DSCC2014-6163

LIGHT FIELD ESTIMATION AND CONTROL USING A GRAPHICAL RENDERING ENGINE

Sina Afshari
Dept. of Electrical, Computer and Systems Eng.
Rensselaer Polytechnic Institute
Troy, New York 12180
Email: afshas@rpi.edu

Li Jia
Dept. of Electrical, Computer and Systems Eng.
Rensselaer Polytechnic Institute
Troy, New York 12180
Email: jial@rpi.edu

Richard J. Radke
Dept. of Electrical, Computer and Systems Eng.
Rensselaer Polytechnic Institute
Troy, New York 12180
Email: radke@ecse.rpi.edu

Sandipan Mishra
Dept. of Mechanical, Aerospace and Nuclear Eng.
Rensselaer Polytechnic Institute
Troy, New York 12180
Email: mishrs2@rpi.edu

ABSTRACT

State-of-the-Art feedback control of lighting depends on point sensor measurements for light field generation. However, since the occupant's perception depends on the entire light field in the room instead of the illumination at a limited set of points, the performance of these lighting control systems may be unsatisfactory. Therefore, it is critical to reconstruct the light field in the room from point sensor measurements and use it for feedback control of lights. This paper presents a framework for using graphical rendering tools along with point sensor measurements for the estimation of a light field and using these estimates for feedback control. Computer graphics software is used to efficiently and accurately model building spaces, while a game engine is used to render different lighting conditions for the space on the fly. These real-time renderings are then used together with sensor measurements to estimate and control the light field in the room using an optimization-based feedback control approach. We present a set of estimation algorithms for this purpose and analyze their convergence and performance limitations. Finally, we demonstrate closed loop lighting control systems that use these estimation algorithms and compare their relative performance, highlighting their benefits and disadvantages.

1 INTRODUCTION

Electric lighting accounts for 16% of the electricity consumption in the commercial sector in the United States, using 850 billion KWh annually. Similarly, in the residential sector, more than 550 billion KWh of electrical energy is used every year, making up 9% of total electrical energy consumption of this sector [1]. With the emergence of light emitting diodes (LEDs) as the new generation of lighting components and new lighting control strategies such as smart lighting systems, these numbers are expected to drop by 50% by 2017 [2]. This goal is achievable only by using new control strategies and cutting-edge technology for lighting simulation and control applications [3].

Prior work on lighting control has focused on the deployment of sensor networks to measure the light in specific locations throughout the illuminated space and using these measurements to drive the lighting system to a desired condition. These approaches have been used to achieve different objectives such as generation of high-quality lighting in terms of color rendering index (CRI) [4] or other metrics [5–7], incorporation of daylight in illumination to achieve energy saving [5, 6, 8, 9], resolution of conflict between occupants [6], and achieving plug-and-play functionality and uniform lighting across the floor by using dis-

tributed and decentralized control schemes [10, 11]. In these approaches, the control problem is typically posed as an optimization problem and is solved based on point sensor measurements in the space using a variety of optimization methods. The drawback to these approaches is that they make use of point sensor measurements instead of calculating and using the entire light field for feedback control. **Measurement of the entire light field and designing control algorithms that make use of the measured light field are among the current challenges of feedback control of the light field in the room.** Previously, Dong et al. proposed a systematic approach for tracking and estimation of the light field in the room using multi-scale representation of the light field [12].

The challenge in measuring the light field in the illuminated space is the need for a large number of sensors. **This paper presents a method to use environmental rendering in conjunction with point sensor measurements to estimate the dynamic light field in a room at discrete locations.** Computer graphics software is used to model the geometry, reflectance properties, and the layout of the space. This model is then used in a game engine platform to simulate the light field in the space in a dynamic fashion. Sensor measurements from the physical space are then used together with this dynamic simulation to estimate the actual light field. The estimated light field is then fed into a closed loop feedback algorithm, which optimizes a composite cost function consisting of a set of metrics including light quality (comfort) and power consumption. The contributions of this work are (1) estimation of the generated light field in the space based on the simulated light field and sensor measurements from the physical space, and (2) design of a closed loop control scheme that uses the estimated light field in order to optimize the generated light field.

The paper is organized as follows: In Section 2, light transport analysis is used to model the generation and measurement of the light field in a space and define the light field estimation and control problems. Section 3 presents a variety of methods to estimate the light field in the physical space based on the graphical rendering tool and sensor measurements from the space. In Section 4, various software packages used in this work for graphical rendering are introduced. Finally, Section 5 demonstrates the results from applying the light field estimation and feedback control to a physical space and analyzes the closed loop performance.

2 Light Field Estimation and Control Problem

The light field estimation problem in this work is posed as an optimization problem to minimize a cost function that evaluates the estimation accuracy based on graphical simulation of lighting and measured data. The light field control problem is posed as a separate optimization problem over the input of light fixtures to minimize a composite cost function. This cost function consists of several metrics evaluating the quality of the generated light field and power consumption. In the rest of this section, the mathematical model for the lighting plant is developed and the light field estimation and control problems are separately defined.

2.1 Light Transport Modeling

The lighting plant in this work is assumed to be a room with n light fixtures and m color sensors. Without loss of generality, one physical light fixture/sensor with p different channels is assumed to be p independent light fixtures/sensors located at the same spot in the room. For example, an RGB color sensor is assumed to be 3 color sensors measuring the red, green and blue contents of the light spectrum. Assuming that $y_k \in \mathbb{R}^m$ is the composite output vector representing the output light measurement at the k-th time-step and $u_k \in \mathbb{R}^n$ is the input light intensity control vector at the k-th time-step, we model the lighting plant as

$$y_k = Pu_k + W_y d_k + v_k \tag{1}$$

where $P \in \mathbb{R}^{m \times n}$ is the *light transport matrix*, or LTM, from the light input to the output measurement. $W_y \in \mathbb{R}^{m \times l}$ is a matrix defining the form of the disturbance light (daylight), and $d_k \in \mathbb{R}^l$ is a vector that specifies the intensity of the disturbance at time-step k. We call W_y a *disturbance spatial variation* matrix, or DSV matrix, for the output. Notice that the (j, i)-th entry in the LTM represents the effect of the i-th light fixture on the j-th sensor. Also $v_k \in \mathbb{R}^m$ is the measurement noise vector at the k-th time-step. In this work, v_k is assumed to have zero-mean uncorrelated Gaussian characteristics.

