Visualisierungsinstitut der Universität Stuttgart

Universität Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Studienarbeit Nr. 2422

Ambient Light Transfer

Manuel Jerger

Studiengang:	Informatik
Prüfer/in:	Jun-Prof. Dr. Martin Fuchs
Betreuer/in:	Sebastian Koch, M.Sc.
Beginn am:	1. April 2013
Beendet am:	30. September 2013
CR-Nummer:	I.3.7

0.1 Abstract

In the following work we present a system for capturing ambient light in a real scene and recreating it in a room equipped with computer-controlled lamps. We capture incident light in one point of a scene with a simple light probe consisting of a camera and a reflective sphere. The acquired environment map is then transferred to a room where an approximated lighting condition is recreated with multiple LED lamps. To achieve this, we first measure the impact each lamp has on the illumination with a light probe and acquire one image per lamp. A linear combination of these images produces a new environment map, which we can recreate inside the room by setting the intensities of the lamps. We employ Quadratic Programming to find the linear combination that approximates a given environment map best. We speed up the optimization process by downsampling the light probe data, which reduces the dimension of our problem. Our method is fast enough for real-time light transfer and works with all types of linear controlled illuminants. In order to evaluate our method, we designed and constructed an omnidirectional lighting system that can spatially illuminate a room in full-color. We first explore several different configurations of our lighting system, our sampling and our optimization algorithm. We then demonstrate our method's capabilities by capturing static and dynamic ambient light in one location and transferring it into a room.

0.2 Kurzfassung

In der folgenden Arbeit stellen wir ein System vor mit dem man das Umgebungslicht an einem Ort aufnehmen und in einen Raum mithilfe computergesteuerter Lampen nachstellen kann. Wir verwenden eine einfache Light Probe, bestehend aus einer Kamera und einer verspiegelten Kugel um das Licht das in einem Punkt einer Szene eingeht aufzunehmen. Dies liefert uns eine Environment Map welche dann in einem Raum durch das Ansteuern von LED-Lampen approximiert wird. Dazu messen wir zuerst die Lichtverteilung jeder einzelnen Lampen im Raum mit einer Light Probe und nehmen ein Bild pro Lampe auf. Durch eine Linearkombination können wir neue Environment Maps erzeugen, welche wir dann im Raum durch Einstellen der Lampenhelligkeit darstellen können. Wir verwenden Quadratic Programming um eine Linearkombination zu finden, welche die zu übertragende Environment Map am besten approximiert. Wir beschleunigen den Optimierungsprozess indem wir die Environment Maps abtasten und somit die Größe des Problems reduzieren. Unsere Methode eignet sich für das Übertragen von Umgebungslicht in Echtzeit und funktioniert mit jeder linear steuerbaren Lichtquelle. Für die Evaluierung haben wir neben einer mobilen Light Probe auch ein Beleuchtungssystem entworfen und aufgebaut. Es handelt sich um eine mobile Konstruktion mit der ein Raum omnidirektional und farbig ausgeleuchtet werden kann. Wir untersuchen zuerst unterschiedliche Konfigurationen unseres Beleuchtungssystems, des Abtastprozesses und der Optimierung. Anschließend demonstrieren wird die Fähigkeiten unseres Systems in dem wir statische und dynamische Umgebungsbeleuchtungen aufnehmen und in einem Raum wiedergeben.

Contents

List of Figures

List of Tables

1 Introduction

Photography has always been very intriguing because it enables us to store, recreate and share visual impressions with others. Since the first grainy black-and-white images the techniques have been greatly improved. Today, photography is mostly digital. Digital cameras and displays have become almost omnipresent in our society and can be found in every household. Many people even carry them around at all times in form of a smartphone.

Digital photography makes it easy to share visual impressions and the Internet has revolutionized the way we share.

However cameras and displays have a severe limitation: They cover only a small field-of-view (FOV) compared to the human eye. We can increase the FOV of a camera with wide-angle lenses or by applying panorama techniques but we cannot do the same with displays.

The size of a display is often constrained by requirements like portability and cost. Most screens, especially in mobile devices, deliver only a very small image to the retina of our eyes. This is perfectly acceptable for displaying high resolution content because only a small region in the center of the retina is capable of perceiving fine details. It does however matter if the user should be immersed in a scene, for example while watching a movie or viewing vacation photos. The so-called *peripheral vision*, which lies in the outer region of the retina, plays a strong role in how we experience our environment. It has a much lower resolution and cannot image fine details, but it perceives the ambient light.

To maximize the user immersion the impression over the whole retina should be consistent. There are several ways to achieve this: *Movie theaters* for example shut out all ambient light and use large screens to diminish false impressions in the peripheral vision. *Head-mounted displays* place small high-resolution screens very close to the eyes. *Immersive spaces* on the other hand completely enclose the user in a box made of screens and achieve a complete immersion [GWN+03]. These methods are either very expensive, too big, or do not allow for a simultaneous experience by multiple users.

There is however a simpler and cheaper solution: Control the ambient light so it matches the displayed scene. Modern LED technology has finally become affordable and powerful enough to be used as room lighting and provides us with digitally controlled full-color illuminants. A room equipped with multiple lamps allows not only for an uniform ambient illumination but also for a spatial one.

In this work we propose a method that recreates approximate lighting conditions in real-time using digital room lighting.

We designed and implemented an omnidirectional RGB lighting system using full-color RGB-LEDs. Our system captures an environment map in a real world scene with a light probe and recreates an approximation in a room by controlling the lamps. We begin with a calibration step and measure the illumination of our lighting system by capturing one light probe image per lamp. The linear combination of these images produces a new environment map, which we can recreate in the room by setting the intensities of the lamps. Our method employs Quadratic Programming (QP) to find the linear combination that approximates a given environment map best. We will show that our system is fast enough to transfer ambient light in real-time from one place to another.

2 Background

This chapter introduces some basic topics necessary to understand this work. We start with a description of the *ambient light* of natural environments. After discussing natural light we talk about artificial lighting and the challenges and benefits of modern LED technology. We introduce the concepts of *environment maps*, *distant light* and *light probes*. We also explain how *Image-based Relighting* exploits the linearity of light and describe the *Inverse Relighting* problem on which we based our method.

2.1 Ambient Light

Our environment is brightly lit by many different kinds of light sources. The most powerful and influential source of them all is the sun. Sunlight is an integral part of our lives and affects us day by day. It defines how we structure our day, when we are awake and when we are asleep. Our body has strongly adapted to the continuously changing illumination and we even use it to synchronize our internal biological clock, the circadian rhythm [EGK+13].

