EUROGRAPHICS '0x / N.N. and N.N.
(Editors)

Volume 0 (1981), Number 0

Global Illumination Compensation for Spatially Augmented Reality

Yu Sheng †, Theodore C. Yapo and Barbara Cutler

Department of Computer Science, Rensselaer Polytechnic Institute

Abstract

When projectors are used to display images on complex, non-planar surface geometry, indirect illumination between the surfaces will disrupt the final appearance of this imagery, generally increasing brightness, decreasing contrast, and washing out colors. In this paper we predict through global illumination simulation this unintentional indirect component and solve for the optimal compensated projection imagery that will minimize the difference between the desired imagery and the actual total illumination in the resulting physical scene. Our method makes use of quadratic programming to minimize this error within the constraints of the physical system, namely, that negative light is physically impossible. We demonstrate our compensation optimization in both computer simulation and physical validation within a table-top spatially augmented reality system. We present an application of these results for visualization of interior architectural illumination. To facilitate interactive modifications to the scene geometry and desired appearance, our system is accelerated with a CUDA implementation of the QP optimization method.

1. Introduction & Motivation

Large scale displays and projection systems are becoming more prevalent and integrated within our daily physical environments. The brightness, contrast, dimensions, and affordability of these displays is increasing, allowing them to be incorporated into immersive display environments that surround the user. In the typical dark theater setting, indirect illumination can be safely ignored because a single planar projection surface is used and all remaining surfaces in the environment are suitably dark and absorptive. When these display surfaces are oriented to face each other, for example in a CAVE [CNSD*92], the secondary scattering of light should no longer be ignored. Doing so may result in final surface illumination with increased brightness and decreased contrast (Figure 1c). Automatic correction for this extra illumination cannot be accurately solved with a constant scaling term since indirect scattering depends on both the total quantity and local distribution of light projected onto the surface.

Spatially Augmented Reality (SAR) [BR07] differs from the traditional CAVE environment by making use of existing surfaces in the environment for projection. These application specific environments can more readily accept and provide immersive experiences to multiple users. Smaller-scale, table top SAR systems can convincingly bring to life static objects by illuminating them with dynamic imagery [RWLB01,RZW02,HKYL06,SYYC09]. In these systems, the position and orientation of each projector and the full geometry of the scene is known a priori, and the surfaces are assumed to be a uniform diffuse material with a known color. In this paper, we tackle the problem of unintended secondary scattering of the projected light for multi-planar or non-planar surfaces. Our goal is to determine what color and intensity values should be projected onto each surface so that the final intensity with scattering matches as closely as possible the desired scene imagery (Figure 1g&h).

2. Related Work

Lighting Design and Common Illumination The inverse lighting problem has previously been investigated using radiosity to determine the relative intensities for a restricted set of light sources [DSG91, SDS*93, Con02] that will best match a desired illumination output. Inverse radiosity is a global illumination technique that recovers reflectance of Lambertian surfaces with known geometry, direct illumination, and final radiance [YDMH99]. The problem is also re-

† e-mail: shengyu@cs.rpi.edu

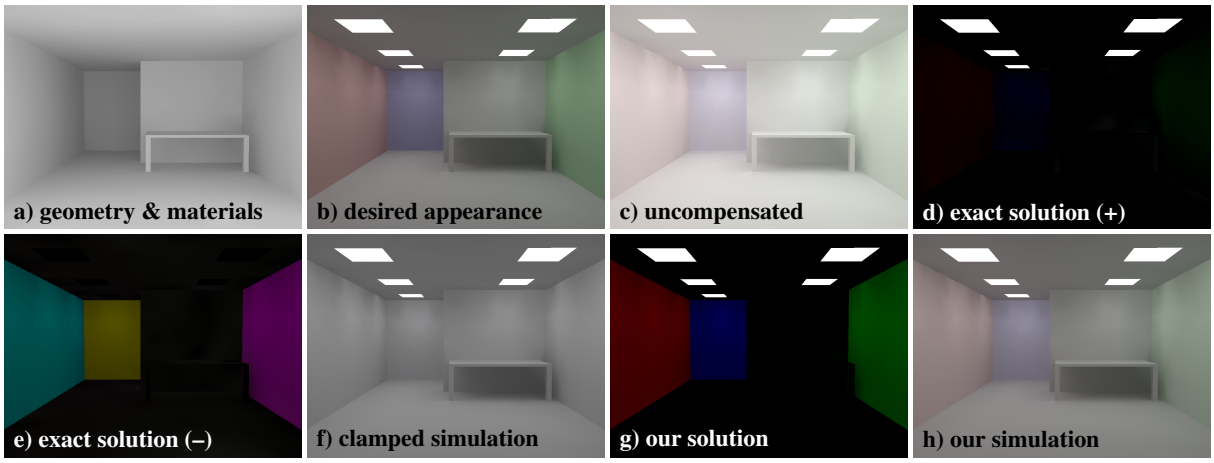


Figure 1: *We aim to use projection to transform (a) this simple L-shaped room with a desk, where all surfaces have uniform diffuse white reflectance ($\rho = 0.6$), into a more colorful room with simulated area lights on the ceiling. (b) The "desired" virtual scene appearance includes subtle color bleeding from global illumination when light from the emitters bounce off the colored walls [red:<0.65,0.4,0.4> green:<0.4,0.65,0.4> blue:<0.4,0.65,0.4> white:<0.6,0.6,0.6>]. (c) Simply projecting this desired imagery onto the all-white surfaces results in a overly bright and washed out scene due to indirect scattering. An exact solution to the compensation problem using IRC requires both (d) positive and (e) negative light. Clamping the negative values of the exact solution to zero (f) results in poor color fidelity. In this paper, we formulate projection compensation as an optimization problem with non-negativity box constraints. (g) Our solution results in a better representation of the intensities, colors, and gradients in the scene when (h) projected onto the same white surfaces.*

lated to work on compensating for the "common illumination" between synthetic and real objects [FGR93, DRB97, GM00, CNR08].

Radiometric Compensation Radiometric compensation is the adaptation of projection imagery to minimize blending artifacts caused by light modulation with local surface attributes, including surface normal, distance from projector, color mismatch of projectors, and additional environmental light [BEK05, NPGB03]. Using a coaxial camera and projector, Fuji et al. demonstrated realtime radiometric compensation for dynamic surfaces and/or moving projectors [FGN05]. Many radiometric compensation methods only take surface reflectance and projector response into consideration. In contrast, Ashdown et al. proposed a robust *content dependent* photometric compensation framework paying careful attention to appropriate choice of color spaces and gamut to produce optimal results for spatially varying or saturated projection surfaces [AOSS06]. However, the optimization method is likely too expensive for interactive projection. Some content dependent compensation methods make use of a perceptually-derived threshold map of the desired imagery [RPG99]; for example, Grundhöfer et al.'s real-time adaptive compensation implemented on the GPU [GB08]. However, without pursuing an optimal result, chrominance clipping errors might occur when projecting onto a highly saturated color surface.