In this work, the light field in the room is assumed to be specified by a set of RGB values denoted by $x \in \mathbb{R}^q$. These are the values that a color sensor would measure if it was placed in each of q locations. The set of locations is assumed to be fixed for a space and can be a spatially uniform grid of points. We assume $q \gg m$ and thus, the light field in the room contains much more information regarding the lighting distribution in the room compared to the output. We also assume there is a linear transformation $C \in \mathbb{R}^{m \times q}$ that maps the light field in the room to the output measurements, $y = Cx$. Similar to (1), we obtain

$$x_k = Gu_k + W_x d_k \tag{2}$$

where $G \in \mathbb{R}^{q \times n}$ is the light transport matrix (LTM) from the input to the light field in the room. $x \in \mathbb{R}^q$ and $u_k \in \mathbb{R}^n$ are the light field vector and the light input vector to the fixtures at the k-th time-step. $W_x \in \mathbb{R}^{q \times h}$ is the DSV matrix for the light field and $d_k \in \mathbb{R}^h$ specifies the intensity of the disturbance at time-step k.

2.2 Light Field Estimation Problem

Based on the light transport analysis approach discussed above, the light generation equations for the actual space and the simulation tool are

$$
\begin{cases} x_k = Gu_k + W_x d_k \\ y_k = Cx_k + v_k \\ y_k = CGu_k + CW_x d_k + v_k \end{cases} \quad \begin{cases} \hat{x}_k = \hat{G}u_k + \hat{W}_x \hat{d}_k \\ \hat{y}_k = \hat{C}\hat{x}_k \\ \hat{y}_k = \hat{C}\hat{G}u_k + \hat{C}\hat{W}_x \hat{d}_k \end{cases} . \quad (3)
$$

The $(\hat{.})$ notation shows the corresponding variables in the simulated environment. In the absence of data from the actual space, these variables can be used as an estimate of the variables in the actual space (an a priori estimate). We use $\tilde{x}_k \in \mathbb{R}^q$ and $\tilde{d}_k \in \mathbb{R}^h$ to denote the estimates of x_k and d_k using both the simulation platform and the sensor measurements from the space (an a posteriori estimate).

The light field estimation problem is posed as an optimization problem over the variable $\tilde{x}$. The cost function, $O(\tilde{x}, \tilde{d}, u, y)$, uses the simulation tool together with various metrics to compare the estimated light field against the sensor measurements in different time-steps. The general estimation problem formulation in this paper is

$$
\begin{aligned} \underset{\tilde{x}_k}{\text{minimize}} \quad & O(\tilde{x}_k, \tilde{x}_{k-1}, \tilde{d}_k, \tilde{d}_{k-1}, u_k, y_k) \\ \text{subject to} \quad & \Omega \tilde{x}_k = \Upsilon_k . \end{aligned} \quad (4)
$$

where $\Omega \in \mathbb{R}^{m \times q}$ and $\Upsilon_k \in \mathbb{R}^m$ are respectively a matrix and a vector defining a linear constraint on $\tilde{x}_k$. Various methods to solve this problem are discussed in Section III.

2.3 Light Field Control Problem

The feedback control of the light field is posed as an optimization problem over the input u to minimize a cost function consisting of a light quality metric, μ_Q, and a power consumption metric, μ_E. Assuming x_{des} is the desired light field, the overall optimization problem is posed as

$$
\begin{aligned} \min_{u_k \in \mathscr{U}} J(u_k) &= \alpha_Q \mu_Q(\tilde{x}, x_{des}) + \alpha_E \mu_E(u_k) \\ \mu_Q &= (\tilde{x} - x_{des})^T \Gamma_Q^{-1} (\tilde{x} - x_{des}) \\ \mu_E &= \Gamma_E^{-1} u \end{aligned} \quad (5)
$$

where $\Gamma_Q \in \mathbb{R}^{q \times q}$ is a diagonal (positive definite) weighting matrix, and $\Gamma_E \in \mathbb{R}^{n \times n}$ is a diagonal matrix with the i-th entry on the main diagonal equal to the efficiency of the i-th light fixture. More details on specification of Γ_Q and Γ_E can be found in [5].

We now describe the feedback update law at the k-th time-step. Note that μ_Q is (locally) convex in u. The stationarity condition for μ_Q, $\nabla_u \mu_Q = 0$, is only satisfied at $\tilde{x} = x_{des}$, and the

