

# Intelligent Lighting Control using Wireless Sensor Networks for Media Production

**Heemin Park<sup>1</sup>, Jeff Burke<sup>2</sup> and Mani B. Srivastava<sup>3</sup>**

<sup>1</sup> Image Development Team, Samsung Electronics, Co., Ltd., Korea  
[e-mail: hmpark@samsung.com]

<sup>2</sup> REMAP, University of California, Los Angeles, Los Angeles, CA 90095 – USA  
[e-mail: jburke@remap.ucla.edu]

<sup>3</sup> Networked & Embedded Systems Lab., University of California, Los Angeles, CA 90095 - USA  
[e-mail: mbs@ee.ucla.edu]

\*Corresponding author: Heemin Park

*Received July 22, 2009; revised September 3, 2009; accepted September 11, 2009;  
published October 30, 2009*

---

## Abstract

We present the design and implementation of a unique sensing and actuation application -- the Illuminator: a sensor network-based intelligent light control system for entertainment and media production. Unlike most sensor network applications, which focus on sensing alone, a distinctive aspect of the Illuminator is that it closes the loop from light sensing to lighting control. We describe the Illuminator's design requirements, system architecture, algorithms, implementation and experimental results. The system uses the Illumimote, a multi-modal and high fidelity light sensor module well-suited for wireless sensor networks, to satisfy the high-performance light sensing requirements of entertainment and media production applications. The Illuminator system is a toolset to characterize the illumination profile of a deployed set of fixed position lights, generate desired lighting effects for moving targets (actors, scenic elements, etc.) based on user constraints expressed in a formal language, and to assist in the set up of lights to achieve the same illumination profile in multiple venues. After characterizing deployed lights, the Illuminator computes optimal light settings at run-time to achieve a user-specified actuation profile, using an optimization framework based on a genetic algorithm. Uniquely, it can use deployed sensors to incorporate changing ambient lighting conditions and moving targets into actuation. Experimental results demonstrate that the Illuminator handles various high-level user requirements and generates an optimal light actuation profile. These results suggest that the Illuminator system supports entertainment and media production applications.

---

**Keywords:** Wireless sensors networks, light control, ubiquitous computing, entertainment and media production

---

A preliminary version of this paper appeared in the ACM/IEEE International Conference on Information Processing in Sensor Networks, 2007 [6]. This version presents further analysis and supporting implementation results.

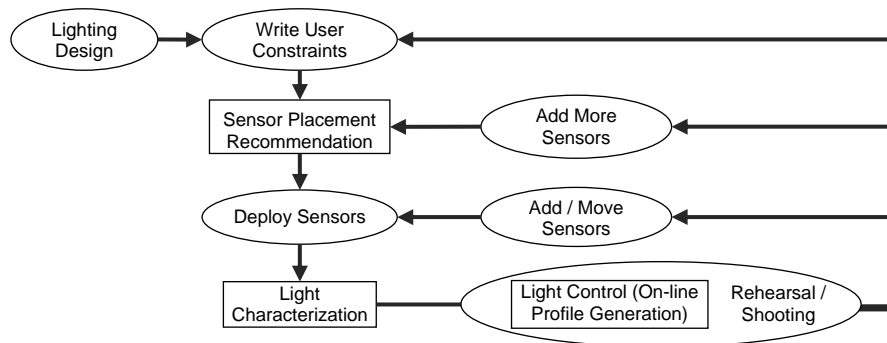
**DOI: 10.3837/tiis.2009.05.001**

## 1. Introduction

Wireless Sensor Network (WSN) technologies have enabled many interesting applications in pervasive and ubiquitous computing [1]. A promising application is UCLA's exploration of WSN for entertainment, multimedia and media production [2]. The Advanced Technology for Cinematography (ATC) [3] project, a collaboration between the Networked and Embedded Systems Laboratory (NESL) and the Center for Research in Engineering, Media and Performance (REMAP), explores WSN support for filmmaking. The ATC project seeks to enhance entertainment production and provide both increased expressive capabilities and significant cost savings by deploying sensor networks on film sets. In the ATC project, we first focused on capturing and archiving sensory data from the set into a database in frame-rate synchronization using WSN. To do so, we developed the Augmented Recording System (ARS) [4] with wireless light sensing module and network platform [5]. With the ARS, we demonstrated the possibilities and usefulness of wireless sensing in filmmaking. Although we showed that wireless sensing itself provides many benefits, it was a one-way data collection. Many types of equipment, such as lights, audio and cameras, are controlled by crews on film sets or in theaters. Many of these use advanced digital control systems but do not incorporate sensing. We wanted to explore the possibilities of using WSN technologies to actuate and control such equipment, not just monitor them. Since lighting is vitally important in film and theater, while being relatively straightforward to control, we sought to develop an intelligent light control system for entertainment and media production using WSN. In [6], we have initially explored the feasibility of using WSN for light control applications, especially, media production and entertainment. In this paper, we present further experiments and explanation of the light control system for media production.

### 1.1 Intelligent Light Control System for Entertainment and Media Production

Although computerized control systems for lights in film and theaters are available as commercial products [7][8], most current systems only provide actuation and do not exploit sensor data. We believe that it is important to know and use the live light information from light sensors deployed on the set. Real-time data accounts for how characteristics, such as light intensity and color temperature, change over time and deployments due to filament aging, supply voltage variation, changes in fixture position, and color filters. Without real-time measurement of light, it is time-consuming to maintain desired light intensities in certain area across many venues and over long periods. Light intensities and color temperature can be measured accurately by currently available handheld manual light meters [9][10]. However, these devices have not been incorporated in computer systems supporting automatic light control and must be manually moved through different points in space. Cameras can provide only reflected light intensity, so we focus on incident light in order to have measurements that are independent of surfaces and materials. On the set or in theaters, many types of equipment, for example, lights, audio and cameras are controlled by crews. Therefore, we wanted to explore the possibilities of using wireless sensors not only to wirelessly collect vital information, but also to actuate and intelligently control such equipment based on sensor readings. As lighting is important in film and media production, and it is relatively easy to control, we have developed an intelligent light control system. Illuminator finds and manages the best light actuation profiles for user requirements using incident light measurements by wireless sensors. This light control framework will also be useful for many illumination



**Fig. 1.** Usage Scenario of the Illuminator System

purposes, such as indoor office, home and museum lighting or outdoor lighting for a stadium and advertisements. However, compared to such illumination applications, the requirements of entertainment and media production lighting are unique and far more challenging, because media production lighting changes dynamically and incorporates aesthetic aspects.

## 1.2 Capabilities of the Intelligent Light Control System

With the intelligent light control system, the proposed system (the Illuminator) finds and maintains the best light settings satisfying the specified user requirements using WSN. Then, the Illuminator becomes a toolset to help media production crews characterize, control and setup lights at the time of performance or filmmaking. The Illuminator should have three key capabilities for this: *given a light setup and user constraints, 1) recommend sensor deployment, 2) characterize the lights, and 3) based on light characteristics and sensor readings, find and manage the best light actuation profiles satisfying user constraints.* These constraints represent requirements of the lighting's aesthetic effect and include desired light intensities in the field and/or a high-level description of lighting conditions. These are discussed further in Section 2.1. We assume that lights have fixed positions over the time of the Illuminator's control; however, our system does not require knowledge of characteristics and locations of lights. Although we could use pan-tilt mounts and automated lighting that can move the direction of lights and control color, we did not consider these features for our system. Tracking and spotlighting using pan-tilt mounts is a well-known technology and can be implemented easily [11]. We do allow mobile stage elements, equipment and actors lit by these fixed lighting instruments, using mobile tags. Tag is a single entity that can sense light intensity and know its location. Brief descriptions of the Illuminator's three capabilities follow.

*Sensor Placement Recommendation:* The Illuminator recommends the best sensor locations using a limited number of available light sensors. Uniform and regular sensor placement will work well in general cases. However, if the user has specific light intensity patterns that they wish to generate, the sensor placement should change accordingly. The Illuminator system parses user constraints and shows the recommended sensor locations in a GUI display.