Inverse Lighting The radiometric compensation methods above assume a simple geometric relationship between projector and camera and do not consider the global light interreflection. When projecting images onto more complex surfaces, effects such as reflection and refraction can be significant and should not be ignored. To correctly compensate for these effects, the desired image should serve as the target result including this unavoidable indirect illumination, and the projection image should be the corresponding direct illumination necessary to produce this result. Seitz et al. proposed the Interreflection Cancellation Operator (IRC), which, in theory, can be used to cancel indirect scattering for general BRDFs [SMK05]. Using active scanning with a single point laser, Seitz et al. demonstrated acquisition of the Impulse Scatter Function (ISF) that is used to represent the IRC for arbitrary unknown geometry. However, this process is time consuming and requires significant storage. If we are only interested in computing the separate contributions of direct and indirect illumination, it is not necessary to compute the entire IRC matrix. Nayar et al. [NKGR06] demonstrated how to extract these components for a wide range of global illumination effects using a small set of structured light patterns. Full light transport between a camera image and projection image can be modeled with the *transport matrix* first introduced by Sen et al. [SCG*05]. Similar to the ISF, this matrix is computed offline using structured light. This type of active scanning is undesirable in online SAR systems with interactively modified geometry; thus, we have selected passive acquisition methods that require additional knowledge about the geometry and materials.

Immersive environments Radiometric compensation and

inverse lighting techniques have been used in immersive environments. The most closely related work to our research is *reverse radiosity*, an iterative method proposed by Bimber et al. [BGZ*06] to compensate for indirect scattering in immersive and semi-immersive projection displays. Similarly, Mukaigawa et al. [MKO06] introduced an analytical solution to compensate for indirect scattering in diffuse projection environments at real time frame rates by reformulating the classical radiosity equation. Habe et al. [HSM07] presented an algorithm to theoretically compensate for specular reflections. These compensation methods are essentially equivalent to the IRC. IRC will always produce an exact solution for the necessary projection imagery, though in some cases the solution requires *negative* emittance (Figure 1e), which is impossible for physical projection systems. Simply clamping negative values to black prior to projection will result in inaccurate color reproduction and make the resulting scene brighter than the desired imagery (Figure 1f).

Wetzstein and Bimber used the transport matrix for radiometric compensation [WB07]. In their work, a least squares error metric is formulated and solved using constrained iterative steepest descent [ZN06]. To achieve real-time video projection, the pseudo-inverse of the transport matrix is used, which may generate values that are negative or greater than the maximum projector intensity. This simple least squares error metric might not ensure color fidelity because the color channels are optimized independently. Furthermore, calibration of the transport matrix is both time and storage consuming and when the projection surfaces or projector configurations are modified, the system must be fully re-calibrated.

3. Our Contributions

The contributions we present in this paper include:

- Prediction of the secondary scattering within a spatially-augmented reality system consisting of multiple projectors and a collection of re-positionable planar and curved diffuse projection surfaces.
- Formulation and solution of an optimization problem to calculate the projection values that when combined with the unavoidable indirect illumination results in a total illumination that best matches the desired imagery.
- An interactive, prototype implementation of the method, both in software and in physical realization that validates the effectiveness of our formulation. We have also implemented the algorithm in CUDA for use in an interactive architectural lighting design system.
- Exploration of a variety of different projection surface albedos and evaluation of the quality of resulting imagery.

4. Compensation Problem Overview

In our examples, we aim to use projection to transform uniform white or grey diffuse geometry into an illusion of a more complex environment including colorful surfaces, area light source emitters, and/or strong shadows of sunlight (e.g., Figure 1). By modeling the diffuse illumination transport in the physical model, we are able to compensate for indirect scattering of the projected light and more accurately control the resulting intensities of patches in the physical model. We outline the details of this problem using a basic patch-based radiosity framework [GTGB84].

4.1. Formulation as a Reverse Radiosity Problem

Initially, we assume that the virtual desired scene, $\mathbf{S_v}$, and the physical scene, $\mathbf{S_p}$, are geometrically the same; that is, they consist of the exact same n patches. We assume that all surfaces in both models are ideal diffuse reflectors, but allow the diffuse color reflectance values of patch i for the two scenes ($\rho_{v,i}$ and $\rho_{p,i}$) to differ. Our goal is to determine what illumination values should be directly projected onto the physical scene to create the target illumination within the virtual scene. The direct illumination from the projector onto a surface is modeled as a diffuse surface emittance per patch and the indirect scattering is modeled using radiosity.

We construct $\mathbf{K_v}$ and $\mathbf{K_p}$, the radiosity matrices for the virtual and physical scenes. Likewise we define $B_{v,i}$ and $B_{p,i}$ to be the radiosities (reflection of both direct and indirect illumination) of patch i in the virtual and physical scenes, respectively, and $E_{v,i}$ and $E_{p,i}$ to be the emitted light (reflection of direct illumination from the projectors) of patch i in their respective scenes. Given these, we have the radiosity equation for each scene:

$$\mathbf{K_v B_v} = \mathbf{E_v} \tag{1}$$

$$\mathbf{K_p B_p} = \mathbf{E_p} \tag{2}$$

where $\mathbf{B_v}$, $\mathbf{B_p}$, $\mathbf{E_p}$, $\mathbf{E_v}$ are $n \times 1$ vectors.

4.2. Exact Solution

The objective of our projection system is to achieve physical radiosity values that match the virtual radiosity values; hence, we set $\mathbf{B_v} = \mathbf{B_p}$. However, our only control over lighting in the physical scene is to set the physical patch emittance values, $E_{p,i}$, by controlling their direct illumination from the projectors. From Equations (1) and (2), we can solve for the required real-scene emittances:

$$\mathbf{E_p} = \mathbf{K_p B_p} = \mathbf{K_p B_v} = \mathbf{K_p K_v}^{-1} \mathbf{E_v}. \tag{3}$$

In practice, $\mathbf{B_v}$ can be approximated more efficiently than the brute force matrix inversion of $\mathbf{K_v}$, and the resulting images derived from $\mathbf{B_p}$. Equation (3) is called the analytical solution of reverse radiosity [MKO06] and $\mathbf{K_p}$ is the IRC. Unfortunately, although the computed emittances will provide an exact solution to Equation (3), this solution may involve *negative* emittances at some physical patches, which is impossible for physical systems. In some cases, the negative quantities are significant and simply clamping these components to zero may result in overly bright and poor color fidelity projections (Figure 1f). Rather than clamping these

terms to zero after-the-fact, one can use constrained least squares to solve equation (3). However, if each color channel is solved independently, this method may also result in low color fidelity (Figure 3c). In the following section we present a new objective function for use in constrained optimization that directly addresses these color fidelity problems.

