search for and use their own images during the study if they were unsatisfied with our images. Example output for each task is shown in Figure 10.

6.2.2. Results

The CSI is a standardized metric that measures creativity support along six factors: Enjoyment, Exploration, Expressiveness, Immersion, Results Worth Effort, and Collaboration. We compute the CSI and the individual CSI factor scores for each interface tested based on the user responses to the standardized CSI survey. The results are summarized in Table 1 and Figure 9.

CSI Factor	Count (σ)	Paired-Sample T-test
Enjoyment	1.7 (1.2)	$p = .0059$
Exploration	3.7 (1.1)	$p = .000098$
Expressiveness	3.0 (1.2)	$p = .00065$
Immersion	1.7 (1.0)	$p = .008$
Results Worth Effort	3.3 (1.5)	$p = .0011$
Collaboration	1.6 (1.8)	$p = .0571$
Overall CSI		$p = .00012$

Table 1: ***CSI Counts and T-tests.*** *CSI factor counts with corresponding t-test. A higher factor count indicates that users valued that dimension more than the others. Counts range from 0 to 5.*

The average CSI score for the visual objectives interface was 80.9 ($\sigma = 12.0$) and the average CSI score for the sliders interface was 49.9 ($\sigma = 28.1$) out of a maximum of 100. We note that the visual objectives interface had a higher CSI score than the sliders interface in 14 out of 15 trials. A paired-sample t-test indicates a significant difference between the two average CSI scores ($p = .00012$). Among individual CSI factors, the visual objectives interface significantly outperforms the sliders interface in Exploration, Expressiveness, and Results Worth Effort ($p << 0.05$). This indicates that the objectives interface betters satisfies our design goals of fast exploration and ease of expression of lighting design ideas compared to the existing control methods. The CSI factor counts indicate which dimensions are important to the users of the interface. We note that the same factors that our system excels in are the most important to the users.

We expect the interface used for each task to be the primary factor affecting the CSI score, but to determine if interface order or scene order also had an effect on the CSI score, we tested the effect of these factors using 3-way ANOVA. The test found no evidence that interface order or scene order ($p >> .05$) had an effect on the CSI score. Additionally, the test found that there is no significant interaction between interface order, scene order, and interface use. The test confirms that the choice of interface configuration is the primary factor that affects the overall CSI score ($p << .05$). Therefore, since the only change in interface configuration was the addition of the visual concepts system on top of the sliders interface, we conclude that the visual concepts system provides significant value to existing systems.

(a) It is a late spring afternoon/evening. A person walks through a garden as the sun sets.

(b) It is a cold winter morning. One person goes about their morning tasks in their home.

Figure 10: ***Example Novice/Intermediate Study Output.*** *Scenes created as part of the CSI user study. Both examples were created by participants using the visual concepts interface. The target scene description is shown above the rendering, and the research images chosen by the participant to create each design are displayed on the bottom of the rendering.*

Collaboration This task involves no collaboration, however in accordance with the CSI protocol, we allowed users to mark the collaboration factor as "not applicable" when completing the surveys. Since the task was framed as presenting design options to a director, approximately half of the users chose to answer the collaboration questions in the context of this scenario. Despite answering the questions as applicable, this only influenced the CSI scores for 4 out of the 15 responses, as the collaboration factor count for many users was 0. Our results suggest that the visual objectives interface improves collaboration, however a follow-up study using a collaborative task should be performed to accurately measure this factor.

7. Discussion and Future Work

Inspired by the workflows of expert theatrical lighting designers, we have created a new lighting design interface that facilitates the exploratory design process. The key idea of the system is to generate lighting design candidates by mixing and matching visual objectives that abstractly model designer intent. We have evaluated the system in a case study and two user studies, which confirm that the system generates good design suggestions, facilitates communication between lighting designers and the design team, and allows designers to create lighting designs more quickly and easily.

While the positive reaction from designers suggests the current system already has sufficient scope to be a useful design tool in the theatrical context, we are interested in extensions that would allow the system to encompass a broader set of stage lighting design scenarios. For example, while the vast majority of theatrical productions use static lighting, the current system does not model properties of modern moving light fixtures such as adjustable beam position, varying beam textures, atmospheric effects, and the capability to program motion over time. Supporting these new features without compromising system performance or the quality of lighting visualization presents a future challenge. The visualization quality can be further improved with better photometric calibration, tone mapping, and gamma correction methods.