*Light Characterization:* Knowledge of the projection pattern of lights and brightness according to dimmer levels is required to generate desired light levels at specific locations. We term this information the *Light Characteristics* and the process of capturing this information is termed *Light Characterization*. Enabled by the additive characteristics of light intensity and for simplicity, the characterization process is achieved by turning on each light one-by-one at each dimmer level and measuring the incident light intensities using wireless light sensors.

*Optimal Light Actuation Profile Generation:* This is the core of the Illuminator light control system. The Illuminator system finds the best light level for each light to meet the user's requirement with the obtained light characteristics information.

In addition to these capabilities, the Illuminator can also be used to reconstruct similar lighting effects in a different physical light setup. Although this requires re-characterization of lights, this feature becomes very powerful, for example, when the same performance needs to be done in different places or at different times. If lights are re-setup or setup in different places, the setup will vary, even though the crew tries to set it up in the same way. We use Illuminator to document the results of a lighting setup, not just its physical assignment.

*Usage Scenario:* Typical usage scenario of the Illuminator system would be as depicted in **Fig. 1**. The oval represents the user's process and step. The rectangle represents the processes that the Illuminator system handles. The user designs lighting based on their purposes and desired aesthetic effects. Their intentions are described in user constraints. Based on the constraints and available light sensors, the Illuminator system recommends sensor placement. Then, the user deploys sensors using Illuminator's recommendation. The Illuminator automatically characterizes lights using deployed sensors. Once the light characterization process is complete, light sensors can be removed from the set, except for ones that are used for consistent illumination or tag tracking. The Illuminator controls lights by on-line light actuation profile generation in the rehearsal process. The user may want to improve lighting design as the rehearsal iterates. Improvement can be done by changing user constraints, adding more sensors for light characterization, and adding or moving light sensors for better illumination results. For example, if user found that the characterization is insufficient for some area due to obstacles, the user would want to deploy more sensors at the area obscured by the obstacles.

### 1.3 Design Requirements

Although the design driver for the Illuminator is entertainment and media production, our system could also be used for other purposes, such as achieving required illuminations in outdoor venues, high-rise buildings, and retail spaces. Design requirements for entertainment and media production are however the most challenging among those applications.

*High Fidelity and Wireless Network-Capable Light Sensing:* The first necessary technology to realize a sensor-supported light control system for entertainment and media production is high performance light sensing. In WSN, the Mica node [12] is the de facto standard platform to form wireless networks. Therefore, a high fidelity and network-capable light-sensing module for Mica platform will offer tremendous advantages. We have developed and used a multi-modal and high-fidelity light sensing module, the Illumimote [5]. The Illumimote is well-suited to light control applications. It provides multi-modal light sensing capabilities (e.g. incident light and RGB color intensities), a wide dynamic range, and fast response time. Therefore, the Illumimote provides the critical sensing substrate for the Illuminator system.

*Understanding High-Level User Constraints:* Expected users of the Illuminator system are the persons who are in charge of setting, controlling and managing lighting of film sets or theaters. Even for them, it is hard to quantify the requirements and constraints of a field's desired illumination. Lighting design is typically an iterative, hands-on process. At a minimum, it is desirable that the users describe their intentions in high-level terms familiar to them; thus the Illuminator should understand and implement high-level user constraints. Examples are: "Maintain 1000 lux brightness at location (10, 20)", "Keep illuminating a moving actor with 1000 lux brightness", "Illuminate an actor with 500 lux brightness from the

front and 1000 lux from the left”, “The difference of illumination on location A and B should be 3000 foot-candle”, “Illuminate an actress with 4:1 contrast ratio from left to right”, and “Evenly illuminate over an area”. User constraints should be able to express desired illumination level and its associated time and locations.

*Event Management:* Some user constraints may overlap in time and locations. Users may also want to use their own light actuation profiles that were generated previously with other constraints. An appropriate framework for data and event management, specification of user constraints, real-time actual profile generation, and light actuation and sensing is required.

*Adaptive to Environment:* In many cases, ambient light from uncontrollable light sources is present in the field. The Illuminator should support a closed-loop control method that incorporates real-time light measurements from sensors

#### 1.4 Related Work

Previous work related to our research can be categorized into five areas: 1) Light Sensor Module Development, 2) Commercial Theatrical Lighting, 3) WSN for Multimedia, Entertainment and Media Production, 4) Sensor Placement, and 5) Light Control using WSN.

*Light Sensor Module Development:* Crossbow MTS310, MTS400 and MTS420 are currently available mote-class light sensor modules for WSN platforms (e.g. Mica2 and MicaZ) [12]. However, they are inadequate for high-fidelity applications, being too sensitive to infrared radiation, and lack the necessary dynamic range. Commercial light meters, such as the Sekonic light meter [9] for incident and spot light intensity measurement, and the Konica Minolta Color Meter IIIf [10] for color temperature measurement, are available from the cinematography community. These commercial meters provide accurate measurements but they are slow and do not provide the wireless networking capabilities essential to our application. The Konica Minolta T-10 [13] light meter supports multi-point measurements by wiring multiple light meters. However, this is an expensive solution, because multiple light meters are required and it is very hard to install and manage due to cumbersome wiring.

*Commercial Theatrical Lighting:* The Cast Software provides commercial lighting software, Wysiyg [8], a 3D CAD tool for designing lighting for theater and performance. This tool supports virtual view and rendering of theatrical lighting in 3D, including both fixture deployment and light profile. Electronic Theatre Controls, Inc. provides a computerized control system, the Eos lighting console [7] that can control up to 5000 channels (devices). However, these products rely on theoretical light characteristic information for each fixture from pre-built libraries, when these can be more accurately obtained by measuring the actual field. However, these tools and consoles cannot utilize active sensing and closed loop control.

*WSN for Multimedia, Entertainment and Media Production:* Major research on this area has been done in UCLA. The Augmented Recording System [4] provides a framework to monitor and archive sensor data from film sets. The data can be used at shooting for decision support for cinematographers, such as directors, script supervisors and directors of photography (DP's). In addition, the data can be utilized for *continuity management* and to synthesize computer generated images with live actors, post-production. Similarly, UCLA REMAP has applied WSN technologies to theater. Burke designed an interactive system for a production of *Macbeth* [14], in which custom software uses wireless sensing devices to dynamically control lighting and sound effects based on performers' positions and movements. Elsewhere, for multimedia, the SEVA system [15] used WSN to determine if objects are in the camera view field. Video footage is annotated with the object IDs and later the IDs are used to search video

the stream that contains those objects. The ACCESS project [16] spotlights an audience using a pan-tilt mount in which the location is computed using a camera and image processing.

One of attractive reasons for using WSN technologies in media production is significant cost saving using small wireless nodes. In many cases, compared to existing (wired) solutions, WSN provides economic benefits for low cost in production, operation, and deployment [17]. For example, WSN and radio-frequency identification (RFID) can be used for continuity management in filmmaking [18], and logistics [19] for cost savings.

*Sensor Placement:* There has been extensive research on sensor placement that deals with sampling problems [20][21][22][23][24][25]. In [21], a sampling method is integrated into a mobile agent. The mobile sensor moves to a location that maximizes the utility function whose objective is to maintain regular sampling and minimize estimation error over distance to that location. In [20], the field is assumed to be Gaussian and the covariance of the field is known. Mutual information between new placement point and existing placement and remaining points are compared, and then choose the point that maximizes the difference. It selects the next point that is most informative to all unsensed locations. The next location is one least correlated to already sensed locations and most correlated to unsensed locations. Since the next sampling location is decided based on the covariance between points, actual measurements from the current sensor deployment are not used to decide the next sampling point; thus, it is not adaptive to live sensor readings. This method is somewhat unrealistic, because it requires a priori knowledge of the covariance of the field. Other research on sensor deployment includes coverage with random deployment policy [22], sensor position assignment [23], and self improvement after initial random deployment [24]. The incremental deployment [25] method uses mobile robots to discover an unknown environment. The goal is to maximize the covered area and to have visibility (connectivity) among robots.