5. Constrained Optimization

To ensure that the projection solution is physically realizable, we reformulate the scenario as a constrained optimization problem. The primary constraint is the physical emittance non-negativity constraint: $\mathbf{E_p} \geq 0$, which means each entry $E_{p,i} \geq 0$. The solution can further be constrained to match the physical limitations of the actual projectors, namely a non-zero black level and a maximum intensity. In the following sections we describe the terms of the objective function we minimize to solve this optimization problem.

We define r_i, g_i, b_i as the red, green, and blue channels of each patch for the virtual "desired" scene, and r'_i, g'_i and b'_i as the three channels for the physical scene. $\mathbf{B_v}$ and $\mathbf{B_p}$ are $3n \times 1$ vectors defined by:

$$\mathbf{B_v} = [r_1,...,r_n,g_1,...,g_n,b_1,...,b_n]^T$$
$$\mathbf{B_p} = [r'_1,...,r'_n,g'_1,...,g'_n,b'_1,...,b'_n]^T$$

and $\mathbf{y} = \mathbf{B_v} - \mathbf{B_p}$ is the difference between these two vectors.

To quantify and improve the color fidelity of our solution, we transform colors in RGB to another linear color space, YPbPr, which separates the luminance and chrominance components, and define our error metrics in this alternate color space. The transformation from (*r*,*g*,*b*) to (*Y*,*Pb*,*Pr*) is accomplished with a 3×3 transformation matrix, $\mathbf{T}$. t_{ij} is the entry in the *i*th row and *j*th column of $\mathbf{T}$. Our metrics will minimize chrominance error as defined by the linear YPbPr color space; however, we note that it does not necessarily minimize human perception of the color error. We will discuss this more in Section 6.

5.1. Minimize Absolute Luminance Error

The first term defines the sum of absolute luminance error of the physical scene over all patches:

$$f_{lumin} = \sum_i (Y_i - Y'_i)^2 = \mathbf{y}^T \mathbf{W_0} \mathbf{y},$$

where

$$\mathbf{W_0} = \mathbf{A}^T\mathbf{A}, \quad \mathbf{A} = [t_{11}\mathbf{I}, t_{12}\mathbf{I}, t_{13}\mathbf{I}],$$

and $\mathbf{I}$ is an $n \times n$ identity matrix.

This term will only minimize the luminance difference; thus, an objective function with this term alone is not sufficient to obtain high quality results for scenes with strong colors, such as the Cornell Box (Figure 2c).

5.2. Minimize Absolute Chrominance Error

Our second term targets the error in color, by minimizing the differences in Pb and Pr:

$$f_{chrom} = \sum_i [(Pb_i - Pb'_i)^2 + (Pr_i - Pr'_i)^2] = \mathbf{y}^T \mathbf{W_1} \mathbf{y},$$

where

$$\mathbf{W_1} = \mathbf{B}^T\mathbf{B} + \mathbf{C}^T\mathbf{C},$$
$$\mathbf{B} = [t_{21}\mathbf{I}, t_{22}\mathbf{I}, t_{23}\mathbf{I}], \quad \mathbf{C} = [t_{31}\mathbf{I}, t_{32}\mathbf{I}, t_{33}\mathbf{I}].$$

By preserving Pb and Pr components, this term will attempt to preserve the color of each patch in the scene.

5.3. Preserving Spatial Luminance Discontinuities

Our third term aims to preserve the gradients and discontinuities in luminance between neighboring patches:

$$f_{nbd\_lumin} = \sum_{(i,j)\in nbd} [(Y_i - Y_j) - (Y'_i - Y'_j)]^2 = \mathbf{y}^T \mathbf{W_2} \mathbf{y},$$

where $(i, j) \in nbd$ means i and j are neighbors and

$$\mathbf{W_2} = \mathbf{E}^T\mathbf{E} = \begin{bmatrix} t_{11}^2\mathbf{L} & t_{11}t_{12}\mathbf{L} & t_{11}t_{13}\mathbf{L} \\ t_{11}t_{12}\mathbf{L} & t_{12}^2\mathbf{L} & t_{12}t_{13}\mathbf{L} \\ t_{11}t_{12}\mathbf{L} & t_{12}t_{13}\mathbf{L} & t_{13}^2\mathbf{L} \end{bmatrix},$$
$$\mathbf{E} = [t_{11}\mathbf{D}, t_{12}\mathbf{D}, t_{13}\mathbf{D}], \tag{4}$$

$\mathbf{D}$ is the incidence matrix of the dual graph of the geometry, and $\mathbf{L}$ is the Laplacian matrix. This term can be further extended to second derivative approximation by replacing $\mathbf{D}$ by the Laplacian $\mathbf{L}$ in equation (4), and even higher.

5.4. Preserving Spatial Chrominance Discontinuities

Our final metric aims to preserve the gradients and discontinuities of chrominance between neighboring patches. The derivation of this term is similar to the previous one.

$$f_{nbd\_chrom} = \sum_{(i,j)\in nbd} [(Pb_i - Pb_j) - (Pb'_i - Pb'_j)]^2 + [(Pr_i - Pr_j) - (Pr'_i - Pr'_j)]^2 = \mathbf{y}^T \mathbf{W_3} \mathbf{y},$$

where

$$\mathbf{W_3} = \mathbf{F}^T\mathbf{F} + \mathbf{G}^T\mathbf{G},$$
$$\mathbf{F} = [t_{21}\mathbf{D}, t_{22}\mathbf{D}, t_{23}\mathbf{D}], \mathbf{G} = [t_{31}\mathbf{D}, t_{32}\mathbf{D}, t_{33}\mathbf{D}].$$

5.5. Complete Objective Function

We assemble the four terms described above to form the objective function of our optimization problem:

$$\begin{aligned} f &= \alpha f_{lumin} + \beta f_{chrom} + \gamma f_{nbd\_lumin} + \delta f_{nbd\_chrom} \\ &= \alpha \mathbf{y}^T\mathbf{W_0}\mathbf{y} + \beta \mathbf{y}^T\mathbf{W_1}\mathbf{y} + \gamma \mathbf{y}^T\mathbf{W_2}\mathbf{y} + \delta \mathbf{y}^T\mathbf{W_3}\mathbf{y} \\ &= \mathbf{y}^T\mathbf{W}\mathbf{y} \end{aligned} \tag{5}$$