Sunlight is very dynamic. As the earth rotates around its axis, the sun travels along an arc over the sky. The axis also tilts over the course of a year and causes the sun to rise much higher in summer than in winter. The lower the suns position, the lower is the light intensity because the path through the atmosphere is much longer and more light is scattered away.

While traveling through the atmosphere, light with a short wavelength is scattered more than light with a longer wavelength. This changes the color of the sunlight depending on the incident angle: With low angles more blue light is scattered away and the color shifts towards red.

The varying sunlight illuminates our nature and produces complex ambient illuminations with different color pallets and light distributions which cannot be captured in a photograph or reproduced by a standard display.

2.2 Room Lighting

Traditional illuminants like halogen, fluorescent and incandescent lamps are designed to simulate sunlight. The have a similar spectrum and are most commonly placed on the ceiling to illuminate the room from above. Compared to sunlight, room lighting is very static: The user can turn it on or off and in some cases control the intensity with a dimmer switch.

Light-emitting diode (LEDs) technology has been greatly improved over the last few years and is on the verge to change the way we think about room lighting. LEDs posses several advantages over traditional illuminants: They have a higher efficacy, produce less heat, are much smaller and more flexible to handle. Because they have extremely high switching speeds, their light output can be precisely and linearly controlled with pulse-width modulation [MSP02]. They also emit monochromatic light with a nearly constant wavelength. By combining multiple primary colors we can build digitally controlled full-color lamps.

Unfortunately, LEDs have an inherent flaw: They are very sensitive to heat and require good heat dissipation [EGK+13]. Their lifespan is related to their inner temperature and decreases rapidly when they overheat. This makes it difficult to build small high output illuminants which the consumer market desires. They are however the ideal component for building large area light sources because heat can be spread out more easily for dissipation. Small LEDs are also cheaper to produce as the yield in production is much higher.

These properties of LEDs have lead to new types of light sources that differ from traditional lamps. The high life-span means that they do not necessarily have to be replaced and can be embedded directly into the walls or furniture. Hundreds of thousands of small LEDs can be spread out on large areas, aggregated in flat panels or linear strips, and provide the ambient light.

2.3 Naming Conventions

This section introduces the naming conventions and assumptions that are used throughout this work.

First of all: We always work in radiometric space. Our cameras and images record relative radiance. An image is represented as a vectors of scalar values. It has three color channels which are concatenated inside the vector. Matrices are denoted by capital letters, vectors and scalars are lowercase. Vectors are written in bold.

The following table shows the notations we use.

Notation	Description
$\boldsymbol{i}_{sum} = a\boldsymbol{i}_1 + b\boldsymbol{i}_2$	linear combination of two images
$A\boldsymbol{x} = \boldsymbol{b}$	matrix-vector multiplication
A^T	transposed matrix or vector
$\lvert\boldsymbol{x}\rvert$	number of elements in a vector or list
$\lvert\lvert\boldsymbol{x}\rvert\rvert$	L_2-norm $\sqrt{x_1^2 + x_2^2 + ...}$

2.4 Light Probe

Environment maps are often used in computer graphics as a simple model for global illumination. They described the incident radiance L_d in a point I, parametrized over the light direction. If the light is distant and directed, the incident light is the same in all points of scene.

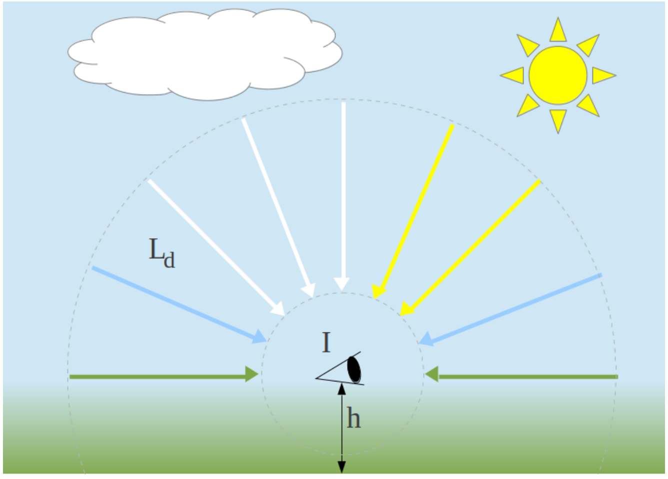


Figure 2.1: Environment maps (a) describe the incident radiance L_d in the point I of a scene.

Environment maps can be used to render specular objects with the reflection mapping technique [BN76], where the direction of reflected light is defined by the orientation of the object's surface. This principle can also be used to acquire environment maps of real scenes: Light probe images [Deb98] are photographs of specular objects that allow us to capture light from almost all directions using a single image. Any specular convex object can be used as a probe object. Spheres are most commonly used for this because their geometry is simple and defined by a single parameter, the radius. Spherical probe objects are also readily available, for example in the form of steel ball bearings or metalized glass spheres. The latter one is cheaper but has a less precise geometry and is more fragile.

A single image taken from a mirrored sphere captures light from almost all directions. Figure 2.2 shows a light probe image (left) and how it was recorded (right). A small cone shaped region O_S lies in the shadow of the sphere S, while some other directions O_C are occluded by the camera and other parts of the mechanical setup. The occlusions can be reduced by increasing the ratio of distance $\overline{CS}$ and sphere diameter, but do not vanish completely. Note that light directions are not imaged uniformly and the lower hemisphere is recorded with a much lower resolution. If we place the sphere close to the floor, we can constrain the incident directions to the upper hemisphere and increase the effective resolution.

With this technique, we can record an omnidirectional view of a scene using a single camera.