*Light Control Using Sensor Networks:* An example using WSN for coordinated light control for building illumination was presented in [26]. It formulated the light control problem, as a trade-off between building occupant's utility function and energy consumption. This utility function indicates the level of satisfaction for each building occupant. However, it does not support representation of illumination quality for cases involving multiple sites; these happen frequently in illumination for entertainment and media production purposes. In addition, the authors assume that the light level at each point of interest from each light source is known. However, they did not show how this information is obtained and validated. Illuminator provides a systematic process to obtain this information via a light characterization phase. In [27], the authors used a similar approach to ours, in that the light is controlled in a closed-loop fashion. However, sensor locations are assumed to be fixed at grids, infeasible in most real cases. Gauger *et al.* [28] uses wireless sensors for home automation. Since the authors used a simple hill climbing algorithm to reach the desired illumination level at points of interest, the target illumination level is not met at first and it takes time for the resultant lighting level to stabilize. Thus, this method cannot be applied to media production or applications requiring fast and real-time response. Our proposed method finds optimal light settings based on light characteristics. Thus, the desired level of illumination can be implemented from the first time.

## 1.5 Contributions and Paper Organization

This paper describes the design challenges, requirements and necessary technologies for light control systems based on WSN technologies, especially for entertainment, multimedia and media production. Our work provides the following contributions. 1) We address unique design requirements and implementation issues for WSN for an application that brings together sensing and control. 2) We identify the required processes and methods for intelligent

light control using WSN: light characterization, parsing of user constraints, optimal light actuation profile generation, and closed-loop control with real-time feedback that accommodates changing ambient lighting conditions and moving targets. A language for user constraints is defined and the light control problem is formulated as an optimization problem. 3) We verify the capabilities of our system with a proof-of-concept testbed. Many real issues, such as source of errors and measurement noise, can be discovered from testbed validation. However, our Illuminator system works appropriately under such harsh conditions thanks to Illuminator's adaptiveness and closed-loop control.

The remainder of this paper is organized as follows. Section 2 presents problem formulations and approaches of the light control system. Section 3 describes system architecture and implementation issues. Section 4 presents the experimental setup and results. Finally, Section 5 discusses future work and conclusions.

## 2. Problem Formulations and Approaches

The following subsections describe the problem formulations and approaches of the major tasks based on the capabilities of a closed-loop light control system described in Section 1.2.

### 2.1 Representation of User Constraints

In theatrical and film lighting design, expression of user's intentions (constraints) may be very complex and hard to describe in terms of absolute number in brightness. This is because the intention of lighting tends to involve multiple points and conditions. For example, in film, a director of photography (DP) may want equal illumination on two points, different illuminations on two points with 200 lux brightness difference, or illumination on two points with contrast ratio of 3:1. We used these examples as an initial set of high-level constraints from which to develop the Illuminator system. We defined a simple but formal language to represent user constraints to provide users the capability of expressing such high-level constraints about lighting and, at the same time, to provide the light control system a way to understand the constraints. There are existing description languages that provide only capabilities of lighting calculation [29] and rendering of virtual reality [30]. The purposes of these languages are rendering and modeling of lighting, and have nothing to do with control. To our best knowledge, this proposed language is the first trial using formal language to represent lighting constraints – the user's intention on studio lighting.

Theatre designers analyze and change position of the light source, intensity, color, contrast and texture of the light to analyze and control stage lighting [31]. Based on these lighting designers' interests, we categorized user's intention regarding the lighting into six types: ABSOLUTE, GREATER, LESS, DIFF, RATIO and EQUAL. The language is formally defined with grammar in [6] to be able to express the six types of user interests. Our language can describe the users' interest for lighting, except for the positions of light sources and texture. The unit of brightness is expressed in either lux or foot-candles. The language also has keywords **place** and **monitor** to indicate preferred sensor locations, and the area of interest for monitoring, respectively. An area can be specified by points or a polygon in this language. Each constraint has associated gamma value that it serves as the objective value in the optimization framework. **Table 1** shows some examples of user constraints, their types and gamma values. The first example represents an illumination of 1000 lux brightness on tag 0 from all lights during 500 ms through 2500 ms. The second represents an illumination of 50 foot-candle (= 538 lux) on tag 1 only from light 0 and light 1, when the tag 1 is at location (20,

30). The foot-candle unit is converted into lux. The DIFF type constraint describes an illumination with relative brightness. The illumination on tag 2 from light 0 and 1 needs to be brighter by 100 lux than that from light 2 and 3.

**Table 1.** Examples of User Constraints

Type	Constraint	<i>gamma</i> value
ABSOLUTE	C1:illuminate 1000 lux at tag 0 from all during [500ms, 2500ms];	1000
ABSOLUTE	C2:illuminate 50 fc at mobile tag 1 from (light0, light1) when tag 1 in (20, 30);	538
DIFF	C3:illuminate tag 2 from (light0, light1); C4:illuminate tag 2 from (light2, light3); C5:set C3 - C4 = 100 lux;	100

Except for the EQUAL type, each constraint has its *gamma* value. Due to space limitations, we do not include the language definition of user constraints. Further examples and the grammar of user constraints can be found in our earlier work [6].

Each constraint can be activated by either or both of two conditions: time and location. With the exception of the EQUAL type, each constraint has an associated objective value, *gamma*, which is the target cost value to be achieved when finding the optimal light actuation profile for the constraint. After parsing the constraints, the type, objective value *gamma*, associated locations or tags, and associated lights of each constraint are stored in a text file in the database directory. The light actuation profile generation task uses this information.

Regardless of the constraint types, the Illuminator system should provide consistent illumination, although there are external uncontrollable lights (e.g. sunlight). Additionally, the tags on actors or props may be moving and the Illuminator system should change the light actuation profile to adapt to the environment as the tags move.

The language can represent 2D coordinates for locations; this means sensors are assumed to be on the floor. This language may have conflicting constraints or impossible constraints under some lighting configurations. However, such conflict or impossible constraints can easily be found in the rehearsal process of Illuminator's operation by checking if the estimated error is too big (e.g. twice the optimization target value).

## 2.2 Sensor Placement Recommendation

Most previous work on sensor placement deals with the coverage issue by regular placement or deployment to better estimate unknown fields [20][21][22][23][24][25][34]. Variance-based approaches are used to estimate the field better by adopting an adaptive and incremental scheme to find the next placement locations of high variance (or entropy) [21]. In this case, the variance is computed based on the measurements by the currently placed sensors. Unlike previous research on sensor placement, in our application, the user knows what the resultant light field should be like. Thus, our system suggests sensor placement to verify if the intended light field is appropriately created. We combine the two typical approaches: the regularity-based technique and the variance-based technique. The placement for the next sensor is calculated in an incremental fashion. We want to place the next sensor at the farthestmost point from the current placement, in terms of both of distance for regularity and variance. In our case, the variance can be computed based on the desired intensities. Let  $Y$  be



```

P = ϕ
Y = all possible locations
for number of available sensors begin
     $\hat{y} = \arg \max_{y \in Y} D(P, y)$ 
    P = P ∪  $\hat{y}$ 
    Y = Y -  $\hat{y}$ 
end

```

**Fig. 2.** Placement Algorithm based on Distance-Variance

the possible placement locations and let  $P = \{p_1, p_2, p_3, \dots, p_n\}$  be the current placement. Then, distance  $D(P, y)$  can be defined in equation (1).

$$D(P, y) = \min_{\forall p_i \in P} \{d(y, p_i) + K \cdot |I(y) - I(p_i)|\} \quad (1)$$