We are encouraged that theatrical designers have been excited to experiment with our interface, and have integrated our interface with ETC's industry standard tools as a proof-of-concept combined interface. We hope our efforts lead to continued collaboration between the entertainment lighting and computer graphics communities. Outside of theatrical lighting design, we believe that similar interface ideas, and our system's core philosophy of using computational techniques to assist a designer explore design possibilities (but not directly solve design problems) may be applicable to other domains such as 3D modeling, photo editing, or other forms of lighting (e.g., image-based lighting for product photography).

Source Code

The source code and lighting scenes used in this paper can be found at github.com/ebshimizu/VisObjInterface.

Acknowledgements

We would like to thank Cindy Limauro, Rob Thomson, and C. Todd Brown from the Carnegie Mellon School of Drama for providing their expertise and access to facilities for this project. Chenxi Liu for providing renderer development support. Dan Efros and Keith Truax for providing feedback for early versions of the tools and insights into their lighting design processes. Marina Mitchell, Matt Halberstadt, Dennis Varian, Chris Mizerak, and all of the Electronic Theatre Controls staff who took part in the demonstration on their demo stage. The design team for Carnegie Mellon's fall 2017 production of *The Matchmaker* for providing us with the resources and will to run a case study, especially Andrew Hunt (Lighting Designer), and Selby Souza (Scenic Designer). Jim McCann for additional revision feedback and advice. Support for this research was provided by the National Science Foundation (IIS-1253530) and by gifts from Adobe.

References

[ADA*04] AGARWALA A., DONTCHEVA M., AGRAWALA M., DRUCKER S., COLBURN A., CURLESS B., SALESIN D., COHEN M.: Interactive digital photomontage. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 294–302. 2

[ADW04] ANRYS F., DUTRÉ P., WILLEMS Y. D.: Image-based lighting design. In *Proceedings of the 4th IASTED International Conference on Visualization, Imaging, and Image Processing* (2004), p. 15. 2

[ALK*03] AKERS D., LOSASSO F., KLINGNER J., AGRAWALA M., RICK J., HANRAHAN P.: Conveying shape and features with image-based relighting. In *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)* (Washington, DC, USA, 2003), VIS '03, IEEE Computer Society, pp. 46–. 2

[BPB13] BOYADZHIEV I., PARIS S., BALA K.: User-assisted image compositing for photographic lighting. *ACM Trans. Graph. 32*, 4 (July 2013), 36:1–36:12. 2

[CAS16] CAST: wysiwyg R38 product website. http://www.cast-soft.com/wysiwyg/, 2016. 3

[CG92] CASELLA G., GEORGE E. I.: Explaining the Gibbs sampler. *The American Statistician 46*, 3 (1992), 167–174. 2, 4

[CKGF13] CHAUDHURI S., KALOGERAKIS E., GIGUERE S., FUNKHOUSER T.: Attribit: content creation with semantic attributes. In *Proceedings of the 26th annual ACM symposium on User interface software and technology* (2013), ACM, pp. 193–202. 2

[CL14] CHERRY E., LATULIPE C.: Quantifying the creativity support of digital tools through the creativity support index. *ACM Trans. Comput.-Hum. Interact. 21*, 4 (June 2014), 21:1–21:25. URL: http://doi.acm.org/10.1145/2617588, doi:10.1145/2617588. 9, 10

[Con17] CONTROLS E. T.: Eos family website. http://www.etcconnect.com/Products/Consoles/Eos-Family/, 2017. 9

[DAG95] DORSEY J., ARVO J., GREENBERG D.: Interactive design of complex time dependent lighting. *IEEE Computer Graphics and Applications 15*, 2 (Mar 1995), 26–36. doi:10.1109/38.365003. 2

[DSG91] DORSEY J. O., SILLION F. X., GREENBERG D. P.: Design and simulation of opera lighting and projection effects. *SIGGRAPH Comput. Graph. 25*, 4 (July 1991), 41–50. 3

[FRS*12] FISHER M., RITCHIE D., SAVVA M., FUNKHOUSER T., HANRAHAN P.: Example-based synthesis of 3d object arrangements. *ACM Trans. Graph. 31*, 6 (Nov. 2012), 135:1–135:11. 2