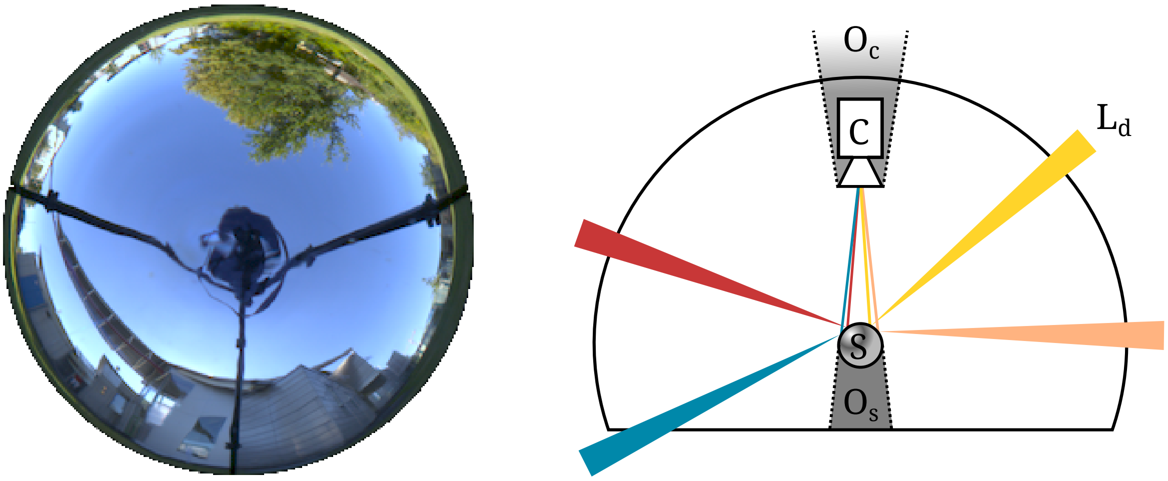


Figure 2.2: A light probe image (left) and how we record it (right): A camera (C) captures the light from all directions via reflection on a mirrored sphere (S). Some directions are occluded by the camera (O_C) or lie in the shadow of the sphere (O_S).

2.5 Linear Light

Light transport is linear in the light sources: The light from multiple sources add together to illuminate a scene. We can observe the light of each source individually and apply the super-position principle to derive new combinations of scene lighting.

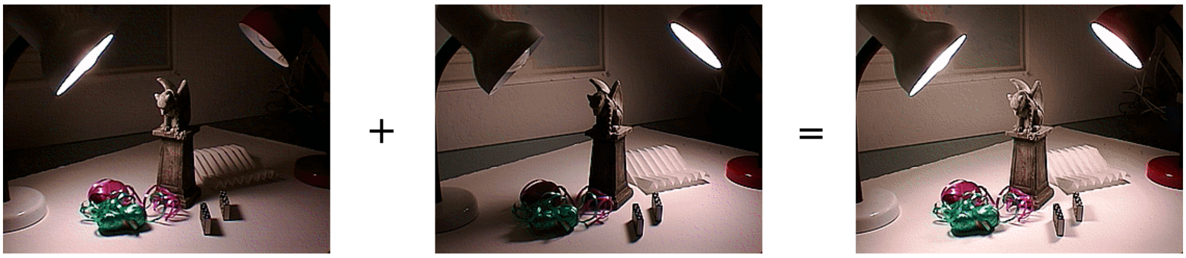


Figure 2.3: Light transport is linear in the light sources. The super-position principle can be applied to photographs of a scene (images from [Hae])

2.5.1 Image-based Relighting

Image-based Relighting [CCH07] employs the linear properties of light in a clever way: A simple weighted sum of differently illuminated photographs can be used to render the scene with a novel illumination.

$$\boldsymbol{i}_{new} = \sum_d w_d \boldsymbol{i}_d \tag{2.1}$$

A set of basis image $\boldsymbol{i}_d$ are taken of an object or scene with a fixed camera, each with a different illumination. Evaluation of the weighted sum results in a new image $\boldsymbol{i}_{new}$. The weights w_d modulate the intensity of the light in the basis images. Note that the weights must be positive to be physically correct. If a lot of images with different illuminations are taken, then one can *relight* a scene with a given environment map. Relighting is especially interesting for rendering objects since the complexity of the rendering process is independent from the complexity of the object geometry. The renderings are also physically correct and reproduce even complex effects like scattering, refractions and caustics easily [CCH07].

The key problem is finding the weights w_d that produce the best approximation of the novel illumination.

In *Explicit Relighting*, the scene is illuminated with well-defined point sources, for example by using a light stage [DWT+02]. The weights can be calculated by sampling the novel environment map at the location that corresponds to the position of the point source.

Implicit Relighting on the other hand can work with any type of light source [CCH07]. The incident light is captured with a *probe object* in addition to the basis images. In this case, the weights are calculated by solving the *Inverse Relighting problem*

$$\sum_d w_d \boldsymbol{p}_d \overset{!}{=} \boldsymbol{p}_{target} \tag{2.2}$$

where $\boldsymbol{p}_d$ are the images of the probe object and $\boldsymbol{p}_{target}$ is the probe image under the novel illumination.

The inverse problem can be written in matrix form as

$$P\boldsymbol{w} \overset{!}{=} \boldsymbol{p}_{target} \tag{2.3}$$

where the matrix P contains all the images $\boldsymbol{p}_d$ as columns, and the vector $\boldsymbol{w}$ contains all the weights w_d. Equation (2.3) has no exact solution because the matrix is not necessarily invertible. We can however calculate an approximate solution by converting it into a minimization problem. By minimizing the differences between both sides of the equation, we can solve for the weights that produce the smallest error. This requires a distance measure for images. If we use the squared L_2 norm, we can formulate the linear least squares problem

$$\underset{\boldsymbol{w}}{\operatorname{argmin}} ||P\boldsymbol{w} - \boldsymbol{p}_{target}||^2 \tag{2.4}$$

which can be solved fast and reliably with convex optimization techniques.

3 Related Work

Digitally controlled full-color lighting can be used to extend the FOV of displays by illuminating the space around the screen with light that matches the displayed scene. We can also apply this idea to a whole room: A spatially illuminated room can supply meaningful information to the peripheral vision of the viewer and can increase the feeling of immersion [GTSH05].

The method we present in this paper is based on the Inverse Relighting problem. Our solution is thus closely related to Implicit Relighting and Lighting Design.