Hessian $\nabla_u^2 \mu_Q$ is positive definite at that point. The power consumption metric μ_E is linear in u and is convex. Therefore, any gradient update law may be used to ensure convergence to the optimum. Similar to the method discussed in [5], we will use gradient update with projection for solving (5), which results in the control law

$$
u_{k+1} = P_{\mathscr{U}} \left[u_k - \varepsilon \left(\alpha_L G^T \Gamma_Q^{-1} (\tilde{x}_k - x_{des}) \right) + \alpha_E \Gamma_E^{-1} \right) \right] \quad (6)
$$

where $P_{\mathscr{U}}$ is the projection to ensure the lighting control is within the allowed range, the k subscript denotes the k-th time-step and ε is a sufficiently small step size. A schematic diagram of the system is shown in Figure 1. Notice that unlike [5], the light field is used here instead of the sensor measurements to generate the input to the light fixtures.
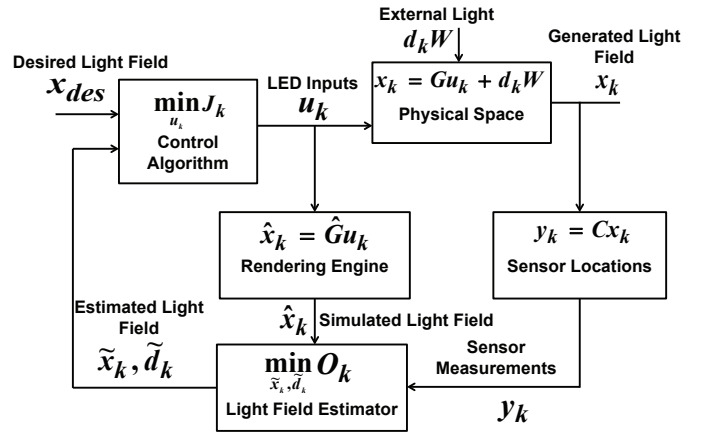

Figure 1. A schematic diagram of the closed loop controller and the estimator working in the feedback loop.

2.4 A Few Remarks

Remark I: Since we assumed $q \gg m$, measurement of the light field in the physical space is not feasible. The actual light field can only be measured in the simulation tool or estimated for the actual space.

Remark II: Unlike (1), there is no sensor noise included in (2). This is because the RGB values for x_k are not measured by actual color sensors.

Remark III: h is a measure of the spatial complexity of the disturbance. The most complex disturbance is $W = I_{q \times q} \Rightarrow h = q$, in which case the value of the disturbance in each of the q locations is independent of the others. However, for lighting applications typically $h \ll q$. This is due to the incoming daylight having a certain pattern which is governed by the number, size and location of the windows and/or the skylights.

3

Remark IV: P and G are assumed to have no dynamics (static). However, the problem cannot be solved offline using optimization techniques due to the dynamics introduced by the disturbance.

Remark V: In the case of P, measured input-output data can be used to identify the LTM. A typical approach is to fit a linear transformation to input-output data using the linear least-squares method [13]. Assuming $U = \begin{bmatrix} u^1 & u^2 & \dots & u^N \end{bmatrix}$ is a sequence of randomly generated light inputs and $Y = \begin{bmatrix} y^1 & y^2 & \dots & y^N \end{bmatrix}$ contains the corresponding sensor measurements, the least squares estimate of P is $\hat{P} = YU^+$ where U^+ is the Moore-Penrose pseudo-inverse of U. This assumes no disturbance. If the noise characteristics of the sensors are different, a weighting matrix must be included in the formulation of the least squares problem. Also, U has to be of full rank and well-conditioned for this method to obtain an accurate P.

3 Light Field Estimation Algorithms

This section presents different methods to estimate the light field generated in the room using the simulated light field in the software platform and sensor measurements from the actual space. Various cost functions and constraints are suggested for the estimation problem and the solution to the resulting optimization problem in each case is presented.

3.1 Minimum-Error Estimation with Output Matching

In order to use both the data obtained from the light field generation simulation in the space (i.e., using the simulation LTM or $\hat{G}$) together with the measured data from the room for estimation, the estimation optimization problem is defined as:

$$\underset{\tilde{x}_k}{\text{minimize}} \quad O_k = \left\| \tilde{x}_k - \hat{G}u_k \right\|_2^2$$
$$\text{subject to} \quad \hat{C}\tilde{x}_k = y_k. \tag{7}$$

The term *minimum-error* is used here since the cost function penalizes the difference between the estimated and the simulated light fields in an effort to minimize the estimation error, $\|x_k - \tilde{x}_k\|_2^2$. Note that this formulation represents a simplified Kalman filter problem with no dynamics. Assuming $M = \hat{C}^T \left(\hat{C}\hat{C}^T \right)^{-1}$, the estimated light field in this case is $\tilde{x}_k = My_k + \left(I - M\hat{C} \right)^{-1} \hat{G}u_k$. Although this method is expected to give an accurate estimation of the light field due to the form of the cost function, it can be sensitive to noisy measurements by the sensor because of the hard constraint. In order to avoid this issue, we introduce the next estimation method.

3.2 Minimum-Error Estimation with Soft Constraint Penalty on Output Matching

The minimum-error estimation with output matching relies heavily on the sensory data from the actual space due to the hard constraint. However, sometimes these data might be noisy. In order to make the estimation method more robust to noisy data, here the estimation problem is posed as an unconstrained optimization problem.

$$\underset{\tilde{x}_k}{\text{minimize}} \quad O_k = \left\| \tilde{x}_k - \hat{G}u_k \right\|_2^2 + \alpha \left\| \hat{C}\tilde{x}_k - y_k \right\|_2^2. \tag{8}$$

Similarly the problem formulation here also represents a simplified Kalman filter problem with no dynamics. The closed form solution in this case is $\tilde{x}_k = \left(I + \alpha \hat{C}^T \hat{C} \right)^{-1} \left(\hat{G}u_k + \alpha \hat{C}^T y_k \right)$. The tuning parameter α is used to determine the influence of the sensor measurements from the actual space on the light field estimate. With a large α, the estimation will be based on the sensor measurements whereas with a small α, the estimation relies on the simulation tool and sensory data is ignored.

