



# Smart Solid-State Lighting Control

## Steuerung eines Intelligenten Belichtungssystems mit LED Lichtquellen

Nan Zhao

Betreuer/-in: Joe Paradiso, Steffen Leonhardt

26. August 2010

Pauwelsstraße 20  
D-52074 Aachen  
Telefon: +49 (0)241 80 -23211  
Fax: +49 (0)241 80 -82442  
E-Mail: [medit@hia.rwth-aachen.de](mailto:medit@hia.rwth-aachen.de)

# Acknowledgment

Meinen herzlichsten Dank an Professor Leonhardt, der mich von den ersten Schritten an begleitet und unterstützt hat, die Schritte, die mich zu dieser einmaligen Erfahrung und einem einzigartigen Umfeld mit großartigen, inspirierenden Menschen führten.

My sincere thanks to Joe. Thank you for your kind, caring and visionary leadership and allowing me to become part of the amazing Responsive Environments Group.

My sincere thanks to Matt. Thank you for teaching me and supporting me. I surely couldn't have done this without you. Thank you, Matt, Becky and Mike for the heart warming welcome, which I will never forget.

My sincere thanks to Mike. Without you my light house would be crooked. Thank you for just being the way you are.

My sincere thanks to Bo. Thank you for being such a patient teacher and a joyful friend, Dudelidu.

My sincere thanks to Laurel. Thank you for filling the air with musik and spending the tiny bit free time you had with us.

My sincere thanks to Alex. Thank you for being such a great cubicle neighbor.

My sincere thanks to Nan-Wei. Thank you for being a lovely, funny, cute, smart, talented, exciting and supportive friend. And thank you Chia-Wei for taking care of this luv.

My sincere thanks to Gershon. Thank you for being the fruits in my pasta, the helado in my banana split, the tree in my Koala forest, the typo in my figure caption, and guess what?

My sincere thanks to Pranav. Thank you for being such a lovely and caring friend and for going through all the delicious food and bubble tea with me.

My sincere thanks to Hannah, Emily and David. You were great, funny and supportive neighbors.

My sincere thanks to Edwina and David R. for the fun time and for being so lovely.

Meinen herzlichsten Dank an meinen Eltern, die mir Alles, sowie auch dieser Teil meines Lebens, ermöglicht haben.

## *Acknowledgment*

---

Meinen liebsten Dank an meinen lieben Freund, Christian, der steht geduldig mit mir ist und mit mir durch Dick und Dünn geht.

My sincere thanks to everyone else, who helped me and shared their knowledge with me.

My sincere thanks to Philips Color Kinetics.

And last but not least, thank you, [dict.leo.org](http://dict.leo.org).

# Erklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen sind, wurden als solche kenntlich gemacht.

---

Ort, Datum

---

Unterschrift



# Abstract

General illumination using LEDs has enabled designers to have precise control of the intensity and color of an illuminated area. In this work, I investigate how these systems can dynamically adjust to create optimal lighting conditions in the built environment. By incorporating environmental feedback into a system control loop, i.e., illuminance or luminance measurements, the precise intensity and color temperature can be generated for an area of interest.

According to the International Energy Agency, 19% of global electricity consumption is due to lighting. Closed-loop solid state lighting technology enables significant power and emission savings. Applications for smart lighting control include large scale office lighting systems, private household lighting, or even theater installations.

Presented in this work is a new approach to optimal lighting control. Unlike previous research that studied constant illuminance, I investigate the incorporation of control elements such as color temperature and color rendering index. In this manner, the control system not only balances lighting intensity and energy, but also attempts to balance acceptable lighting quality. Using online light signal calibration, there is no need for user localization. Two feedback methods are investigated: illuminance lux feedback using an environmental sensor and luminance feedback using a commercial webcam. A testbed is built that enables evaluation and demonstration of the control system. Three short-term user studies show the energy saving potential and the user experience of the test system.



# Table of Contents

<b>Acknowledgment</b>	<b>iii</b>
<b>Erklärung</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>Table of Contents</b>	<b>ix</b>
<b>List of Symbols</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 State of the art</b>	<b>3</b>
<b>3 Commercial Solid State Lighting System</b>	<b>5</b>
3.1 Light Grid Model . . . . .	7
3.2 Closed-Loop Control . . . . .	12
3.2.1 Sensor Board Based Control . . . . .	12
3.2.2 Webcam Based Control . . . . .	27
3.2.3 Adaptive Control . . . . .	32
3.3 GUI and Software . . . . .	33
3.3.1 Input Processing and PI Controller . . . . .	35
3.3.2 Calibration Routine . . . . .	40
3.3.3 Optimization Routine . . . . .	41
3.4 Results . . . . .	45
3.4.1 System Model Evaluation . . . . .	45
3.4.2 Calibration Evaluation . . . . .	46
3.4.3 Controller Evaluation . . . . .	53
3.4.4 Short-Term User Study . . . . .	57
<b>4 Pentachromatic Solid State Lighting System</b>	<b>67</b>
4.1 Light Grid Model . . . . .	68
4.2 Open-Loop Control . . . . .	70
4.2.1 Behavior of Pentachromatic Solid State Light Source . . . . .	70
4.2.2 Identify Transfer Function . . . . .	73
4.2.3 Optimization Routine . . . . .	73
4.3 Results . . . . .	75
<b>5 Conclusion and Future Work</b>	<b>81</b>
<b>Bibliography</b>	<b>83</b>



# List of Symbols

## Abbreviation

CCT	Color temperature
CRI	Color rendering index
LED	Light emitting diode
PSO	Particle swarm optimization
LP	Linear programming
ALADIN	Ambient lighting assistance for an ageing population
WSN	Wireless sensor network
RF	Radio frequency
PWM	Pulse-width modulation
UDP	User datagram protocol
CFF	Critical flicker frequency
CKGUI	Intelligent lighting control software user interface
GUI	Graphical user interface
VLC	Visible light communication

## Variables

$E_\lambda$	Spectral irradiance
ratio	Cold and warm color temperature intensity ratio for white LED system, Channel intensity and total intensity ratio for pentachromatic system
CCT	Color temperature in Kelvin
int	Intensity, illuminance
channel	PWM duty cycle, normalized channel intensity with maximum channel intensity as normal intensity
CRI	Color rendering index
T	Time of periode, camera exposure time
k	Ratio of channel brightness and total brightness of the light source
d	Contribution factor
dmx	Vector of PWM duty cycle settings of every channel and light source inside the system
c	Cost vector
x	Position vector, subject of optimization
fitness	Fitness of the optimization objective
RMSE	Root mean square error
R – square	Coefficient of determination
Bl	Brightness level
L	Luminance
G	Gain of automatic gain control for cameras
F/#	Camera aperture
Powersaving	Power saving
E	Power consumption in Watts
bfs	Basic feasible solution of LP optimization
t	Time
$T_s$	Settling time

## Indexes

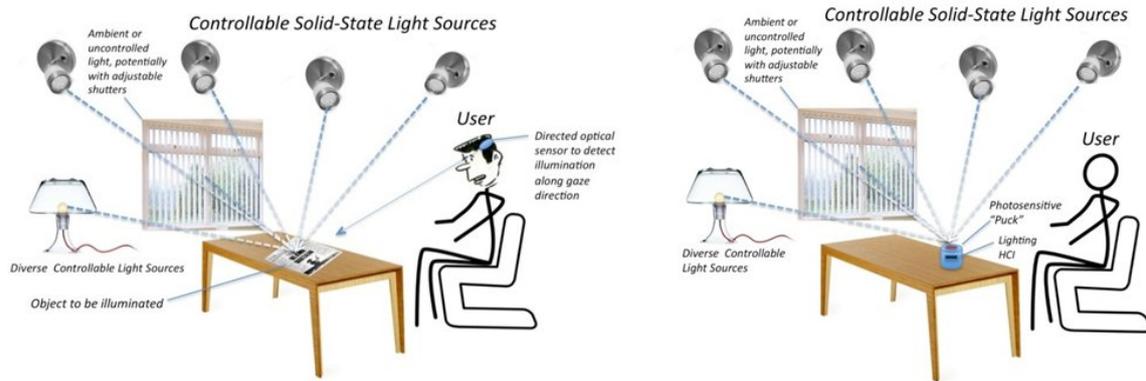
$n$	Number of solid state light sources
$i$	Solid state light source index
$j$	Solid state light source channel index

# 1 Introduction

Solid state lighting is the choice of lighting technologies in the future for many reasons. First, there is a high potential for energy and GHG emissions savings. [13] „Lighting accounts for 19% of global electricity consumption and gives rise to CO<sub>2</sub> emissions that are equivalent to 70% of those from the world’s light-duty vehicles.“ [13] „A large proportion of total lighting energy is used by inefficient and outdated technologies (...)“ „Over the next 25 years, global electricity consumption for lighting with current socioeconomic trends and policies is projected to rise to over 4250 TWh, an increase of 60% overall at an average rate of 1.9% per annum“ [13] Considering these predictions, significant reduction of energy consumption is achievable using smart lighting control and light-emitting diodes (LEDs). LEDs with high efficacy are commercially available, and LED-based systems enable precise control of the intensity and color of an illuminated area. A single lighting device is able to fulfill a variety of user requirements, as relatively few changes in the lighting control can produce a wide variety of lighting scenarios. Finally, using a high switching frequency, it is possible to transfer information from a light source to a movable or fixed light sensor with a high data rate, in a technique known as visible light communication (VLC). Each of these qualities of solid state lighting makes LEDs very compelling engines for intelligent and adaptive lighting systems.

One application for smart lighting control is a large-scale lighting system in an office environment. Individual lighting profiles for each participant create a pleasant working area and increase productivity [4, page 10]. By incorporating environmental feedback from illuminance and luminance measurements into a system control loop, the precise intensity and color temperature can be generated for any area of interest. Adaptive lighting also enables reduction of the total energy consumption and operating expenses. The ALADIN (ambient lighting assistance for an ageing population) project is one example of such a system for a private household application. In this work, [6] studied the psychophysiological effect of smart lighting control for the elderly and found that low color temperature (2700 K) and low illumination (100 lux) are subjectively experienced as relaxing, while medium color temperature (4000 K) and medium illumination (1300 lux) are best suited to mentally demanding tasks.

My approach to an intelligent lighting system is to provide maximized lighting quality and user comfort while minimizing power consumption. An intuitive user interface gives the user control over the lighting system. Simple handling and automation make it possible to save energy and create suitable lighting for different situations with little effort. Using a puck, a small movable sensing and input device, the user is able to specify a point of interest by placing it at the desired position, or to specify an area of interest by moving the puck around the desired area. The intelligent lighting controller receives and processes the information and calculates the most efficient settings, which also satisfy the user’s



**Figure 1.1:** Intelligent solid state lighting system

color and illuminance preferences. The puck can be considered a remote light switch that simplifies interaction with lighting and enables active feedback to maintain lighting quality despite environmental changes. In another approach, a simple CMOS camera sensor, like a webcam or cell phone camera, could be used in place of the puck. In this case, the user is able to select the area of interest by simply drawing it onto the camera image. One main advantage of this approach is that CMOS cameras are widely available and embedded in the office environment. Smart phones, netbooks and laptops, many of which have built-in cameras and support WIFI communication, are potential platforms for mobile smart lighting control. As opposed to the puck, a camera measures reflected light. As such, it is possible to track the user's vision and measure the approximate amount of light that the user perceives with his eyes, using a head mounted camera, for example. Yet more control options are possible because the camera captures a scene and can detect multiple objects within it. Luminance ratios between surfaces and intered context (e.g. reading a book or paper, looking at a computer monitor) can become control criteria. Gesture recognition is also possible using camera based feedback. The user could adjust the level of intensity or color temperature with simple gestures.

In the following chapters, I will introduce two intelligent lighting systems using preexisting custom pentachromatic solid state lighting and a commercial white LED lighting system, respectively. The pentachromatic system is an open loop control system, which is implemented and evaluated using MATLAB Simulink. This part of my work investigates optimization and control of pentachromatic solid state light sources. The dichromatic commercial lighting system is a closed loop system. I constructed a testbed with four light fixtures and used both a webcam and an illuminance sensor board designed by the MIT Media Lab's Responsive Environments Group to realize real-time feedback and online system calibration. Online system calibration means online user localization using the illuminance or luminance sensor. This part of my work focuses on optimization and control of a lighting system with multiple light sources, different sensing methods and interactive lighting. The design of the closed-loop control and calibration routine can be applied to the pentachromatic system in the future. Several user studies show the achieved power saving and evaluate interaction between users and the system.

## 2 State of the art

Lighting control methods now in common use include on/off switches, occupancy recognition, daylight harvesting, scheduling, task tuning, and demand control [4, page 3 ff.]. The on/off switch is the most common way to control lighting. Occupancy recognition and scheduling are used to provide illumination only when users are detected or are scheduled to be present, respectively. Daylight harvesting leverages natural daylight to reduce artificial light as much as possible. The intensity of lighting can be tuned according to a given task in order to improve worker productivity, aesthetics, or even to create a certain mood. Demand control is used to prevent peak demand.

Research papers [12, 16, 14, 7, 8, 11] analyze strategies for lighting control in context of intelligent buildings. Increasing research and commercial deployments of wireless sensor networks (WSN) have motivated the use of WSNs to monitor lighting conditions and the development of closed-loop lighting control systems. In these systems, illuminance sensors are placed in the area of interest in order to detect the illuminance surface and feed back the lighting information. [12] gives a brief recommendation of sensor placement for better estimation of the fields. [16] uses Bayes' Net to find the optimal, or the most efficient and satisfying intensities for the available light sources. [7] and [11] use the linear programming optimization technique to calculate the intensity setting. [11] considers the power consumption as the objective and the user-preference as constraints, whereas [7] aims to optimize the tradeoff between energy consumption and user preference for multiple users. Both algorithms require the knowledge of the positions of the occupants, which can be detected using RFID tags or other RF-based user localization systems. [11] introduces a scenario with two control levels, or discrete lighting devices, like fluorescent lights and local sources, such as table lamps. Daylight is monitored for daylight harvesting. Another optimization technique that has been tested in this application is the stochastic hill climbing method. [14] and [8] introduce an automated distributed control method based on this approach. By testing the influence of each light source on each sensor node, an appropriated range for the proximate lights or a control range for each sensor node is created. Within this range, random settings of intensity are generated and saved if the objective is improved according to the objective of the previous settings. This process is repeated until the improvements become insignificant. [14] and [8] prove convergence of the illuminance. [7] introduces an algorithm for active sensing, which is developed to reduce energy consumption of the sensing device by taking light samples only when the predicted usage is high.

None of the aforementioned papers investigate solid state lighting as the light source. Using dichromatic, trichromatic or other multichromatic LED light sources, control elements beyond intensity, such as color temperature (CCT) and color rendering index (CRI), can be adjusted according to user preference. A higher efficacy is achievable and synchro-

nized measurements can be taken with high frequency, which is not detectable with the human eye. Solid state lighting products with white and colored LEDs are commercially available. Matt Aldrich from the Responsive Environments Group at the MIT Media Lab developed a pentachromatic solid state light source, which contains blue, cyan, green, amber and red LEDs. Each color-channel can be controlled separately through 16bit PWM. Multichromatic lighting devices make possible the generation of white points with any color temperature and many color rendering indices. CRI of up to 90 is achievable with the 5 wavelength LED lighting device, where 100 is the maximum. As a point of comparison, a cool daylight fluorescent lamp has a CRI of 76.

The authors from [12, 16, 14, 7, 8, 11] used light sensors to measure direct light. It is also possible to use reflected light measurement as the control feedback. The process of detecting reflected light corresponds more closely to the function of the human eye. Luminance detection is possible using cameras or light sensors pointed at the reflecting object. [5] investigates the applicability of cameras as sensors for lighting control. The authors experiment with different methods of luminance measurement and compare the results with illuminance measurements. In their case, the camera is at a fixed position and the measurement is taken from a calibrated surface. They make the conclusion that multiple illuminance sensors can be replaced with a single camera, but point out privacy concerns associated with camera observation.

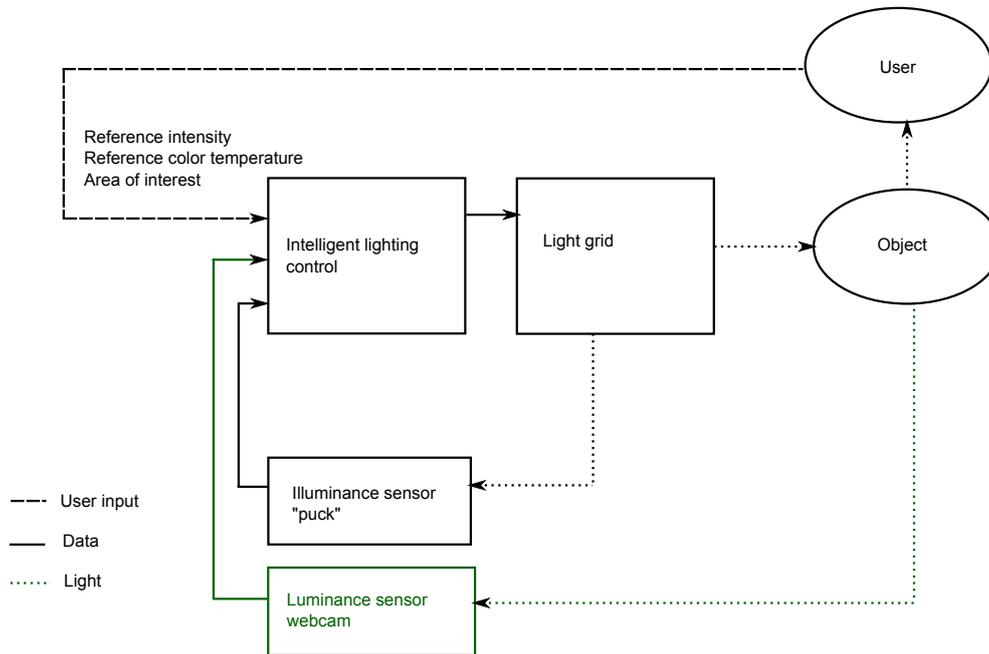
### 3 Commercial Solid State Lighting System

The commercial solid state lighting testbed consists of 4 iW BLAST 12's, which are white LED light fixtures designed by Philips. Each fixture has two sets of LEDs with warm and cold color temperatures. The LEDs are divided into three channels, which can be controlled separately. Two warm white channels emit white light with 3000 K and one cold white channel provides white light with 6500 K. The combination of all three channels makes it possible to create any CCT from 3000 K up to 6500 K. The maximal power input per light fixture is 50 W. The power supply unit is the Philips PDS-150e, a 24 VDC, 150 W power source, which can be wired to maximum three iW BLAST 12 light fixtures. It is outfitted for ethernet communication using a Philips proprietary protocol based on UDP. The intensity can be dimmed with 8-bit resolution.



**Figure 3.1:** Commercial solid state lighting testbed

The testbed is located in the 5th floor of the MIT Media Lab Extension Building close to the north-west glass façade. Besides sunlight coming in from the north-west, fluorescent light sources provide constant lighting inside the office area. Four iW BLAST 12 light fixtures, arranged in a 2-by-2 grid, are mounted above an office cubicle. The cubicle area is about 1.9 m long and 1.6 m wide. It contains a white writing desk, a white frame, a gray separating wall and two gray chairs. The floor is covered with deep gray carpet. The light fixtures are mounted on a wooden frame, which has the same size as the cubicle and is about 2.10 m high. The distance in between the iW BLAST 12 light sources is approximately 1 m and the distances from the light sources to the floor and to the level of the writing desk are 1.8 m and 1.1 m respectively.



**Figure 3.2:** Intelligent lighting system block diagram

There are two kinds of feedback devices, which can be used with the commercial solid state lighting test system. The sensor board designed by the Responsive Environments Group [1] is integrated into the test system and can be used as an input and feedback device. The other feedback device is a Logitech Quickcam Pro 9000, which is a commercial CMOS webcam with  $1600 \times 1200$  pixels video and image resolution and a maximum frame rate of 30 frames per second. Since the webcam measures reflected light, the measured light intensity is highly dependent on the surface of the captured object. Both feedback devices and two PDS150e power supplies are wired to a personal computer. The computer is equipped with an Intel Core i7 2.66 GHz Processor, X58i Motherboard and 6 GB RAM. The operating system is Windows 7 64Bit.

As described in the introduction, the user is able to select the area of interest, which in this case should be inside the cubicle area, and adjust reference intensity and CCT. The intelligent lighting system interprets the user input and optimizes the lighting output accordingly.

## 3.1 Light Grid Model

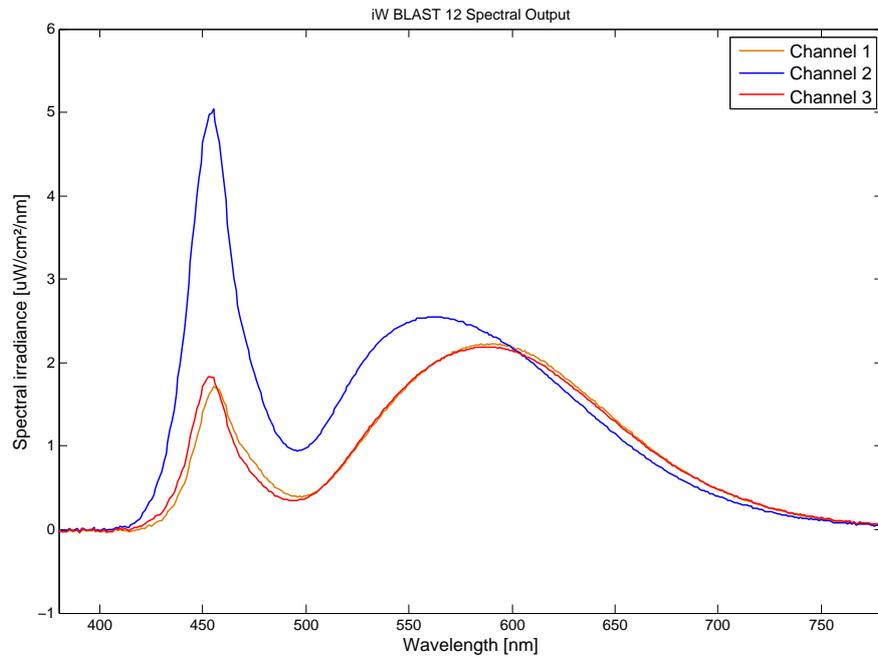
In this chapter I introduce two mathematical models for the dichromatic solid state lighting system. Based on the models, I am able to predict the behavior of the system and simulate the light grid. The models are also very useful for developing and understanding the intelligent lighting controller. The first model is a spectrum based model, similar to the pentachromatic system model described in chapter 4.1. The second is a simplified model.

Both models are based on the following system description. The commercial lighting test-bed consists of  $n = 4$  light fixtures arranged in a  $m$ -by- $l$  grid, where  $m = l = 2$ . Besides the solid state light sources, sunlight and fluorescent lighting are present. We assume that the intensities of sunlight and fluorescent light are known and uniform within the area of interest. The illuminated room is empty, which means that there is no shadow in the area of interest. Finally, the solid state light sources are assumed to be point sources. Under these conditions, intensity, CRI, CCT and other lighting characteristics are predictable at any given position and for any intensity settings of the solid state light grid.