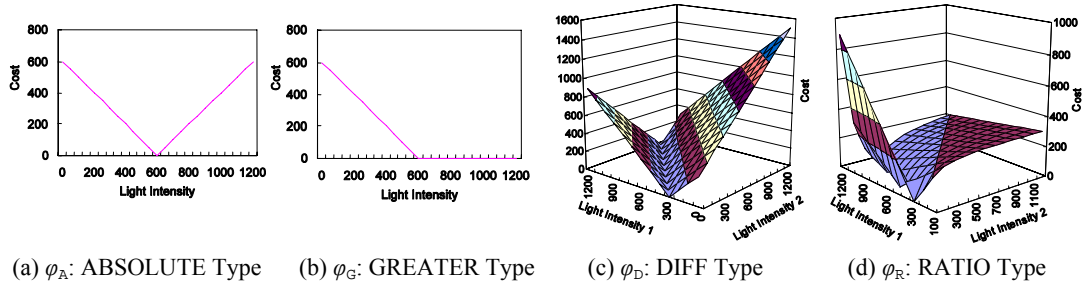
$D(P, y)$  is the minimum combined distance of the Euclidean distance on a 2D plan and the distance (difference) in their desired value (light intensities) from new location  $y$  to the current placement  $P$ . The distance  $D(P, y)$  represents how far the sensor at location  $y$  is from the current placement  $P$ , where  $d(y, p_i)$  is the Euclidean distance between  $y$  and  $p_i$ ,  $K$  is the weight for linearly combining the two terms, and  $I(y)$  and  $I(p_i)$  are the desired light intensities at location  $y$  and  $p$ , respectively. The two terms represent regularity and variance by the distance and the difference of the desired intensities between  $y$  and the closest sensor in  $P$ , respectively. The next best sensor location would be  $y$  that maximizes  $D(P, y)$ . The incremental placement algorithm is shown in **Fig. 2**. The variance is only affected by the ABSOLUTE type of constraints. If there are multiple constraints, the maximum variance across the constraints is used. Recommended optimal sensor locations and the current deployment are displayed on a GUI display, providing real-time feedback of the current deployment.

### 2.3 Light Characterization

The transfer function from all possible combinations of dimmer settings and locations to light intensities must be known to compute the intensity of illumination at a given location. This knowledge is equivalent to the *Light Characteristics* for a given lighting deployment and is a function of dimmer setting and target location. Let  $\lambda : X \times Y \rightarrow I$  be the light characteristics, where  $X$  is the set of possible light actuation profiles (combinations of settings of dimmers),  $Y$  is the set of locations of interest, and  $I$  is the generated light intensity. Then, the straightforward light characterization process involves  $\lceil n(Y)/s \rceil \cdot d^l$  steps, where  $l$ ,  $d$  and  $s$  are the number of lights, dimmer levels and available sensors, respectively. This is of exponential complexity and thus practically infeasible.

Fortunately, it is already known that light intensity is additive [26] and we exploit this property for the light characterization process. As the total incident light intensity at a location is summation of light intensities at that point from all sources illuminating the location, we can characterize lights one-by-one. We sweep through all intensities for a given light and measure light intensities. Then, we can redefined the light characteristics function as  $\lambda : L \times D \times Y \rightarrow I$ , where  $L$  is the set of lights, and  $D$  is the set of possible different levels of a dimmer. The complexity becomes  $\lceil n(Y)/s \rceil \cdot ld$  and this is practically feasible.

This characterization process in the field is more accurate than pre-characterization beforehand (e.g. at factory). Side effects are caused by objects and walls in the room or studio.



**Fig. 3.** Cost Functions of Constraint Types

As the light intensity is measured in the presence of the objects and walls, lights due to side effects, such as reflection and diffusion, are included in the measurements when characterizing each light source. This approach would produce less error, because side effects are already taken into account. We assume that the area is divided into fixed-size grids. For other locations, not characterized by the sensors, we estimated intensities by applying the Natural Neighbor Interpolation scheme [33], an interpolation technique based on Voronoi diagrams.

## 2.4 Light Actuation Profile Generation

The main function of the Illuminator system is to generate optimal light actuation profiles (value for each dimmer over time) for user constraints. As there are typically many lights creating the field, light intensity at a location is affected by multiple sources and thus the lighting should be done in a coordinated fashion. The inputs for actuation profile generation are the light characteristics  $\lambda$  and user constraints, and the output is the optimal light actuation profile that satisfies user constraints. Let  $\mathbf{x} = \{x_1, x_2, x_3, \dots, x_n\}$  be a light actuation profile (a set of dimmer values), where  $n$  is the number of lights. Then, the incident light intensity  $I_y^{x,L}$  at location  $y$  from lights  $L$  with the light actuation profile  $\mathbf{x}$  can be computed using equation (2).

$$I_y^{x,L} = \sum_{\forall i \in L} \lambda(i, x_i, y) \quad (2)$$

We defined a set of cost functions  $\Phi = \{\varphi_A, \varphi_G, \varphi_L, \varphi_D, \varphi_R, \varphi_E\}$  to indicate how well the lighting matches the user constraints. Examples of the cost function  $\varphi \in \Phi$  of constraint type ABSOLUTE, GREATER, DIFF and RATIO are shown in Fig. 3. Fig. 3(a) and (b) show the cost functions when the required brightness ( $\gamma$ ) is 600 lux, Fig. 3(c) shows the cost when the required brightness difference of two illuminations is 300 lux, and Fig. 3(d) shows that when the desired contrast ratio is 3:1. The cost functions for ABSOLUTE, GREATER and LESS type have two arguments, since they compare generated light intensities and desired ones. The DIFF and RATIO type have three arguments, as these functions calculate the difference and ratio of the first two input arguments, then compare the result with the third value. The EQUAL type may have more than two arguments, as this type assesses evenness of the light intensities across many points. The evenness is determined by computing the standard deviation of the light intensities at the locations of interest.

Let  $c \in C$  be a user constraint and  $\{y_c^1, y_c^2, y_c^3, \dots, y_c^m\}$ ,  $\gamma_c$  and  $L_c$  be the associated sensor locations, objective value and associated lights with the constraint  $c$ , respectively, where  $m$  is the number of associated locations. Constraint type DIFF and RATIO have two associated locations,  $y_c^1$  and  $y_c^2$ . The EQUAL type may have more than two associated locations and it does not need the  $\gamma$  value. Then, the cost for a constraint  $c$  with light actuation profile  $\mathbf{x}$ ,

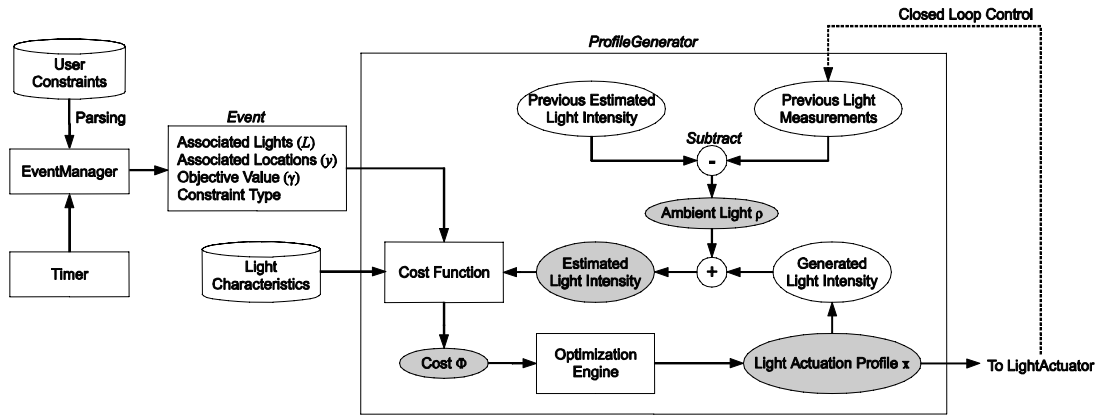


Fig. 4. Data Flow Diagram of Light Actuation Profile Generation

$\varphi(c, \mathbf{x})$  can be computed, as in equation (3), where  $type(c)$  represents constraint type  $c$ .

$$\varphi(c, \mathbf{x}) = \varphi_{type(c)} \left( \{I_{y_c^1}^{x, L_c}, I_{y_c^2}^{x, L_c}\}, I_{y_c^3}^{x, L_c}, \dots, I_{y_c^m}^{x, L_c}\}, \gamma_c \right) \quad (3)$$

We use the root-mean-square for total cost to minimize the costs of constraints altogether. Then, finding the best light actuation profile  $\hat{\mathbf{x}}$  for given constraints,  $C$  can be formulated as an optimization problem in equation (4).

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \sqrt{\sum_{c \in C} \Phi^2(c, \mathbf{x})} \quad (4)$$

If the solution of the above optimization problem (4) is sought,  $\hat{\mathbf{x}}$  would be the optimal dimmer values that minimize total cost.