3.3 Minimum-Error with Spatial Disturbance Model

The minimum-error estimation with soft constraint penalty on output matching does not incorporate the disturbance (external light in the actual space) into the estimation and is expected to give an erroneous estimate in the presence of disturbance. Therefore we now estimate not only the light field in the space ($\hat{x}$), but also the disturbance ($\hat{d}$). By incorporating a disturbance model, a more accurate estimate of the light field is expected to be achieved which can be used in closed loop to drive the generated light field in the room to a desired light field. The new optimization problem is defined as

$$\underset{\tilde{\theta}_k}{\text{minimize}} \quad O_k = \left\| A\tilde{\theta}_k - \hat{G}u_k \right\|_2^2 + \alpha \left\| B\tilde{\theta}_k - y_k \right\|_2^2 \tag{9}$$

where $\tilde{\theta}_k \in \mathbb{R}^{q+l}$ is a matrix equal to $[\tilde{x}_k \quad \tilde{d}_k]^T$, $A \in \mathbb{R}^{q \times q+l}$ is a matrix equal to $[I_{q \times q} \quad -W]$, $B \in \mathbb{R}^{m \times q+l}$ is $[\hat{C} \quad \bar{0}]$ and $\bar{0} \in \mathbb{R}^{m \times l}$ is a matrix with all entries equal to zero. Similarly, α serves as a tuning parameter and is specified by the trust level in the sensor measurements. The optimum in this case is obtained from $\tilde{\theta}_k = \left(A^T A + \alpha B^T B \right)^{-1} \left(A^T \hat{G}u_k + \alpha B^T y_k \right)$.

3.4 Minimum-Error with Spatial and Temporal Disturbance Model

The static estimation algorithms proposed in Sections 3.1 to 3.3 all suffer from sensitivity to noise. In order to filter the high temporal frequency content of the noise and add memory to the estimation method, the current data as well as past data have to

4

be taken into account to determine the new light field estimate. Here, the new estimation problem is defined as

$$\underset{\tilde{\theta}_k}{\text{minimize}} \quad O_k = \left\|A\tilde{\theta}_k - \hat{G}u_k\right\|_2^2 + \alpha \left\|B\tilde{\theta}_k - y_k\right\|_2^2 \tag{10}$$
$$+ \beta \left\|d_k - d_{k-1}\right\|_2^2 .$$

This yields

$$\underset{\tilde{\theta}_k}{\text{minimize}} \quad O_k = \left\|A\tilde{\theta}_k - \hat{G}u_k\right\|_2^2 + \alpha \left\|B\tilde{\theta}_k - y_k\right\|_2^2 \tag{11}$$
$$+ \beta \left\|F\tilde{\theta}_k - F\tilde{\theta}_{k-1} - \hat{G}u_k + \hat{G}u_{k-1}\right\|_2^2 .$$

$F \in \mathbb{R}^{q \times q+l}$ is $[I \quad \bar{0}]$ and $\tilde{\theta}$, A and B are defined similarly to the minimum-error with spatial disturbance model method. Also, the subscripts k and $k-1$ denote the current and the previous time-steps respectively. Notice that this formulation represents a moving horizon estimator problem [14]. Assuming $N = \left(A^T A + \alpha B^T B + \beta F^T F\right)^{-1}$, the estimated light field can be obtained recursively from

$$\tilde{\theta}_k = N\left((A^T + \beta F^T)\hat{G}u_k + \alpha B^T y_k + \beta F^T(F\tilde{\theta}_{k-1} - \hat{G}u_{k-1})\right). \tag{12}$$

Claim: Using the recursive algorithm in (12) and assuming u_k to be bounded, $\tilde{\theta}_k$ always converges.

Proof. From (12),

$$\tilde{\theta}_k = \left((A^T A + \alpha B^T B + \beta F^T F)^{-1}(\beta F^T F)\right)\tilde{\theta}_{k-1} \tag{13}$$
$$+ \text{other independent terms.}$$

Therefore, $\tilde{\theta}$ converges if and only if $\sigma_{max} = \bar{\sigma}\left((A^T A + \alpha B^T B + \beta F^T F)^{-1}(\beta F^T F)\right) < 1$ where $\bar{\sigma}(.)$ denotes the largest singular value. Note that from the definition of F, $\bar{\sigma}(F) = \underline{\sigma}(F) = 1$ where $\underline{\sigma}(.)$ denotes the smallest singular value. Thus,

$$\sigma_{max} = \frac{1}{1 + \underline{\sigma}(A^T A + \alpha B^T B)} \tag{14}$$

Given that $A^T A + \alpha B^T B$ is a positive definite matrix, $\underline{\sigma}(A^T A + \alpha B^T B) > 0$ and therefore, $\sigma_{max} < 1$.

3.5 Sensitivity to Model Uncertainty
The above estimators use $\hat{G}u_k$ and y_k as input to estimate the light field. Since in the absence of disturbance in simulation, $\hat{G}u_k = \hat{x}_k$, and since y_k is obtained directly from the sensors, no identification of the plant is required for the light field estimation. However, the drawback is that accurate modeling of the physical space in the rendering software is required. Let the light transport matrix of the physical system be $\hat{G} + \Delta\hat{G}$, where $\hat{G}$ is the light transport matrix in the graphical rendering model. Using the minimum-error with spatial and temporal disturbance model estimator,

$$(\Delta\theta_k)_{ss} = (A^T A + \alpha B^T B + \beta F^T F)^{-1}(A^T \Delta\hat{G}u_{ss}) \tag{15}$$

where $(\Delta\theta_k)_{ss}$ is the steady state error in the light field estimate.

4 Using Graphical Rendering Software for Finding $G\hat{u}$
In the previous section, different algorithms were suggested to use $\hat{G}u$ together with data from the physical space to obtain an a posteriori estimate for the generated light field. In this work, we use computer simulations via graphics software to calculate $\hat{G}u$. This section introduces the software we used for this task and describes the simulation procedure.