3.1 Relighting and Lighting Design

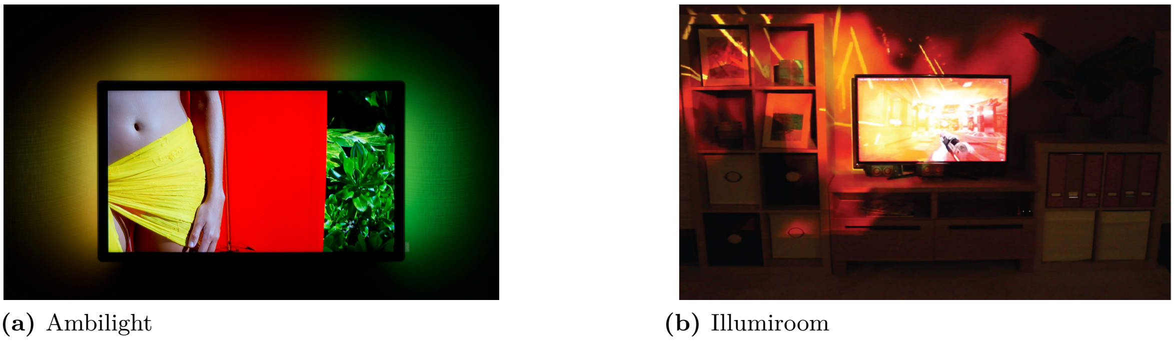
There are several Implicit Relighting methods [MLP04] [FBS05] that use probe objects in order to do relighting with low-frequency illuminations. They could be directly applied to our problem by *relighting the room* using the lamps. The rendering-part can be omitted as we are only interested in the image weights, which correspond to the brightness of the lamps. However, those techniques are designed to render *good* images and are not suited for real-time applications. They also do not use constrained weights which are required in our application, because real lamps have a maximum brightness.

The Inverse Relighting problem is also addressed in Image-Based Lighting Design [ADW04] by Anrys et al.: They use a light stage to illuminate an object with RGB point-sources and capture the effect of each lamp on the object with a camera from a fixed view-point. The user can then specify the desired illumination of the object by drawing on it in image-space. This serves as a target for the minimization algorithm which finds the weights for the lamps that reproduce the desired illumination. They employ Sequential Quadratic Programming (SQP) to minimize the least squares equation (2.4). Analytical representations of the Hessian and Jacobin matrices are employed for an additional speedup.

Our method is closely related to the work of Anrys et al: Instead of illuminating an object, we illuminate the walls of a room and capture omnidirectional images with a light probe. This is basically the concept of a light stage turned inside out. Instead of SQP, we apply Quadratic Programming (QP) to solve the same constrained least squares problem, and reduce the problem size beforehand by performing a downsampling on the light probe data.

3.2 Extending Displays

Movie and home theaters benefit from dim ambient light and large screens because this setup increases the immersion of the viewer. The screen produces a large image on the viewers retina and the absence of ambient light diminishes false impressions in the peripheral vision. Rather than shutting down the ambient light, one can also actively control it to match the brightness and color of the displayed images.



(a) Ambilight

(b) Illumiroom

Figure 3.1: Extending the apparent size of a display by augmenting the surrounding space with light. Philips Ambilight (a) uses LEDs and Microsoft's Illumiroom uses a projector (b).

Philips created their Ambilight TVs around this idea: Lines of RGB-LEDs are placed on the edges of a screen and illuminate the wall behind it. The LEDs are controlled to match the average color in the border regions of the image and thus extend the screen onto the wall. This gives the impression of a larger television screen and also supplies the room with ambient light of an appropriate hue. A study showed that their implementation is well received by the user and greatly improves the feeling of immersion [Beg05]. The Illumiroom concept [JBOW13] follows a similar idea: A projector is used instead of LEDs to augment the the area around a screen with content of lower resolution.

Just recently Philips introduced wireless controlled LED-based full-color lamps to the consumer market [HUE]. The Philips hue bulbs are shaped like traditional lamps and fit in standard sockets. They can be individually controlled via a wireless station and a smartphone which gives the user full control over hue and brightness. More interestingly, they can be linked to the Philips Ambilight TVs so all of the ambient illumination in a room is under control of the television. The user can define the approximate location of the lamps and the display in the room. The Ambilight TV uses this information to extract appropriate lamp colors from the video stream. How exactly this is done has not been published.

3.3 Real Illumination from Virtual Environments

Gosh et al. addresses the same problem we do in their work *Real Illumination from Virtual Environments* [GTSH05]. They too illuminate a room with multiple RGB lamps to reproduce an illumination given by an environment map. However, their work is focused on the effects of ambient light on the user. Their solution is not general and requires an optimal room and lighting setup.

The proposed method works like this: First, they switch the lamps to white one at a time and capture the light spots on the walls with a light probe. They fit a Gaussian to each spot: The center is placed at the the brightest point and the standard deviation is adjusted until the error is minimal. They then calculate the color for the lamps by sampling the environment map with the fitted Gaussian kernel. This technique only works with calibrated RGB Lamps with a known white point. Gosh et al. link the color channels of the LEDs to the color channels of the display, which prohibits the use of white LEDs or other broad-spectrum illuminants. There are also cases where a Gauss kernel cannot be fitted with a small error, for example if a lamp causes multiple light spots.

4 Hardware

Our method required the construction of two devices: A light probe for capturing ambient light and a computer controlled lighting system for recreating it. This chapter describes their construction and implementation and explains the challenges we faced.

4.1 Lighting System

We designed and constructed a lighting system that is capable of illuminating a medium-sized room omnidirectional in full-color. We evaluated several LED products and came to the conclusion that low-power LEDs provide more light per cost than high-power LED lamps.

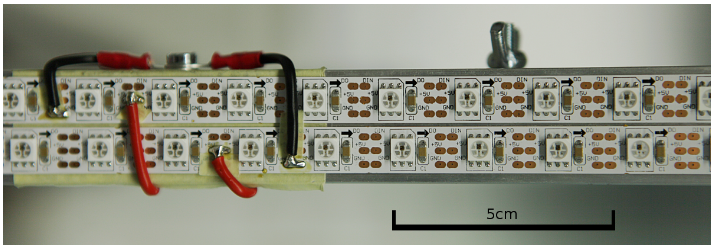


Figure 4.1: Two strips of WS2812-LEDs and the power wires, mounted on an L-shpaed aluminum rail.

We used WS2812 RGB-LEDs from Worldsemi [WS2] which have an interesting property: they are digital. Each single device has a built-in chip that adjusts the light output of the three color channels linearly with an 8 bit pulse-width modulation. They communicate over a very simple timing-based protocol that allows them to be arranged in a *daisy-chain* configuration. Hundreds of LEDs can be connected in a linear chain and addressed individually with just one input: The data for all LEDs is shifted in on the input, the first LED takes three bytes and forwards the rest to its successor which then does the same. After the data has propagated, all LEDs latch and start displaying the new light color simultaneously. They were originally designed to be used in large LED displays, which is also the reason for their wide beam angle of 120 degree and the low price per piece. They are available as pre-assembled strips with 30 or 60 LEDs / meter.