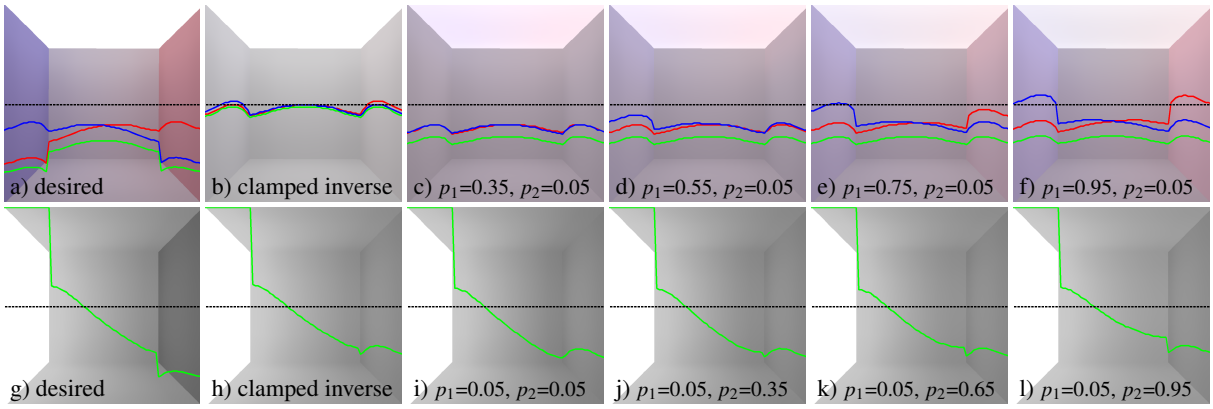


Figure 2: *In the top row we transform an all-white box ($\rho = 0.8$) into (a) the classic Cornell box, with a top face emitter, and materials red:<0.85,0.45,0.45>, blue:<0.45,0.45,0.85>, and white:<0.8,0.8,0.8>. Each image is overlaid with a plot of the RGB channels of the pixels on the midline (dashed black). (b) The clamped inverse method results in an overall brighter image and very little color on the left and right walls, because this exact solution relies heavily on negative light. In images (c-f) we show our solutions for different settings of the p_1 parameter: $p_1 = 0$ minimizes the per-patch difference in luminance and $p_1 = 1$ minimizes the per-patch difference in chrominance. In the bottom row we aim to transform an all-white box ($\rho = 0.5$) into (g) a scene with the emitter on the left:<0.9,0.9,0.9> and other surfaces top/back/bottom:<0.8,0.8,0.8> and right:<0.3,0.3,0.3>. In images (i-l) we show our solutions for different settings of the p_2 parameter: $p_2 = 0$ minimizes the absolute per-patch difference and $p_2 = 1$ minimizes the spatial difference. The green line in each plot in the bottom row shows a plot of the intensity value for each pixel along the midline. Of the 5 simulations (h-l), image (i) achieves the minimum luminance value at the pixel between the back and right walls and image (l) achieves the greatest luminance discontinuity at that same point.*

where α, β, γ, and δ are nonnegative weighting parameters that sum to 1, and $\mathbf{W} = \alpha\mathbf{W_0} + \beta\mathbf{W_1} + \gamma\mathbf{W_2} + \delta\mathbf{W_3}$. Since Laplacian matrices are all positive semi-definite, matrix $\mathbf{W}$ is guaranteed to be positive definite if $\mathbf{T}$ is a non-singular matrix, which is always true for any color space that can be linearly transformed from RGB.

We define parameters p_1 and p_2 to provide user control of the objective function in an application specific manner. We set the four coefficients in equation (5) from p_1 and p_2,

$$\alpha = (1-p_1)(1-p_2), \qquad \beta = (1-p_2)p_1,$$
$$\gamma = (1-p_1)p_2, \qquad \delta = p_1 p_2.$$

p_1 controls the relative importance of luminance vs. chrominance and p_2 adjusts the relative importance of absolute vs. spatial error. In general, p_1 determines the color fidelity, and p_2 preserves the smoothness and discontinuity in the desired virtual environments. We present results for different relative weightings in Figure 2.

6. Optimization in Perceptually Uniform Color Space

Ideally, a solution to the compensation problem should reduce the *perceived* error between the desired and physical scenes. The natural way to ensure this result is to minimize error in a color space which attempts to be perceptually-uniform, in that Euclidean distance in the space corresponds roughly to perceived differences between observed colors. However, the inherent non-linearity of the human visual system, as captured in the CIE L*a*b* color space [McL76], precludes efficient optimization of such metrics. To explore the substitution of a linear YPbPr space for the more perceptually-uniform L*a*b*, we used MATLAB **fmincon** to minimize L*a*b* ΔE for a scene of 96 patches (Figure 3e). Even with this reduced geometry, the L*a*b* calculation takes more than 30 minutes to converge; indicating that this approach is likely to be impractical. Running our QP-based compensation algorithm in YPbPr on the same scene (Figure 3d) produces results visually similar to the L*a*b* algorithm in seconds (Section 7.2).

For each of the simulations shown in Figure 3, we estimated the perceptual difference by using CIE L*a*b* ΔE calculations [McL76]. The results from our YPbPr optimization are very similar to those of L*a*b* and significantly better than both clamped inverse and non-negative least squares. Note that about 2.3 ΔE's are equivalent to one Just Noticeable Difference (JND).

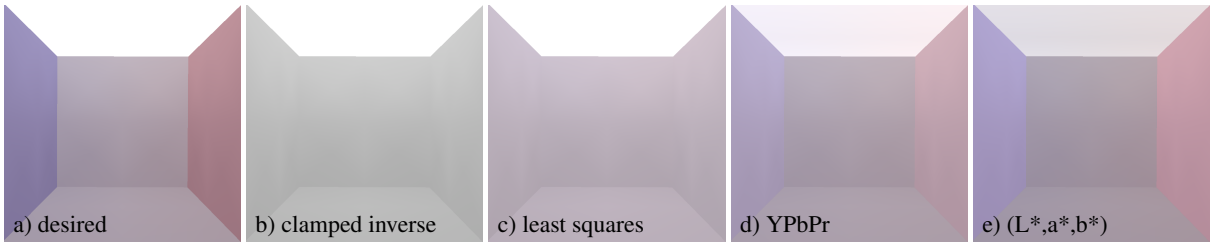
7. Implementation Details

In this section we discuss methods to solve the optimization problem outlined in Section 4 and the implementation details and resulting performance for each method.

7.1. Quadratic Programming

Equation (5) can be further derived to:

$$f = (\mathbf{B_v} - \mathbf{K_p}^{-1}\mathbf{E_p})^T\mathbf{W}(\mathbf{B_v} - \mathbf{K_p}^{-1}\mathbf{E_p})$$
$$= \mathbf{B_v}^T\mathbf{W}\mathbf{B_v} - 2\mathbf{B_v}^T\mathbf{W}\mathbf{K_p}^{-1}\mathbf{E_p} + \mathbf{E_p}^T\mathbf{K_p}^{-T}\mathbf{W}\mathbf{K_p}^{-1}\mathbf{E_p}.$$



$\overline{\Delta E} = 16.4,\ \Delta E_{max} = 33.2$ $\overline{\Delta E} = 11.4,\ \Delta E_{max} = 25.4$ $\overline{\Delta E} = 6.24,\ \Delta E_{max} = 14.9$ $\overline{\Delta E} = 5.94,\ \Delta E_{max} = 13.2$