Various simulation programs are used by architects and lighting designers to examine and evaluate the performance of lighting systems in buildings [15–17]. One of the most commonly used software packages for lighting design, simulation and analysis purposes is Radiance [18]. However, a critical functionality for our application is high-quality, high-accuracy dynamic rendering to represent the reactions of the lighting control system to external disturbance in real time. The command-based structure of Radiance and its requirement of detailed inputs to set up a scene made this functionality infeasible.

In this work, the 3ds Max and Maya software packages by Autodesk were used to create the environmental model. These tools enable high-quality and high-speed visualization of detailed environments with complex texturing and lighting [19,20]. Critical factors in our choice were Maya's capability to realistically simulate lighting panels that evenly emit photons and 3ds Max's powerful and detailed daylighting system. Moreover, in order to add various capabilities to our software platform such as convenience of environmental change, importing external data and measuring and controlling object properties in real time, we investigated computer game design tools. The Unity3D Pro game engine is well-suited for these purposes and it also enables human-machine interaction via scripting. Previously, Jia and Radke [21] used Unity3D Pro to virtually collect time-of-flight data in a synthetic environment and validate occupancy tracking algorithms.

Our simulation methodology is to first create detailed and accurate room models in Maya, taking into account the geometry and the material properties of the surfaces. Next we create light objects in the same positions as the real lights in the room models in Unity3D Pro. These directional, point, and spot lights are able

to generate illumination at real-time frame rates (i.e., 30Hz or more), but only for rays directly from a light source to a surface. Thus, reflections from surfaces are not captured in these renders. The next step is building off-line renders of the lighting in the room due to the light fixtures (using Maya) and daylight (using 3ds Max). These renders can be saved and reused as texture images called *lightmaps*, which are images of all the geometric surfaces in the space with their illumination characteristics. Finally, the lighting in the room is simulated by linearly combining the individual lightmaps with weightings based on the light fixture inputs and importing the composite lightmap into Unity3D Pro. For specification of the disturbance (daylight), various data such as time of the day and weather condition can be used. The process is further detailed in [22].

5 Simulation Results and Discussion

This section presents the performance results for the closed loop control algorithm described in Section II-C coupled with each of the estimation methods discussed in Section III. In order to save space, the short notations presented in Table 1 are used to refer to each of the methods.

Estimation Algorithm	Notation
Minimum-Error Estimation with Output Matching	III.A Method
Minimum-Error Estimation with Soft Constraint Penalty on Output Matching	III.B Method
Minimum-Error with Spatial Disturbance Model	III.C Method
Minimum-Error with Spatial and Temporal Disturbance Model	III.D Method

Table 1. The notation used to refer to each of the estimation methods.

A three dimensional model of the Adaptive Lighting Testbed, a testbed within the Smart Lighting Engineering Research Center at Rensselaer Polytechnic Institute, was made using Autodesk Maya and Autodesk 3ds Max Design [23]. This space is a $12' \times 8' \times 12'$ furnished room with windows and a simulated skylight. There are twelve color-tunable RGB LED light fixtures from Acuity [24] on the ceiling. Also, twelve wireless color sensors from Ocean Optics [25] are attached to the ceiling near the light fixtures and provide RGB measurements from different points in the room. Figure 2 illustrates the actual testbed and the model created by the software. Figure 3 depicts a layout plan of the room from the top. This Figure illustrates the location of the light fixtures (u), RGB sensors (y) and the location of

the RGB values that characterize the light field ($\hat{x}$). It can be assumed that $\hat{x}$ is measured by a set of (virtual) RGB color sensors attached to the ceiling at the locations depicted in Figure 3.

The control algorithm was implemented in MATLAB. The Unity3D Pro game engine was used to realize different lighting conditions in the simulation. Moreover, light transport analysis was used to mathematically model the testbed. This model was implemented in MATLAB to simulate the sensor readings from the actual testbed. The simulation results are shown using three graphs illustrating the 2-norm of the estimation error (i.e., $\|x - \tilde{x}\|_2^2$), light field convergence (i.e., x compared to x_{des}) and the estimation cost function (i.e., O).
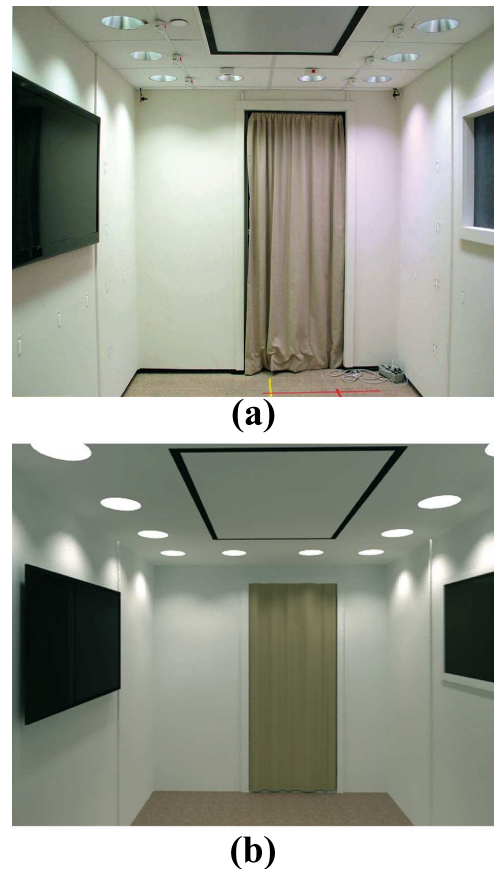

(a)

(b)

Figure 2. The Adaptive Lighting Testbed used to simulate the performance of the light field estimator and closed loop control algorithm, (a) actual testbed, (b) simulated three dimensional model.