The first model is spectrum based model, and therefore requires spectral information about the light sources. The spectral output of the iW BLAST 12 was measured using a spectrometer that was calibrated against a radiometric standard. The measurement is performed in a dark room for different PWM settings. In the following, the visible spectra of the three channels diagramed in figure 3.3 are referred as standard spectral irradiances  $E_{\lambda,j,0}(\lambda)$ .  $d_0 = 0.3$  m is the normal distance at which the normal spectral irradiance is measured. Assuming that the light source is a point source, the relationship between irradiance and the distance to the light source is described by the power law. The spectral irradiance  $E_\lambda$  at position  $x$  is

$$E_\lambda(x, \lambda) = \sum_{i=1}^4 \frac{d_0^2}{(|x - x_i|^2)} * \sum_{j=1}^3 E_{\lambda,j,0}(\lambda) + E_{\lambda,ambient}(\lambda). \quad (3.1)$$

$x_i$  is the position vector of light fixture  $i$ .  $E_{\lambda,ambient}$  is the spectral irradiance of ambient light, which, for simplicity, we assume to be independent of position. For example, daylight spectrum can be generated using a daylight simulator as described in [10, page 237 ff.]. Using colorimetric functions [17], formula 3.1, I am able to calculate CRI, CCT and illuminance at position  $x$ . CCT is estimated by finding the minimum distance between the white point and the iso-temperature lines along the black body curve in the CIE domain. The corresponding estimate is the CCT of the white point. CRI is calculated by comparing the color difference  $\Delta E_s$  with 15 representative color samples. CRI is defined as  $R_s = 100 - 4.6 * \Delta E_s$ , and the total CRI is  $R_a = \overline{R_s}$ . Illuminance is the integral of irradiance weighted by the luminous efficiency function  $V(\lambda)$ . More details about the colorimetry functions can be found in [1].

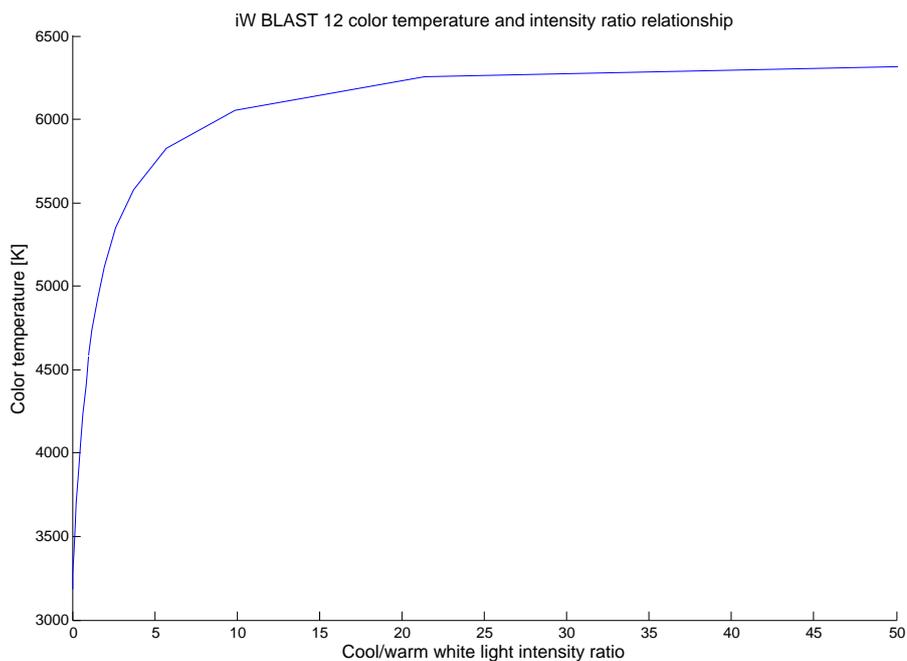


**Figure 3.3:** iW BLAST 12 visible spectrum, each of the three channels is measured separately with 100% duty cycle and a distance of 30 cm

The simplified system model does not need  $E_\lambda$  to estimate CRI, CCT or intensity. For the simplified model, I assume that ambient light is not present. Only the quality of solid state illumination is considered. Since the dichromatic system contains only white lights with two different CCTs I am able to identify CCT explicitly using the intensity ratio of warm and cool white light. Figure 3.4 shows the unique relationship between CCT and the intensity ratio of warm and cool white light.

$$ratio = int_{cold}/int_{warm} \quad (3.2)$$

Formula 3.3 describes the polynomial fit.



**Figure 3.4:** iW BLAST 12 color temperature and cool/warm white light intensity ratio relationship for simplified system model

$$CCT(x) = \begin{cases} p1A * (1/ratio(x))^3 + p2A * (1/ratio(x))^2 ... \\ + p3A * (1/ratio(x)) + p4A, & \text{for } int_{6000K} \geq int_{3000K}, \\ p1B * ratio(x)^3 + p2B * ratio(x)^2 ... \\ + p3B * ratio(x) + p4B, & \text{for } int_{6000K} < int_{3000K}. \end{cases} \quad (3.3)$$

$$p1A = -1175, p2A = 3283, p3A = -3950, p4A = 6429$$

$$p1B = 325.6, p2B = -1164, p3B = 2185, p4B = 3260$$

The root mean square error of the curve fit and the coefficient of determination for  $int_{6000K} \geq int_{3000K}$  are RMSE = 8.368, R-square = 0.9998, and RMSE = 2.718, R-square = 1 for  $int_{6000K} < int_{3000K}$ .

The intensity ratio at position x is

$$\begin{aligned} ratio(x) &= int_{6500K}(x)/int_{3000K}(x) \\ &= \frac{\sum_i \left( \frac{d_0^2}{|x-x_i|^2} * (channel_{2,i}) * int_{max,2,0} \right)}{\sum_i \left( \frac{d_0^2}{|x-x_i|^2} * (channel_{1,i} * int_{max,1,0} + channel_{3,i} * int_{max,3,0}) \right)} \end{aligned} \quad (3.4)$$

$channel_{j,i}$  is the normalized intensity.  $int_{j,i}$  is the light intensity of channel  $j$  and light fixture  $i$  and  $int_{max,j,i}$  is the maximal achievable intensity of channel  $j$  and light fixture  $i$  at the same position. It follows that

$$channel_{j,i} = \frac{int_{j,i}(x)}{int_{max,j,i}(x)} \text{ with } j = 1, 2, 3 \text{ and } i = 1, 2, 3, 4. \quad (3.5)$$

$int_{max,j,0}$  is the maximum intensity of channel  $j$  which is measured with standard distance  $d_0$ . It is identical for all light sources.

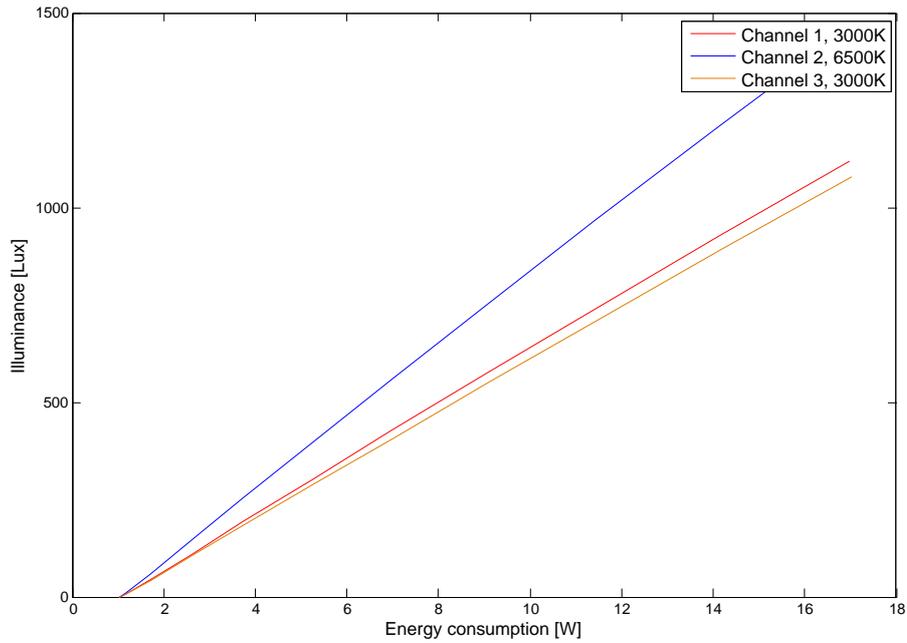
Intensity at position x is simply the sum of warm white light intensity and cool white light intensity.

$$\begin{aligned} int(x) &= (int_{6500K} + int_{3000K}) \\ &= \sum_i \frac{d_0^2}{|x-x_i|^2} * \left( \sum_{j=1}^3 channel_{j,i} * int_{max,j,0} \right) \end{aligned} \quad (3.6)$$

The cool white LED produces higher CRI than the warm white LED. The highest CRI measured during an experiment examining 100 white points, is 83, and the lowest is 73. The used white points are chosen to be evenly distributed in the intensity and CCT range. The root mean square deviation is 3.465. Since the variance is not significant, CRI is considered as constant in the simplified system model.

$$CRI = 78 \quad (3.7)$$

The function describing energy consumption is the same for both the spectrum based and the simplified system model. The current rises linearly with the intensity output, and the



**Figure 3.5:** iW BLAST 12 energy consumption

cool white LEDs have higher efficacy than the warm white LEDs. The linear fit of the relationship between power consumption and the normalized channel intensities is

$$\begin{aligned}
 E_i(\text{channel}_{1,i}, \text{channel}_{2,i}, \text{channel}_{3,i}) \\
 = 15.7779 * \text{channel}_{1,i} + 15.9031 * \text{channel}_{2,i} + 15.8613 * \text{channel}_{3,i} + 1.022.
 \end{aligned} \tag{3.8}$$

$E_i$  is the power consumption of light source  $i$  and  $E$  is the total power.

$$E = \sum_i (E_i) \quad (3.9)$$

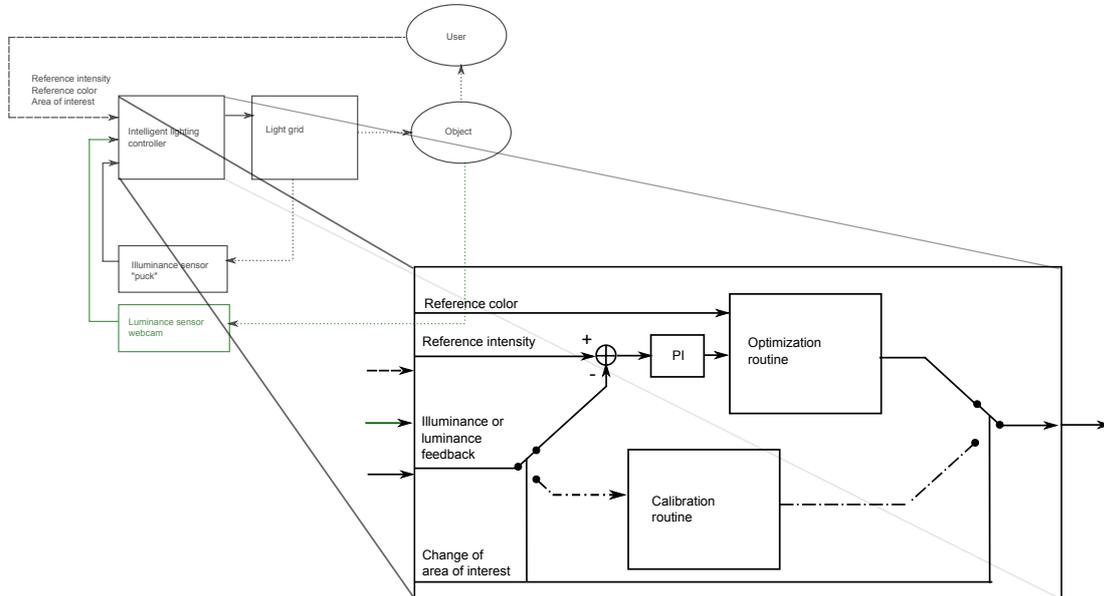
## 3.2 Closed-Loop Control

The closed-loop control is implemented with two different measuring systems. One is illuminance measurement using a linear analog lux sensor and the second method is luminance measurement using a commercial CMOS webcam. The closed-loop control includes three major components, the calibration routine, optimization routine, and a PI controller. The calibration routine measures the transfer function, which describes the influence of light source  $i$  for  $i = 1, 2, 3, 4$  at the area of interest. In other words, the calibration routine detects each light fixture's contribution to the light condition at the position of the area of interest. The calibration technology I am going to introduce in the following does not require additional hardware besides the luminance or illuminance sensor to locate the user. Neither information about location of the light sources nor any other information about the environment is needed. Since the transfer function at the current moment is measured, shadows that are caused by objects in between light source and area of interest or a defect in the light source will be recognized. The optimization routine solves the energy efficiency optimization problem for the specified area of interest and the user's intensity and CCT preferences.

### 3.2.1 Sensor Board Based Control

The sensor board based control uses the sensor board as its feedback and input device. Communication between sensor board and PC takes place via serial port. The sensor board is equipped with 3 analog linear lux sensors, one digital color sensor, two sliders and one push button. Table 3.1 shows the characteristics of the sensors. Since the embedded sensors have different qualities, they are dedicated for different tasks. The calibration routine needs high bandwidth lux feedback for fast detection. The low range, high sensitivity 50 Hz filtered lux sensor allows stable and moderate closed-loop intensity control. The digital color sensor enables color-temperature feedback for closed-loop color control.

Figure 3.7 shows the setup and figure 3.8 shows the result of the analog low range, low bandwidth lux sensor offline calibration, which is distinguished from the online calibration routine. The sensor was placed 30 cm in front of the iWBLAST 12 in the middle of the light cone inside a black box. The inside of the box was covered with black fleece. The measured sensor values are compared to lux values, which were measured under similar conditions using the spectrometer. The sensor saturates for intensities higher than 2000 lux and has a gain of 1.89.



**Figure 3.6:** Intelligent lighting controller block diagram

Sensor	Type	Bandwidth	Range	ADC Res.	mlx / mV
ISL29006, Intersil	analog, linear	50 Hz	2000 lux	$2^{12}$	606 mlx/mV
ISL29006, Intersil	analog, linear	600 Hz	2000 lux	$2^{12}$	60 mlx/mV
ISL29006, Intersil	analog, linear	50 Hz	10000 lux	$2^{12}$	3030 mlx/mV
TCS3414CS, Taos	digital, linear	50 Hz	5000 lux	$2^{16}$	1515 mlx/mV

**Table 3.1:** Sensor board characteristics [1]

The sensor board also functions as an input device. The built-in sliders are used for intensity and CCT adjustment and the pushbutton triggers the calibration routine.

To achieve the desired settings the user simply marks the area which needs to be illuminated using the sensor board. During this process each lighting device sends out defined light signals and the intensity of these light signals are synchronously measured using the high bandwidth lux sensor. This process is called the calibration routine. There are three versions of the calibration routine: single point calibration, multipoint calibration and area average calibration.



**Figure 3.7:** Sensor board low range high sensitivity analog lux sensor offline calibration setup

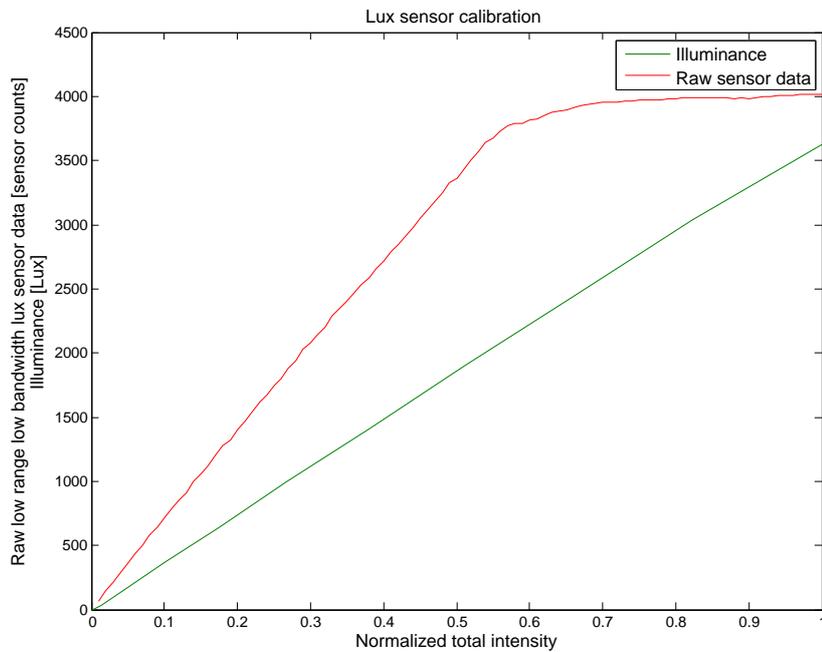
For single point calibration, the user places the sensor board at the point of interest and does not move it during the calibration process. The transfer-function for one single point is estimated. Figure 3.10 shows the light signals of light number 1 to 4 and the trigger signals for the unfiltered lux sensor. The light fixtures switch between on-state, in which all three channels provide maximum intensity, and off-state, where all three channels are turned off. The high bandwidth lux sensor is used in this case because it enables detection of fast illuminance changes. The illuminance difference  $\Delta int_i$  between on and off states of light source  $i$  expresses how much light fixture  $i$  distributes to the point of interest. In the following, the factor  $d_i = \Delta int_i / int_{norm}$  is called the contribution factor.

$$int_{meas,1} = \Delta int_2 + \Delta int_3 + \Delta int_4 + int_{ambient} \quad (3.10)$$

$$int_{meas,2} = \Delta int_1 + \Delta int_3 + \Delta int_4 + int_{ambient} \quad (3.11)$$

$$int_{meas,2} = \Delta int_1 + \Delta int_2 + \Delta int_3 + int_{ambient} \quad (3.12)$$

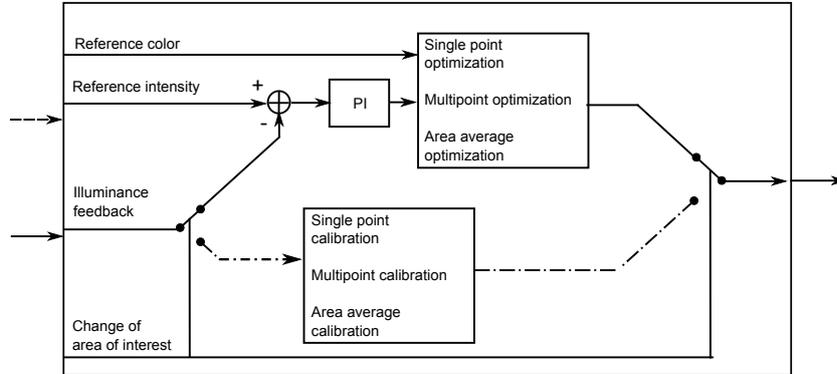
$$int_{meas,2} = \Delta int_1 + \Delta int_2 + \Delta int_4 + int_{ambient} \quad (3.13)$$



**Figure 3.8:** Sensor board low range high sensitivity analog lux sensor offline calibration result

$$\begin{aligned} \Delta int_1 = & \frac{1}{3} * (int_{meas,2} - int_{ambient}) - \frac{2}{3} * (int_{meas,1} - int_{ambient}) \\ & + \frac{1}{3} * (int_{meas,4} - int_{ambient}) + \frac{1}{3} * (int_{meas,3} - int_{ambient}) \end{aligned} \quad (3.14)$$

$$\begin{aligned} \Delta int_2 = & \frac{1}{3} * (int_{meas,1} - int_{ambient}) - \frac{2}{3} * (int_{meas,2} - int_{ambient}) \\ & + \frac{1}{3} * (int_{meas,4} - int_{ambient}) + \frac{1}{3} * (int_{meas,3} - int_{ambient}) \end{aligned} \quad (3.15)$$



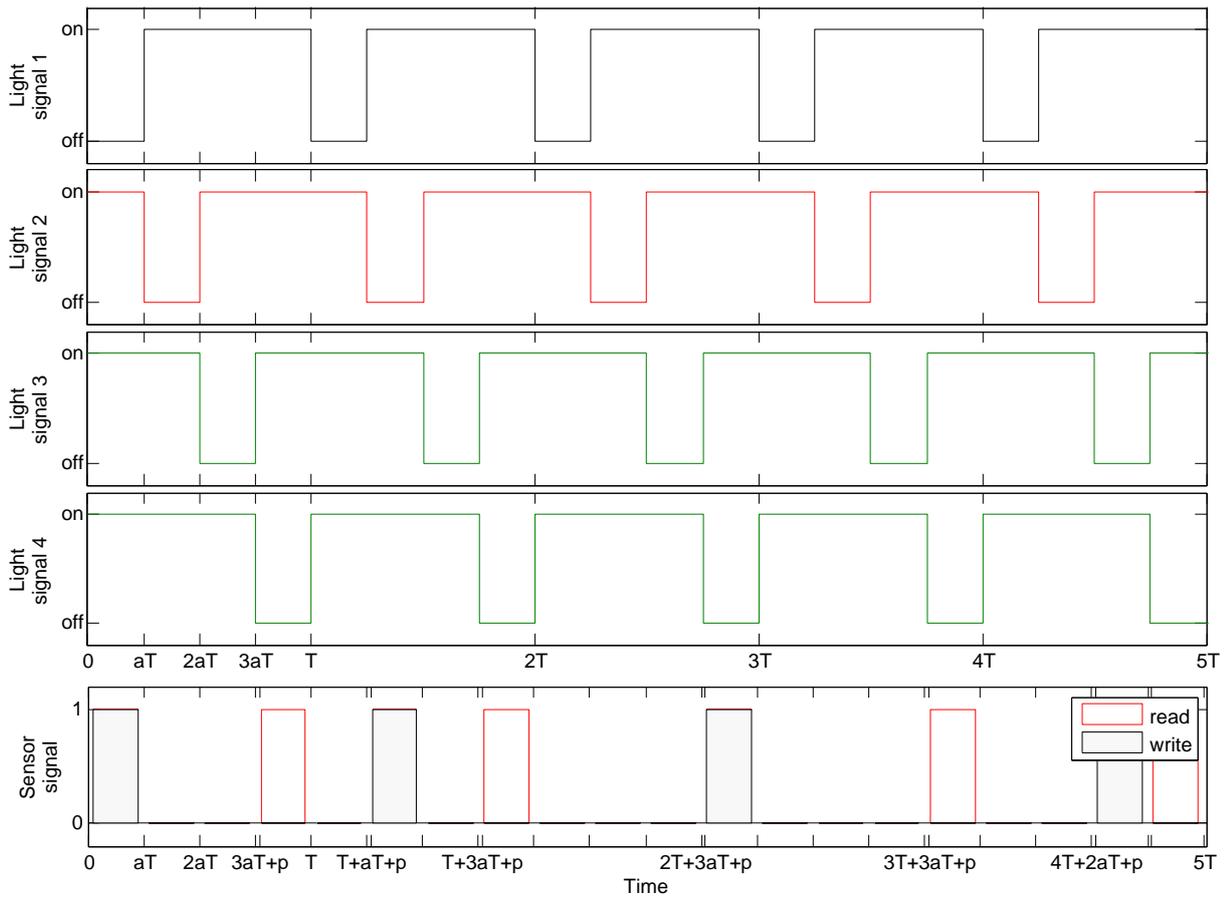
**Figure 3.9:** Sensor based intelligent lighting controller block diagram

$$\begin{aligned} \Delta int_3 = & \frac{1}{3} * (int_{meas,1} - int_{ambient}) + \frac{1}{3} * (int_{meas,2} - int_{ambient}) \\ & - \frac{2}{3} * (int_{meas,4} - int_{ambient}) + \frac{1}{3} * (int_{meas,3} - int_{ambient}) \end{aligned} \quad (3.16)$$

$$\begin{aligned} \Delta int_4 = & \frac{1}{3} * (int_{meas,1} - int_{ambient}) + \frac{1}{3} * (int_{meas,2} - int_{ambient}) \\ & + \frac{1}{3} * (int_{meas,4} - int_{ambient}) - \frac{2}{3} * (int_{meas,3} - int_{ambient}) \end{aligned} \quad (3.17)$$

$int_{meas,k}$ ,  $k = 1, 2, 3, 4$  is the asserted illuminance values during the  $k^{th}$  measurement. The first measurement takes place between  $t = 0$  and  $t = aT$ , where  $t = 0$  is the moment when the calibration routine starts. The second, third and fourth measurements are taken respectively between  $T + aT$  and  $T + 2aT$ ,  $2T + 3aT$  and  $3T$  and between  $4T + 2aT$  and  $4T + 3aT$ .  $int_{ambient}$  is the ambient illuminance without solid state lighting. The ambient measurement is called dark measurement and happens at the end of the calibration routine. Since all measurements are taken within a very short time, we assume that ambient lighting does not change during measurement number 1 to 4 and when the dark measurement is made.