We decided to use large amounts of these LED strips to build a lighting system. We constructed eight gibbet-shaped sticks, two meters tall, to carry the LEDs and distribute them throughout a room. The sticks are arranged in circular fashion so the fronts face the wall and their tops

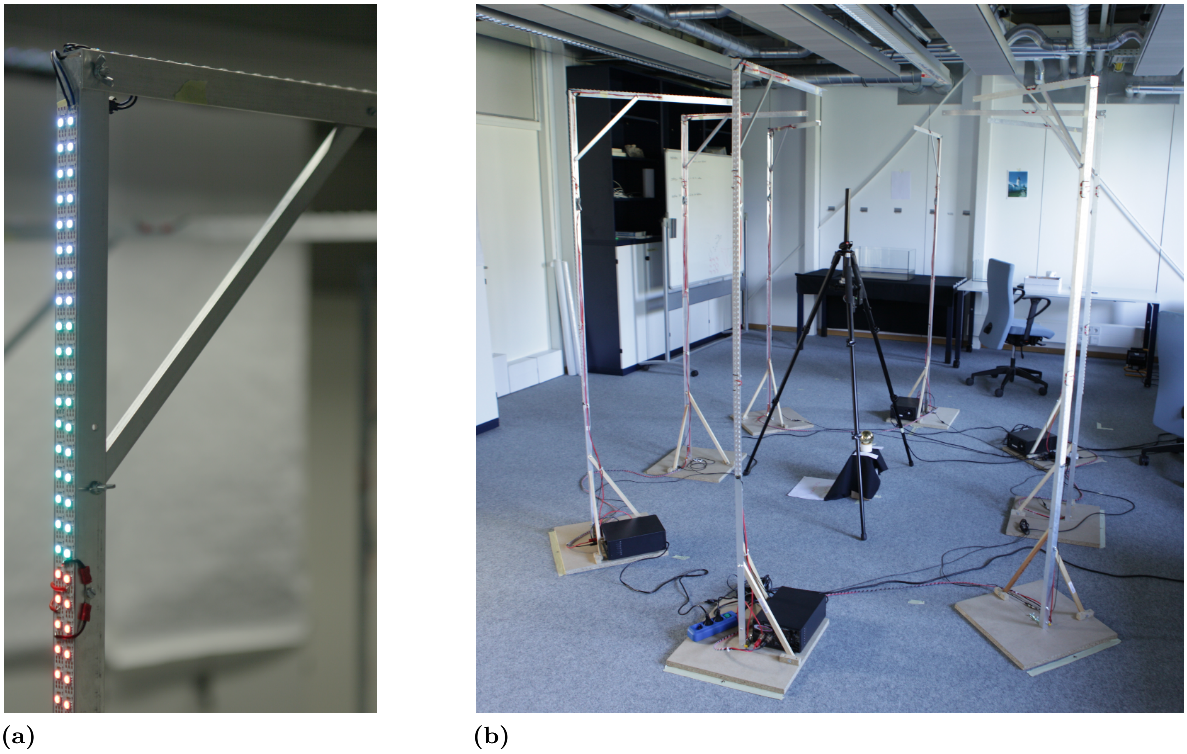


Figure 4.2: LED-strips mounted on a gibbet-shaped construction made of aluminum (a). Eight sticks are set up in a room (b) and illuminate the walls and ceiling. A light probe is placed in the center of the room for the calibration step.

face the ceiling. Each stick is lined with four meters of LED-Strip (240 LEDs), arranged in two parallel columns for increased light output. We supply both strips with the same data. This gives us 120 controllable lamps per stick, and 960 lamps on all eight sticks.

The lighting system was powered by four 30A 5V power supplies and consumed about 460 watts on full power. The LED's low voltage of 5V turned out to be impractical because our spacious setup required the transport of high currents over a distance of several meters. We had to use heavy 6mm^2 copper wires between the power supply and the sticks to keep the voltage drop to a minimum. We used L-shaped aluminum rails, 30x30mm wide and 3mm thick, as construction material for the sticks because it is light, sturdy and can also act as a heatsink for the LEDs. Aluminum is a good electrical conductor too, so we used it as part of the wiring and replaced nearly half of all copper wires with it.

The lighting system cost less than 1200 Euro and took about 45 hours to build. The mechanical drawings and the bill of material can be found in the appendix A.

The WS2812 LEDs can be controlled with an Arduino, a cheap microcontroller platform. We used a Teensy 3.0 [TEE] which is special third-party Arduino board that has a faster CPU and a better USB controller than the original Arduino hardware. It also comes with many specialized libraries. One of them is the OctoWS2811 library [OCT] which can be used to communicate with the WS2812 LEDs. We used it to quickly write our own firmware and turn the Teensy into an USB interface for our lighting system. We can control our sticks reliably with refresh rates up to 40 Hz.

The wide beam angle causes two successive LEDs on the strip to produce nearly the exact same illumination pattern in the room. For this reason we decided to reduce the vertical resolution by dividing the strips into equal sized segments and combine multiple LEDs into virtual lamps. Clustering several LEDs together is also beneficial in that it averages manufacturing-related variations of the light output. It also makes it possible to select the number of lamps without changing the setup itself. After evaluating configurations with different segment sizes (see 6.1) we found that we can choose fairly large segments and still produce good results.

4.1.1 Limitations

The LEDs did not behave as linearly as we originally thought. Although the response of each single color channel was perfectly linear in the input, they showed an interdependency. The red channel was especially affected and its light output dropped measurably when the output on the green and blue channel was increased. We suspect that this effect is either caused by an insufficient interconnection inside the LED which limits the available peak power, or a poor choice of decoupling capacitors by the manufacturer of the strips.

The influence of the temperature was also measurable. The light output decreased with increased temperature, which is typical for LEDs. Even though our aluminum construction provided a large surface area for heat dissipation, it reached up to 55 ° C after thirty minutes on full power. Again, the red channel was affected the most. This is effect is very hard to compensate for because it is always present, even with low temperatures. It would require either a constant temperature or an active compensation inside the LED to achieve a stable light output.

4.2 Light Probe

Our method utilizes a simple light probe to capture the incident radiance in a scene that we want to reproduce. Furthermore, it is used in the calibration step 5.2 to record the light distribution of each individual lamp.