[ILM14] ILM: Openexr v2.2.0. http://www.openexr.com/index.html, 2014. 8

[KP09] KERR W. B., PELLACINI F.: Toward evaluating lighting design interface paradigms for novice users. *ACM Trans. Graph. 28*, 3 (July 2009), 26:1–26:9. 2

[KPC93] KAWAI J. K., PAINTER J. S., COHEN M. F.: Radioptimization: Goal based rendering. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1993), SIGGRAPH '93, ACM, pp. 147–154. 2

[Lig16] LightConverse 3D Show v58 product website. http://lightconverse.com/, 2016. 3

[LMGH*13] LOPEZ-MORENO J., GARCES E., HADAP S., REINHARD E., GUTIERREZ D.: Multiple light source estimation in a single image. *Computer Graphics Forum 32*, 8 (2013). 2

[LRFH13] LIN S., RITCHIE D., FISHER M., HANRAHAN P.: Probabilistic color-by-numbers: Suggesting pattern colorizations using factor graphs. *ACM Trans. Graph. 32*, 4 (July 2013), 37:1–37:12. 2, 7

[LSK*10] LEE B., SRIVASTAVA S., KUMAR R., BRAFMAN R., KLEMMER S. R.: Designing with interactive example galleries. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2010), CHI '10, ACM, pp. 2257–2266. 2

[MAB*97] MARKS J., ANDALMAN B., BEARDSLEY P. A., FREEMAN W., GIBSON S., HODGINS J., KANG T., MIRTICH B., PFISTER H., RUML W., RYALL K., SEIMS J., SHIEBER S.: Design galleries: A general approach to setting parameters for computer graphics and animation. In *Transactions on Graphics* (New York, NY, USA, 1997), SIGGRAPH '97, ACM Press., pp. 389–400. 2

[ML16] MA LIGHTING I. G.: MA 3D product website. http://www.malighting.com/en/products/control/control/ma-lighting/ma-3d/ma3d-ma-3d.html, 2016. 3

[PBMF07] PELLACINI F., BATTAGLIA F., MORLEY R. K., FINKELSTEIN A.: Lighting with paint. *ACM Trans. Graph. 26*, 2 (June 2007). 2

[Pel10] PELLACINI F.: envylight: An interface for editing natural illumination. *ACM Trans. Graph. 29*, 4 (July 2010), 34:1–34:8. 7

[PF92] POULIN P., FOURNIER A.: Lights from highlights and shadows. In *Proceedings of the 1992 Symposium on Interactive 3D Graphics* (New York, NY, USA, 1992), I3D '92, ACM, pp. 31–38. 2

[PTG02] PELLACINI F., TOLE P., GREENBERG D. P.: A user interface for interactive cinematic shadow design. *ACM Trans. Graph. 21*, 3 (July 2002), 563–566. 2

[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 24*, 3 (Aug. 2005), 464–470. doi:10.1145/1073204.1073214. 2, 8

[SC97] STRAUSS A., CORBIN J.: *Grounded Theory in Practice*. SAGE Publications, 1997. 10

[SDS*93] SCHOENEMAN C., DORSEY J., SMITS B., ARVO J., GREENBERG D.: Painting with light. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1993), SIGGRAPH '93, ACM, pp. 143–146. URL: http://doi.acm.org/10.1145/166117.166135, doi:10.1145/166117.166135. 2

[SSCO09] SHAPIRA L., SHAMIR A., COHEN-OR D.: Image Appearance Exploration by Model-Based Navigation. *Computer Graphics Forum* (2009). doi:10.1111/j.1467-8659.2009.01403.x. 2

[SW14] SCHWARZ M., WONKA P.: Procedural design of exterior lighting for buildings with complex constraints. *ACM Trans. Graph. 33*, 5 (Sept. 2014), 166:1–166:16. 2

[Vec16] VECTORWORKS I.: Vectorworks Spotlight product website. http://www.vectorworks.net/spotlight, 2016. 3

[YAMK15] YUMER E., ASENTE P., MECH R., KARA L. B.: Procedural modeling using autoencoder networks. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (2015), ACM, pp. 109–118. 2