The transfer-function is



**Figure 3.10:** Solid state light source driver signal and sensor board trigger signal for single point calibration routine

$$int(x, channel_{j,i}) = \sum_{i=1}^4 (d_i(x) * \sum_{j=1}^3 (channel_{j,i}) * k_j) * int_{norm}, \quad (3.18)$$

$$k_j = \frac{int_{max,j}}{\sum_{j=1}^3 int_{max,j}}. \quad (3.19)$$

$channel_{j,i}$  is the driver signal for channel  $j$  and light fixture  $i$ , which is the input of the controlled system. It also describes the normalized intensity of the  $i^{th}$  light and the  $j^{th}$  channel.  $k_j$  describes the brightness of channel  $j$  compared to the total brightness. The switching pattern, as you can see in figure 3.10, repeats consistently in order to create continuous illuminance by means of PWM light control. The PWM carrier frequency in this case is  $f_{calibration} = 1/T$ . This PWM carrier frequency must be distinguished from the PWM carrier frequency outside the calibration routine, which is used to control the light intensity. The maximal possible frequency,  $f_{calibration} = 1/T$  is 62 Hz, which is achievable using the recent firmware version of the sensor board and serial communication with our testbed pc based on the windows.h library in the Windows 7 environment (see chapter 3.3.2). This limitation is due to kernel limitations for serial communication. The frequency must be greater than the critical flicker frequency (CFF) in order to create a flicker-free lighting experience. For this reason, the calibration signals are designed in a way that requires the least number of serial commands within one period  $T$ . The pattern of the signal also minimizes the number of necessary driver commands. The PWM duty cycle is chosen to be as high as possible to reduce flicker. Ideally the calibration should run with  $f_{calibration} = f$  so that the PWM carrier frequency does not have to be changed from  $f$  to  $f_{calibration} \neq f$  during the calibration process (see chapter 5). Change of switching frequency is visible to the user and therefore undesirable. In order to mitigate this effect, carrier frequency changes only happen once at the beginning and once at the end of the calibration routine and occur at the same time for every light fixture. Intensity fade-out takes place at the end of the calibration routine. The dark measurement is taken as soon as all the lights reach the off-state. The calibration runs for  $P=500$  periods, which means it has a total duration of  $\Delta T_{calibration} = T * 500 = 8s$ . An appropriate duration  $\Delta T_{calibration}$  is important in this case, because the intensity level changes from the previous lighting condition to 75% duty cycle during the calibration. If the intensity level does not remain with 75% duty cycle for a sufficient amount of time before the fade-out, this change of intensity appears as a flash. In contrast, a persistence of 8 seconds creates a pleasant visual feedback, which identifies that the calibration routine is running, and ensures sufficient uniform lighting over the entire area during the calibration process. This restriction of the duty cycle during the calibration also results from the performance limitation of serial communication and the priority to surpass the frequency threshold for flicker-free lighting.

The major disadvantage of single point calibration is that the optimization is performed for only a single point of interest. Thus, multipoint and area average calibration are developed based on the single point calibration routine.

Multipoint calibration performs single point calibration for multiple points of interest. These points represent an area of interest. They can be evenly distributed along the edge of the area and in the middle of the area, depending on its size. The estimation of the transfer-function for each of the  $n$  selected points occurs in the same way as the single point calibration. This means that the user places the sensor board at one point, executes single point calibration, and after its completion he moves the sensor to the next point and repeats the calibration. Each point of interest could be calibrated within approximately  $5 * T = 0.08$  seconds. This means that the movement from point 1 via point 2, point 3 and so on, to point  $n$  could be a fluent motion if single point calibration is synchronously triggered.

Area average calibration estimates the average of the distribution factors  $d_i(x_l)$  within the area of interest. In this case, the user moves the sensor evenly inside the area of interest during the calibration. Single point calibration is executed for a high number of different points within the area of interest, and the software processes the average of the measured intensities.

$$d_i(x) = \overline{(d_i(x_l))} \quad (3.20)$$

The three different calibration routines require three distinct optimization routines since they provide different information about the area of interest. The optimization routines are single point optimization, multipoint optimization and area average optimization respectively for the corresponding calibration routines.

The input values of the single point optimization routine are light contribution factors  $d_i$  with  $i = 1, 2, 3, 4$ , reference intensity and reference CCT. In order to determine the most efficient settings for the lighting system, I use linear programming to solve the optimization problem. The objective is the energy consumption and the constraints manifest the intensity and CCT preferences at the area of interest. As described in chapter 3.1, the total energy consumption is

$$\begin{aligned} E(dm x) &= \sum_{i=1}^4 (E_i(channel_{1,i}, channel_{2,i}, channel_{3,i})) \\ &= \sum_{i=1}^4 (15.7779 * channel_{1,i} + 15.9031 * channel_{2,i} + 15.8613 * channel_{3,i} + 1.022) \end{aligned} \quad (3.21)$$

with

$$x = dm.x = \begin{bmatrix} channel_{1,1} \\ channel_{1,2} \\ channel_{1,3} \\ channel_{1,4} \\ channel_{2,1} \\ channel_{2,2} \\ \vdots \\ channel_{4,3} \\ channel_{4,4} \end{bmatrix}$$

and

$$channel_{j,i} \in [0, 1]$$

The cost vector is accordingly

$$c = [15.7779 \quad 15.7779 \quad \dots \quad 15.9031 \quad 15.9031 \quad \dots \quad 15.8613 \quad 15.8613 \quad \dots \quad ].$$

It follows that

$$fitness = c * x.$$

The objective is

$$min\{c * x\}, x \geq 0. \quad (3.22)$$

The constraints are

$$x = dm.x \leq 1, \quad (3.23)$$

$$\sum_{i=1}^4 \left( \sum_{j=1}^3 (d_i * channel_{j,i} * k_j) \right) = [d * k_1 \quad d * k_2 \quad d * k_3] * x = \frac{int_{ref}}{int_{norm}}, \quad (3.24)$$

and

$$\begin{aligned} \frac{\sum_{i=1}^4 (int_{6000K,i})}{\sum_{i=1}^4 (int_{3500K,i})} &= \frac{\sum_{i=1}^4 (d_i * channel_{2,i} * k_2)}{\sum_{i=1}^4 (d_i * (channel_{1,i} * k_1 + channel_{3,i} * k_3))} \\ &= ratio(CCT_{ref}) \end{aligned} \quad (3.25)$$

With

$$\sum_{i=1}^4(int_{6000K,i}) + \sum_{i=1}^4(int_{3500K,i}) = int_{ref} \quad (3.26)$$

the last constraint becomes

$$\begin{aligned} \sum_{i=1}^4(int_{6000K,i}) &= \sum_{i=1}^4(d_i * channel_{2,i} * k_2) * int_{norm} \\ &= \frac{int_{ref}}{\left(1 - \frac{1}{ratio(CCT_{ref})}\right)} \end{aligned} \quad (3.27)$$

and

$$\begin{aligned} \sum_{i=1}^4(int_{3500K,i}) &= \sum_{i=1}^4(d_i * (channel_{1,i} * k_1 + channel_{3,i} * k_3)) * int_{norm} \\ &= \frac{int_{ref}}{(1 + ratio(CCT_{ref}))} \end{aligned} \quad (3.28)$$

$ratio(CCT_{ref})$  is the inverse function of formula 3.3. The cubic polynomial fit is

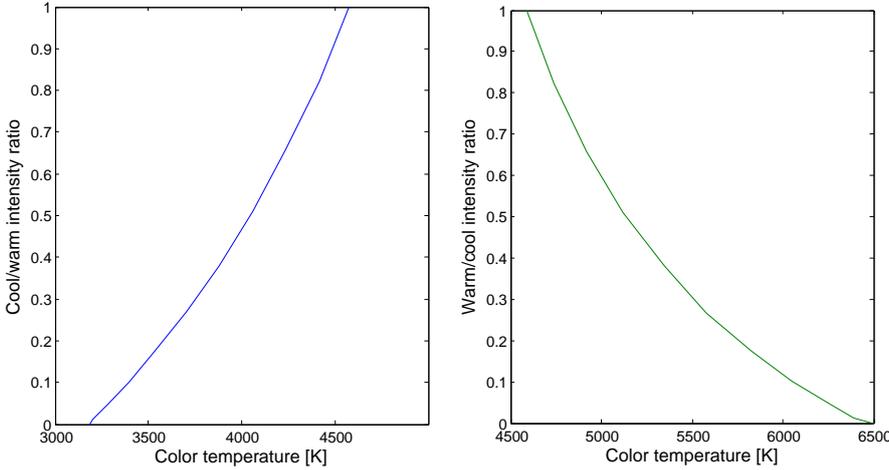
$$ratio(CCT) = \begin{cases} p1A * cct_{ref}^3 + p2A * cct_{ref}^2 + p3A * cct_{ref} + p4A, \\ \text{for } 6500 \geq CCT \geq 4600, \\ p1B * cct_{ref}^3 + p2B * cct_{ref}^2 + p3B * cct_{ref} + p4B, \\ \text{for } 4600 > CCT \geq 3000 \end{cases} \quad (3.29)$$

with

$$p1A = -1.031e - 010, p2A = 1.951e - 006, p3A = -0.01255, p4A = 27.48$$

and

$$p1B = 1.011e - 010, p2B = -9.238e - 007, p3B = 0.003274, p4B = -4.321.$$



**Figure 3.11:** Color temperature and cool/warm white light intensity relationship

The root mean square error and the coefficient of determination of the curve fit are RMSE = 0.002239, R-square = 1 for  $6500 \geq CCT \geq 4600$  and RMSE = 0.008697, R-square = 0.9996 for  $4600 > CCT \geq 3000$ .

Since the optimization algorithm minimizes the total power consumption, the constraints 3.24, 3.27 and 3.28 lead to equivalent results as

$$\sum_{i=1}^4 \sum_{j=1}^3 (d_i * channel_{j,i} * k_j) \geq int_{ref}/int_{norm} \quad (3.30)$$

$$\sum_{i=1}^4 (int_{6000K,i}) \geq \frac{int_{ref}}{(1 - \frac{1}{ratio(CCT_{ref})})} \quad (3.31)$$

and

$$\sum_{i=1}^4 (int_{3500K,i}) \geq \frac{int_{ref}}{(1 + ratio(CCT_{ref}))}. \quad (3.32)$$

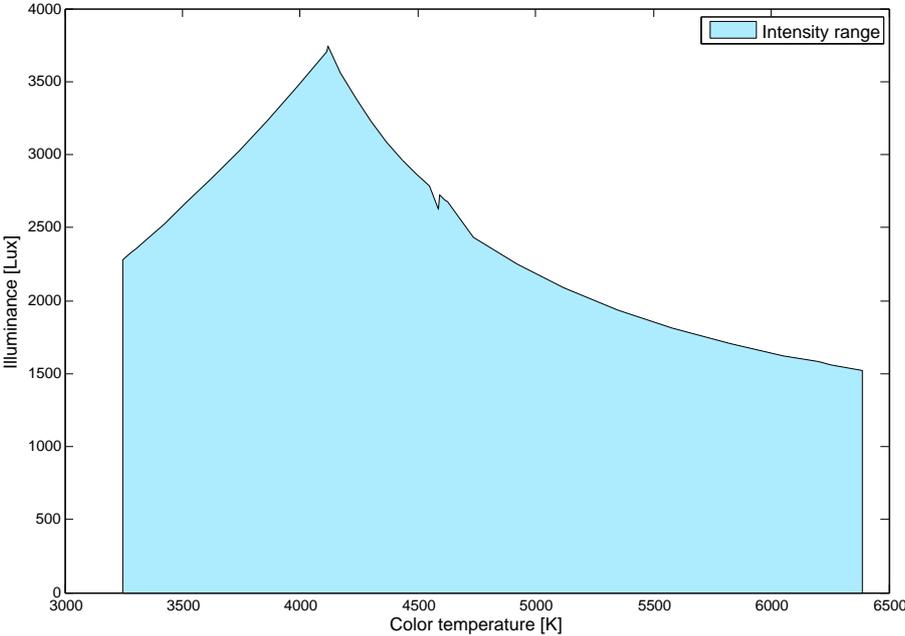
Except for the case where required intensity exceeds the intensity range of white points with reference CCT, the constraints 3.24, 3.27 and 3.28 lead to different results than 3.30, 3.31 and 3.32. The achievable intensity range is tied to the choice of CCT because of the design of the lights, as shown in figure 3.12. In this exceptional case the advantage of using 3.30, 3.31 and 3.32 is that the intensity constraint has a greater priority than the color temperature constraint. The required intensity is provided, but the accuracy of the produced color temperature is reduced. The optimization algorithm chooses a white point that achieves the preferred intensity and has a color temperature as close as possible to reference CCT. The color temperature will be shifted toward 4100 K.

The formal statement of the LP-problem is

$$\begin{aligned} \min \{c * x\}, x \geq 0 \\ A * x \leq b \end{aligned} \quad (3.33)$$

with

$$c = [15.7779 \quad 15.7779 \quad \dots \quad 15.9031 \quad 15.9031 \quad \dots \quad 15.8613 \quad 15.8613 \quad \dots \quad ],$$



**Figure 3.12:** Intensity range vs. color temperature, the brightest white points have a color temperature around 4100 K, the achievable intensity decreases with increasing color temperature difference from 4100 K

$$A = \begin{bmatrix} I_{12 \times 12} & & \\ -d * k_1 & -d * k_2 & -d * k_3 \\ -d * k_1 & 0 & -d * k_3 \\ 0 & -d * k_2 & 0 \end{bmatrix}$$

and

$$b = \left[ 1 \quad 1 \quad \cdots \quad 1 \quad 1 \quad \frac{int_{ref}}{int_{norm}} \quad \frac{int_{ref}/int_{norm}}{(1+ratio(CCT_{ref}))} \quad \frac{int_{ref}/int_{norm}}{(1-\frac{1}{ratio(CCT_{ref}))})} \right].$$

Multiple point optimization solves the LP-optimization problem for multiple points of interests. For  $n = 2$  points the constraints would be

$$x \leq 1 \tag{3.34}$$

$$\sum_{i=1}^4 \left( \sum_{j=1}^3 (d_{1,i} * channel_{j,i} * k_j) \right) = [d_1 * k_1, d_1 * k_2, d_1 * k_3] * x = \frac{int_{ref}}{int_{norm}}, \tag{3.35}$$

$$\sum_{i=1}^4 \left( \sum_{j=1}^3 (d_{2,i} * channel_{j,i} * k_j) \right) = [d_2 * k_1, d_2 * k_2, d_2 * k_3] * x = \frac{int_{ref}}{int_{norm}}, \tag{3.36}$$

$$\sum_{i=1}^4 (d_{1,i} * channel_{2,i} * k_2) = \frac{int_{ref}/int_{norm}}{(1 - \frac{1}{ratio(CCT_{ref})})} \tag{3.37}$$

$$\sum_{i=1}^4 (d_{2,i} * channel_{2,i} * k_2) = \frac{int_{ref}/int_{norm}}{(1 - \frac{1}{ratio(CCT_{ref})})} \tag{3.38}$$

$$\sum_{i=1}^4 (d_{1,i} * (channel_{1,i} * k_1 + channel_{3,i} * k_3)) = \frac{int_{ref}/int_{norm}}{(1 + ratio(CCT_{ref}))} \tag{3.39}$$

$$\sum_{i=1}^4 (d_{2,i} * (channel_{1,i} * k_1 + channel_{3,i} * k_3)) = \frac{int_{ref}/int_{norm}}{(1 + ratio(CCT_{ref}))} \tag{3.40}$$

From this it follows, that

$$A = \begin{bmatrix} I_{12 \times 12} & & \\ -d1 * k_1 & -d1 * k_2 & -d1 * k_3 \\ -d2 * k_1 & -d2 * k_2 & -d2 * k_3 \\ -d1 * l_1 & 0 & -d1 * k_3 \\ -d2 * k_1 & 0 & -d2 * k_3 \\ 0 & -d1 * k_2 & 0 \\ 0 & -d2 * k_2 & 0 \end{bmatrix}$$

and

$$b = \begin{bmatrix} 1 & \dots & 1 & \frac{int_{ref}}{int_{norm}} & \frac{int_{ref}}{int_{norm}} & \frac{int_{ref}/int_{norm}}{(1+ratio(CCT_{ref}))} & \frac{int_{ref}/int_{norm}}{(1+ratio(CCT_{ref}))} & \frac{int_{ref}/int_{norm}}{(1-\frac{1}{ratio(CCT_{ref})})} & \frac{int_{ref}/int_{norm}}{(1-\frac{1}{ratio(CCT_{ref})})} \end{bmatrix}.$$

The cost vector remains the same as for single point optimization. The number of constraints increases linearly with the number of points multiplied with factor 3. Area average calibration allows the consideration of many points within the area of interest without increasing the number of constraints. Since it returns an average distribution vector, solving the LP-optimization does not guarantee uniform illuminance within the area of interest. Without any additional processing, the optimization algorithm activates the lights one after another, sorted from high to low distribution factor to provide the desired amount of light. The active light fixture might only partially illuminate the area of interest or create very uneven illumination within the area of interest. The LP-optimization does not take this into account. In order to reduce this effect, size and position of the area are examined in more detail below. The variance of the contribution factors is a good way to characterize size and position. If the variance is small, it means the user marked a bigger area or the area of interest is located in between light fixtures, and that in order to provide uniform lighting, several lights need to be turned on. If the variance is high, the marked area is small or located right under a light fixture. The area average optimization thus involves two steps. The first step is solving the linear program. The second step is to compare the contribution factors  $d_i$ . If the difference between  $d_i$  and  $d_j$  with  $i, j \in [1, 2, 3, 4]$ ,  $i \neq j$  is smaller than the predefined, relative threshold, both lights will be forced to turn on. The total intensity remains, but it is provided by both sources weighted by their contribution factors. If the threshold value is not reached, the LP solution does not need to be changed.

Once the optimal setting is found, the light sources are actuated accordingly. The illuminance at the sensor position, which ideally is the point of interest or a point inside the area of interest, is fed back to controller. The low bandwidth lux sensor is used in this case, because the unfiltered lux sensor is sensitive in the frequency band up to the PWM carrier frequency  $f$ , whereas the filtered lux sensor measures the average illuminance. The controller is a PI controller with  $P = 0.001$  and  $I = 0.01$ . The integrator eliminates stationary control deviation. The constants are tuned to create asymptotic transient behavior without oscillation or peak overshoot.

Sensor format	CMOS
Resolution	RGB
Exposure mode	640 × 480
Exposure	manual
Backlight compensation	-5, -6, -7, -8
Focus mode	off
Focus	off
White balance mode	inf
Frame rate	manual
Gain	30 frames per second
	0

Table 3.2: Webcam configuration

### 3.2.2 Webcam Based Control

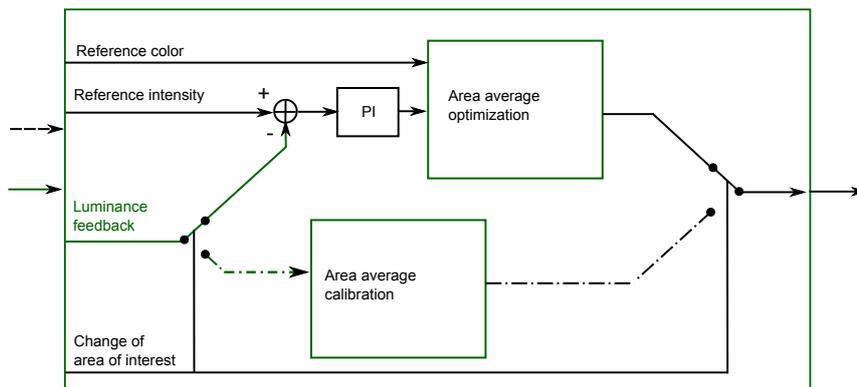


Figure 3.13: Webcam based intelligent lighting controller block diagram

The design of the camera based controller is equivalent to the sensor board based controller. The feedback device is a commercial USB webcam, the Logitech Quickcam Pro 9000. Table 3.2 contains the camera configurations.

Figure 3.15 shows the setup and figure 3.14 shows the result of the webcam offline calibration, which is distinguished from the online calibration routine. The experiment is performed in a dark enclosed box. The inside of the box is covered with black fleece. A sheet standard 216 mm × 279 mm copy paper with TAPPI Brightness 92 and ISO Brightness 104 is

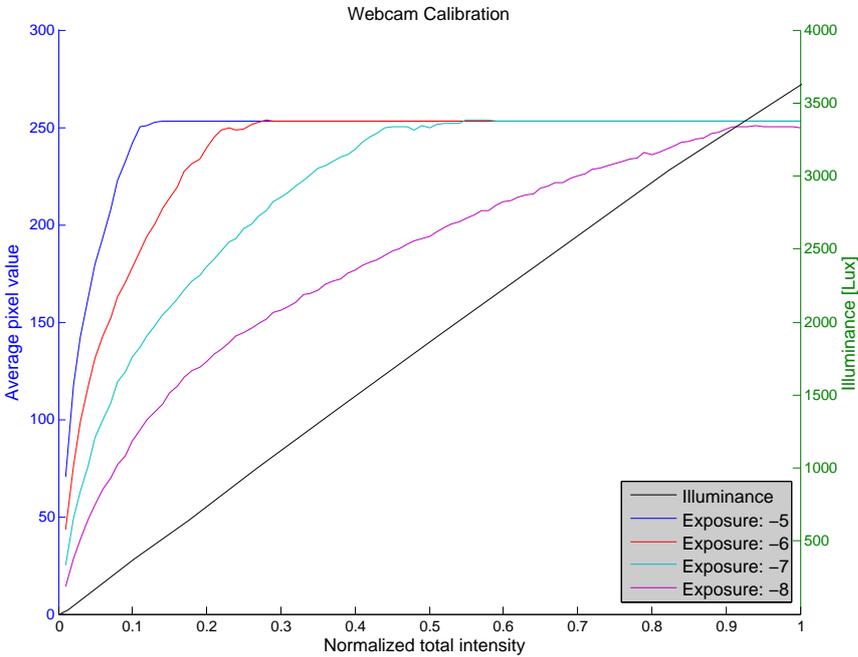


Figure 3.14: Webcam offline calibration result



**Figure 3.15:** Webcam offline calibration setup

attached to the box 30 cm in front of the iW BLAST 12. Copy paper is selected as the calibration object because our testbed is set up as an office environment. The webcam is mounted on the light fixture facing the sheet of paper with a distance of 30 cm. Since the green channel of the RGB image mostly contributes to the luminance of the image [15], the calibration is only performed for the green channel and all luminance measurement in the following refer to the green component. According to [15], the brightness level of a webcam image is

$$Bl = k * L * G * T * (F/\#)^{-2} \quad (3.41)$$

with

$Bl$ : brightness level

$k$ : constant

$L$ : luminance

$G$ : gain of automatic gain control

$T$ : integration time

$F/\#$ : aperture

$k$ ,  $T$ ,  $G$  and  $F/\#$  are constant. Since the illuminated object does not change during the experiment, luminance should be proportional to the normalized total intensity of the light source. Figure 3.14 shows the average pixel value of 29 pixels within an area with the size of the spectrometer head located in the middle of the light cone. As shown, camera feedback is non-linear. I assume that the webcam performs gamma correction, which is done in most

webcams, to mimic the non-linear perception of human eye. It follows that

$$\log(L_{eye}) = \log(Bl) - \log(k) - \log(G) - \log(T) - \log(F/\#^{-2}) \quad (3.42)$$

$$L_{eye} = f_{GammaCorrection}(L). \quad (3.43)$$

Analogous to our sensor board based controller, the webcam based controller contains a calibration routine, optimization routine and a PI controller. After the calibration routine is initiated, the user is instructed to draw the area of interest into the camera image. A mask is created according to the selection.