5.1 Feedback Control Using III.A

In the absence of disturbance, the estimation error ($\|x - \tilde{x}\|_2^2$) is trivially always equal to zero. This is due to the cost function
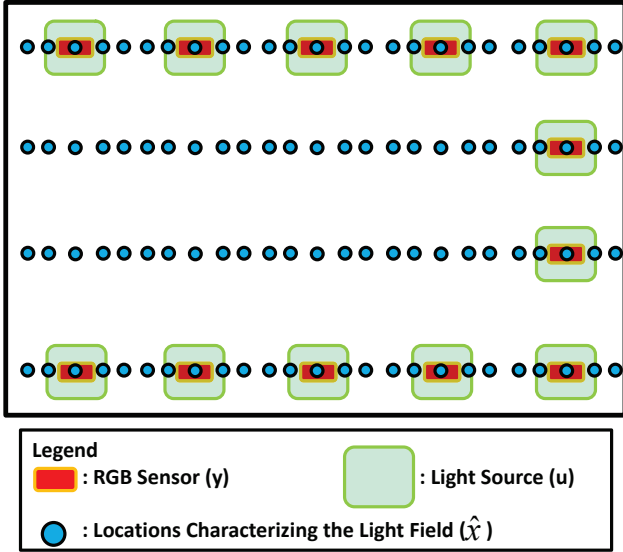
Figure 3. A layout plan of the Adaptive Lighting Testbed together with locations of the RGB sensors, light fixtures and the simulated virtual sensors that measure the light field in the simulation tool.
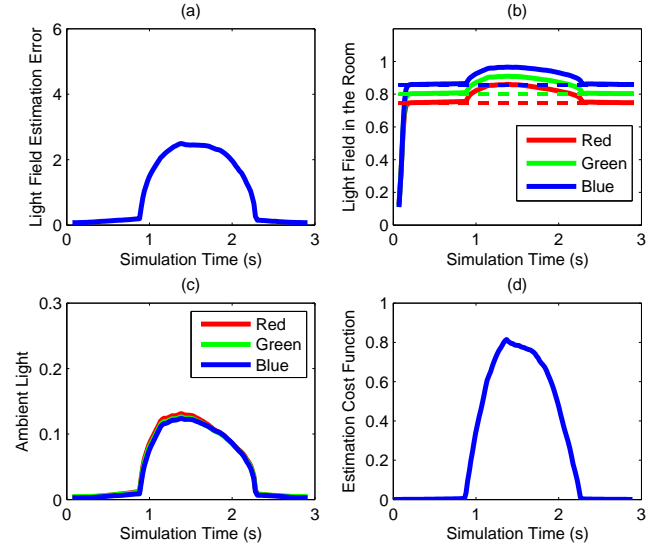

Figure 4. Simulation results for the III.A and III.B methods in the presence of disturbance, (a) 2-norm of the estimation error (i.e., $\|x - \tilde{x}\|_2^2$), (b) light field convergence (i.e., x compared to x_{des}) for a sample location, (c) ambient light disturbance (i.e., Wd) for a sample location, (d) the estimation cost function (i.e., O).

penalizing the difference norm between the simulated and estimated light fields. Using this estimate for feedback control, the generated light field converges to the desired light field.

The simulation results using this method in the presence of disturbance are shown in Figures 4. In this case, there exists a large estimation error as shown in Figure 4(a). Here, the ambient light is added to the light generated by the fixtures to make up the total light field in the room. In the rendering tool, the ambient light is not modeled and therefore the rendered light field only represents the light generated by the fixtures. This makes the a posteriori estimate highly erroneous. Using this estimate for feedback control results in a large difference between the desired light field and the generated light field, as shown in Figure 4(b). Notice that the light field is two dimensional in nature. However, in order to save space, only the RGB values for one location are illustrated to represent the estimator and controller performance.

5.2 Feedback Control Using III.B

The simulation results for this method are identical to the III.A method for both of the scenarios with and without disturbance and therefore, are not repeated here. In the absence of disturbance, the estimated light field converges to the generated light field and using this estimate for feedback control, the generated light field converges to the desired light field ($\tilde{x} \rightarrow x$ and $x \rightarrow x_{des}$). In the presence of disturbance, the light field estimate has a large error compared to the light field in the room as shown in Figure 4(a) (due to the effect of ambient light). Using this estimate for feedback control, the generated light field fails to

converge to the desired light field (Figure 4(b)).

5.3 Feedback Control Using III.C

In the absence of disturbance, the simulation results for this method are identical to the III.A method and therefore are not repeated here. In the presence of disturbance, the estimation error is much smaller than the previous approaches but still not zero, as shown in Figure 5(a). The reduced estimation error is due to the fact that in this approach, the ambient light is separately estimated based on the measurements from the actual system and therefore, it does not distort the a posteriori light field estimate. This method uses the DSV matrix to generate the control input. In practice, calculating this matrix with high accuracy can be challenging. We assumed 20% variation in this matrix to account for the error resulting from measurement difficulty. The nonzero estimation error is due to this variation assumption. The result for using this estimate for feedback control of light field is shown in Figure 5(b). Using this method reduces the difference between x and x_{des} significantly.