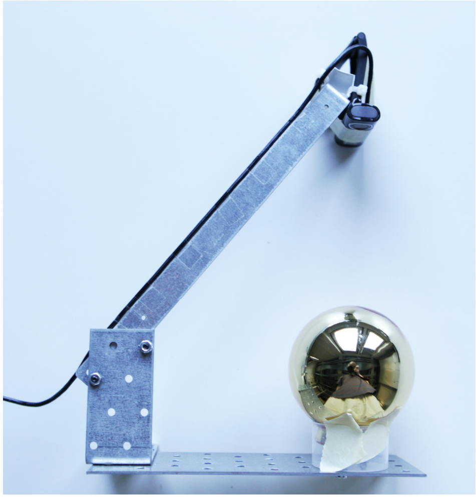


Figure 4.3: Our mobile light probe consists of a consumer-grade HD webcam mounted above a mirrored sphere.

We based our light probes on reflection mapping as explained in section 2.4. Our method does not require high resolution light probe images or precise incident directions. We thus assume a perfect, distortion-free pinhole camera with an infinitely small hole. The sphere is assumed to be perfectly shaped and 100% reflective. We correct any color tint by performing a white balance and scaling the color-channels linearly.

4.2.1 Model

Every pixel of our light probe camera captures light that originates from a different direction. These directions can be calculated from the pixel positions with a simple geometric model. For this, we first need to know the position of the sphere in the image by letting the user select three points that define the circular area. We also need several geometric parameters of the probe setup: The sphere diameter, the distance between sphere center and camera pinhole, the cameras rotation around the sphere center and the rotation of the probe with respect to the room. We assume that the sphere's center is aligned with the optical axis of the camera.

Our geometric model is shown in figure 4.4. We place the virtual image plane (red) through the points where the marked circle touches the sphere. Note that the image plane not lie at the origin of the sphere but is shifted towards the camera by a distance h. This distance is given by the y-coordinate of the contact point P between sphere and the tangent t. To calculate the reflected direction, we cast a ray i from the pinhole I through each pixel on the image plane, intersect it with the sphere and determine the surface normal n at the intersection point P. On a sphere, the surface normal at a point P is the vector from the center C of the sphere to P. If C is placed at the origin, P can be directly interpreted as the normal. We then reflect the the ray on the surface using the normal, normalize it and rotate the resulting direction according to the rotation of the camera.

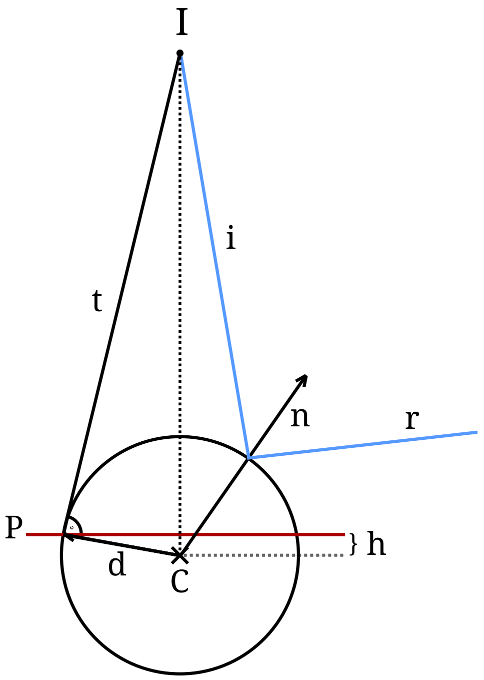


Figure 4.4: Simple geometric model for calculating the direction of the light reflected by a mirrored sphere. The camera I records the light coming from direction r.

We mask out all obstructed light directions in image space and constrain the sampling range to the upper hemisphere. We precalculate all available directions and save them together with their corresponding pixel positions in a list D for the sampling step explained in section 5.1.

4.2.2 Probe Implementations

We constructed two devices: The *Webcam Probe* is a mobile light probe based on a consumer webcam. The second device, the *Canon Probe*, uses a more sophisticated DLSR camera.

The Webcam Probe consisted of a simple consumer grade HD webcam, mounted above a Christmas ornament. The probe object was 8 cm in diameter and had a yellow color tint. This light probe was very portable and had almost no obstructed light directions but the camera had a color-depth of only 8 bit and limited exposure capabilities. It was also impossible to capture raw images because of the unknown preprocessing steps that happen inside the device.

For this reason, we decided to set up the Canon Probe, consisting of a Canon 5D MarkII and the same glass sphere. We placed the camera on a tripod above the sphere, looking down. This setup was much more inflexible and the tripod obstructed more light directions, but it gave us 14 bit of high resolution raw images and full control over the exposure. In order to convert the raw data into relative radiance, we performed a radiometric calibration and recovered the response curve with the method from Robertson et al. [RBS99], which is conveniently available in the open source package *pfshdrcalibrate* [PFS].

We used the canon probe for most of our experiments and the webcam probe only for the recording of dynamic environment maps.