Since the communication to the camera is fairly slow (see chapter 3.3.2), the performance is not tuned to reach CFF. The calibration signal has a very simple structure and contains redundant measurements. Similar to the sensor board based calibration, each light is put into an on and off-state, and the luminance difference is measured. This is only performed for one period. The contribution factor is derived from the luminance difference. The contribution factors are

$$d_i = \Delta Bl_i. \quad (3.44)$$

$$L_{eye} \sim Bl. \quad (3.45)$$

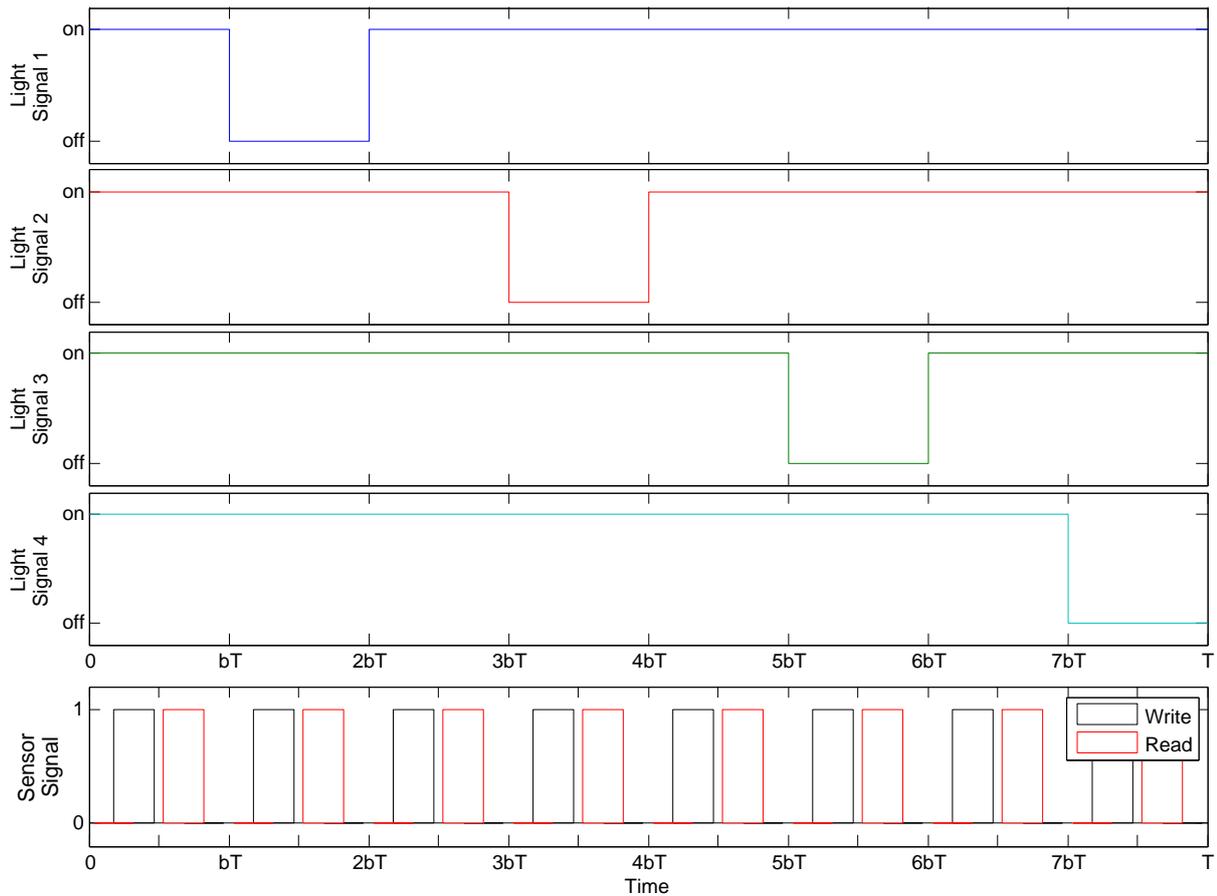
$$\Delta(Bl_1) = Bl_{meas,1} - Bl_{meas,2}, \quad (3.46)$$

$$\Delta(Bl_2) = Bl_{meas,3} - Bl_{meas,4}, \quad (3.47)$$

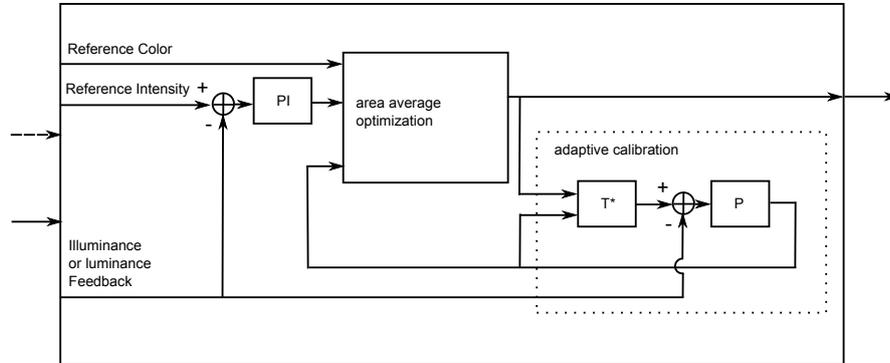
$$\Delta(Bl_3) = Bl_{meas,5} - Bl_{meas,6}, \quad (3.48)$$

$$\Delta(Bl_4) = Bl_{meas,7} - Bl_{meas,8} \quad (3.49)$$

$Bl_{meas,k}$  is the measured brightness level during the  $k^{th}$  measurement. The first measurement is performed between  $t = 0$  and  $t = bT$ , where  $t = 0$  is the moment when the calibration routine starts. The second measurement is taken between  $t = bT$  and  $t = 2bT$ , and so on.  $\Delta Bl_i$  is the brightness difference of light source number  $i$  with gamma correction.  $Bl$  could be the mean  $Bl_{mean}$  or the median  $Bl_{med}$  of the pixel values in the area of



**Figure 3.16:** Solid state light source driver signal and webcam trigger signal for webcam calibration routine



**Figure 3.17:** Adaptive controller block diagram

interest. The benefit of using the median instead of the mean value is that the median is less sensitive to outliers. For instance, a small black object inside a bright area will not change the total output as much as the average measurement. But on the other hand, if the image has very high contrast,  $Bl_{mean}$  is more stable than  $Bl_{med}$ . In the following I will use  $Bl = Bl_{med}$ .

The webcam based optimization routine is identical to the sensor board based area average optimization routine. The PI controller is tuned considering the same criteria as the sensor board based controller for slower update rate. The settling time is higher compared to the sensor board based controller, because in contrast to illuminance measurement, luminance feedback does not only depend on the lighting condition, but also on the scene that the camera is capturing. Therefore luminance feedback is not as constant as illuminance feedback. The settling time is increased to reduce the influence of noise caused by activities in the area of interest. The constants  $P=0.01$  and  $I=0.01$  meet the tradeoff between adjustment latency and user friendly moderate light transitions.

### 3.2.3 Adaptive Control

The adaptive control does not need a calibration routine. The user does not have to actively calibrate the system and recalibrate after each change in the area of interest. He simply moves the sensor board to the area of interest or selects an area inside the camera image. The adaptive controller contains an adaptive calibration unit, which is a separate

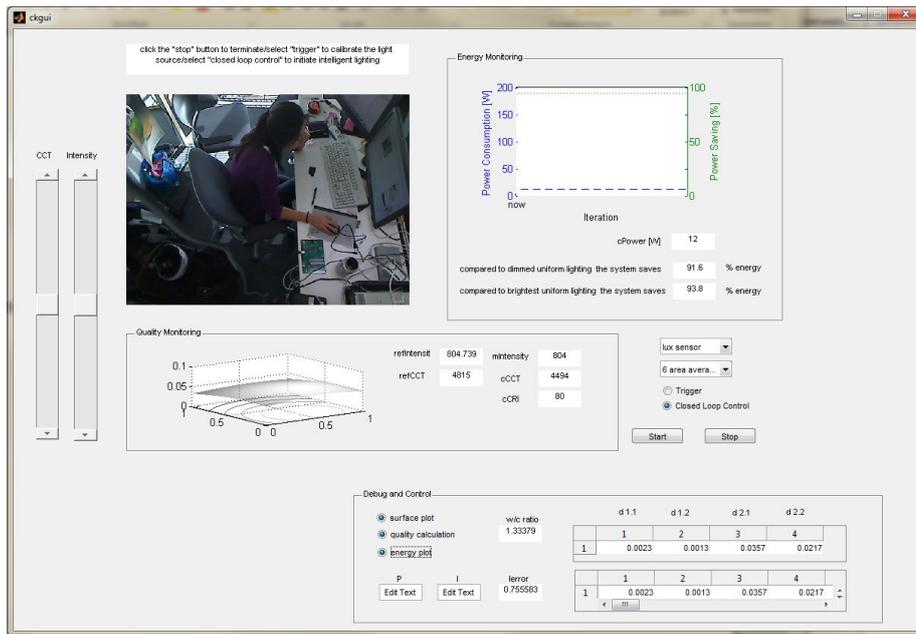


Figure 3.18: Intelligent lighting graphical user interface

closed-loop that adaptively guesses the transfer function. It observes the illuminance or luminance feedback and compares it with a prediction based on the last guess of the transfer function. It corrects the internal model depending on the error of the prediction. A simple proportional controller is used at this point.

### 3.3 GUI and Software

*CKGUI* is the intelligent lighting control software user interface. As shown in figure 3.18, the GUI contains input elements similar to the sensor board, which are sliders and radio buttons. The sliders enable intensity and CCT adjustment. The buttons activate detailed monitoring options and initiate the calibration and optimization routine. There are three panels on the GUI which represent different aspects of the controller. The *Energy Monitoring* panel provides information about energy consumption and savings. A figure and three text fields are displayed. The figure shows the energy consumption (blue) and energy savings (green) of the last 10 update cycles. The energy saving describes achieved savings in percent compared to constant maximum uniform lighting, which consumes  $E_{max} = 200$  W. In the text field  $cPower$  [W], you can see the estimated power consumption at the current moment in watts. The estimation of the power consumption is based on formula 3.8. The two lower text fields show the current power savings in percent compared to constant maximum uniform lighting and dimmed uniform lighting.

$$Powersaving_{max}(t) = \frac{(E_{max} - E(t))}{E_{max}} \quad (3.50)$$

$$Powersaving_{dim}(t) = \frac{(E_{dim}(t) - E(t))}{E_{dim}(t)} \quad (3.51)$$

$$E_{dim}(t) = \begin{cases} E_{max} * \frac{int_{ref}(t)}{int_0}, & \text{for } int_{ref} \leq int_0, \\ E_{max}, & \text{for } int_{ref} > int_0. \end{cases} \quad (3.52)$$

$$int_0 = 1100 \text{ [sensor counts]} \quad (3.53)$$

$int_0$  is the illuminance on the desk, when every light source is contributing maximum intensity and ambient lighting is zero. Here I assume that the area of interest is within a plane, which has the height of the working desk, and that the reference intensity is the needed intensity.

refIntensity	Reference intensity
refCCT	Reference CCT
mIntensity	Feedback intensity
cCCT	Calculated CCT using spectrum based system model without consideration of ambient light
cCRI	Calculated CRI using spectrum based system model without consideration of ambient light

**Table 3.3:** Quality monitoring panel text fields

The *Quality Monitoring* panel shows the illuminance characteristic. Table 3.3 shows the meaning of the variables.

Reference CCT and intensity are tied to the software or hardware slider input values. Software and hardware sliders are synchronized. The 3-D plot on the left shows the approximate illuminance surface without ambient light. The calculation is based on the simplified system model. The illuminance is normalized to  $int_{norm} = 3000$  lux. The x axis is normalized to 1.9 m and the y axis is normalized to 1.6 m. The coordinates (1, 1) represents the upper left corner of the desk and (0, 1) the upper right corner of the desk.

The Debug and Control panel include radio buttons to activate the illuminance surface plot, the energy consumption and saving plot and CCT and CRI calculation. Since these calculations and visualizations are computationally intensive, enabling them causes reduction of the update rate. Besides the buttons, there are several text fields to display parameters for the debugging process.

1 area average controller mode with observer (testing, lux sensing only)	Sensor board based adaptive calibration control, testing stage
2 data logging	controller mode with data logging functionalities for short term studies
3 single point controller mode	Single point sensor board based control
4 single point constant scan controller mode (not complete)	Single point Sensor board or webcam based control with continuously triggered calibration, not completed
5 multi point controller mode (lux sensing only)	Multipoint sensor board based control
6 area average controller mode	Area average sensor board or webcam based control

**Table 3.4:** Controller modes and project status

Outside the panels, the webcam image is shown in the upper left corner. This image is a live preview of the camera stream. If the user chooses to use camera based control, he can draw the area of interest into the image window at the beginning of the calibration routine. Above the image, there is a text field which shows instructions to help the user understand the GUI. On the left hand side you can find sliders to adjust reference intensity and CCT. The popup-menus on the right provide options for the sensing device (either the sensor board or the webcam) and controller options. The controller options are listed in table 3.4. In order to start the calibration routine the radio button, *Trigger*, needs to be selected. The *Closed-Loop Control* radio button initiates closed-loop control. Pressing the *Start* and *Stop* button activates and deactivates the interactive lighting system.

The GUI is linked to a MATLAB function, which executes the smart lighting control. Figure 3.19 and 3.20 show the main components.

### 3.3.1 Input Processing and PI Controller

The start button calls function *start\_Callback(hObject, eventdata, handles)* which executes a while loop until the stop button is pressed. At each iteration reference values and feedback sensor values are updated. If a sensor board based control mode is selected, the reference values are either the slider positions of the sensor board or the GUI. Since the user is able to use both input devices, the input values need to be synchronized after each change. The most recently updated value becomes the currently valid input. If webcam based control is selected, the GUI is the default interface. The hardware sliders have an accuracy of 0.01, and the GUI sliders a resolution of 0.001. Both input signals are in a range from 0 to 1. To compensate the noise of the analog signal of the sensor board sliders, the signal is filtered using a 10-order FIR Filter. The filter function is  $y(n) = \sum_{i=0}^n b_i * x(n - i)$  with  $n = 10$  and  $b_i = 0.1$ .

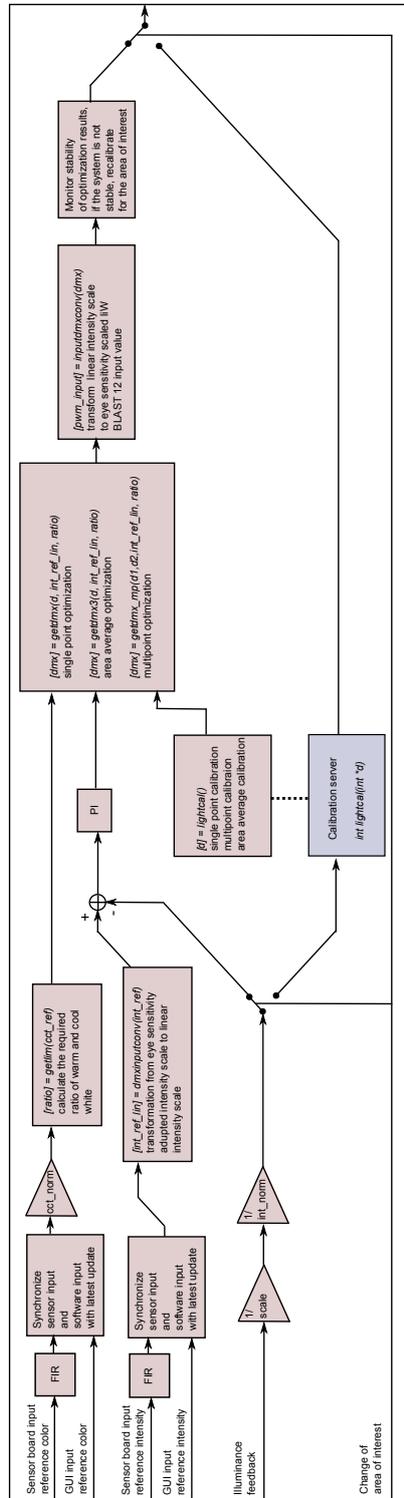


Figure 3.19: Sensor board based software block diagram

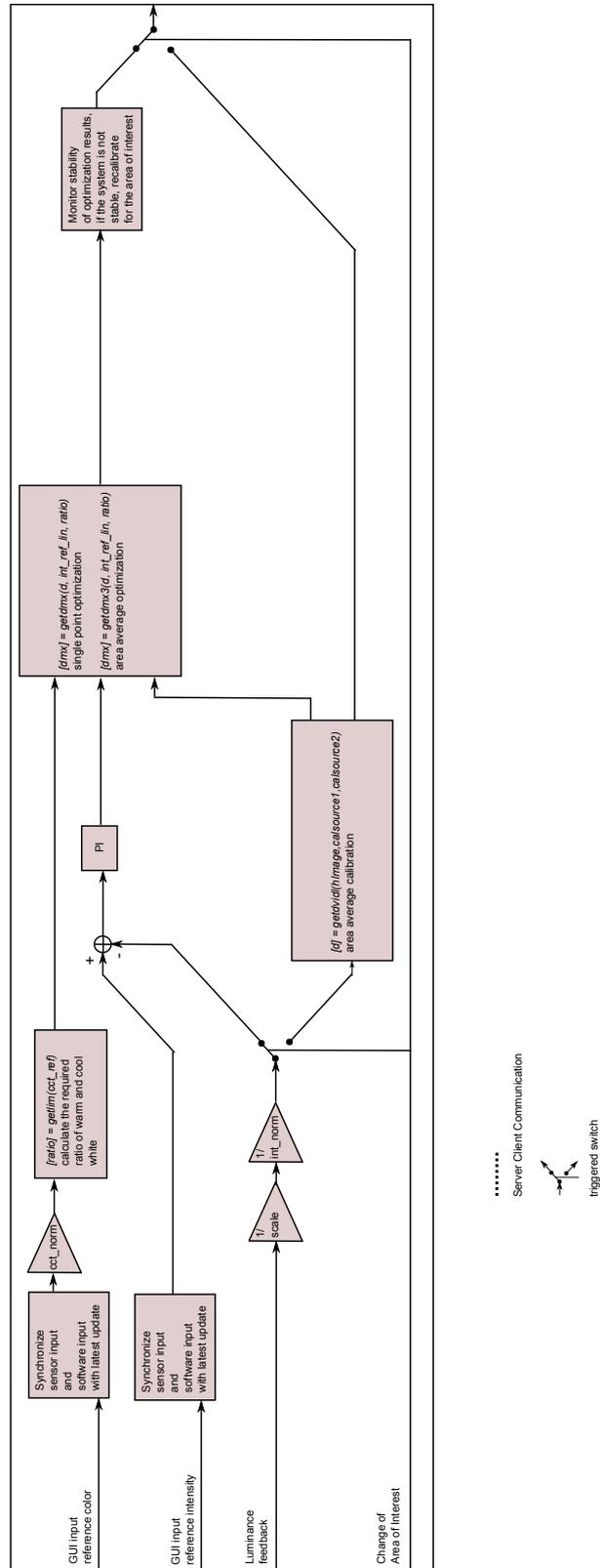
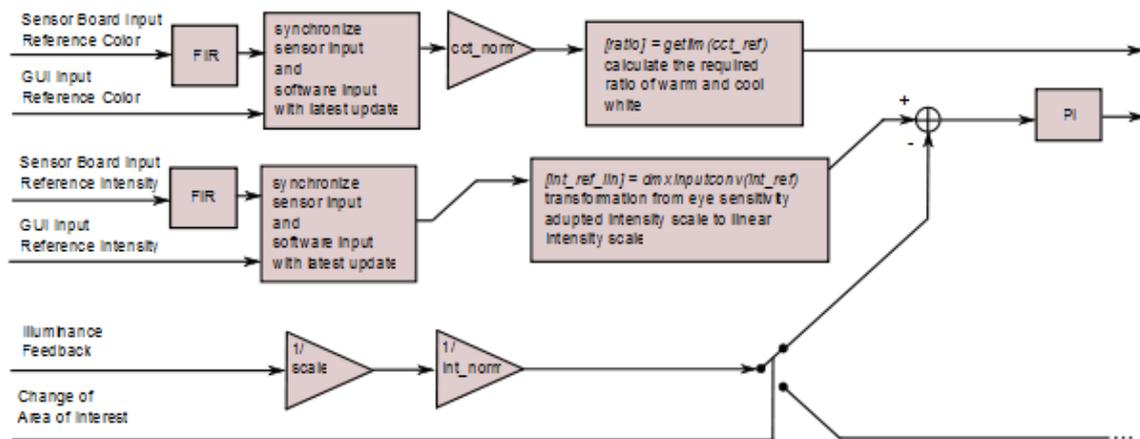
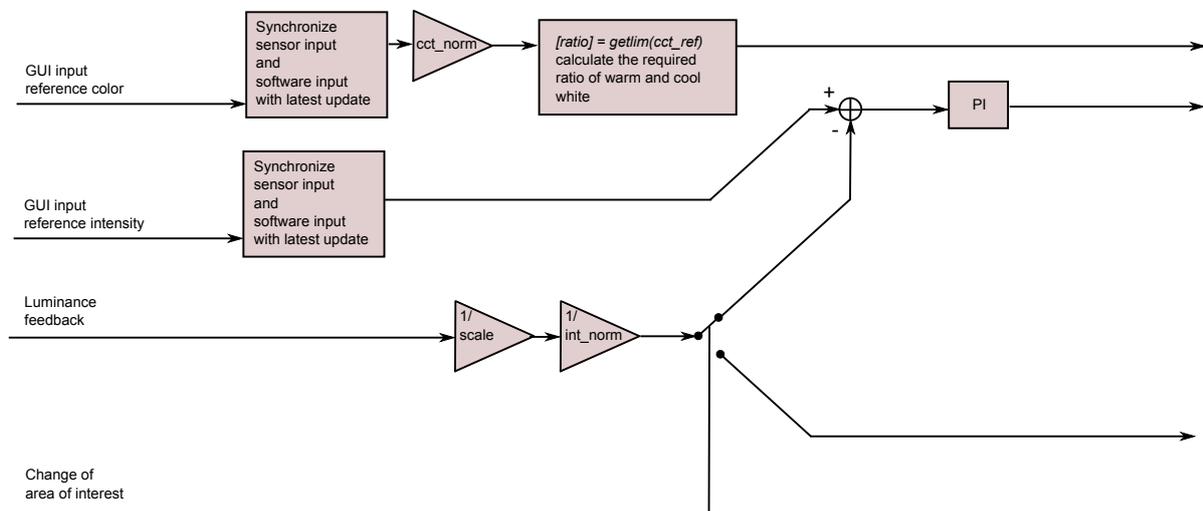


Figure 3.20: Webcam based software block diagram



**Figure 3.21:** Sensor board based input processing and PI controller software block diagram



**Figure 3.22:** Webcam based input processing and PI controller software block diagram

The reference input values are normalized to  $CCT_{norm} = CCT_{max} - CCT_{min}$ ,  $CCT_{max} = 6500$  K,  $CCT_{min} = 3000$  K and  $int_{norm} = 3000$  lux. The absolute reference CCT value is

$$CCT_{ref} = CCT_{ref,norm} * (CCT_{max} - CCT_{min}) + CCT_{min} \quad (3.54)$$

$$CCT_{ref,norm} = CCT_{slider} \quad (3.55)$$

$getratio(cct\_ref)$  calculates the required intensity-ratio of warm and cool white light using formula 3.29. The reference intensity input is aligned to the non-linear perception of the human eye. This means the intensity slider is scaled to follow the power law. The function  $dmxinputconv(int\_ref)$  converts the non-linear user input into linear and normalized lux values. Linear conversion is not necessary if webcam based control is selected because the webcam provides perception adapted feedback. The achievable illuminance or luminance depends on the position of the area of interest and ambient lighting. The upper limit is  $int_{max}(x) = \sum_{i=1}^4 d_i(x) * int_{norm} + int_{ambient}$  and the lower limit is  $int_{min}(x) = int_{ambient}$ .