5.4 Feedback Control Using III.D

The simulation results for this method in the absence and presence of disturbance are identical to the III.C method and therefore are not repeated here. Figures 6(a) and (b) show simulation results for using the III.D method in the presence of distur-
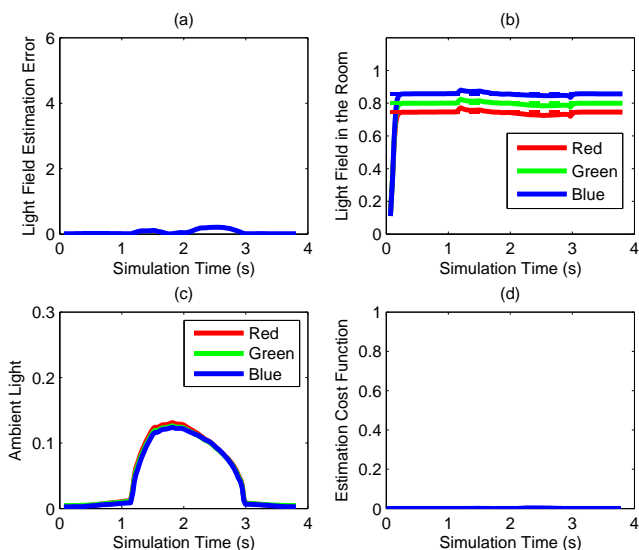
Figure 5. Simulation results for the III.C method in the presence of disturbance, (a) 2-norm of the estimation error (i.e., $\|x - \tilde{x}\|_2^2$), (b) light field convergence (i.e., x compared to x_{des}) for a sample location, (c) ambient light disturbance (i.e., Wd) for a sample location, (d) the estimation cost function (i.e., O).
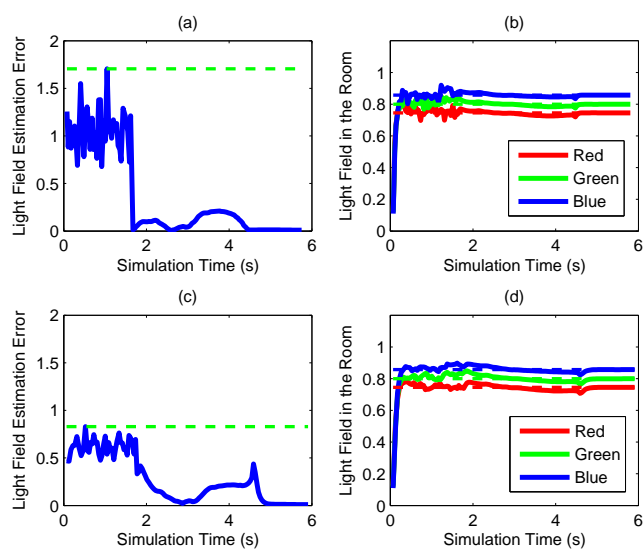

Figure 6. Simulation results in the presence of disturbance and measurement noise, (a) 2-norm of the estimation error (i.e., $\|x - \tilde{x}\|_2^2$) using the III.C method, (b) light field convergence (i.e., x compared to x_{des}) for a sample location using the III.C method, (c) 2-norm of the estimation error using the III.D method, (d) light field convergence for a sample location using the III.D method.

bance and measurement noise. The noise is assumed to be zero-mean uncorrelated Gaussian with standard deviation of 0.05 and is present in the time interval $[0 \quad 1.8]$ seconds. The maximum estimation error in this case is 1.7049, as shown in Figure 6(a). Using the III.D method in the presence of similar disturbance and noise reduces the maximum estimation error to 0.8294 as shown in Figure 6(c). Therefore, penalizing the high temporal variation in the light field estimate reduced the estimation error by more than a half.

5.5 Estimation and Control with Model Uncertainty

In this section, we first simulated a lighting scenario where the LTM of the physical space from the input to the light field is different from that of the graphical rendering engine ($\Delta\hat{G} > 0$) using three different approaches and compared the results. The purpose is to observe the effect of uncertainty in the LTM on the estimator and controller performance. In order to do this, the top half entries of the LTM of the physical space are assumed to be 10% smaller than those in the LTM of the rendering tool. Also, the sensor measurement noise is assumed to be a white Gaussian noise with $\sigma = 0.1$.

In the first simulation, the estimator is taken out of the loop and the perturbed model is directly used to perform the control (no estimation). In the second simulation, the estimator is in the loop, but uses the perturbed model instead of the real model to estimate the light field without using sensory data (only a priori

estimate). This estimate is then used to close the loop for control. In the third simulation, the estimator uses the perturbed model together with sensory data from the testbed for light field estimation (a posteriori estimate). The results from these simulations are shown in Table 2.

Estimation Algorithm	Steady-State Estimator Error	Steady-State Controller Error
No Light Field Estimation	NA	88.9837
Estimation Based only on the Simulation Tool	0.5891	63.0339
Estimation Based on the Simulation Tool and Sensory Data	0.3362	42.6046

Table 2. Simulation results with a fixed 10% variation in the first half entries of the LTM for the physical plant compared to the LTM for the rendering tool.

The second column shows the steady-state two-norm of the

difference between the estimated light field and the measured light field vectors. Since there is no estimation in the first case, this difference does not exist. The third column shows the steady-state value of the control cost function introduced in (5). These simulations show that even in the presence of model uncertainty, using the simulation platform for light field estimation and control results in a better control performance compared to not using any estimation. Also, among the two estimation approaches, the one that uses both the simulation tool and the sensory data results in a better estimation and control performance.

Next, two lighting scenarios, one with large model uncertainty and small sensor measurement noise and the other with small model uncertainty and large sensor measurement noise, are simulated. Each of these scenarios is simulated using various values for α. Here, all the simulations use the minimum-error with spatial and temporal model estimator. The purpose of these simulations is to show that the parameter α can be used to tune the level of trust to either the model or the sensory data. As shown in Table 3, with large model uncertainty and small sensor noise, the large α results in a better estimation. However, with small model uncertainty and large noise, the small α performs better.