5 Implementation

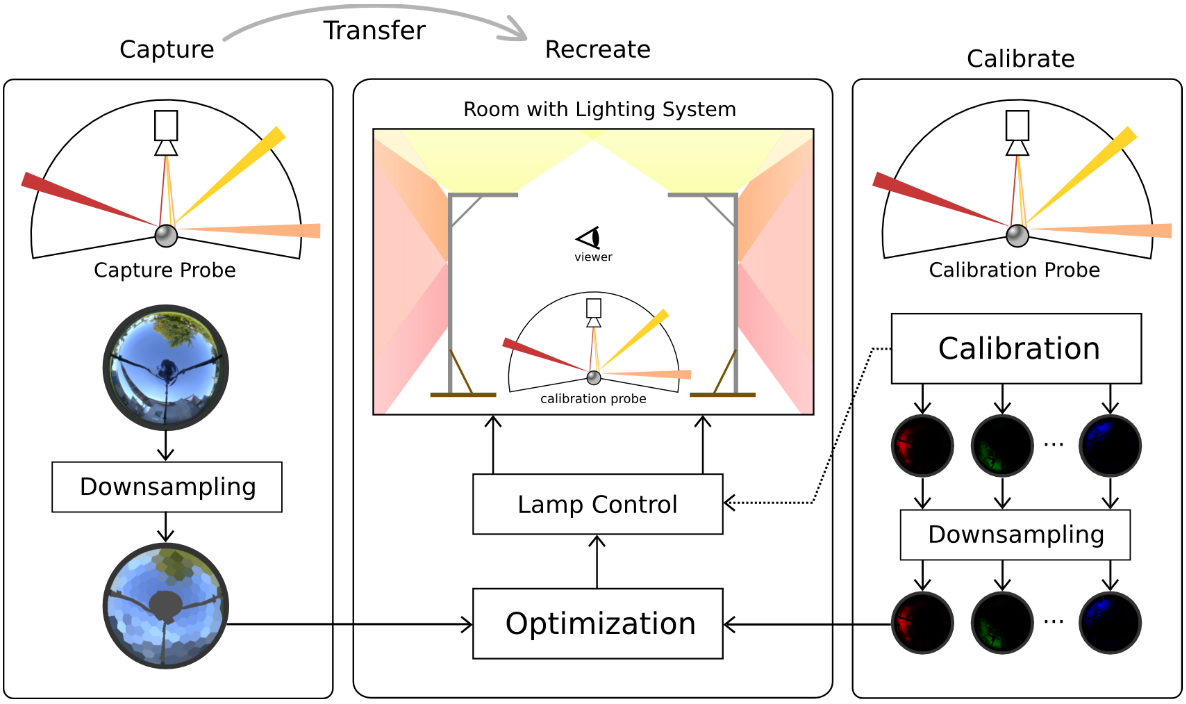


Figure 5.1: Overview of our system including the transfer- and calibration-pipeline. We first calibrate the room by capturing the illumination of each lamp with the Capture Probe. The acquired images are used by the optimization process to find the lamp values that reproduce the best approximation of a target environment map given by the Capture Probe.

Figure 5.1 shows an overview of the Ambient Light Transfer pipeline we devised. The incident radiance in a real scene is recorded by the *Capture Probe* and transferred to the viewing room. The room is illuminated by our lighting system and isolated from all other light sources. Our method utilizes information about how each lamp interacts with the room. This is done in the *calibration phase* where the lamps are switched on one at a time and the incident radiance is captured by the *Calibration Probe* in the center of the room. In our case a lamp is defined as a monochromatic light source and we treat the LED's color channels as individual lamps.

The *optimizer* uses the calibration images together with a target given by the *Capture Probe* to find the brightness values for the lamps. We employ Quadratic Programming to find a linear combination of calibration images that approximates the target best, and use the resulting weights to set the brightness of the lamps. We speed up the optimization process by performing a downsampling on the light probe data, which reduces the problem size and allows us to achieve real-time speeds.

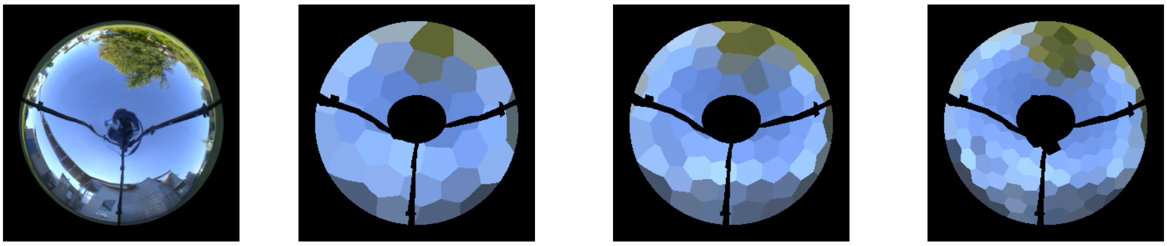
We implemented our system on a Linux-machine using the GNU tool-chain and C++.

5.1 Sampling

Our method leverages the fact that indirect illumination of a room causes light patterns with a low spatial frequency. We assume that effects like specular reflections are small in size compared to the diffuse reflections from large uniform colored areas. We thus can perform a downsampling on the light probe images and consider only the average radiance coming from a relatively small set of uniformly distributed directions. Downsampling not only reduces the size of our problem and the amount of data the optimizer has to handle, but also removes some of the image noise from the camera. It has a regularization effect, too: Highlights and small shadows are averaged and the variance of the data-set is reduced.

We select a set of sampling directions D, uniformly distributed in angular space. Recall that directions can be expressed as vectors of fixed length and the points that correspond to a vector lie on the hull of a sphere. The uniform selection of directions in angular space is thus equal to the uniform distribution of points on the surface of a sphere. The problem of creating uniform distributions of points on a sphere is complex has been thoroughly studied in mathematics. It is vital for all applications that require precise numeric integrations over the surfaces of a sphere [SK97]. We use uniform distributions calculated by Fliege et al. [FM99] as sampling directions and omit all directions that lie outside the sampling range.

We define the *neighborhood* of a sampling direction d as the cone of incident light that contains all directions that are closest to d. The distance between two directions is given by the enclosed angle. The cones and neighborhoods can be visualized in image-space as the *Voronoi Cells* that correspond to the points on the sphere.



(a) all directions **(b)** 33 directions **(c)** 63 directions **(d)** 128 directions

Figure 5.2: Neighborhoods in image space for different numbers of sampling directions. Each neighborhood displays the average radiance of the sampled directions.

We sample each direction by averaging the values of the pixels contained in the neighborhood. Figure 5.2 shows the sampled neighborhoods for several number of sampling directions as seen by the light probe camera. The Voronoi Cells assume a hexagonal shape which becomes more uniform with more sampling directions.

To find the neighbors, we use a naive approach: For each pixel, search for the closest sampling direction. The high run-time complexity is acceptable because the calculations have to be carried out only once for each light probe configuration and can thus be precalculated.

This sampling is not optimal, especially if we select a small number of directions: Neighborhoods are never of exactly the same size. Their shapes are irregular, they have hard edges and they do not overlap. Pixels that are marked as occluded are not sampled which reduces the size of nearby neighborhoods.

We investigated Gaussian distributions for weighting the pixels in a neighborhood. We experimented with different parameters for variance and kernel size but were not able to find a single optimal kernel for all sampling directions because of their irregular shapes. There are also other problems to consider: The border regions must be handled correctly because we do not sample the whole sphere and the sampling kernels are very large compared to the surface of the sphere. A Gaussian sampling is also much more expensive to evaluate than the nearest-neighbor approach because the sampling kernels overlap.

We decided to keep the sampling as simple as possible and use the nearest-neighbor approach. Not only is it very fast, but also independent from the number of sampling directions $|D|$ because a pixel belongs to exactly one direction and is queried only once per sampling. We select $|D|$ so the Voronoi Cells are much smaller than the light spots caused by the lamps. See section 6.2 for an comparison of various sizes and their influence on the quality of our results.