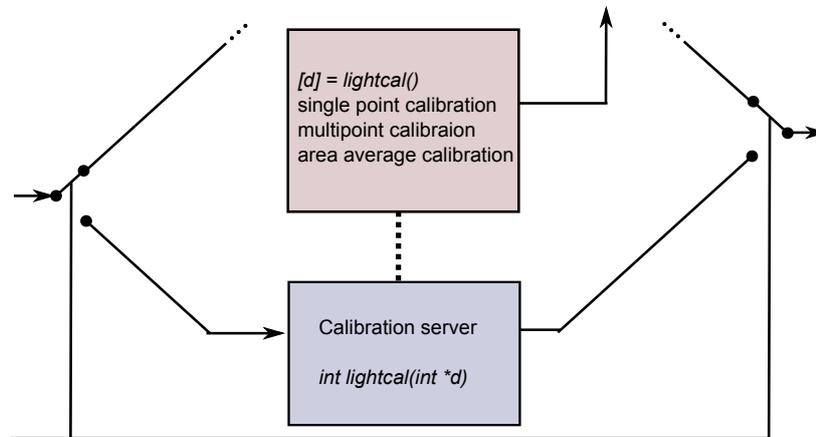
The sensor board feedback returns values in sensor counts. Since the lux sensor is linear, in order to estimate the lux value the sensor feedback only needs to be divided by the factor  $scale = 1.89$ . The converted lux value is then normalized with  $int_{norm}$ . The webcam feedback is an 8 bit pixel value. As described above, the camera returns non-linear feedback. The scale between non-linear lux reference input and the non-linear feedback is  $scale = 0.1$ . This is an approximation, but since the absolute lux value is not important for the user, this approximation does not affect the functionality of the system. The PI controller is reset after every execution of the calibration routine and when the closed-loop control is disabled.

### 3.3.2 Calibration Routine

To initiate the calibration routine, the user can either press the button on the sensor board when the sensor board is the active sensor or otherwise by selecting the trigger radio button.

For sensor board based control, the calibration routine calls the function  $lightcal()$ . This function initiates a TCP/IP connection to the calibration server, which is a local host, over port 12345. Once the connection is established, the calibration server executes the calibration routine as described in 3.2.1, returns the result to the client and disconnects the client. The calibration server then returns to listening mode. The server is programmed in C++ using a WINSOCKET tutorial template [3]. UDP and serial connections are programmed using WINSOCKET and windows.h library. The calibration routine is not developed in MATLAB because timer accuracy and performance of the serial communication in MATLAB were not sufficient to achieve CFF.

If webcam based control is selected, the calibration routine calls the function  $getdvid(hImage, calsource1, calsource2)$ .  $hImage$  is the camera stream object and  $calsource1$  and



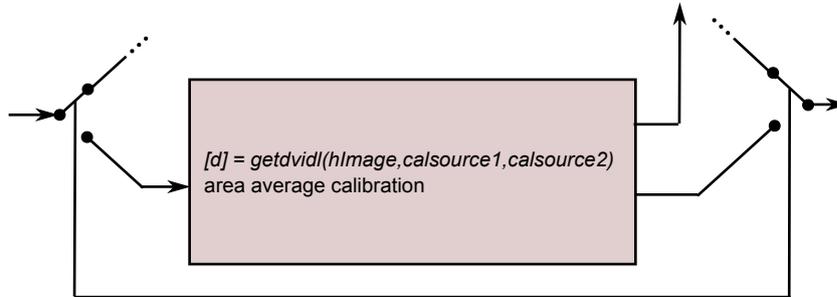
**Figure 3.23:** Sensor board based calibration software block diagram

*calsource2* are UDP objects representing the PDS-150e power supplies. *getdvid()* executes the calibration as described in 3.2.2 and shown in figure 3.16. The MATLAB instrument control toolbox is used to create the UDP connection and the image acquisition toolbox is used to access the webcam. The preview window in the upper left corner of the GUI shows the live webcam stream. The live preview is created using *preview()*. The function *update\_liveint(obj,event,hImage)* is defined as the preview update function. It calculates the median and the average of the pixel values inside the area of interest. Unfortunately there is no guarantee that the preview update function is executed for every frame. If the previous execution is not finished before the next frame is loaded, the update function will not be executed for the new frame. The area of interest is defined using *imfreehand()* and saved through a binary mask created using *createMask()* and the image processing toolbox. The camera configurations can be found in table 3.2. This application is developed with the goal to show the possibilities and prospects of camera based control. It is not tuned for high performance image acquisition or image processing. The calibration frequency does not reach CFF.

### 3.3.3 Optimization Routine

The optimization routine requires the intensity ratio *ratio*, the intensity control value and the contribution factors  $d_i$ .

The function *getdmx(d, int\_lin, ratio)* executes the optimization routing for single point



**Figure 3.24:** Webcam based calibration software block diagram

optimization. It uses the function  $rsm(c,A,d,eps,index\_B,index\_V,bfs)$  to solve the optimization problem.  $rsm()$  implements the revised simplex method and is originally programmed by [9]. Several modifications were necessary to solve the linear program as describe in 3.2.1. For example, the solution basis vector,  $index\_B$ , and the vector containing the auxiliary variables,  $index\_V$ , need to be defined explicitly. The input variables are  $c = [15.7779 \ \cdots \ 15.9031 \ \cdots \ 15.8613 \ \cdots \ 0 \ \cdots \ 0]$ ,

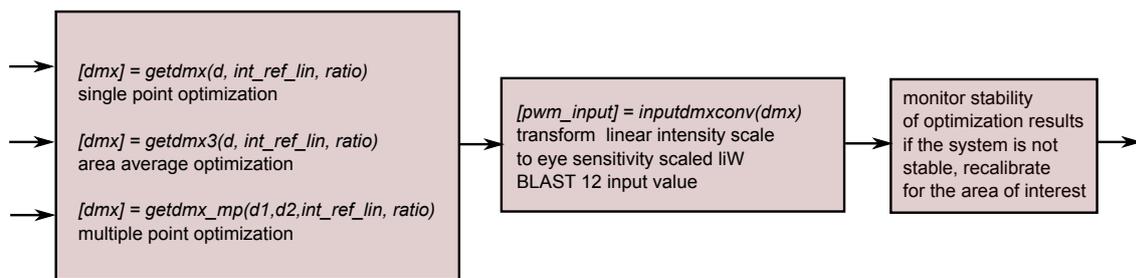
$$A = \begin{bmatrix} I_{12 \times 12} & & & \\ -d * k_1 & -d * k_2 & -d * k_3 & I_{15 \times 15} \\ -d * k_1 & 0 & -d * k_3 & \\ 0 & -d * k_2 & 0 & \end{bmatrix},$$

$$b = \left[ 1 \ \cdots \ 1 \ \frac{int_{ref}}{int_{norm}} \ \frac{int_{ref}/int_{norm}}{(1+ratio(CCT_{ref}))} \ \frac{int_{ref}/int_{norm}}{(1-\frac{1}{ratio(CCT_{ref})})} \right]$$

and

$$bfs = \left[ [1 \ \cdots \ 1]_{1 \times 12} \ [0 \ \cdots \ 0]_{1 \times 15} \right].$$

$bfs$  is the basic feasible solution and  $eps$  is a very small number, which describes the threshold for the approximation of zero. The function  $getd-mx3(d,int\_ref,ratio)$  solves the area average optimization problem. It uses the same function  $rsm(c,A,d,eps,index\_B,index\_V,bfs)$ . In a second step, it estimates the variance of the contribution factors and distributes duty among light sources with similar contributi-



**Figure 3.25:** Sensor board based optimization software block diagram



**Figure 3.26:** Webcam based optimization software block diagram

on factors described in 3.2.1 to ensure quasi-uniform lighting within the area of interest. The function *getdmx\_mp()* solves the multipoint optimization problem by extending the constraints to multiple points. In this case, two points are used to describe the area of interest. After the optimization the calculated, normalized intensities need to be transformed into corresponding driver inputs for the iW BLAST 12. This is done using *inputdmx\_conv(dmx)*.

Error is introduced to the system if the measured transfer function, or contribution vector, is not valid. This could happen if the adaptive controller is not used and the user does not recalibrate the system after moving the sensor to a different place or after changing the binary mask. If the change is significant, the system is put into an unstable state, and the illumination begins to oscillate. In order to prevent the system from remaining in this state, the stability monitoring block observes the spectral energy of the driver input signal. 10 samples of the driver signal are buffered and the spectral energy density is calculated. In an unstable state, the spectral energy of the middle frequency range does not converge to zero. The product of the spectral energy from the middle frequency range *PoS* is therefore an indicator for the stability of the system. If the system is stable, the spectral energy will converge to zero and therefore the product of its sequence will converge to the minimum.

$$PoS(n) = \begin{cases} PoS(n-1) * spectralEnergy(n), & \text{for } spectralEnergy(n) < spectralEnergyThres \\ & \text{and } PoS(n-1) * spectralEnergy(n) > 1, \\ PoS(n-1) * spectralEnergyThres, & \text{for } spectralEnergy(n) \geq spectralEnergyThres \\ & \text{and } PoS(n-1) * spectralEnergyThres > 1, \\ 1, & \text{for } PoS(n-1) * spectralEnergyThres \leq 1. \end{cases} \quad (3.56)$$

If the system is not stable, the product will increase very quickly. The software detects an unstable state if the spectral energy has been higher than a predefined threshold over a certain number of cycles, which means that the product of the sequence exceeds the stability threshold. After the detection of an unstable state, the software triggers the calibration routine to correct the false transfer function.

## 3.4 Results

### 3.4.1 System Model Evaluation

100 white points were chosen with different intensities for the evaluation of the system model. CCT and CRI were measured, as well the intensity at a defined position for the

white test points. The experiment was performed in a dark room. Only one light fixture was active. The distance between the active light source and sensor position was 0.3m. Figure 3.27 to figure 3.30 show the predicted values using the described system model in comparison to the actual measured values under the described circumstances.

### 3.4.2 Calibration Evaluation

In this section I show the results of four experiments. Experiment one shows the absolute intensity contribution of each light source in sensor counts measured using the sensor board based single point calibration. The calibration routine was performed 15 times at the same point of interest. The mean sensor values and standard deviations are shown in table 3.5. Multipoint calibration performs single point calibration for each point of interest, and thus it has the same accuracy as single point calibration.

	$d_1$	$d_2$	$d_3$	$d_4$
average	298.87	10.53	20.67	7.53
standard deviation	1.36	0.52	0.62	0.52

**Table 3.5:** Average intensity contribution and standard deviation of 15 calibration measurements at a constant point of interest using single point calibration

Experiment two shows results of 15 area average calibrations using the sensor board for a constant area of interest. The area of interest is linear and about 80 cm long. It is located on the writing desk in the middle of light source 1 and 3. Area average calibration has low repeatability compared to single point or multipoint calibration, because the measured contribution factors are highly dependent on the movement of the sensor during the calibration process. If the speed of movement changes during the calibration process, the area that is measured with lower speed becomes more dominant. The mean sensor values in sensor counts and standard deviations are shown in table 3.6.

	$d_1$	$d_2$	$d_3$	$d_4$
average	194.13	4.6	145.67	6.33
standard deviation	35.77	5.74	36.51	5.49

**Table 3.6:** Average contribution factors and standard deviation of 15 calibration measurements at a constant area of interest using area average calibration

The third experiment investigates the uniformity of illumination at the area of interest using sensor board based area average and multipoint calibrations. The area of interest is the same for both calibration methods. It is linear, about 80 cm long and located in between and underneath light fixture 1 and light fixture 3 on the working desk. After the calibration, the sensor board is placed in the middle of the area of interest and the reference intensity is set to  $400 \frac{\text{lux}}{\text{scale}}$ . Once the illuminance reaches the reference value, 5

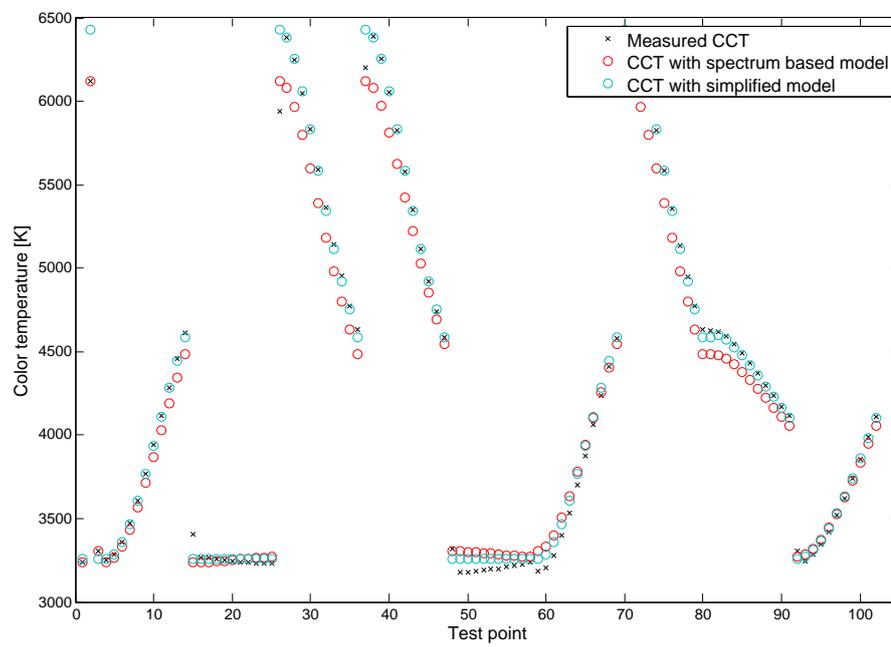


Figure 3.27: Model evaluation for color temperature

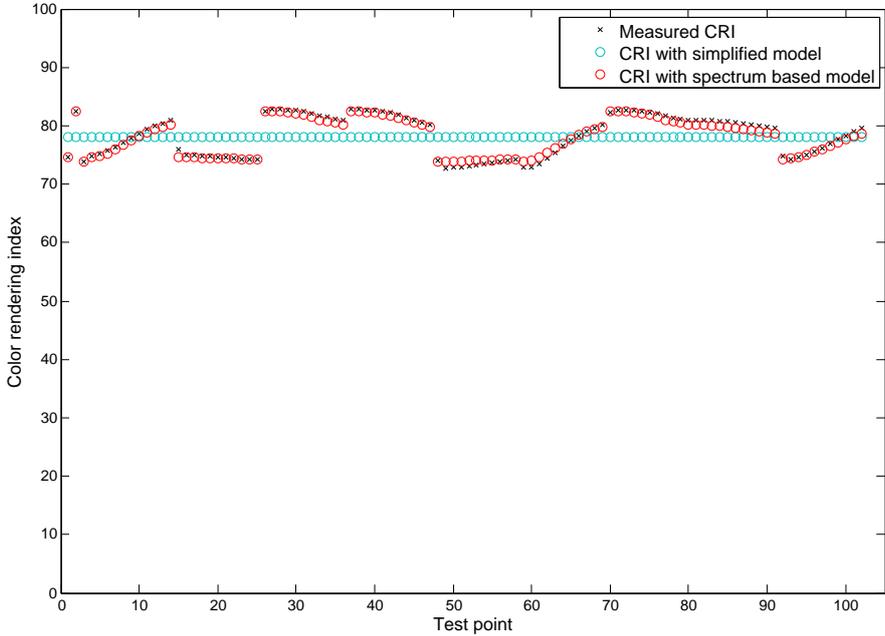
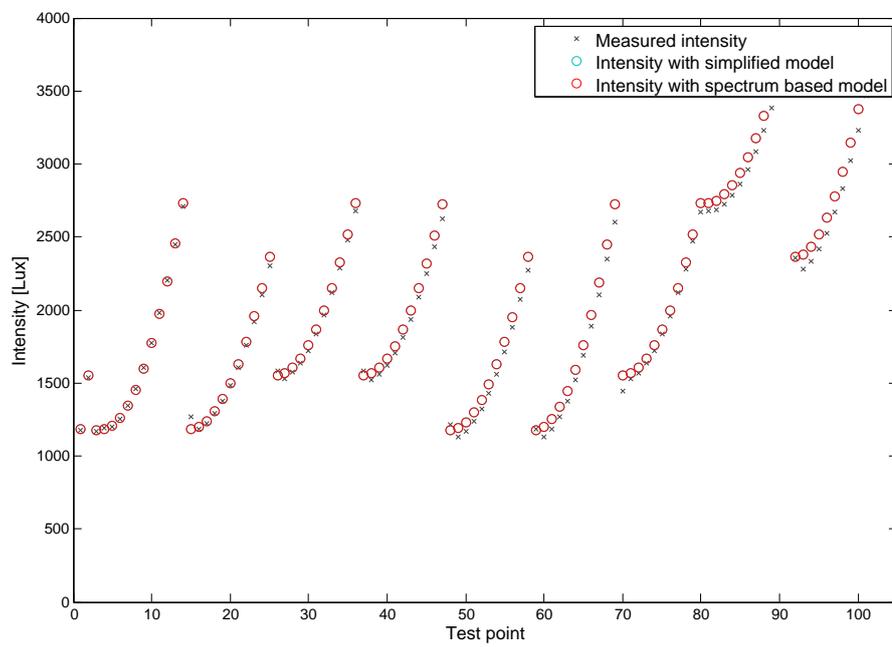


Figure 3.28: Model evaluation for color rendering index



**Figure 3.29:** Model evaluation for intensity in lux

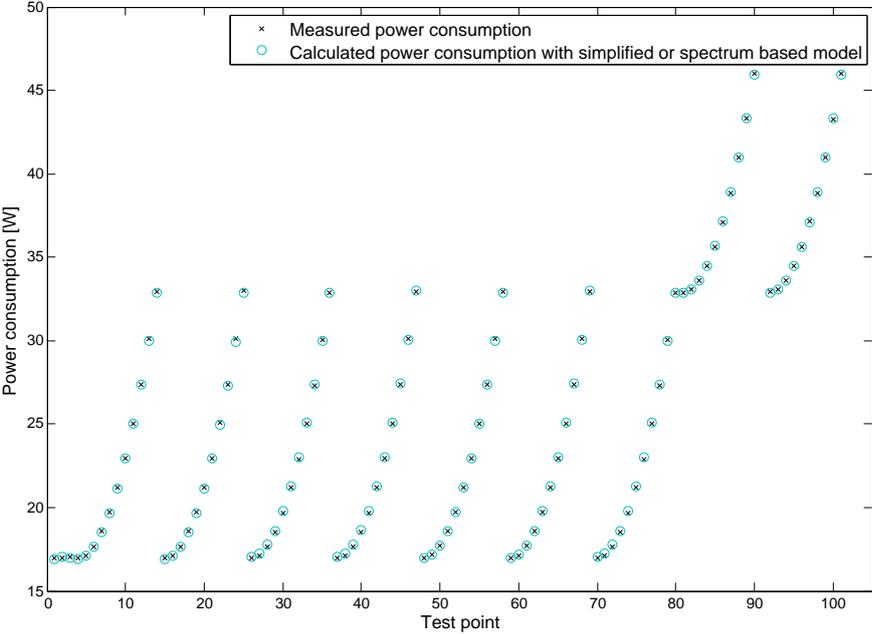
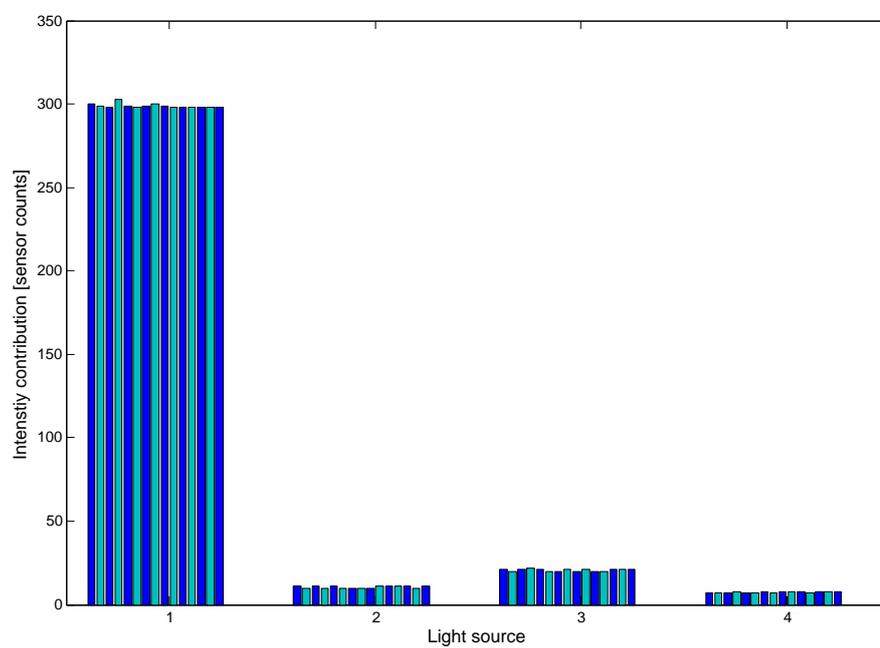
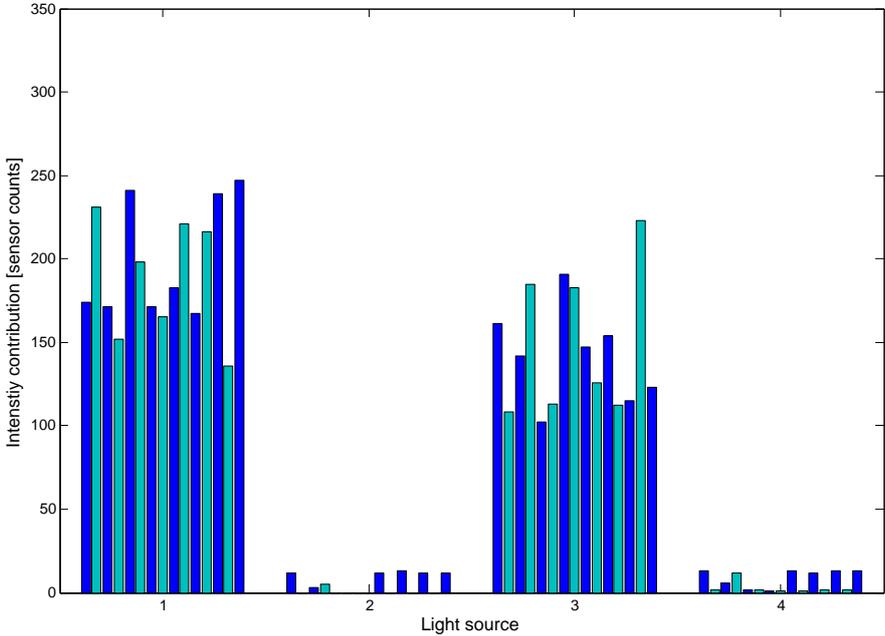


Figure 3.30: Model evaluation for power consumption



**Figure 3.31:** Intensity contribution measured using single point calibration routine, the measurement is repeated 15 times at the same point of interest



**Figure 3.32:** Intensity contribution measured using area average calibration routine, the measurement is repeated 15 times at the same area of interest

points among the area of interest are measured. The results are shown in table 3.7 and 3.8. Both methods achieved similar uniformities.