We can extend this approach to allow the Illuminator to control lights, where there are uncontrollable external lights, such as sunlight. The external lights are regarded as ambient light and assumed to change slowly. If the ambient light at location  $y$  at time  $t$  is  $\rho_y(t)$ , the measured incident light intensity is  $I_y^*(t)$ , and actuation profile is  $\mathbf{x}$ . The following equation holds, as light intensity is additive.

$$I_y^*(t) = I_y^{x, L}(t) + \rho_y(t) \quad (5)$$

Using the sensors' light measurement and estimated light intensity from the light characteristics, we can estimate the ambient light intensity  $\rho_y(t)$  by subtracting generated light intensity  $I_y^{x, L}$  from the measured light intensity  $I_y^*(t)$ . Then, the incident light intensity at location  $y$  with actuation profile  $\mathbf{x}$  can be redefined as the following equation.

$$I_y^{x, L} = \sum_{i \in L} \lambda(i, x_i, y) + \rho_y(t) \quad (6)$$

The ambient light intensity may change over time, so whenever the light measurement is updated,  $\rho_y(t)$  needs to be updated. Although the ambient light changes over time and locations, this does not require re-characterization of lights, as long as the light setup remains same. Fig. 4 shows the overall data flow of the light actuation profile generation. Illuminator's closed loop control is implemented by the path through the generated light actuation profile, LightActuator, dimmer, actual lighting and actual light measurements. If there are some mismatches (i.e.  $\rho$ ) between the actual light measurements and estimated light intensities, the next estimated light intensity will accommodate the difference as additional ambient light or

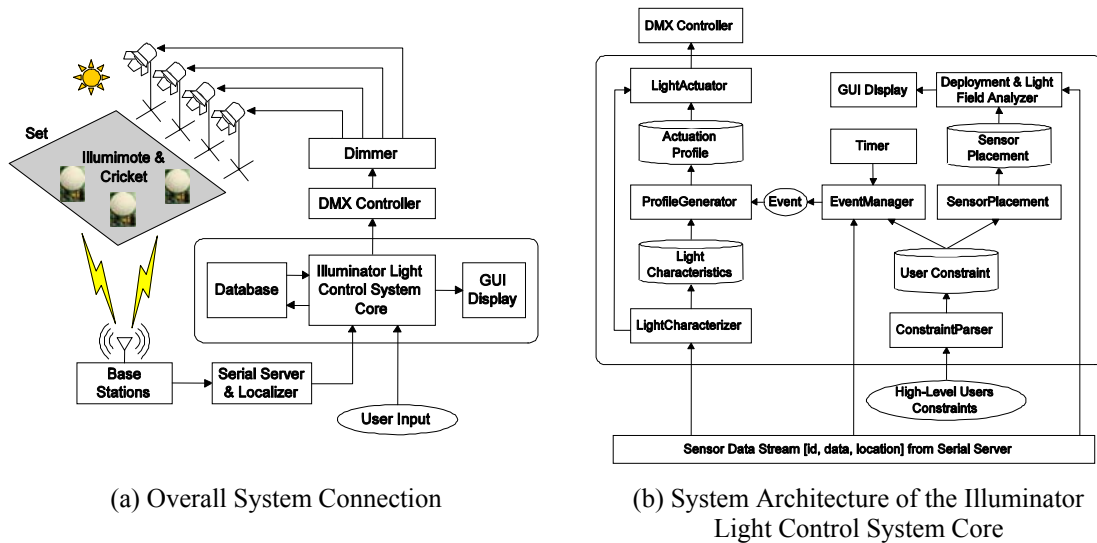


Fig. 5. Illuminator Light Control System

estimation error.

### 3. System Architecture and Implementation

The Illuminator light control system can be divided into three subsystems: sensor network, Illuminator core, and DMX controller & dimmer. The overall system connection is shown in Fig. 5(a). The sensor network, base station nodes, and DMX controller & dimmer are external components outside the laptop gateway computer. In our implementation, the serial server & localizer, database, Illuminator core, and GUI display run on the gateway computer.

#### 3.1 Sensor Network

The sensor network measures the light intensities and sensor locations. It consists of two sub-networks: the Cricket localization system [34] and the single-hop MicaZ [12] network with the Illuminote light sensing board [5]. Although any localization system can be adopted, as the Illuminator system is not confined to particular localization methods, we used the Cricket nodes for range measurements due to their simplicity and availability. Although it is simple, it is effective, in that it can provide ranging accuracy within a few centimeters. Three Cricket nodes were deployed as beacons that are pre-calibrated with their locations. We coupled a Cricket with Illuminote to localize the light sensor module. The Cricket nodes run on TinyOS [35] and measure distances between them using ultrasound. The Illuminote node runs the SOS [36] environment on a MicaZ node to achieve higher bandwidth. The Illuminote provides rapid and accurate measurements of incident light intensity and RGB color intensities (for color temperature). Two Java modules, SerialServer to interface between the Illuminator core and sensor networks, and Localizer to compute the Cricket node positions based on the ultrasound range measurements to manage two sensor network platforms running concurrently.

#### 3.2 DMX Controller and Dimmer

We used a Leprecon 6-channel dimmer [37] and ENTTEC DMXEtherGate MK2 [38], as the DMX-controlled dimming system. DMX refers to the entertainment industry standard control

signal for lights, 8-bit dimmer levels multiplexed on an RS485 serial link. The Illuminator core sends 521-byte control packets to the DMX controller via the Ethernet. Then, the DMX controller sets the dimmer based on the control packets. The dimmer has six 100 W channels and can control each light at 8-bit resolution.

### 3.3 System Architecture & Components of the Illuminator Core

The system architecture of the Illuminator core is shown in [Fig. 5\(b\)](#), with subtasks as rectangles. Modules are implemented as separate Java threads and thus can run concurrently. The intermediate data are kept as files in the database directory and are shown as cylindrical shapes. The Illuminator can operate in different modes: Sensor Placement Recommendation, Deployment Assistance, Light Characterization, and Off-line & On-line Profile Generation. As each module is an independent thread, each operational mode only needs to activate the necessary modules. For example, the Light Characterization mode only activates the LightCharacterizer, LightActuator and SerialServer module. The On-line Profile Generation mode activates the ProfileGenerator, LightActuator, EventManager, Timer, ConstraintParser, and SerialServer module.

*ConstraintParser*: We used the Java Compiler Compiler (JavaCC), a parser generator for Java [\[39\]](#) to implement the grammar of the language for user constraints. The parser module reads the user constraints and constructs the UserConstraints database in tuples of [type, activation conditions, associated tags and locations, associated lights,  $\gamma$  value].

*LightCharacterizer*: This module controls all lights in turn for available dimmer settings. Each sensor reads incident light intensities at its location. The sensor readings are stored in the LightCharacteristics database in tuples of [light id, dimmer level, location, incident light intensity]. Each intensity reading will be subtracted by the intensity reading at the dimmer value 0 (i.e. reference black level) to consider ambient lights. Then, the Natural Neighbor Interpolation scheme [\[33\]](#) is applied to estimate light characteristics for unsensed locations.

*EventManager*: Execution of the actuation profile generation is triggered by events. The EventManager module periodically looks into the activation conditions (sensor location and current time) of the UserConstraints database. If the activation conditions are met, the EventManager constructs event objects based on the user constraints to be satisfied and signals the ProfileGenerator module with the activated events.

*ProfileGenerator*: As explained in Section 2.4, light actuation profiles are found by a combinatorial optimization framework. The ProfileGenerator module finds the best combination of dimmer values for lights to meet the user's desired intensity level at the sensor locations and other requirements. We implemented a genetic algorithm [\[40\]](#) for the optimization method. A brief explanation of its implementation follows. First, 100 random solutions are generated to create the initial population. In the case of on-line profile generation, the current actuation profile from the ActuationProfile database is added to the initial population to avoid searching from beginning. The population's costs are computed and solutions are sorted by cost. Then, the worst 20% of solutions are removed. Mutation and crossover are performed on the best 20% of the solutions, and the new solutions added to the population. In crossover, dimmer values of lights are exchanged with each other. Mutation perturbs the dimmer values. Then, the cost is recomputed and above procedure repeats, until a satisfied solution is found or the maximum number of searches is reached.