	$\alpha = 10$	$\alpha = 1$	$\alpha = 0.1$	$\alpha = 0.01$
$\Delta \hat{G} = 50\%$ $\sigma = 0.01$	$e_{ss} = 1.10$	$e_{ss} = 1.11$	$e_{ss} = 1.15$	$e_{ss} = 1.21$
$\Delta \hat{G} = 40\%$ $\sigma = 0.1$	$e_{ss} = 1.25$	$e_{ss} = 1.09$	$e_{ss} = 1.06$	$e_{ss} = 1.03$

Table 3. The steady state 2-norm of the estimation error (e_{ss}) for the second set of simulations with model uncertainty. The first row represents large model uncertainty and small sensor noise while the second row represents small model uncertainty and large sensor noise.

6 Conclusions and Future Work

In this work, we investigated the problem of estimating the light field in a space using a three-dimensional environmental model and measurements from a sensor network. Graphics and game design software were used to simulate a physical lighting testbed. Different estimation approaches were investigated and their performances were compared in a closed loop lighting control algorithm. It was shown that by using the appropriate estimator, the generated light field in the room can be driven to a desired light field even in the presence of the disturbance and sensor noise. Future directions include implementation of these estimation and control algorithms in the actual Adaptive Lighting Testbed.

REFERENCES

[1] EIA, 2012. Annual energy outlook 2013 early release. Tech. Rep. DOE/EIA-0383ER(2013), Dec.

[2] DOE, 2012. Life-cycle assessment of energy and environmental impacts of led lighting products. Tech. rep., Feb.

[3] Lee, E., and Selkowitz, S., 1994. "The design and evaluation of integrated envelope and lighting control strategies for commercial buildings". In American Society of Heating, Refrigerating and Air- Conditioning Engineers (ASHRAE) winter meeting and exhibition,Chicago, IL (United States),28 Jan - 1 Feb 1995.

[4] Aldrich, M., Zhao, N., and Paradiso, J., 2010. "Energy efficient control of polychromatic solid state lighting using a sensor network". In Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 7784, pp. 778408–778408–15.

[5] Afshari, S., Mishra, S., Julius, A., Lizarralde, F., Wason, J. D., and Wen, J. T., 2013. "Modeling and control of color tunable lighting systems". *Energy and Buildings,* **68, Part A**, pp. 242 – 253.

[6] Singhvi, V., Krause, A., Guestrin, C., Garrett Jr., J., and Matthews, H., 2005. "Intelligent light control using sensor networks". In Proceedings of the 3rd international conference on Embedded networked sensor systems, SenSys '05, ACM, pp. 218–229.

[7] Park, H., Burke, J., and Srivastava, M., 2007. "Design and implementation of a wireless sensor network for intelligent light control". In Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on, pp. 370 –379.

[8] Afshari, S., Mishra, S., A., J., Lizarralde, F., and Wen, J., 2012. "Modeling and feedback control of color-tunable led lighting systems". In Proceedings of the 2012 American Control Conference (ACC), pp. 3663–3668.

[9] Wen, Y., Granderson, J., and Agogino, A., 2006. "Towards embedded wireless-networked intelligent daylighting systems for commercial buildings". *Sensor Networks, Ubiquitous, and Trustworthy Computing, International Conference on,* **1**, pp. 326–331.

[10] Afshari, S., and Mishra, S., 2013. "Decentralized feedback control of smart lighting systems". In Proceedings of the ASME Dynamic Systems and Control Conference.

[11] Koroglu, M., and Passino, K., 2014. "Illumination balancing algorithm for smart lights". *Control Systems Technology, IEEE Transactions on,* **22**(2), March, pp. 557–567.

[12] Dong, F., Hombal, V., and Sanderson, A., 2012. "Adaptive light field sampling and sensor fusion for smart lighting control". In Information Fusion (FUSION), 2012 15th International Conference on, pp. 40–47.

[13] Sen, P., and Darabi, S., 2009. "Compressive dual photography". *Computer Graphics Forum,* **28**(2), pp. 609–618.

[14] Rao, C., Rawlings, J., and Mayne, D., 2003. "Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations". *Automatic Control, IEEE Transactions on,* **48**(2), Feb, pp. 246–258.

[15] Krietemeyer, E., 2013. "Dynamic design framework for mediated bioresponsive building envelopes". PhD thesis, Rensselaer Polytechnic Institute, Troy, NY.

[16] Sheng, Y., 2013. "Interactive daylight visualization in spatially augmented reality environments". PhD thesis, Rensselaer Polytechnic Institute, Troy, NY.

[17] Clarke, J., 2012. *Energy simulation in building design.* Routledge.

[18] Larson, G. W., Shakespeare, R., Ehrlich, C., Mardaljevic, J., Phillips, E., and Apian-Bennewitz, P., 1998. *Rendering with Radiance: the art and science of lighting visualization.* Morgan Kaufmann San Francisco, CA.

[19] Cusson, R., and Cardoso, J., 2012. *Realistic Architectural Visualization with 3ds Max and mental ray.* Focal Press.

[20] Lanier, L., 2006. *Advanced Maya texturing and lighting.* Sybex.

[21] Jia, L., and Radke, R. J., 2014. "Using time-of-flight measurements for privacy-preserving tracking in a smart room". *IEEE Transactions on Industrial Informatics,* **10**(1), February, pp. 689–696.

[22] Jia, L., Afshari, S., Mishra, S., and Radke, R. J., 2014. "Simulation for pre-visualizing and tuning lighting controller behavior". *Energy and Building,* **70**, pp. 287–302.

[23] Afshari, S., Mishra, S., Wen, J., and Karlicek, R., 2012. "Demo abstract: An adaptive smart lighting system". In Proceedings of the 2012 Buildsys Conference, pp. 201–202.

[24] LED downlight, vivia round. `http://www.gothamlighting.com/product/detail.aspx?id=119170`.

[25] Ocean optics, colorbug color measurement appliance. `http://www.oceanoptics.com/products/colorbug.asp`.