Control Point	very left	left	central	right	very right
Area Average Calibration	700	490	400	698	517
Multipoint Calibration	690	480	400	700	526

**Table 3.7:** Comparison of illuminance uniformity between area average calibration routine and multipoint calibration at five different control point inside a 80 cm long area of interest on the writing tabel

	$d_1$	$d_2$	$d_3$	$d_4$
Area Average Calibration	0.1956	0	0.1978	0.0011
Multipoint Calibration	0.0267	0.0033	0.4067	0.01
	0.3311	0.01222	0.0233	0.0078

**Table 3.8:** Contribution factors measured using the area average calibration routine and multipoint calibration routine for the same area of interest

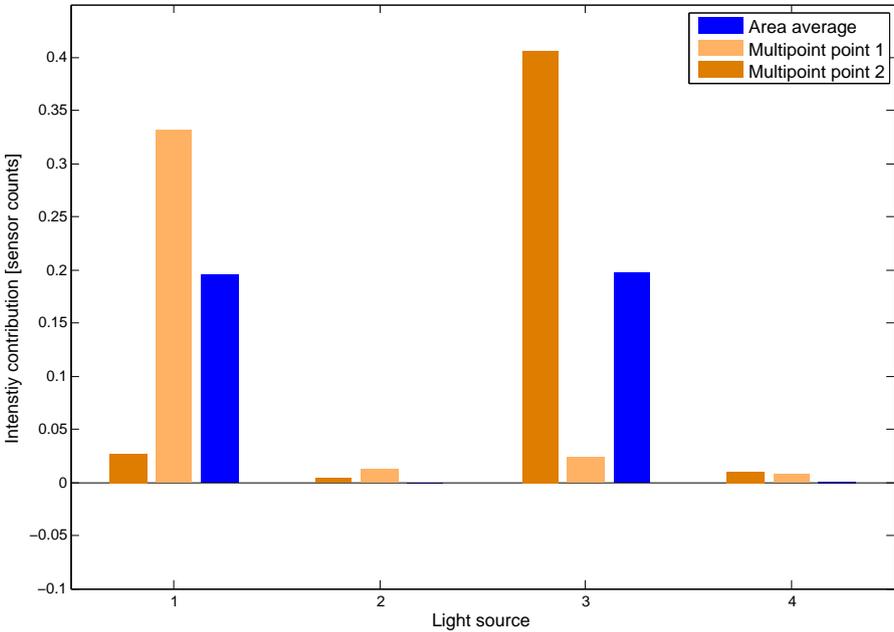
The last experiment shows the repeatability of the webcam based calibration routine. 15 calibrations are executed at a constant area of interest similar to the area average calibration experiment. The average contribution factors and standard deviations are in table 3.9. Since the area is selected with freehand drawing, inconsistencies result from variance in the position and shape of the area of interest.

	$d_1$	$d_2$	$d_3$	$d_4$
average	0.6223	0.0895	0.6725	0.1583
standard deviation	0.0122	0.0058	0.0174	0.0064

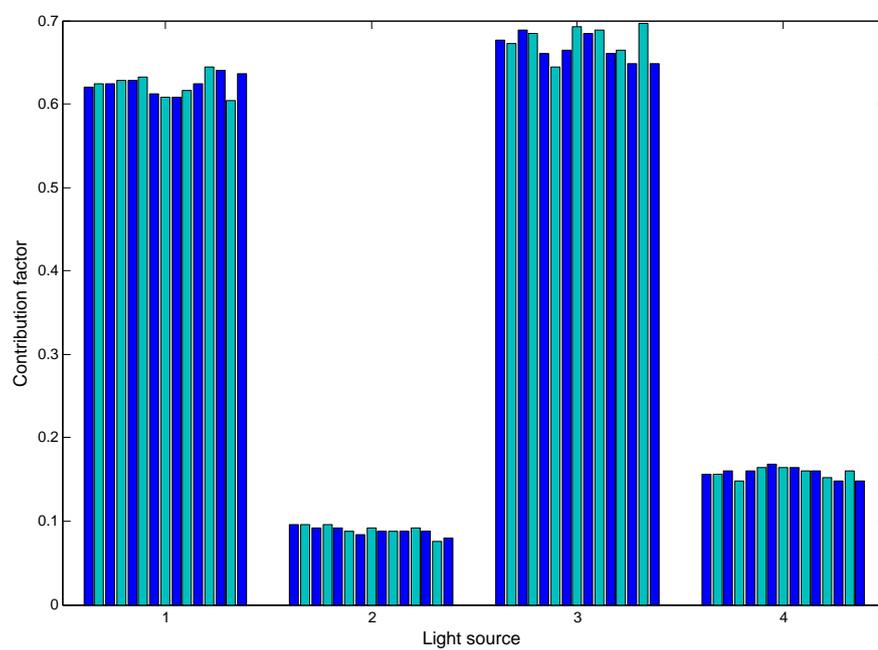
**Table 3.9:** Average contribution factors and standard deviation of 15 calibration measurments at a constant area of interest using webcam based calibration

### 3.4.3 Controller Evaluation

The step response of the sensor based single point closed-loop control shows that the reference value is achieved within the settling time  $T_s = 8.44$  s. The error tolerance is 0.4%. The illuminance at the area of interest slowly converges towards the reference value without overshoot or oscillation. The asymptotic step response is important, because fast illumination changes are unpleasant, whereas transient changes create a comfortable feeling. Very short illuminance changes, for example caused by passing shadows, do not affect the system. The reference color temperature is  $CCT = 6000$  K. The average update rate of the controller is 20 Hz without and 10 Hz with spectrum based CCT and CRI calculation. If the energy plot and surface plot are enabled, the update rate of the controller decreases further.



**Figure 3.33:** Contribution factors measured using the area average calibration routine and multipoint calibration routine for the same area of interest



**Figure 3.34:** Contribution factors measured using the webcam based calibration routine; the measurement is repeated 15 times at the same point of interest

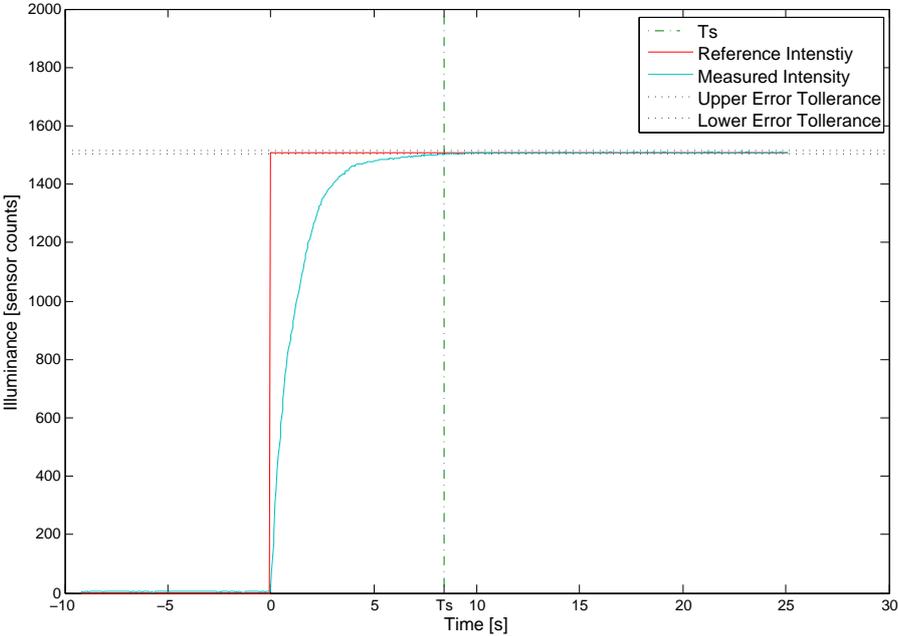


Figure 3.35: Step response of sensor board based single point closed-loop control

The step response of the multipoint sensor board based control is measured for a linear area of interest, about 80 cm long and located in between and underneath light fixture 1 and light fixture 3 on the working desk. After the calibration, the sensor board is placed in the middle of the area of interest and the reference intensity is set to  $900 \frac{\text{lux}}{\text{scale}}$ . The reference color temperature is  $CCT = 4000 \text{ K}$ . In this experiment, the settling time is  $T_s = 13.5 \text{ s}$ .

The step response of the sensor board based controller with area average calibration and optimization routines is measured in the same way for the same area of interest. The settling time is  $T_s = 11.7 \text{ s}$ .

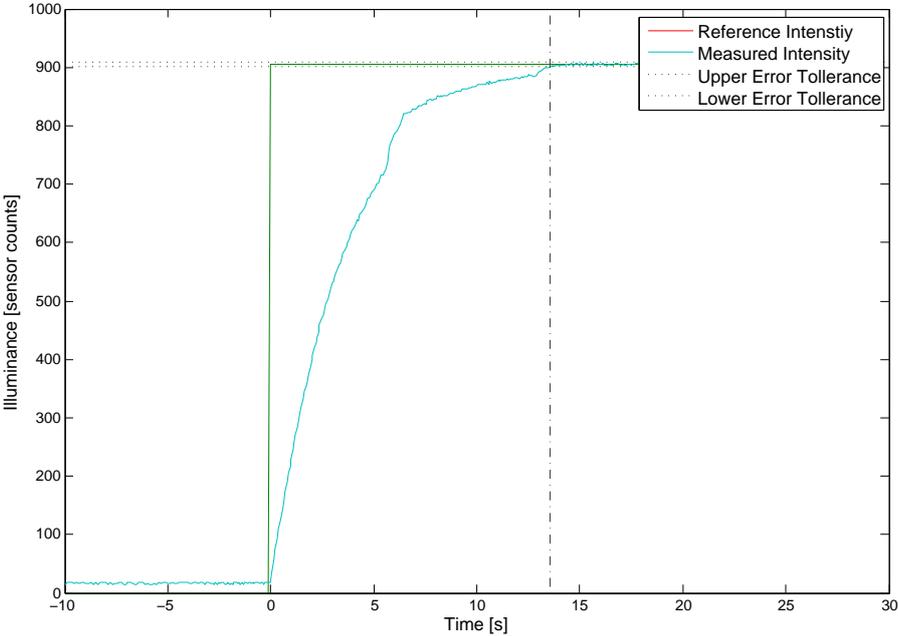
The step response of the webcam based controller is also measured for the same area of interest. The median of the pixel values of the entire area of interest is fed back. The settling time,  $T_s = 32.3 \text{ s}$ , is greater compared to the sensor board based controller, as explained in 3.2.2. The update rate without enabling the energy monitoring and illuminance surface plots is approximately 30 Hz.

#### 3.4.4 Short-Term User Study

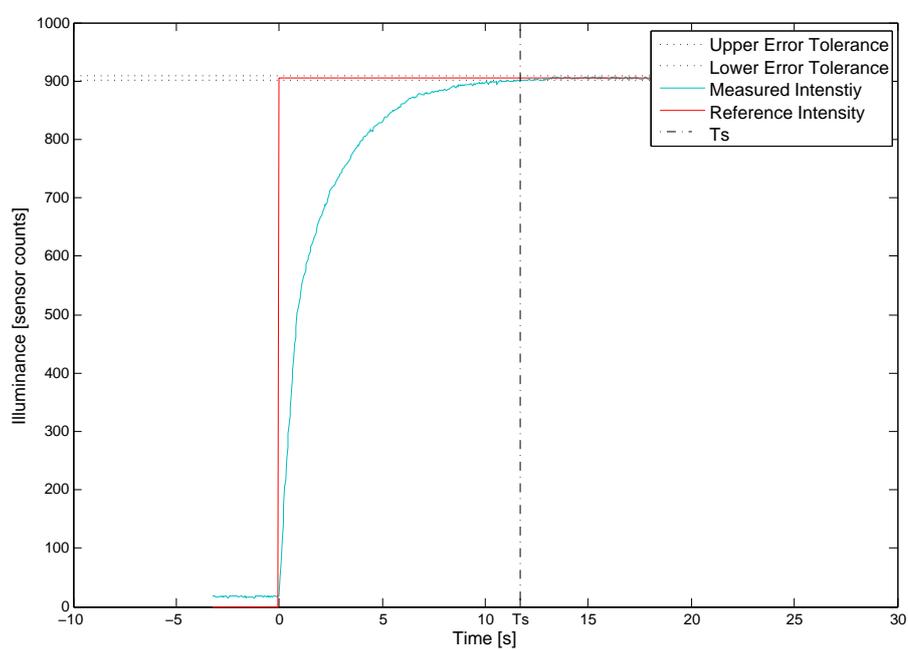
I set up three short-term user studies to evaluate the achievable energy saving compared to a static system and to investigate interaction between user and the intelligent lighting system. These studies took place during 10 working hours from noon to nightfall.

In the first and second study, the sensor board was the selected feedback device. The controller performed single point calibration and optimization. Spectrum based CCT and CRI calculations were enabled and quality and power saving plots were disabled. The selected software mode was *2 data logging*. Every 60 seconds, time, date, intensity reference, CCT reference, intensity feedback, light source intensity settings, contribution factors, stability, calculated CRI, calculated CCT, calculated power consumption and power savings were logged to the hard disk. A dark measurement was taken at regular intervals to capture the ambient lighting condition, requiring all the light fixtures to be turned off. In order to keep the disturbance for the user at an acceptable level, this measurement was only performed every 10 minutes.

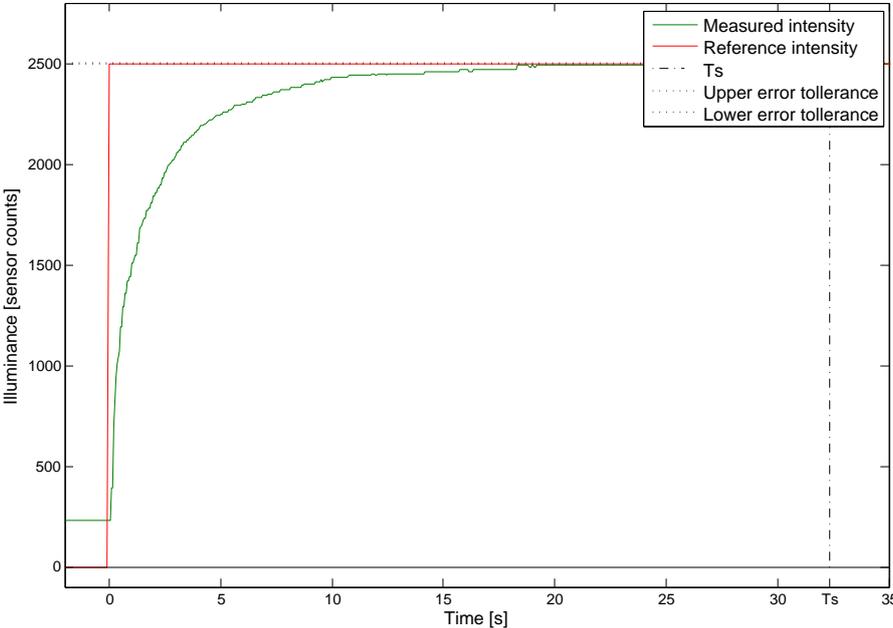
The first short-term study was performed on me, subject #1. I selected reference intensity and CCT according to my preference. I calibrated the sensor at the beginning of the experiment and didn't move it during the entire experiment, since I was satisfied with my initial choice. The sensor board was placed on the work table. Figure 3.39 shows that during early afternoon from about 14:30 to 16:30, ambient light intensity was above the reference intensity. During these hours 100% power saving compared to static maximum uniform lighting was achieved as the electric lights didn't need to augment local lighting to meet the reference illumination. From 16:30 to 19:00, passing clouds and sunset led to many illuminance changes. The controller successfully compensated intensity deficits. After



**Figure 3.36:** Step response of sensor board based multipoint closed-loop control



**Figure 3.37:** Step response of sensor board based closed-loop control using area average calibration and optimization routines



**Figure 3.38:** Step response of webcam based closed-loop control

19:30, since the ambient lighting was not changing rapidly anymore, the error between reference intensity and measured intensity was constantly zero. During the experiment 100% power saving was the highest and 67% the least achievement. Power saving was calculated by comparing the power consumption with a dimmable system as described in equation 3.52. An average of 90% power saving during the day is very significant. Power saving in comparison with maximum constant uniform lighting (generally the case in commercial buildings) is 95%.

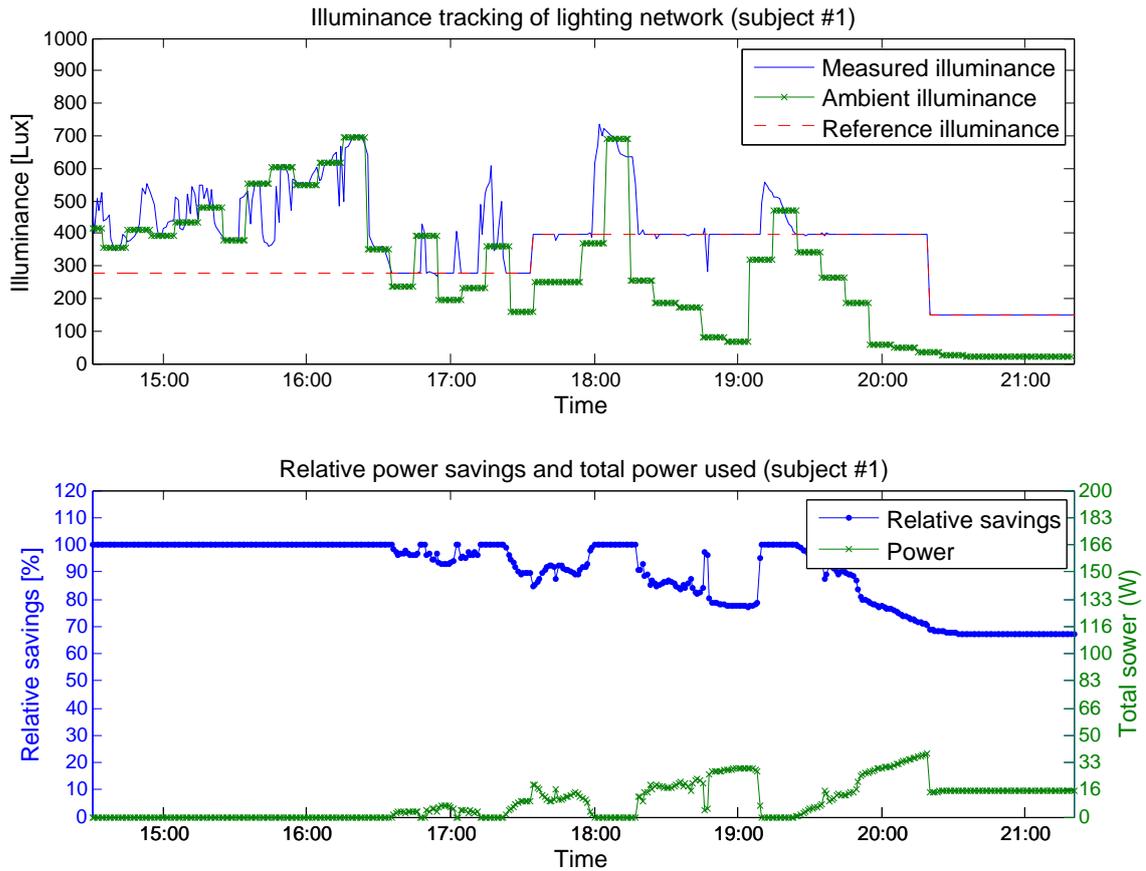
The second experiment was performed on a 24-year-old student, subject #2. After a short introduction to the functionality of the system, the test-user obtained freedom to use the system according to his desire. A short debrief after the study documented the user's experience and opinion about the intelligent lighting system.

At the beginning of the experiment, the reference intensity signal shows that the user was testing the behavior of the system and changed the reference values many times. The behavior repeats at around 12:45, 13:45, 17:30 and 18:30. The black dashed vertical lines in figure 3.40 represent executions of the calibration routine. The red dashed vertical lines represent changes in the sensor position. In some cases, the user moved the sensor and recalibrated the system, normal behavior when the area of interest changes. But in other cases, the user moved the sensor but did not recalibrate the system or moved the sensor away from the area of interest right after the calibration. The first case occurred, at around 12:30, causing a step in the feedback value. The second case took place at around 18:50. This kind of behavior occurs because placing the sensor into the area of interest is disturbing. The test-user needed the space in the area of interest and thus he had to remove the sensor. This problem could be solved using a much smaller design of the sensor board (see chapter 5).

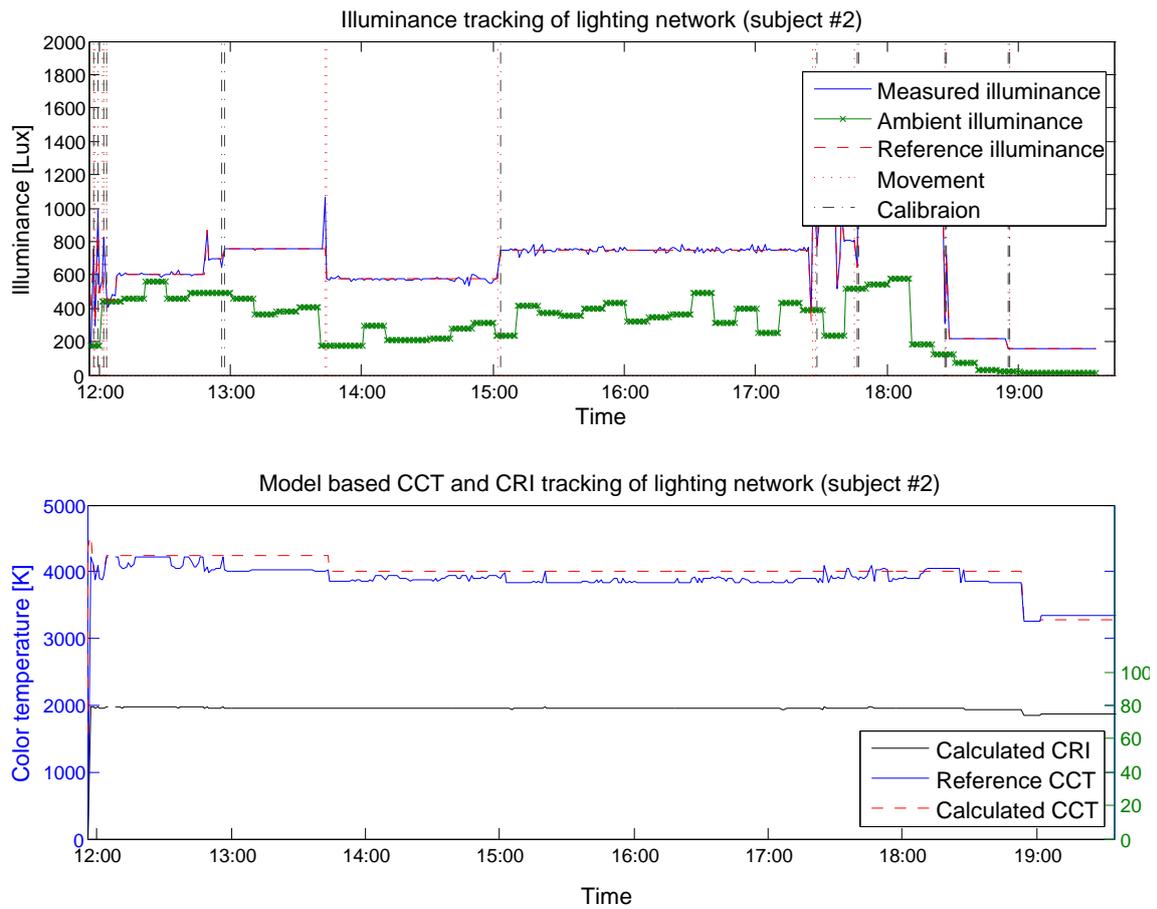
Comparing the measured ambient light intensity and the user preference, you can see that the reference intensity follows the ambient light level. It seems that the test user wanted to keep the ratio between ambient light and artificial light constant. In the discussion at the end of the experiment the test subject explained that ambient light fell mostly from the left side into the office area. Creating artificial light accordingly from the right side increased the lighting quality, because the artificial lighting not only made the illumination more consistent in the area of interest, but also more consistent among different incidence angles. Consistency of the illumination angle and stable ratio between ambient light and artificial lighting are two important aspects for a new design of the sensor board.