*LightActuator*: The Actuation Profile database is a set of tuples consisting of [start time, dimmer level for light 1, light 2, ..., light n]. The LightActuator module reads the current actuation profile corresponding to the current time and actuates lights based on this profile by

sending control packets to the DMX controller. The LightActuator module is directly controlled by the LightCharacterizer module at the light characterization stage.

*GUI Display:* A graphical display has been implemented using Processing [41]. In combination with the Illumimote's graphic user interface (GUI), it shows the current locations of the sensors for placement recommendation as rectangles. Sensor nodes on the GUI are highlighted when their locations are inside the recommended regions.

## 4. Experimental Results

### 4.1 Experimental Setup

To evaluate the capabilities of the Illuminator light control system, we set up a proof-of-concept testbed in a laboratory environment. The setup has a similar structure to that shown in Fig. 5(a). We set up a small, 102 x 82 cm, stage on the floor. Four halogen lamps were used for light sources at the upper side of the stage. Four halogen lamps on the wall in a line were connected to the dimmer: the leftmost lamp is light 1 and the rightmost lamp is light 4. (All of the following figures of the stage are drawn upside down so that lights are placed at the bottom.) Although four beacon nodes are required for unique localization in 3-D, we used only three Cricket nodes, because we assumed that sensors are on the floor and thus the Z-coordinate is always zero. Due to the limited availability of Illumimote prototypes, we could only deploy one Illumimote node and one Cricket node. We slightly modified the Illuminator core and measured 25 points to emulate 25 sensors. This paper mainly focuses on light field sensing and actuation using wireless sensor technologies; moreover, the scalability itself of deploying multiple wireless nodes is purely a networking issue, and out of scope of this paper. Therefore, with one wireless light sensor, the capabilities of sensing light fields and actuating lights could be sufficiently explored.

We set the maximum brightness for the bitmap image file as 2000 lux. Then, maximum variance is 2000 and the maximum distance between sensors is about 130 cm. Based on this,

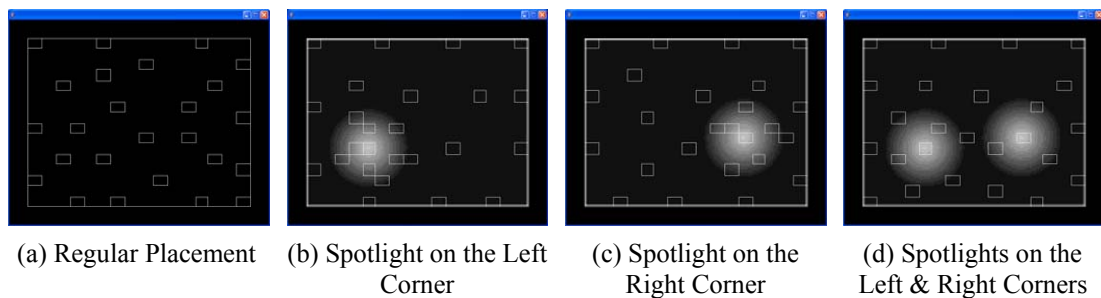
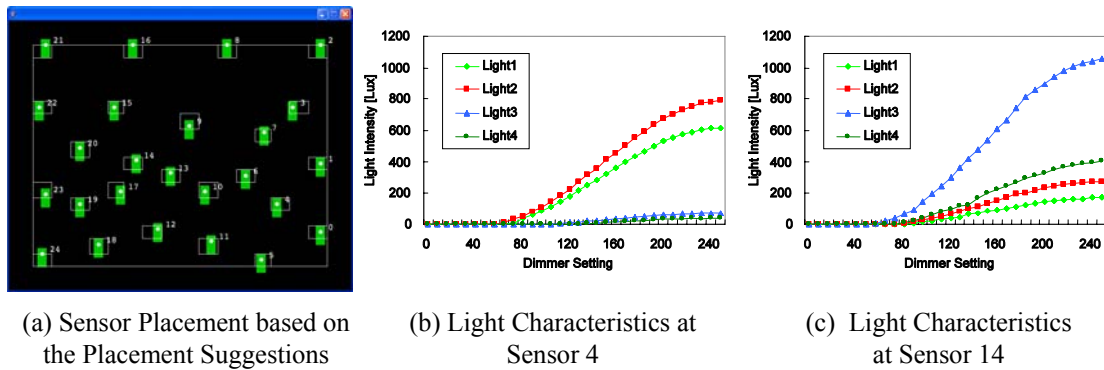


Fig. 6. Placement Suggestion for 25 Sensors

we set  $K = 0.03$  for equation (1) to give twice the weight to regularity compared to variance.

### 4.2 Sensor Placement Recommendation

We first tested the capability of guided sensor placement. We assumed that the user wants to illuminate based on light intensity maps, in which spotlights on the left corner, the right corner, and both of them are illuminated, respectively. Using the maps, recommended locations for 25 sensors were obtained, as shown in Fig. 6. The recommended locations are shown in rectangular areas instead of points to provide flexibility in physical deployment. Denser placement is recommended for areas of light intensity changes, as shown in Fig. 6(b), (c), and



(a) Sensor Placement based on the Placement Suggestions (b) Light Characteristics at Sensor 4 (c) Light Characteristics at Sensor 14  
**Fig. 7.** Sensor Placement based on Suggestions and Light Characteristics at Sensor 4 and 14 (d).

### 4.3 Sensor Deployment and Light Characterization

We characterized lights for 32 dimmer levels at 25 points, as recommended in **Fig. 6(d)**. **Fig. 7(a)** shows the figure of the actual deployment that we characterized lights based on the recommendations. The characteristics are different for each light and for each sensor. **Fig. 7(b)** and **(c)** show examples of light characteristics of sensor 4 and 14. As sensor 14 is deployed on the left, we can guess it is affected by lights 3 and 4 more than by lights 1 and 2.

	Light1	Light2	Light3	Light4
(a) Photographs of the Real Illumination				
(b) Voronoi Tessellations				
(c) Natural Neighbor Interpolation				

**Fig. 8.** Light Characteristics at Maximum Dimmer Level

We conducted this characterization process for four lights at each of 25 points. We took photographs of the stage with illumination at the maximum dimmer level to demonstrate the quality of the light characterization results after the interpolation. A Canon EOS 10D digital SLR camera was used with lens aperture setting F/4, focal length 17mm, and shutter speed 1/125 second at ISO 400. **Fig. 8(a)** shows the photographs of the stages with illumination from each light. **Fig. 8(b)** shows the Voronoi tessellation based on light measurements from 25 sensors. Based on this Voronoi tessellation, Natural Neighbor Interpolation [33] was applied. **Fig. 8(c)** shows the interpolated light characteristics including unsensed area. The photographs are a little bit brighter, because the stage is white and thus highly reflective. The interpolated characterization results only show the incident light intensities, so they look a little darker. However, **Fig. 8(a)** and **(c)** show high correlations.

#### 4.4 Response Time Analysis

We generated a series of light pulses and measured them using the Illuminator system to evaluate the response time of the currently implemented light control system. Fig. 9 shows the generated light pulses and measurements. The total turnaround time was approximately between 100 ms and 200 ms. That is, the current prototype can control and update light settings about 5 to 10 times per second. The total turnaround time includes transmission delay from the Illuminator to the Ethernet port of the DMX controller, from the DMX controller to the dimmer, time to actuate lights, sensing delay, transmission delay from the Illumimote to the base station node, and serial communication delay from the base station to the gateway computer. We believe the turnaround time can be optimized below 100 ms; this is an acceptable delay in practical light control in the theater and filmmaking. For example, further optimization can be done by improving transmission delay from gateway to dimmer using a faster interface.