Figure 3: *We compare the quality of the result using d) our linear YPbPr metric to e) the optimal solution within the nonlinear (L*,a*,b*) perceptually-uniform color space. The a) desired image, b) clamped inverse solution, and c) box constrained least squares solution are also included. The global illumination and compensation results for all images are computed using a coarse 96 triangle mesh. The physical surface albedo, $\rho = 0.8$, and the virtual materials are red:<0.8,0.5,0.5> blue:<0.5,0.5,0.8> white:<0.8,0.8,0.8>. Note that the ceiling polygon is brighter and more accurate in the YPbPr image yet the color fidelity is better with (L*,a*,b*). The mean and max $L^*a^*b^*$ ΔE for each solution is listed below the image.*

Thus, our compensation formulation maps to a Quadratic Programming (QP) problem:

$$f = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{c}^T\mathbf{x} \tag{6}$$

subject to $\mathbf{x} \geq 0$ (or the appropriate minimum and maximum intensities limits of the projectors) by letting $\mathbf{x} = \mathbf{E_p}$, $\mathbf{c} = -2{\mathbf{K_p}}^{-T}\mathbf{W}\mathbf{B_v}$, and $\mathbf{Q} = 2{\mathbf{K_p}}^{-T}\mathbf{W}{\mathbf{K_p}}^{-1}$. Note that the term ${\mathbf{B_v}}^T\mathbf{W}\mathbf{B_v}$ is disregarded in Equation (6) since it is constant and will not affect the optimization result.

There are several algorithms available to solve QP problems, such as the active set strategy [GMSW84] and preconditioned conjugate gradients [Kny91]. Zhang and Nayar [ZN06] used a steepest descent algorithm to solve a similar box constraint optimization problem. However this method has poor convergence performance for some functions. Because the Hessian matrix $\mathbf{Q}$ in equation (6) is positive definite, and the constraint is a box constraint, we are able to use the two-metric projection method [Ber82] to attain a quadratic convergence rate.

7.2. CUDA

We have implemented our optimization algorithm via the two-metric projection method using NVIDIA's CUDA framework [NVI09]. CUDA provides an interface for software programmers to access the parallel computing ability of modern GPUs. It also provides libraries for high performance numerical computing, including Linear Algebra and FFT. Our implementation makes use of the CUBLAS library. We also use the Cholesky factorization code provided by Volkov and Demmel. [VD08].

We achieve a significant performance improvement from the GPU on a modern desktop with a 512 megabyte NVIDIA 8800GT graphics card and Intel Core 2 QUAD Q9450 CPU. For a Cornell Box scene with about 1200 patches, once the precomputation of form factor and Laplacian matrices is finished, it takes 0.28 seconds for the algorithm to converge for a physical environment with $\rho = 0.642$ and 1.29 seconds for $\rho = 0.9$. The corresponding implementation in MATLAB runs for about 15.65 and 17.76 seconds respectively.

To reduce the running time, our implementation caches 4 critical matrices in graphics card memory. Each one is a $3n \times 3n$ single precision floating point matrix, where n is the number of patches in the scene. For our Cornell Box scene, this cached memory is approximately 200 megabytes. Thus, we are currently limited to scenes with at most 1600 patches in the physical environment, $\mathbf{S_p}$. This problem can be alleviated by optimizing the memory usage, which is an interesting topic for future work. In addition, rapid advances in graphics hardware and memory size will further alleviate these restrictions.

8. Physical Validation

To validate our compensation we have recreated the Cornell Box scene within a five-sided physical environment where all the surfaces have the same diffuse neutral-colored (white or grey) reflectance properties. Our physical test setup consists of opaque projection surfaces on a table top surrounded by four projectors mounted above the user's head and directed toward the table (Figure 4a). We tested a variety of matte papers and paints on the interior of this box with albedos ranging from 0.06 to 0.94, which were measured using an optical reflectometer. In the mathematics of our compensation algorithm, we assume that all physical surfaces are perfectly Lambertian. We also created variations of the original Cornell Box to serve as test cases for the virtual "desired" scene. Figure 5 presents a small selection of these tests using the different compensation strategies. For each example we show a simulated projection as well as a photograph of the actual projection within our physical scene for validation.

As the average albedo of the surfaces in the physical scene increases, the indirect component becomes a larger fraction of the total illumination. On the more challenging test cases (when the physical surfaces have higher albedo than the vir-

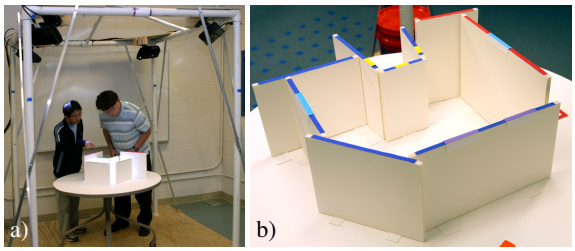


Figure 4: *The overall setup of our table-top spatially augmented reality system is shown in the left image. A single camera above the scene detects the current geometry that is then communicated to four calibrated projectors which display imagery on the scene. The right image shows an example physical scene with uniform matte white projection surfaces. Note: Both images were captured with bright room lighting. During normal use the room lighting is turned off.*

tual scene) our method performs significantly better than the clamped inverse solution used in previous work (Section 2). For scenes with darker physical albedo, the QP solution and the clamped solution are more similar and both match the desired scene quite well, as expected.

9. Application to Architectural Lighting Design

The application that initially motivated our work is the simulation of global illumination within a physical small scale architectural model [SYYC09]. The geometry of the architectural design is sketched with diffuse white foam-core walls (shown in Figure 4b). The user specifies windows and the desired material properties with additional color-coded tokens. The ability to change the apparent surface reflectance properties within an immersive augmented reality environment is a powerful tool for interior design. The walls and tokens are detected by a calibrated camera above the table. The closed 3D geometry of the intended architectural model must be inferred from these walls, filling in the gaps between the walls and adding a ceiling [SYYC10]. Thus, neither the geometry nor the materials of the physical scene match the intended design. The goal is to project imagery onto the interior of the physical walls to mimic the appearance of the intended design under a specified illumination condition.

Geometric differences between $\mathbf{S_v}$ and $\mathbf{S_p}$ An important modification to the initial problem described in Section 4 is to allow geometric differences between $\mathbf{S_v}$ and $\mathbf{S_p}$. In fact, in many augmented or mixed reality systems it is necessary to omit portions of the physical scene to allow unobstructed views for the users and projectors, except when rear-projection displays are used. In other words, while a target application (e.g., global illumination calculations using radiosity) might require a "watertight" virtual scene, often the physical scene is not watertight. We call these missing patches, which exist in $\mathbf{S_v}$but not $\mathbf{S_p}$, *fillin geometry*. Furthermore, the physical scene may have additional geometry that will cause unintended inter-reflections of light or obstruct the viewpoint of the projectors. We call this *extra geometry* (patches that exist in $\mathbf{S_p}$ but not $\mathbf{S_v}$). Patches that exist in both scenes are called *projection surfaces*.