Figure 3.40 shows reference CCT compared to calculated CCT using the spectrum based model. Spectrum based CRI prediction is also shown in Figure 3.40. The achieved average power savings compared to maximum constant lighting is 68%. Comparison of power consumption with the dimmed lighting condition is misleading and not meaningful in this case because the sensor board was not placed in the area of interest during the entire study. Thus, reference intensity is not equal to the actually desired intensity level.

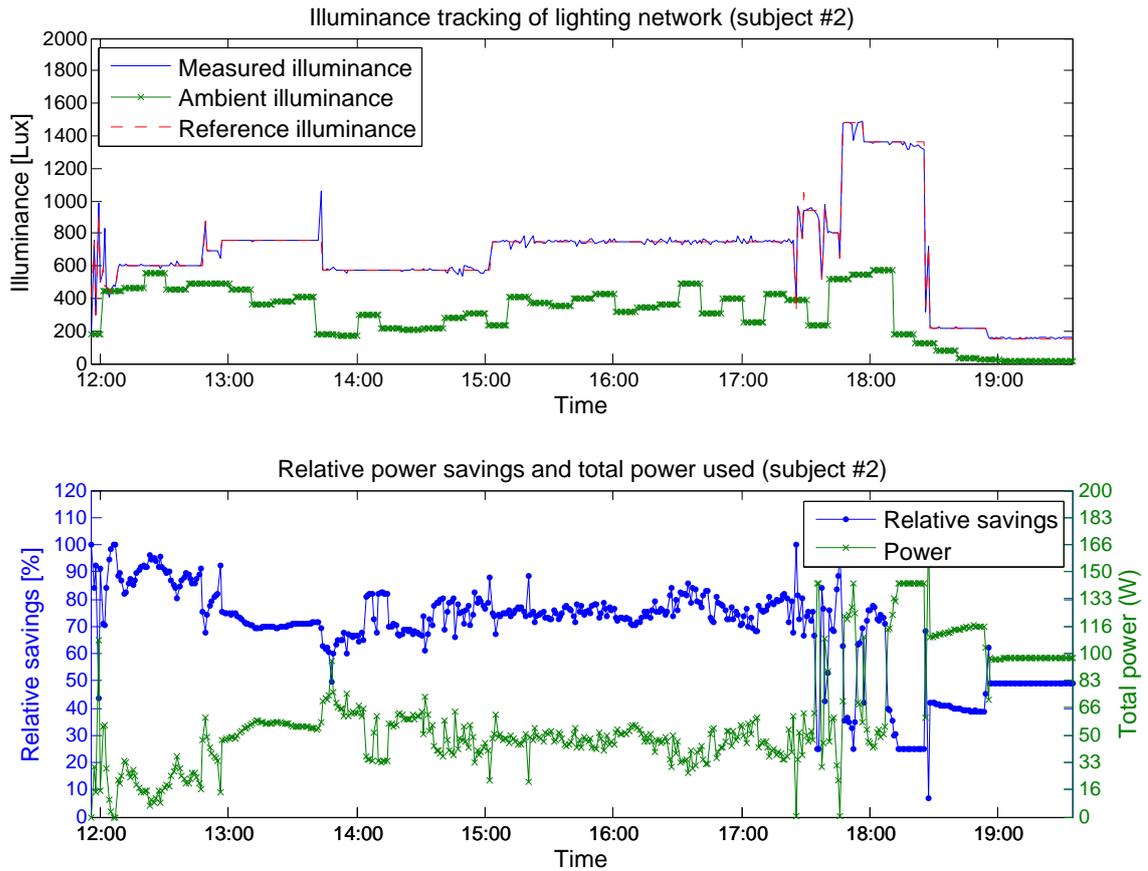
The third study was performed with luminance feedback. Thus, the webcam was the se-



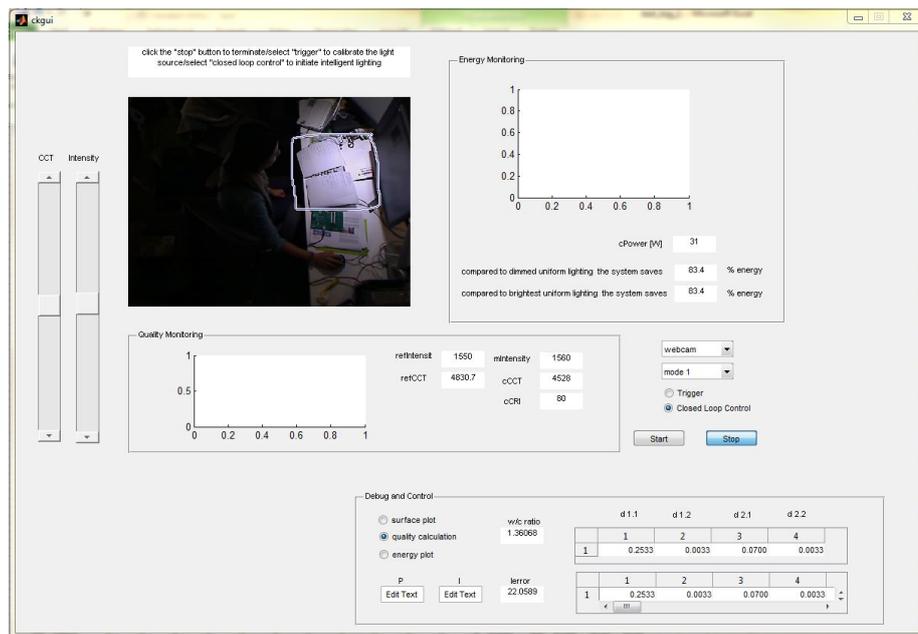
**Figure 3.39:** Short term user study with subject #1 and sensor board based single point calibration and optimization, (top) reference, measured and ambient illuminance at the point of interest, (bottom) energy consumption and energy saving in comparison to a dimmed static uniform lighting system



**Figure 3.40:** Short term user study with subject #2 and sensor board based single point calibration and optimization, (top) reference, measured and ambient illuminance at the point of interest, time of calibrations and sensor movements (bottom) reference CCT and model based CCT and CRI

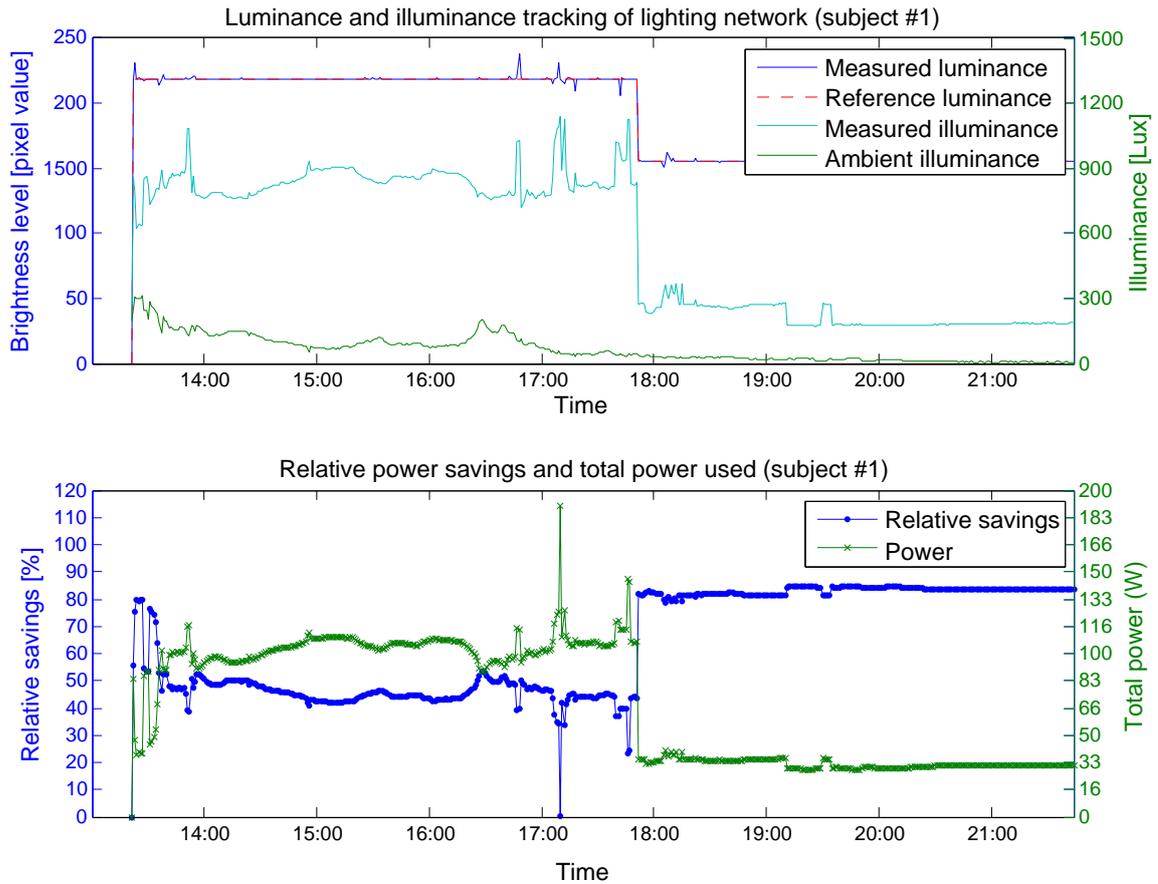


**Figure 3.41:** Short term user study with subject #2 and sensor board based single point calibration and optimization, (top) reference, measured and ambient illuminance at the point of interest, (bottom) energy consumption and energy saving in comparison to a static uniform lighting system with constant maximum power consumption of 200 W



**Figure 3.42:** Short term user study with subject #1 and webcam based area average calibration and single point optimization GUI snapshot

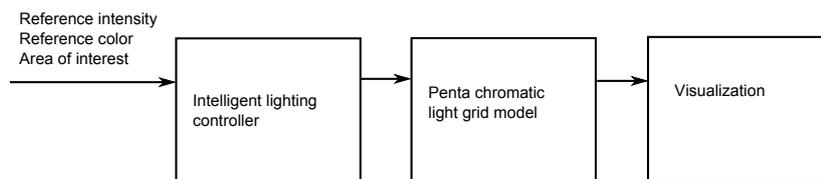
lected sensor, and area average calibration and single point optimization routines were used. Spectrum based CCT and CRI calculations were enabled and quality and power saving plots were disabled. The software was in mode 2 *data logging*. Every 60 seconds, time, date, intensity reference, CCT reference, illuminance and luminance feedback, light source intensity settings, contribution factors, stability, calculated CRI, calculated CCT, calculated power consumption and power savings were logged to the hard disk. The dark measurement was performed every 60 seconds using the sensor board. Figure 3.42 shows the selected area of interest and the position of the sensor board. The sensor board was outside of the area of interest here, and did not affect the luminance measurement. The lux sensor was placed very close to the area of interest in order to get a good approximation of the illuminance inside the area of interest. The position of the camera was constrained during the experiment. It was attached to the top of the wooden frame in order to capture the entire cubicle area. The distance between area of interest and the webcam was approximately 1.5 m. Exposure was set to  $-5$ . The area of interest was selected at the beginning of the experiment and remained constant. Figure 3.43 shows that the ambient lighting was low but fairly consistent. Despite nearly constant ambient lighting, there are peaks in the luminance measurement. These peaks were caused by changes in objects within the area of interest. The measured illuminance approximation shows that remaining constant luminance leads to illuminance changes. The achieved average energy saving in comparison to maximum constant lighting was 63%.



**Figure 3.43:** Short term user study with subject #1 and webcam based area average calibration and single point optimization, (top) reference and measured luminance, measured and ambient illuminance at the area of interest, (bottom) energy consumption and energy saving in comparison to a static uniform lighting system with constant maximum power consumption of 200 W

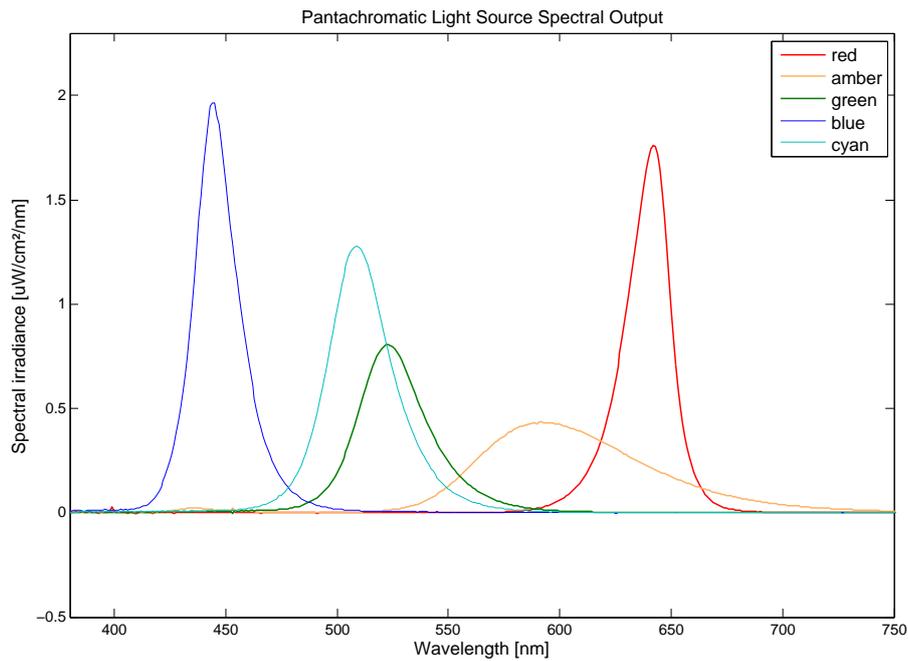
## 4 Pentachromatic Solid State Lighting System

Based on existing hardware I created a simulink model of a virtual pentachromatic solid state lighting system. The simulink model, which is to distinguish from the mathematical light grid model described in chapter 4.1, consists of 4 major components. These are the input and feedback device, the intelligent lighting controller, the pentachromatic light grid model and the visualization block. The virtual pentachromatic light sources are predicated on the LED ring described in chapter 2, which includes blue, green, cyan, amber and red colored LEDs, a temperature sensor and digital color sensor. Detailed information about the design of the LED light source can be found in [1]. The virtual light grid comprises  $n$  pentachromatic LED lighting devices arbitrarily arranged in a  $m$ -by- $l$  grid. Sunlight and fluorescent lighting are present, but they cannot be controlled through the system. Using the Simulink model I am able to develop and test the open-loop controller, which can be applied to a physical system in the future. In the following chapters I introduce a mathematical model of the pentachromatic light grid, based on which the light grid model in Simulink is developed, the open-loop controller and evaluation of the light grid model and the stochastic optimization, which is a part of the open-loop control.



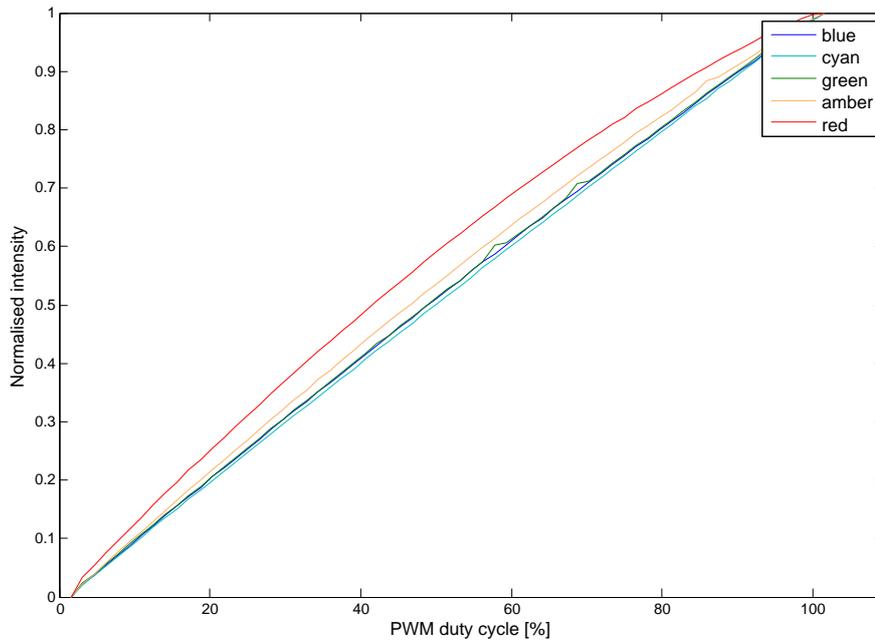
**Figure 4.1:** Pentachromatic solid state lighting system

## 4.1 Light Grid Model



**Figure 4.2:** Visible spectrum of each channel of the pentachromatic light source with 100% PWM duty cycle

In this chapter I introduce a mathematical model of a pentachromatic solid state light grid similar to the spectrum based model described in chapter 3.1. Analogous to the assumptions made for the dichromatic system model, the intensity of ambient light, such as sunlight or fluorescent light, is known. Ambient light is uniform within the area of interest. The illuminated room is empty. The solid state light sources are point sources. The grid consists of  $n$  lights in an  $m$ -by- $l$  grid. The arrangement of the lighting devices is arbitrary, but the position of each light source is known.



**Figure 4.3:** Intensity and PWM duty cycle relationship of the pentachromatic light source

The spectral irradiance and illuminance of each color channel for different PWM settings in a dark room are measured with the spectrometer. The spectrometer head is placed in the middle of the light cone with a distance of 30 cm to the light source. The results in figure 4.3 show that the intensity of amber and red LED do not change linearly according to the PWM input. The reason is temperature drift for long duty cycles. In this lighting model I assume that a proper control method is used to operate the solid state lighting devices. For example local closed-loop proportional control can be used to resolve temperature and depreciation hazards. Figure 3.3 shows the standard spectral irradiances  $E_{\lambda,j,0}(\lambda)$ .  $d_0 = 0.3$  m is the normal distance. Illuminance, CCT and CRI are calculated with colorimetric functions and formula 3.1, as described in chapter 3.1.

Since the intensity of each channel and each lighting device is known, prediction of power consumption is possible. The power consumption of each LED is proportional to the duty cycle of the PWM control. The total power consumption is the sum of the power consumption of each channel. The linear fit of the measured power consumption is

$$E = \begin{cases} 2.0, & \text{for } channel_j = 0, j \in [1, 2, 3, 4, 5] \\ d_{blue} * channel_1 + d_{green} * channel_2 \dots \\ + d_{cyan} * channel_3 + d_{red} * channel_4 \dots & \text{otherwise} \\ + d_{amber} * channel_5 + offset, & \end{cases} \quad (4.1)$$

with

$$d_{blue} = 3.7741, d_{green} = 3.7453, d_{amber} = 3.5808, d_{red} = 4.2483, d_{cyan} = 3.3687, \\ offset = 2.4062.$$

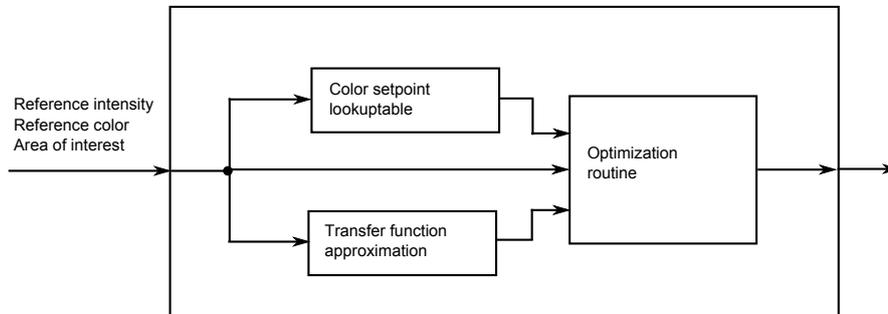
## 4.2 Open-Loop Control

The tasks of the open-loop control are to receive and interpret the user's requirements, set the PWM duty cycles for each channel of the solid state lighting devices accordingly and minimize the energy use at the same time. In order to simplify the control problem, the open loop control is broken down into 3 separate control and optimization problems.

1. Create (offline) and use (online) lookup table to match reference CCT with channel intensity ratios (see chapter 4.2.1)
2. Measure transfer function and receive user input (online) (see chapter 4.2.2)
3. Use linear programming to solve energy optimization problem (online) (see chapter 4.2.3)

### 4.2.1 Behavior of Pentachromatic Solid State Light Source

An infinite number of white points can be created using additive mixing and 5 different colors. Different ratios of blue, cyan, green, amber and red colors create different intensities, CCTs and CRIs. It is not obvious how CRI, CCT, intensity and efficacy are related to each other. This problem is investigated in experiment #4 in chapter 4.3.



**Figure 4.4:** Pentachromatic intelligent lighting controller block diagram

Illuminance and CCT are input parameters. Therefore the system must be able to create white points within a range of intensities and CCTs. A reasonable range for CCT would be 2000 K to 10000 K. The range of intensity is limited by the color gamut. For CCTs from 2000 K to 10000 K, I find white points with the highest intensity, CRI and efficacy. By maximizing efficacy and intensity, I find the white points which consume the least energy and allow the broadest range of intensity. By optimizing CRI, I find white points which have similar color rendering characteristics to sunlight so as to provide high lighting quality.

Since CRI and CCT cannot be described by closed-form expressions, it is difficult to formulate the optimization problem as analytical mathematical programming. A suitable method to solve this problem would be a stochastic optimization technique. Research at the Responsive Environments Group already investigated the brute force method [1]. I proposed using particle swarm optimization technique to improve the performance. Particle swarm optimization is a stochastic optimization technique motivated by social behaviour of bird flocking or fish schooling. A population of particles is generated initially. The particles are arranged randomly in the space of solutions. Each particle can calculate its fitness with the fitness function and it also knows the fitness of the other particles. By following simple rules, the population converges toward the global optimum without any central control. There are three components that influence the velocity of each particle. The first component is the velocity of the last iteration weighted with the inertia  $w$ . The second component is the personal best solution  $P_{best}$ .  $P_{best}$  is updated every iteration. The distance from the last position to the personal best solution is multiplied by the gain  $c_1$

and a random factor  $rand_1$ . The third component is the global best solution  $G_{best}$ .  $G_{best}$  is also updated in each iteration by comparing the personal best solutions. The distance from the last position to  $G_{best}$  is multiplied by the gain  $c_2$  and random factor  $rand_2$ . The velocity  $v_j$  of the  $j^{th}$  particle is accordingly

$$v_j(k) = w*v_j(k-1) + c_1*rand_1*(P_{best,j} - x_j(k-1)) + c_2*rand_2*(G_{best}(k) - x_j(k-1)) \quad (4.2)$$

$k$  is the time index and  $x_j$  is the position of the  $j^{th}$  particle. The random factors  $rand_1$  and  $rand_2$  prevents convergence to a local optimum.  $c_1$  and  $c_2$  defines the weight of the local best and global best influence. The inertia  $w$  controls the speed of the particles. To improve the convergence behaviour, a fourth parameter can be introduced, which is the neighbourhood best solution. But this is not further considered. Other parameters, which can be modified to adjust the behaviour of the swarm and the accuracy of the optimization, are the number of iterations and the size of population. In this case the fitness function is

$$fitness(channel) = \begin{cases} k_{cct} * \left( \frac{CCT(channel) - CCT_{ref}}{CCT_{ref}} \right)^2 & , \text{for } CCT_{ref} - \Delta CCT \geq CCT(channel) \\ & \text{or } CCT_{ref} + \Delta CCT \leq CCT(channel) \\ k_{dvw} * \left( \frac{dvw(channel)}{5.4e-3} \right)^2 & , \text{for } dvw(channel) > 5.4e-3 \\ k_{cri} * \frac{Ra(channel)}{100} + k_{int} * \frac{int(channel)}{int_{norm}} \dots & , \text{otherwise} \\ + k_{efficacy} * \frac{int(channel)}{E(channel) - E_0} * \frac{E_{norm} - E_0}{int_{norm}} & \end{cases} \quad (4.3)$$

$$channel = \begin{bmatrix} channel_{1,i} \\ channel_{2,i} \\ channel_{3,i} \\ channel_{4,i} \\ channel_{5,i} \end{bmatrix}$$

In order to make the efficacy comparable for different intensities, it is necessary to eliminate  $E_0$ , which is the standby power consumption. The efficacy is defined as  $int/(E - E_0)$  in chapter 4.  $k_{cct}$ ,  $k_{cri}$ ,  $k_{dvw}$ ,  $k_{efficacy}$  and  $k_{int}$  are weight factors for CCT, CRI, distance to the black body curve in the CIE domain, efficacy and intensity respectively.  $channel$  describes the PWM duty cycles for each channel of light fixture  $i$ .  $int_{norm}$  and  $E_{norm}$  are standard intensity and energy consumption. Using this optimization method, I am able to create one or more lookup tables with intensity ratios sorted by CCT. Using this lookup table, reference CCTs are transformed into channel intensity ratios  $ratio_j = int_j / \sum_{j=1}^5 int_j$ , which are used to control the light sources. I propose using a lookup table because the performance of the PSO does not allow real-time optimization. Using online particle swarm optimization, more satisfying white points could be found, but the controller will be more complex and need higher calculation capacity. The benefit may or may not be significant.