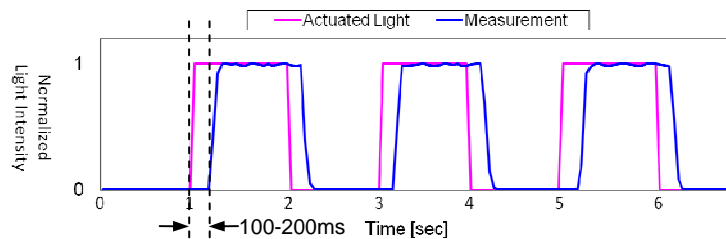


Fig. 9. Light Intensity Measurements for Light Pulses

#### 4.5 Light Actuation Profile Generation

Once the light characteristics are obtained, a light actuation profile for any constraints can be generated. We tested the Illuminator system in four categories: Absolute Brightness, Even Illumination, Relative Brightness, and Contrast Ratio. For absolute brightness, consistent illumination with a mobile tag and a random ambient light, and five intensity patterns described by bitmap images are tested.

*Absolute Brightness:* For the first test item, we tested *consistent illumination* with and without random ambient light. A user constraint in Fig. 10(a) that requires absolute and consistent illumination of 500 lux at tag 0 was used. We continued generating the light actuation profile, even after the brightness requirement was met, to see the behavior of our system. Fig. 10(b) shows the experimental results of the consistent illumination for the mobile sensor. The **Generated** curve shows the estimated brightness by the light characteristics,  $I_y^{x,L}$ .

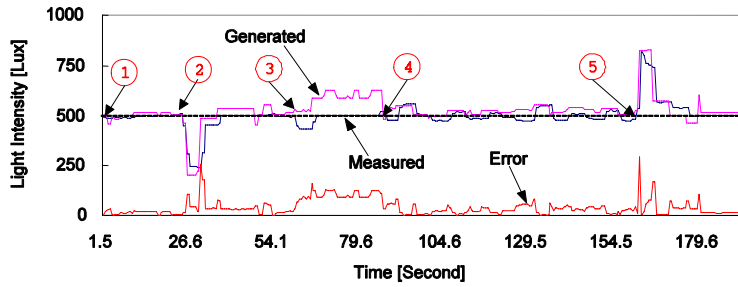
**Measured** curve shows the brightness measured by the Illumimote light sensor. There is a difference between generated brightness and measurement, which is the error. Sources of this error could be: sensing error, interpolation error in light characteristics, localization error and ambient light intensity. Fig. 10(c) shows the moving path of the mobile sensor through five locations. Whenever the sensor moved to new locations, the measured light intensity changes. As the sensor moves, the generated intensities were updated accordingly, based on the 2-D interpolated light characteristics information. As one can see in Fig. 10(b), although there is some fluctuation, our system maintains consistent brightness for the mobile sensor.

We also tested the case with ambient lights. One additional halogen lamp was connected to the dimmer and set to random intensities to generate random ambient lighting. Fig. 11 shows

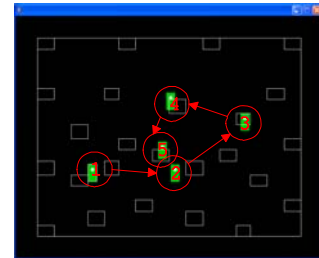


```
C0: illuminate 500 lux at mobile tag 0 from all;
```

(a) User Constraint for Consistent Illumination

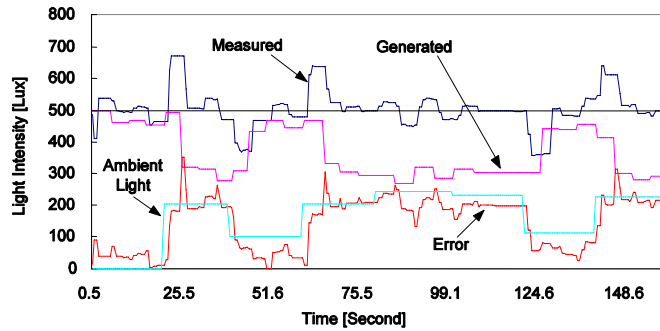


(b) Illumination Results



(c) Path of the Moving Tag

**Fig. 10.** Experimental Results of Consistent Illumination for a Mobile Sensor



**Fig. 11.** Experimental Results of Consistent Illumination with Ambient Light



(a) User Constraints – Intensity Profiles



(b) Estimated Intensities based on Light Characteristics



(c) Photographs of the Real Illuminations

**Fig. 12.** Experimental Results of Illumination for the Fixed Intensity Profiles

the experimental results with such random ambient light. Please note that the unit of the ambient light in Fig. 11 is not lux, but dimmer value. Whenever there are changes in ambient light, there are also changes in the measurement. However, the measured brightness converges

to 500 lux soon after. The Error curve appropriately tracks the ambient light changes. That is, the Illuminator adapts to the environment correctly.

In the experiments of consistent illumination, the genetic algorithm for optimization searched from 100 to 800 solutions for real-time generation of one light actuation profile. It took at most 5 milliseconds on a Pentium 1.5GHz laptop computer. This speed is satisfactory for the film or video frame rate (33 milliseconds).

Second, to evaluate the capabilities of illuminating according to intensity map images, we tested our Illuminator with five intensity maps, of  $128 \times 97$  pixels; each pixel represents intensity from 0 to 2000 lux. User constraints were to illuminate as similarly as the associated bitmap images of **Fig. 12(a)** as possible. The five bitmap images represent the user's desired light intensities: left spotlight, right spotlight, front light, left light, and right light, respectively, from the leftmost column. The Illuminator generates the light actuation profiles that match with the given bitmap images the most. **Fig. 12(b)** shows the estimated light intensities generated by the Illuminator.

**Table 2.** Summary of Experimental Results

Test Item	Average Error
Consistent Illumination for Mobile Sensor	17.3 lux (3.5%)
Consistent Illumination with Ambient Light	20 lux (4%)
Bitmap Images (leftspot / rightspot / frontside / leftside / rightside)	112.3 / 96.6 / 402.6 / 373.9 / 323.5 lux
Even Illumination	177 lux (Std Dev)
Relative Brightness	25.1 lux (6%)
Contrast Ratio (Varying Ratio)	6%
Contrast Ratio (Varying Location)	8.9%

For comparison, we took photographs of the real illuminations by the generated actuation profile, as shown in **Fig. 12(c)**. As in **Fig. 8**, the real illumination and estimated illumination are very similar and have high correlations.

We performed other types of illumination tests to test the capabilities of understanding various types of user constraints. These included even illumination, relative brightness and contrast ratio. Due to the limited space of the paper, we briefly explain the experimental results on relative illumination and contrast ratio. The capability of relative illumination was tested using user constraints that describe illumination difference on two locations from -500 lux to 500 lux. Experimentation on relative brightness was successful in that two illuminations changed accordingly to satisfy the illumination difference requirements. They maintained relative brightness between two tag locations with a 6% average error. We also evaluated the capability of illuminating with a varying contrast ratio with and without moving tag. For this, we prepared user constraints of contrast ratio from 1:1 through 7:1 between illuminations on tags 10 and 14. The Illuminator performed well for the contrast ratio illumination in both cases with average errors 6% and 8.9%, respectively. **Table 2** summarizes the experimental results, calculating the average errors for the test items. The average errors and their percentage over target value are also shown for each case; abnormal values at transient period are excluded when averaging. Only the error of achieved contrast ratio compared to target ratio are calculated in the percentage for contrast ratio tests. In the case of even illumination, the standard deviation was used, as there is no absolute target value. As shown in **Table 2**, our Illuminator system can provide consistent illumination with less than 4% error on average. For

fixed illumination with bitmap image constraints, errors were a little bigger in some cases (frontside, leftside and rightside), because the target illumination maps were not achievable with a set of four lights. This can be seen in the photographs in Fig. 12. In addition to the experiments we have described so far, we performed experiments on even illumination and contrast ratio for mobile tags. The Illuminator found correct light actuation profiles for those cases.

## 5. Conclusions

We have presented design requirements, system architecture, and implementation of the Illuminator. It is an intelligent light control system for entertainment and media production using WSN. We used the Illumimote, a multi-modal and high fidelity light sensor module designed for the Mica family, to satisfy the high-performance light sensing requirements. Then, we created a toolset to characterize lights, generate desired lighting effects for user constraints and help deploy lights in entertainment illumination applications. A formal language was developed to describe the users' high-level constraints. An optimization framework was applied to find optimal light settings under specified light setups. In experimental results, the Illuminator was shown to appropriately handle various types of user constraints and optimal light actuation profile generation, on-line and off-line.