To determine the desired global illumination solution, we construct the virtual form factor matrix, $\mathbf{K_v}$, with the projection and fillin surfaces (omitting the extra geometry). Since the fillin geometry does not have a physical counterpart, we disregard the computed $B_{v,i}$ for these patches. To calculate the compensated projection values, we construct the physical form factor matrix, $\mathbf{K_p}$, with the projection and extra surfaces (omitting the fillin geometry). We do not have target values for $B_{p,i}$ values for the extra patches from the virtual scene; thus, in our examples we set the target radiosity value for these patches to be the minimum intensity of our projectors.

Iterative Architectural Design Figure 6 and our companion video present examples of architectural designs that were modeled with our table-top spatially augmented reality system for daylighting visualization. The system is sufficiently fast and practical for interactive visualization and analysis of lighting conditions during architectural design. The video includes animations of a single day (March 21st) within each space. As the sun moves across the sky from east to west the shadows of bright sunlight through the windows play across several brightly colored walls.

Our implementation runs sufficiently fast to be useful in iterative, interactive visualization. When the geometry of the physical scene is modified, the system must perform image processing and mesh generation [SYYC10], calculate form factor matrices, and initialize matrices for QP. Depending on the complexity of the scene, the total time for these steps is 11-16 seconds, for the examples shown in this paper and in the video. After this precomputation, the desired lighting condition and optimal projection imagery (according to our YPbPr metrics) are computed in 1-1.5 seconds.

Radiosity Simulation with High Frequency Detail In our daylighting simulation system, direct sun illumination is an important visual cue for users to understand the geometry of their design. To capture the hard shadow boundaries cast by sunlight, we use shadow volumes to render direct sun illumination and add it to the indirect illumination computed by radiosity. This hybrid rendering algorithm is also applied to the compensation results. For each patch directly illuminated by the sun, we first calculate the fraction s of the total illumination of the patch that comes directly from the sun. We also compute a, the fraction of the patch area illuminated by the sun. Then the compensated direct illumination d can be computed by multiplying s with the compensated value computed from our QP optimization. The projection value for each patch is d/a. These patch values are interpolated to determine the values for each vertex and used to render the stencil shadow volumes. Results are shown in Figure 6.

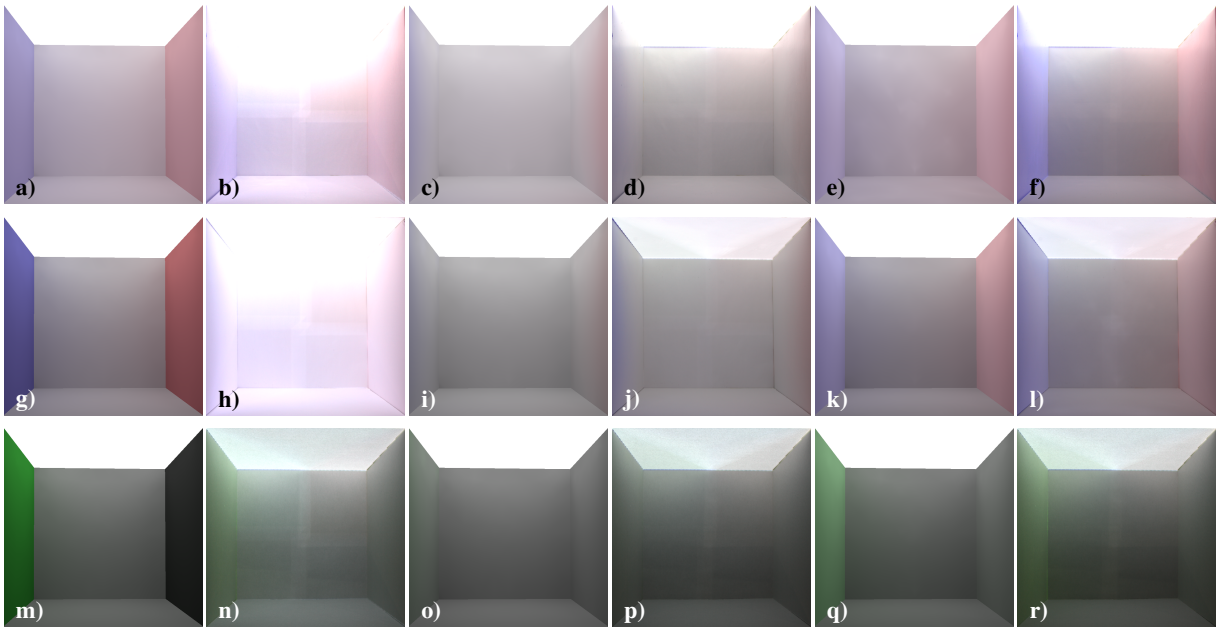


Figure 5: *Photographic comparisons within five-sided physical boxes with different reflectances. (a-f) Light pastel-colored virtual Cornell Box scene [red:<0.5,0.5,0.8> white:<0.836,0.836,0.836> blue:<0.5,0.5,0.8>] projected on very pale grey ($\rho = 0.836$) physical material. (g-l) Darker-colored virtual Cornell Box scene [red:<0.6,0.2,0.2> white:<0.615,0.615,0.615> blue:<0.2,0.2,0.6>] projected on light grey ($\rho = 0.615$) physical material. (m-r) A scene with a very dark virtual wall [green:<0.05,0.4,0.05> white:<0.4,0.4,0.4> black:<0.05,0.05,0.05>] projected on medium grey ($\rho = 0.365$) physical material. The physical and virtual camera exposures within each row are equivalent. The uncompensated projection images are overexposed due to indirect scattering and the colors are dull. The differences in color fidelity between the clamped inverse method and our QP optimization method are apparent and similar in both computer simulation and physical projection.*

10. Conclusions & Future Work

We have presented a constrained optimization framework to compensate for the inter-reflection of light in complex physical projection environments, such as Spatially Augmented Reality. Our method preserves the average intensity, color fidelity, and discontinuities of the desired imagery. Physical environments with different reflectivity are studied. We also demonstrate our compensation algorithm applied to architectural daylighting design and achieve interactive rendering speeds for the visualization of a variety of different direct lighting conditions.

We foresee a variety of continuing studies related to this projection compensation technique. Our current absolute error metrics do not apply with Weber-Fechner law, and relative error is a better metric. We find that the relative error can also be reformulated to a QP problem whose Hessian is positive definite. Comparing and refining results generated by these two metrics is an interesting future work area. We would also like to extend the compensation algorithm to hierarchical radiosity to further accelerate the computing speed. It may also be useful to extend the method to model and compensate for non-diffuse light transport within the system. Finally, for specific applications, such as architectural daylighting design, we can study the typical range of geometry and desired reflectance properties. This information can be used to recommend the best compromise surface albedos for the physical surfaces within the table-top SAR system.