## 4.2.2 Identify Transfer Function

Since the pentachromatic system is a virtual system at this point, the system model is used to approximate the transfer function. According to the distance between the point of interest and the light sources, contribution factors for each light source is

$$d_i = \frac{d_0^2}{|x - x_i|^2} * int_{max,0} \quad (4.4)$$

It follows that

$$int(dm.x) = \sum_{i=0}^n d_i * \sum_{j=1}^5 k_j * channel_{j,i} \quad (4.5)$$

$channel_{j,i}$  is the normalized intensity of the  $j^{th}$  channel and  $i^{th}$  light source, which is also the PWM duty cycle assuming a linear relationship between PWM input and intensity output.  $k_j$  is the ratio between the maximum total intensity and the maximum intensity of channel  $j$  according to fomula 3.19. In the physical system, this calculation will be replaced by the online calibration routine as described in 3.2.1 or 3.2.2. A challenge using pentachromatic system is signal aliasing, which is likely since 5 channels of each fixture need to be synchronized. Therefore an asynchronous calibration method, as described in chapter 5, might be more applicable.

## 4.2.3 Optimization Routine

In a last step based on the reference values, the intensity ratios and the transfer function optimal intensity settings for each light source are calculated. Since the energy optimization problem is a convex linear problem, I am able to use linear programming to express and solve the problem. Energy consumption is the objective, and the constraints ensure intensity and color temperature observation. The optimization routine receives the intensity reference value, the color ratios from the lookup table, match to reference CCT and the contribution factors  $d_i$ . The cost vector is

$$c = [c_1 \quad c_2 \quad \cdots \quad c_n] \quad (4.6)$$

with

$$c_i = [d_{blue} \quad d_{cyan} \quad d_{green} \quad d_{amber} \quad d_{red}] \quad (4.7)$$

and

$$\begin{aligned}
 d_{blue} &= 3.7741, \\
 d_{green} &= 3.7453, \\
 d_{amber} &= 3.5808, \\
 d_{red} &= 4.2483, \\
 d_{cyan} &= 3.3687, \\
 c_1 &= c_2 = \dots = c_n = c_i.
 \end{aligned}$$

The constrains are

$$channel_{j,i} \geq 1 \tag{4.8}$$

$$\sum_{i=1}^n d_i * channel_{j,i} * k_j = ratio_j * int_{ref} \tag{4.9}$$

$$\sum_{i=1}^n d_i * \sum_{j=1}^5 channel_{j,i} * k_j = int_{ref} \tag{4.10}$$

The formal statement is

$$min\{c * x\} A * x \geq b \tag{4.11}$$

with

$$c = [c_1 \quad c_2 \quad \dots \quad c_n], \tag{4.12}$$

$$A = \begin{bmatrix}
 I_{n*5 \times n*5} & & & & \\
 d & 0 & 0 & 0 & 0 \\
 0 & d & 0 & 0 & 0 \\
 0 & 0 & d & 0 & 0 \\
 0 & 0 & 0 & d & 0 \\
 0 & 0 & 0 & 0 & d \\
 d & d & d & d & d
 \end{bmatrix}, \tag{4.13}$$

and

$$b = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ ratio_1 * int_{ref} \\ ratio_2 * int_{ref} \\ ratio_3 * int_{ref} \\ ratio_4 * int_{ref} \\ ratio_5 * int_{ref} \\ int_{ref} \end{bmatrix}. \quad (4.14)$$

The Simulink optimization block is programmed based on the revised simplex method and the MATLAB function *rsm()*.

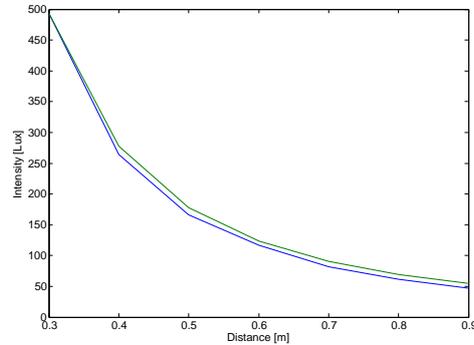
### 4.3 Results

In this chapter, I will only present the evaluation of the system model and the results of the PSO optimization. Calculations in experiment #1 through #4 were executed in MATLAB using a modified version of the PSO toolbox [2] and the colorimetry toolbox developed by [1].

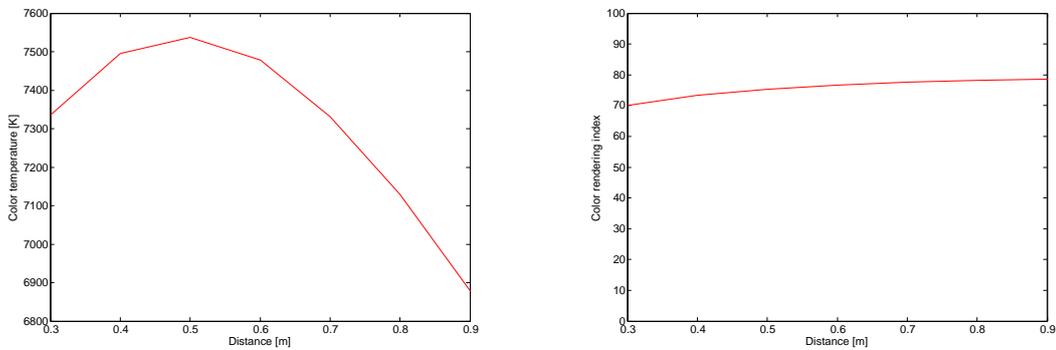
Experiment #1 was performed to evaluate the distance dependency of irradiance, color temperature and color rendering index. The experiment took place in a dark room. Irradiance, CCT and CRI were measured using the spectrometer for seven different distances and one LED ring. The spectrometer head was placed in the middle of the light cone. Figure 4.5 shows intensity and distance relationship. The prediction was made using the first measurement, with 0.3m distance as the reference intensity. Error increases with the distance difference. The wrong assessment results from the assumption that the LED source is a point source, which is not true for the LED ring. The highest error is about 16%. CRI and CCT do not remain constant as assumed, because the five different spectra are not perfectly defused.

Experiment #2 evaluates the accuracy of the colorimetry functions and power calculation. Intensity, CCT, CRI and power consumption were measured for 34 different white points. The experiments were taken in a dark room with one LED ring and the spectrometer, which was placed in the middle of the light cone with a distance of 30 cm. Figure 4.7 to figure 4.8 show the comparison of measured and predicted values.

Experiment #3 shows the outcome of the monte carlo simulation in comparison to the results of the particle swarm optimization. 35 white points with the highest CRI and matching color temperature are selected. The configuration of the particle swarm optimization is shown in table 4.1. The initial particle positions are white points generated using a linear

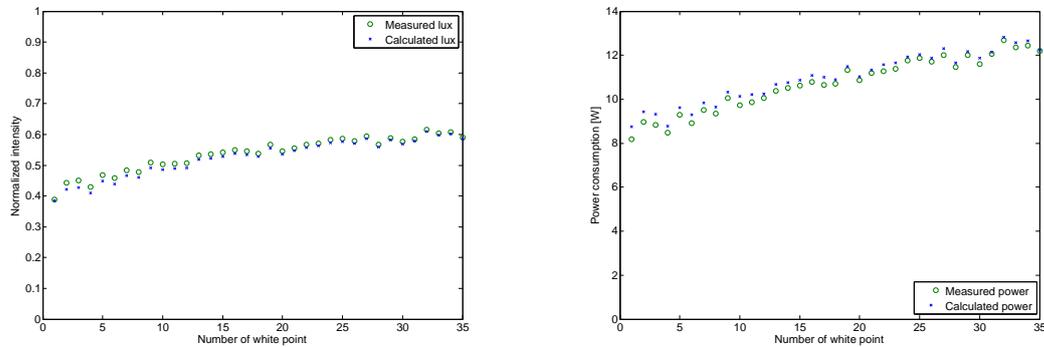


**Figure 4.5:** Experiment #1: evaluation of predicted distance and illuminance relationship

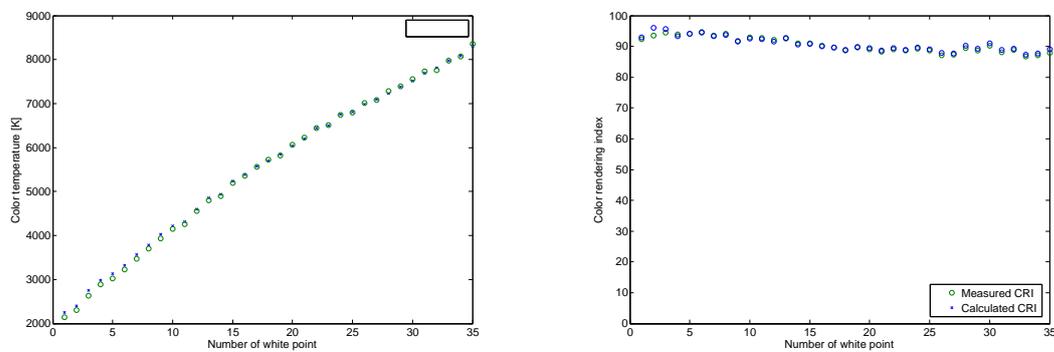


**Figure 4.6:** Experiment #1: measured distance and CCT relationship (left) and measured distance and CRI relationship (right)

transformation [1] for the range of desired color temperature. Besides significant performance improvement, the white point found using PSO offer even higher CRI, intensity and efficacy. Particle swarm optimization can be concluded within minutes whereas the monte carlo simulation lasts for several hours.

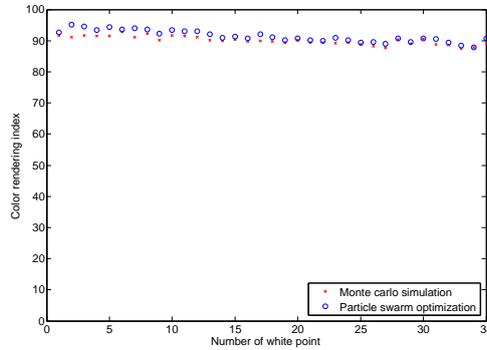


**Figure 4.7:** Experiment #2: comparison measured and predicted normalized intensity (left) and power consumption (right) based on the pentachromatic light grid model

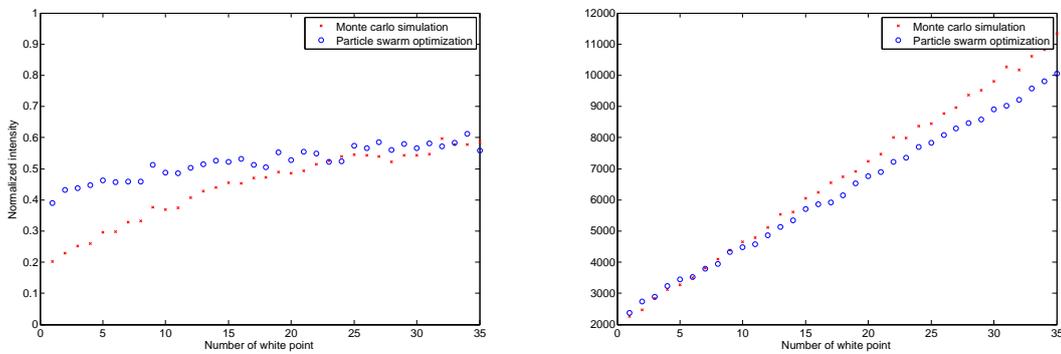


**Figure 4.8:** Experiment #2: comparison measured and predicted color temperature (left) and color rendering index (right) based on the pentachromatic light grid model

Experiment #4 shows results of particle swarm optimization for different objectives. The objectives are efficacy, intensity and CRI. The weight factors for the objective function are  $k_{efficacy} = 1$ ,  $k_{int} = 0$ ,  $k_{cri} = 0$  for efficacy optimization,  $k_{efficacy} = 0$ ,  $k_{int} = 1$ ,  $k_{cri} = 0$  for intensity optimization and  $k_{efficacy} = 0$ ,  $k_{int} = 0$ ,  $k_{cri} = 1$  for CRI optimization. The initial



**Figure 4.9:** Experiment #3: comparison of CRI for 35 CRI optimized white points using monte carlo simulation and particle swarm optimization



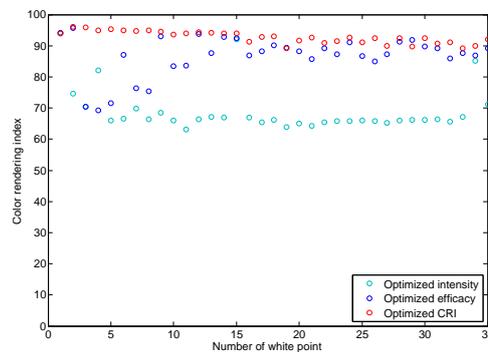
**Figure 4.10:** Experiment #3: comparison of normalized intensity (left) and color temperature (right) for 35 CRI optimized white points using monte carlo simulation and particle swarm optimization

particle positions are identical to experiment #3. The PSO configuration parameters are presented in table 4.1. This experiment is very important for the generation of the color temperature and channel intensity ratio lookup table as described in chapter 4.2.1. As you

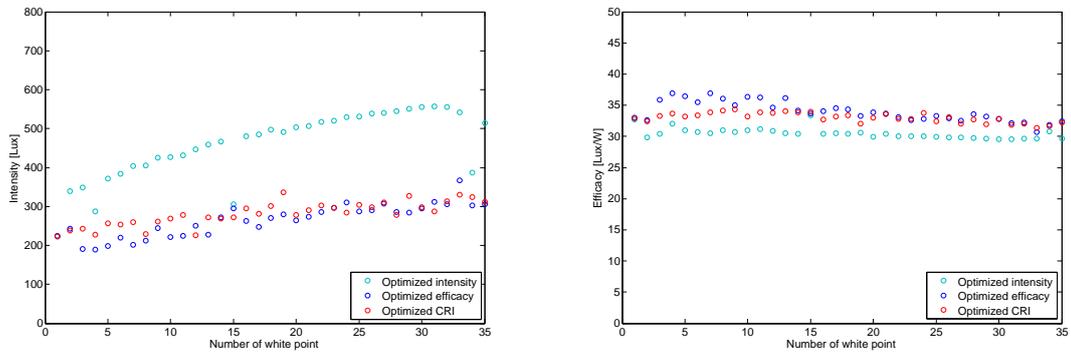
Max partical velocity	0,1
Local best influence	15
Global best influence	7
Number of particles	5
Maximum number of iterations	2000
$\Delta CCT$	120 K

**Table 4.1:** Particle swarm optimization configuration parameters

can see in figure 4.11 and figure 4.12 optimizing intensity range leads to tradeoffs in CRI and efficacy and vice versa. If efficacy is the objective, CRI only differs from the optimal CRI in the low color temperature range. The following conclusion can be drawn from this experiment. If the required intensity does not exceed the achievable intensity range for the efficacy optimized white points, the efficacy optimized white points for CCT above 4000 K should be used. If the required intensity does not exceed the achievable intensity range for CRI optimized white points, the CRI optimized white points for CCT below 4000 K should be used. In any other cases, intensity optimized white points should be used.



**Figure 4.11:** Experiment #4: comparison of color rendering index of white points generated using particle swarm optimization and different objectives such as intensity, efficacy and CRI



**Figure 4.12:** Experiment #4: comparison of intensity (left) and efficacy (right) of white points generated using partial swarm optimization and different objectives such as intensity, efficacy and CRI

## 5 Conclusion and Future Work

In the previous chapters, I presented design and control techniques for an intelligent lighting system using commercial white LED light sources, and a model of a virtual pentachromatic intelligent lighting system based on preexisting hardware.

The intelligent lighting system proposed in chapter 3 is still at an early stage of development. A testbed with 4 light fixtures was built, and illuminance and luminance feedback were integrated and evaluated. The results of experiments and studies show that light based localization, LP optimization and PI closed-loop control are technically feasible and applicable, as proposed. The prototype of the controller and the operating software were programmed in MATLAB for the most part, limiting their performance. Because of this limitation, the calibration process is visible and cannot be continuously executed; further work in this area is needed to improve the performance of the system.

The pentachromatic system is currently in a conceptual state. The control techniques are developed, and should be implemented and tested on a real system.

Future work for the sensor board control system includes embedded calibration, redesign of the sensor board and closed-loop color temperature control. Significant improvement is achievable if the calibration routine is incorporated into the embedded software, eliminating the bottlenecks caused by limited speed of communication. It is possible to use light source driver signals, as described in figure 3.16, but with variable duty cycles. The sensor board executes the calibration routine constantly in the background, triggering measurements with constant frequency and then estimating and returning the distribution factors to the controller. In this case, the fundamental frequencies  $f$  and  $f_{calibration}$  are identical. Using this method, the duty cycle cannot be 100% or 0%, because under those conditions it is not possible to perform the off-state or on-state measurement, respectively. Therefore there is a tradeoff between wide intensity range and high carrier frequency. It is also possible to use different carrier frequencies for each light fixture. The advantage of this method is that the measurements do not need to be synchronized with the driver signals and the driver signals of different light sources do not need to be synchronized to each other. The measured illumination is filtered with bandpass filters in order to identify the source. Also, in this case, the sensors board scans constantly and returns the distribution factors to the controller. If these proposed changes are realized, the user will no longer have to calibrate the system manually. Changes of sensor position would be recognized automatically, though multipoint and area average calibration would still be possible. Another necessary step is the redesign of the sensor board. A smaller sensor board could increase the quality of the closed-loop control because the user could place the sensor within the area of interest without tradeoffs. In addition, an array of light sensors could be integrated to detect illumination angles. Finally, wireless communication should replace

wired serial communication to improve mobility and ease of use in future deployments.

Closed-loop color temperature control is not currently implemented in the system because there is no reliable color sensor input to the color temperature conversion algorithm. The transformation function for sunlight, for example, is different than the transformation for LED light. Since both spectra are present, it is difficult to estimate the exact color temperature. One solution might be to add more color sensors. Another solution is to take dark measurements in order to detect the amount of sunlight, and choose between sunlight and LED light transformation functions depending on the ratio between the two.

Future work on the webcam based control system includes luminance surface approximation, performance tuning, color temperature feedback and gesture recognition. The current calibration method for the webcam based system is area average calibration. In future work, multipoint calibration, optimization and feedback should be implemented and tested for luminance based closed-loop control. In contrast to the sensor board, the camera makes it possible to monitor the illumination of an area. Without a camera, this could only be realized by a network of light sensors. One way to realize light surface approximation is to divide the image of the area of interest into sections and measure luminance for each section. It is also useful to measure the contributing factors for each section and to constrain each section with reference intensity and color. The performance of the calibration routine could be improved using a different programming platform and asynchronous measurements, in which light source driver signals and measurement trigger signals are unsynchronized. This approach requires less communication between the controller and the webcam. Color temperature feedback for the webcam based system is possible using an auto white balance algorithm to estimate the color temperature, as described in [15].

The intelligent lighting system should be extended to a multi user system, as provisions for this transformation have already been made. The variable *nou*, representing the number of users, is used in many parts of the software. If the variable is used in a function, the function is implemented for a multiuser application. The estimation of illumination angle should be further investigated. Finally, closed-loop control and online calibration need to be applied to the pentachromatic system and tested on a large scale lighting system. Long-term research goals for interactive lighting include visual light communication and distributed control.

# Bibliography

- [1] ALDRICH, Matt: *Dynamic Solid State Lighting*, Massachusetts Institute of Technology, Diplomarbeit, 2010
- [2] BIRGE, Brian: *Particle Swarm Optimization Toolbox*, 2005
- [3] C-WORKER.CH (Hrsg.): *Grundlagen und TCP*. c-worker.ch. – <http://www.c-worker.ch/tuts.php>
- [4] DILLOUÏE, C.: *Advanced lighting controls: energy savings, productivity, technology and applications*. Fairmont Pr, 2006
- [5] HUO, J. ; CHANG, Y. ; WANG, J. ; WEI, X.: Robust automatic white balance algorithm using gray color points in images. In: *IEEE Transactions on Consumer Electronics* 52 (2006), Nr. 2, S. 542
- [6] IZSO, L. ; LANG, E. ; LAUFER, L. ; SUPLICZ, S. ; HORVATH, A.: Psychophysiological, performance and subjective correlates of different lighting conditions. In: *Lighting Research & Technology* 41 (2009), Nr. 4, S. 349
- [7] MIKI, M. ; AMAMIYA, A. ; HIROYASU, T.: Distributed optimal control of lighting based on stochastic hill climbing method with variable neighborhood. In: *IEEE International Conference on Systems, Man and Cybernetics, 2007. ISIC, 2007*, S. 1676–1680
- [8] MIKI, M. ; HIROYASU, T. ; IMAZATO, K. ; YONEZAWA, M.: Intelligent lighting control using correlation coefficient between luminance and illuminance. In: *Proc IASTED Intelligent Systems and Control* 497 (2005), Nr. 078, S. 31–36
- [9] MORGAN, Steven S.: *A Comparison of Simplex Method Algorithms*, University of Florida, Diplomarbeit, 1997
- [10] OHTA, N. ; ROBERTSON, A.R.: *Colorimetry: fundamentals and applications*. John Wiley New York, 2005
- [11] PAN, M.S. ; YEH, L.W. ; CHEN, Y.A. ; LIN, Y.H. ; TSENG, Y.C.: Design and Implementation of a WSN-Based Intelligent Light Control System. In: *28th International Conference on Distributed Computing Systems Workshops, 2008. ICDCS'08, 2008*, S. 321–326
- [12] PARK, H. ; BURKE, J. ; SRIVASTAVA, M.B.: Design and implementation of a wireless

- sensor network for intelligent light control. In: *Proceedings of the 6th international conference on Information processing in sensor networks* ACM, 2007, S. 379
- [13] PROFILES, E.E.P.: LIGHT'S LABOUR'S LOST. (2006)
- [14] SINGHVI, V. ; KRAUSE, A. ; GUESTRIN, C. ; GARRETT JR, J.H. ; MATTHEWS, H.S.: Intelligent light control using sensor networks. In: *Proceedings of the 3rd international conference on Embedded networked sensor systems* ACM, 2005, S. 229
- [15] VUONG, Q.K. ; YUN, S. ; KIM, S.: A New Auto Exposure and Auto White-Balance Algorithm to Detect High Dynamic Range Conditions Using CMOS Technology. In: *Relation* 10 (2009), Nr. 1.5, S. 9064
- [16] WEN, Y.J. ; GRANDERSON, J. ; AGOGINO, A.M. ; BERKELEY, UC: Towards embedded wireless-networked intelligent daylighting systems for commercial buildings. In: *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006* Bd. 1, 2006
- [17] WYSZECKI, G. ; STILES, W.S.: *Color science*. Wiley New York, 1982