Future work can see the deployment of the Illuminator in sound stages or theaters. Other localization techniques, such as [42] or commercial systems may be used, to extend the scale of this system and the accuracy of sensor locations. Applying more sophisticated multi-input multi-output control algorithms would be another area for future work. Currently, the Illuminator uses optimization techniques to find the best light setting and relies on the light characteristics. However, when the number of lights increases to hundreds, the optimization framework might not function sufficient for real-time light control. Then, it would be desirable to approach the light control problem from a control theoretic perspective, where it is not required to know light characteristic information beforehand. Finally, the Illuminator's initial development focused on controlling light intensity. However, modern lighting control systems are capable of changing color, beam spread and other characteristics on-the-fly. Future versions of the Illuminator may explore handling color and color temperature. The Illumimote light sensing module already has the capability of sensing RGB colors and color temperature. Therefore, the Illuminator can extend the characterization process to RGB color space and thus the control algorithm can be extended to three color channels.

## References

- [1] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 51-58, 2000.
- [2] J. Burke et al., "Embedding expression: Pervasive computing architecture for art and entertainment," *Elsevier Journal of Pervasive and Mobile Computing*, vol. 2, pp. 1-36, Feb., 2006.
- [3] F. Wagnister et al., "Advanced technology for cinematography," 2002, <http://remap.ucla.edu>.
- [4] N. M. Su et al., "Augmenting film and video footage with sensor data," *PerCom '04*, 2004.
- [5] H. Park et al., "Illumimote: Multimodal and high-fidelity light sensor module for wireless sensor networks," *IEEE Sensors Journal*, vol. 7, no. 7, pp. 996-1003, July 2007.
- [6] H. Park, J. Burke, and M. B. Srivastava, "Design and implementation of a wireless sensor network for intelligent light control," *Information Processing in Sensor Networks*, pp. 370-379, 2007.
- [7] Electronic Theatre Controls, Inc, "Lighting control products," <http://www.etccconnect.com>.
- [8] Cast Software, "Wysiwyg," website, <http://www.castlighting.com>.

- [9] Sekonic, "Sekonic L-558Cine DualMaster," <http://www.sekonic.com/Products/L-558Cine.html>.
- [10] Konica Minolta, "Minolta Color Meter IIIf," <http://konicaminolta.com>.
- [11] Spotlight, website, <http://www.spotlight.it>.
- [12] CrossBow, "Mica data sheets," website, <http://www.xbow.com>.
- [13] Konica Minolta, "Illuminance meter t-10," 1996.
- [14] J. Burke, "Dynamic performance spaces for theater production," *Theatre Design & Technology (U.S. Institute of Theatre Technology)*, vol. 38, no. 1, 2002.
- [15] X. Liu et al., "Seva: Sensor enhanced video annotation," *ACM Multimedia*, Nov., 2005.
- [16] M. Sester, "ACCESS Project," website, <http://www.accessproject.net>.
- [17] I. Khemapech et al., "A survey of wireless sensor networks technology," *PGNET*, 2005.
- [18] A. Marianantoni et al., "Sensor networks for media production," *Demos in SenSys*, 2004.
- [19] E. Fleisch and M. Dierkes, "Ubiquitous computing: Why auto-id is the logical next step in enterprise automation," *White Paper, STG-AUTOID-WH-004*, Auto-ID Center, 2003.
- [20] C. Guestrin et al., "Near-optimal sensor placements in Gaussian processes," *ICML*, 2005.
- [21] M. Rahimi, M. Hansen, W. J. Kaiser, G. S. Sukhatme, and D. Estrin, "Adaptive sampling for environmental field estimation using robotic sensors," *IROS '05*, 2005.
- [22] Y. Zou and K. Chakrabarty, "Uncertainty-aware sensor deployment algorithms for surveillance applications," *Global Telecommunications Conference (GLOBECOM'03)*, 2003.
- [23] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, "Grid coverage for surveillance and target location in distributed sensor networks," *IEEE Transactions on Computers*, Dec. 2002.
- [24] N. Heo and P. K. Varshney, "A distributed self spreading algorithm for mobile wireless sensor networks," *Wireless Communications and Networking (WCNC)*, 2003.
- [25] A. Howard, M. J. Matari'c, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, no. 2, pp. 113-126, 2002.
- [26] V. Singhvi et al., "Intelligent light control using sensor networks," *SenSys '05*, 2005.
- [27] M.-S. Pan, L.-W. Yeh, Y.-A. Chen, Y.-H. Lin, and Y.-C. Tseng, "Design and implementation of a wsn-based intelligent light control system," *ICDCSW '08*, pp. 321-326, 2008.
- [28] M. Gauger, D. Minder, P. J. Marr'on, A. Wacker, and A. Lachenmann, "Prototyping sensor-actuator networks for home automation," *REALWSN '08*, pp. 56-60, 2008.
- [29] P. Hanrahan and J. Lawson, "A language for shading and lighting calculations," *SIGGRAPH Computer Graphics*, vol. 24, no. 4, pp. 289-298, 1990.
- [30] R. Carey and G. Bell, *The annotated VRML 2.0 reference manual*, UK, Addison-Wesley Longman Ltd., 1997.
- [31] N. H. Morgan, *Stage Lighting for Theatre Designers*. The Herbert Press, 1995.
- [32] S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," *Wireless Communications and Networking (WCNC)*, 2003.
- [33] R. Sibson, A brief description of the natural neighbor interpolation, In: D.V. Barnett, editor, *Interpreting Multivariate Data*. Chichester: John Wiley & Sons, 1981.
- [34] A. Smith et al., "Tracking Moving Devices with the Cricket Location System," *Mobisys*, 2004.
- [35] TinyOS, website, <http://www.tinyos.net>.
- [36] C.-C. Han et al., "A dynamic operating system for sensor nodes," in *MobiSys '05*, 2005.
- [37] Leprecon, "Leprecon Id-360 users manual," Feb. 2000.
- [38] ENTTEC Pty Ltd, "ENTTEC DMX512 Ethernet Gateway MK2 User Manual," Dec. 2005.
- [39] Java Compiler Compiler, <http://javacc.dev.java.net>.
- [40] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, pp. 44-50, 1992.
- [41] B. Fry and C. Reas, "Processing," <http://www.processing.org>.
- [42] A. Ledeczi et al., "Towards precise indoor RF localization," *HotEmNets '08*, June 2008.



**Heemin Park** received his B.S. and M.S. degrees in CS from Sogang University, Korea in 1993 and 1995, respectively. Then, he joined Samsung Electronics, Korea. In 2001, he was granted full support for his Ph.D. study from Samsung. He received his Ph.D. degree in EE from UCLA in 2006. While at UCLA, he worked as a GSR for NESL, CENS, and REMAP. He is currently principal engineer and technical leader of the image processor design group for mobile phone cameras in Samsung Electronics. His research interests include a broad range of ubiquitous and mobile imaging, ubiquitous computing for multimedia, video compression, and video coding.



**Jeff Burke** is Executive Director of the Center for Research in Engineering, Media and Performance (REMAP) at UCLA. He has co-authored, designed, coded or produced performances and new genre installations that have been exhibited in eight countries, coordinating diverse teams, spanning the arts and engineering. Burke has taught in the UCLA School of Theater, Film and Television, as well as in the graduate industrial design program at the Art Center College of Design.



**Mani B. Srivastava** received the B.Tech. degree from I.I.T. Kanpur, India, in 1985, and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1987 and 1992, respectively. He is on the faculty at the University of California at Los Angeles (UCLA). His research interests are in embedded systems, low-power design, wireless networking, and pervasive sensing. He received the President of India's Gold Medal in 1985, the NSF Career Award in 1997, and the Okawa Foundation Grant in 1998. He currently serves as the Editor-in-Chief of the *ACM Sigmobile Mobile Computing and Communication Review*, and as Associate Editor for the *IEEE/ACM Transactions on Networking*, and the *ACM Transactions on Sensor Networks*.