Acknowledgments

The authors wish to thank Jie Chen for his suggestion to use the two-metric projection method and Chao Chen for valuable discussions about the error metrics. We also thank the anonymous reviewers for their helpful comments and corrections. This work was supported by NSF CMMI 0841319, NSF IIS 0845401, and a grant from IBM.

References

[AOSS06] ASHDOWN M., OKABE T., SATO I., SATO Y.: Robust content-dependent photometric projector compensation. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop* (Washington, DC, USA, 2006), IEEE Computer Society, p. 6.

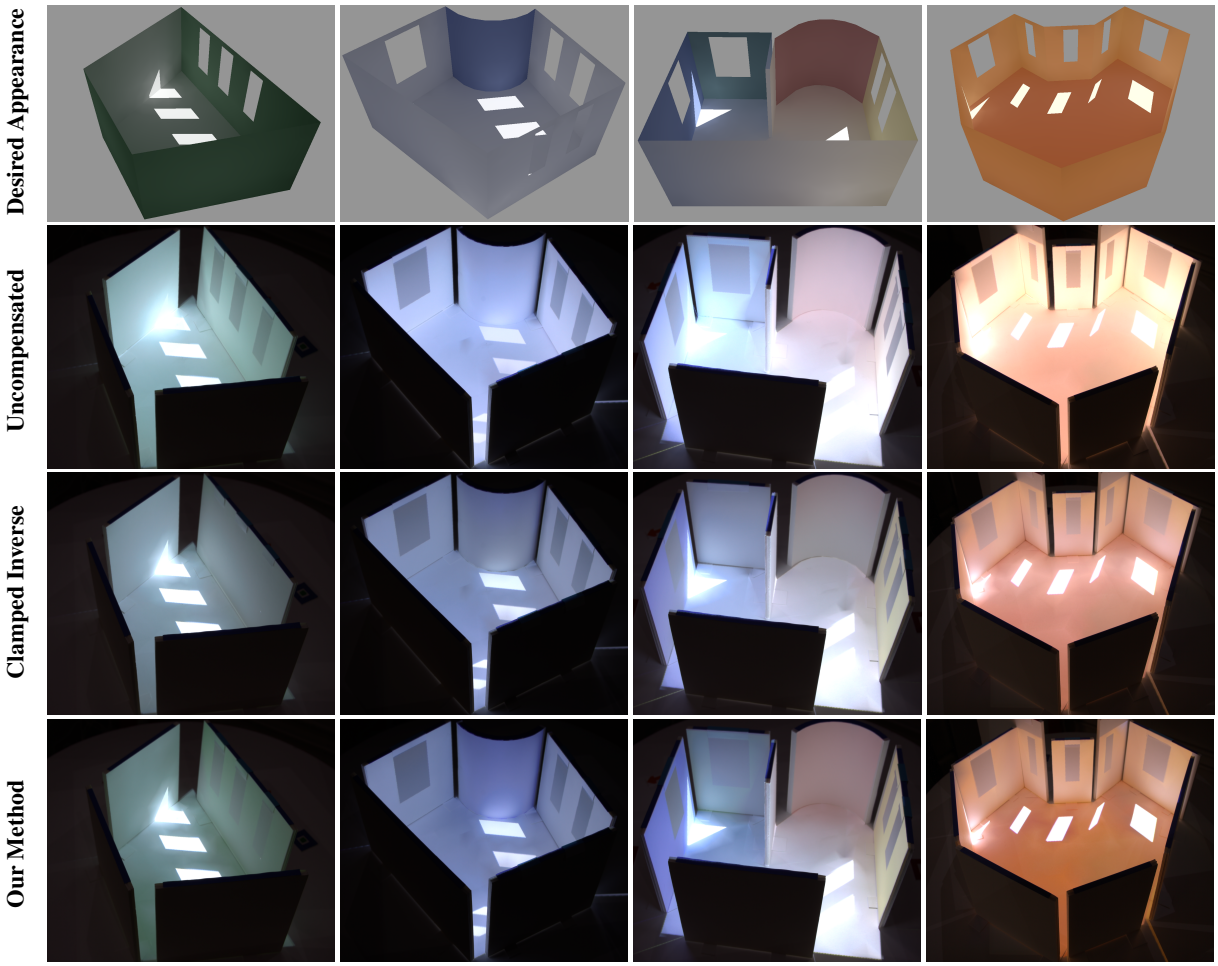


Figure 6: *Visualization of daylighting on March 21st at noon within four architectural designs. The top row presents a computer visualization of the virtual geometry (ceiling has been omitted) and desired global illumination simulation. Sun and skylight enter the model through simple rectangular windows within one or more walls. A variety of wall and floor colors have been specified by the user. The three bottom rows contain photographs of the corresponding physical scene. The table surface ($\rho = 0.95$) and the walls ($\rho = 0.9$) are constructed of bright white diffuse paper. The second row shows the result of simply projecting the desired imagery onto the scene. The third and fourth rows show the clamped inverse method and our QP optimization, respectively. To facilitate comparison of color fidelity, the camera exposure for the second row has been decreased by 25% relative to the third and fourth rows. Virtual scene materials from left to right: [green:<0.333,0.529,0.376> white:<1,1,1>], [blue:<0.306,0.38,0.659> white:<1,1,1>], [blue:<0.325,0.416,0.706> green:<0.322,0.482,0.439> red:<0.627,0.345,0.369> yellow:<1,0.973,0.624> white:<1,1,1>], and [floor:<0.7,0.5,0.4> walls:<0.8,0.7,0.5>]. In the clamped inverse simulation for the first and third scenes, no green color is projected on the green walls and thus the windows on those walls are not visible. In the second scene, the bottom of the blue curved wall is most vividly presented in our QP solution. Similarly, in the fourth scene, the saturated color of the floor and bottom portion of the walls is most accurately reproduced in with our method. The color of the red curved wall in the third scene is barely visible in the clamped inverse solution.*

[BEK05] BIMBER O., EMMERLING A., KLEMMER T.: Embedded entertainment with smart projectors. *Computer 38*, 1 (2005), 48–55.

[Ber82] BERTSEKAS D. P.: Projected newton methods for optimization problems with simple constraints. *SIAM Journal on Control and Optimization 20*, 2 (1982), 221–246.

[BGZ*06] BIMBER O., GRUNDHÖFER A., ZEIDLER T., DANCH D., KAPAKOS P.: Compensating indirect scattering for immersive and semi-immersive projection displays. In *VR '06: Proceedings of the IEEE conference on Virtual Reality* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 151–158.

[BR07] BIMBER O., RASKAR R.: Modern approaches to augmented reality, 2007. ACM SIGGRAPH 2007 Course Notes.

[CNR08] COSSAIRT O., NAYAR S., RAMAMOORTHI R.: Light-field transfer: Global illumination between real and synthetic objects. *ACM Transactions on Graphics 27*, 3 (July 2008).

[CNSD*92] CRUZ-NEIRA C., SANDIN D., DEFANTI T., KENYON R., HART J.: The cave: Audio visual experience automatic virtual environment. *Communications of the ACM 35*, 6 (June 1992), 65–72.

[Con02] CONTENSIN M.: Inverse lighting problem in radiosity. *Inverse Problems in Engineering 10*, 2 (January 2002), 131–152.

[DRB97] DRETTAKIS G., ROBERT L., BOUGNOUX S.: Interactive common illumination for computer augmented reality. In *Eurographics Rendering Workshop 1997* (June 1997), pp. 45–56.

[DSG91] DORSEY J. O., SILLION F. X., GREENBERG D. P.: Design and simulation of opera lighting and projection effects. In *Computer Graphics (Proceedings of SIGGRAPH 91)* (July 1991), pp. 41–50.

[FGN05] FUJII K., GROSSBERG M. D., NAYAR S. K.: A projector-camera system with real-time photometric adaptation for dynamic environments. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 814–821.

[FGR93] FOURNIER A., GUNAWAN A. S., ROMANZIN C.: Common illumination between real and computer generated scenes. In *Graphics Interface* (1993), pp. 254–263.

[GB08] GRUNDHÖFER A., BIMBER O.: Real-time adaptive radiometric compensation. *IEEE Transactions on Visualization and Computer Graphics 14*, 1 (Jan./Feb. 2008), 97–108.

[GM00] GIBSON S., MURTA A.: Interactive rendering with real-world illumination. In *Eurographics Workshop on Rendering (Rendering Techniques)* (June 2000).

[GMSW84] GILL P. E., MURRAY W., SAUNDERS M. A., WRIGHT M. H.: Procedures for optimization problems with a mixture of bounds and general linear constraints. *ACM Trans. Math. Softw. 10*, 3 (1984), 282–298.

[GTGB84] GORAL C. M., TORRANCE K. E., GREENBERG D. P., BATTAILE B.: Modelling the interaction of light between diffuse surfaces. In *Computer Graphics (Proceedings of SIGGRAPH 84)* (July 1984), vol. 18(3), pp. 213–222.

[HKYL06] HAN S. H., KIM J. H., YUN T. S., LEE D. H.: Extensible interface using projector-based augmentation. In *International Conference on Computer Graphics & Virtual Reality (CGVR)* (2006), pp. 132–137.

[HSM07] HABE H., SAEKI N., MATSUYAMA T.: Inter-reflection compensation for immersive projection display. In *Proceedings of IEEE International Workshop on Projector-Camera Systems (ProCams) (poster)* (2007).

[Kny91] KNYAZEV A. V.: A preconditioned conjugate gradient method for eigenvalue problems and its implementation in a subspace. In *International Ser. Numerical Mathematics, v. 96, Eigenwertaufgaben in Natur- und Ingenieurwissenschaften und ihre numerische Behandlung, Oberwolfach, 1990.* (1991), pp. 143–154.

[McL76] MCLAREN K.: The development of the CIE 1976 (L*a*b*) uniform colour-space and colour-difference formula. *Journal of the Society of Dyers and Colourists 92* (1976), 338–341.

[MKO06] MUKAIGAWA Y., KAKINUMA T., OHTA Y.: Analytical compensation of inter-reflection for pattern projection. In *VRST '06: Proceedings of the ACM symposium on Virtual reality software and technology* (New York, NY, USA, 2006), ACM, pp. 265–268.

[NKGR06] NAYAR S. K., KRISHNAN G., GROSSBERG M. D., RASKAR R.: Fast separation of direct and global components of a scene using high frequency illumination. *ACM Transactions on Graphics 25*, 3 (July 2006), 935–944.

[NPGB03] NAYAR S., PERI H., GROSSBERG M., BELHUMEUR P.: A Projection System with Radiometric Compensation for Screen Imperfections. In *ICCV Workshop on Projector-Camera Systems (PROCAMS)* (Oct 2003).

[NVI09] NVIDIA: CUDA Programming Guide 2.2, 2009. `http://www.nvidia.com/object/cuda_home.html`.

[RPG99] RAMASUBRAMANIAN M., PATTANAIK S. N., GREENBERG D. P.: A perceptually based physical error metric for realistic image synthesis. In *Proceedings of SIGGRAPH 99* (Aug. 1999), Computer Graphics Proceedings, Annual Conference Series, pp. 73–82.

[RWLB01] RASKAR R., WELCH G., LOW K.-L., BANDYOPADHYAY D.: Shader lamps: Animating real objects with image-based illumination. In *Rendering Techniques 2001: 12th Eurographics Workshop on Rendering* (June 2001), pp. 89–102.

[RZW02] RASKAR R., ZIEGLER R., WILLWACHER T.: Cartoon dioramas in motion: Apparent motion effects on real objects with image-based illumination. In *International Symposium on Non-photorealistic Animation and Rendering (NPAR)* (June 2002).

[SCG*05] SEN P., CHEN B., GARG G., MARSCHNER S. R., HOROWITZ M., LEVOY M., LENSCH H. P. A.: Dual photography. *ACM Transactions on Graphics 24*, 3 (Aug. 2005), 745–755.

[SDS*93] SCHOENEMAN C., DORSEY J., SMITS B., ARVO J., GREENBERG D.: Painting with light. In *Proceedings of SIGGRAPH 93* (Aug. 1993), Computer Graphics Proceedings, Annual Conference Series, pp. 143–146.

[SMK05] SEITZ S. M., MATSUSHITA Y., KUTULAKOS K. N.: A theory of inverse light transport. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 1440–1447.

[SYYC09] SHENG Y., YAPO T. C., YOUNG C., CUTLER B.: Virtual heliodon: Spatially augmented reality for architectural daylighting design. In *Virtual Reality Conference, 2009. VR 2009. IEEE* (March 2009), pp. 63–70.

[SYYC10] SHENG Y., YAPO T. C., YOUNG C., CUTLER B.: A spatially augmented reality sketching interface for architectural daylighting design. *To appear in IEEE Transactions on Visualization and Computer Graphics* (2010).

[VD08] VOLKOV V., DEMMEL J. W.: Benchmarking gpus to tune dense linear algebra. In *SC '08: Proceedings of the 2008 ACM/IEEE conference on Supercomputing* (Piscataway, NJ, USA, 2008), IEEE Press, pp. 1–11.

[WB07] WETZSTEIN G., BIMBER O.: Radiometric compensation through inverse light transport. In *PG '07: Proceedings of the 15th Pacific Conference on Computer Graphics and Applications* (Washington, DC, USA, 2007), IEEE Computer Society, pp. 391–399.

[YDMH99] YU Y., DEBEVEC P., MALIK J., HAWKINS T.: Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Proceedings of SIGGRAPH 99* (Aug. 1999), Computer Graphics Proceedings, Annual Conference Series, pp. 215–224.

[ZN06] ZHANG L., NAYAR S.: Projection defocus analysis for scene capture and image display. *ACM Transactions on Graphics 25*, 3 (July 2006), 907–915.