5.2 Calibration

Our method employs information about how the lamps illuminate the room. In the calibration step, we place the Canon Probe in the center of the room and capture the illumination each lamp produces. This step requires a very sensitive camera because our lighting system illuminates the room with indirect light, which is reflected multiple times before it reaches the sensor. Our Canon camera had a color-depth of 14 bit per channel and provided enough dynamic range to capture all of the reflected light without too much noise.

The calibration loop is straight forward: Set a lamp to maximum brightness, record its light pattern with a light probe image and turn it off again. This is repeated for all lamps. We additionally capture a dark-frame image with all lamps turned off and subtract it from the calibration data. This removes the ambient light that may be present in the almost but not completely dark room, and also handles the floor-noise from the image sensor. Our method does not utilize prior knowledge about the color channels of our lamps and treats them as individual, monochrome light source. The calibration step has to be repeated every time the lighting system changes or large objects inside the room are moved.

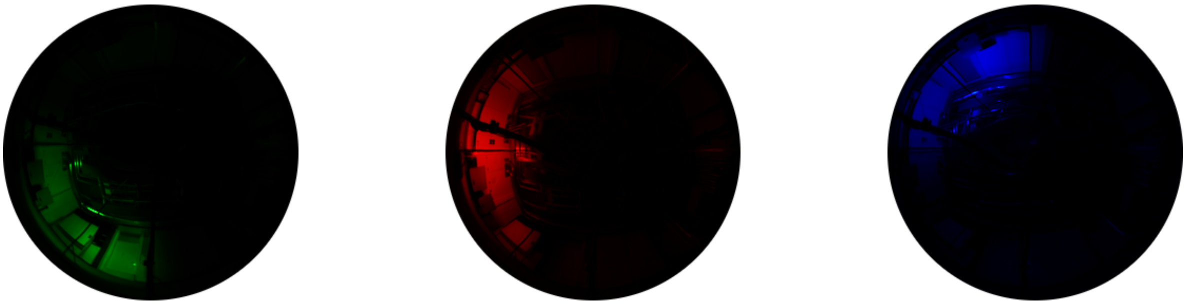


Figure 5.3: Three light probe images acquired in the calibration step. We treat each color of our RGB-LEDs as an individual, monochrome lamp.

5.3 Optimization

Our problem is formulated as follows: Given a target light probe image T and a set of calibration images I_l (one for each lamp l), find the weights w_l so that the difference $(\boldsymbol{T} - \sum_{l \in L} w_l I_l)$ is as small as possible. The weights w_t correspond to the brightness of the lamp l. The downsampling reduces the light probe images to a vector of size $3|D|$ containing the average radiance for all color channels. For the sake of simplicity we hereafter assume monochromatic images.

We take the sum of squared differences and formulate the minimization term

$$\underset{w}{\operatorname{argmin}} \, ||s\boldsymbol{t} - \sum_{l \in L} w_l \boldsymbol{i}_l||^2 \tag{5.1}$$

where $\boldsymbol{i}_l$ are the downsampled calibration images and $\boldsymbol{t}$ is the downsampled target.

The resulting weights w_l define the intensity of the lamps and are constrained to a range of [0..1], where 1.0 is the maximum brightness. The factor s is the *target scale* which relates the exposure of the target to the exposure of the calibration images. A bigger value increases the overall brightness of the result, but can lead to color changes if it is too large and the target cannot be reached by the optimizer. The choice of the scaling factor is not trivial and depends on the setup of the lamps and their light distribution in the room because the illumination is not completely uniform and the maximum brightness varies between sampling directions. See 6.4 for an evaluation of the scaling factor.

We use Quadratic Programming (QP) to minimize the term (5.1) with respect to the weights. The solver is provided by the CVXOPT library [CVX]. We transform our minimization term into the QP matrix form (5.2) required by the solver.

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{w}^T Q \boldsymbol{w} + \boldsymbol{c}^T \boldsymbol{w} \\ \text{subject to} \quad & U\boldsymbol{w} = \boldsymbol{b} \end{aligned} \tag{5.2}$$

The vector $\boldsymbol{w}$ contains our weights w_l and the Q matrix is filled with the calibration data. U is the unit matrix and $\boldsymbol{b}$ contains the target data. We expanded our minimization term (5.1), as well as the QP term (5.2) and derived the values of the Q matrix by comparison of coefficients. The QP solver has to be set up only once per calibration and the vector $\boldsymbol{b}$ can be updated with the new target data in every iteration.

The optimizer utilizes only a single CPU-core and converges fast enough for real-time applications. See section 6.3 for an evaluation of the run-time with respect to the number of lamps and sampling directions. The optimization converges reliably. Small changes in the target produce small changes in the result and lamp values do not jump or jitter. This is important for the transfer of slow varying environment maps because the created illumination should behave smoothly.

We defined the normalized *optimization error*

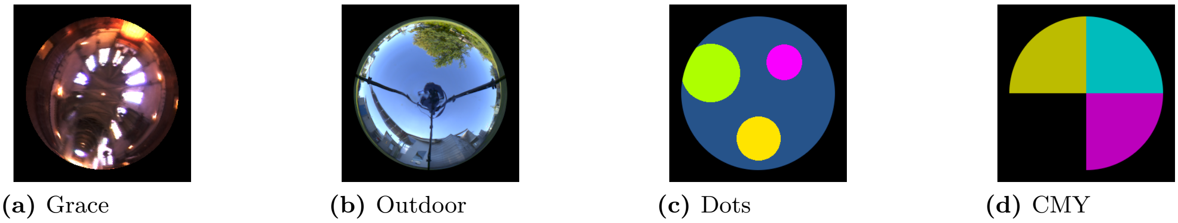
$$e_o = \frac{1}{|D|s^2} ||s\boldsymbol{t} - \sum_{l \in L} w_l \boldsymbol{i}_l||^2 \tag{5.3}$$

in order to be able to compare the minimization error of different setups. This measure is independent from the number of sampling directions and corrected for the scaling factor s.

6 Evaluation

For evaluation we set up our lighting system in a medium sized room. The room was not optimal and had rather bad reflective properties: The whole ceiling was lined with metal ducts and reflectors of the original room lighting which caused strong specular reflections. Several dark-blue shelves absorbed so much light that we had to drape them in white table-cloth. Figure 4.2 shows the placement of the lighting system in the room.

All experiments were conducted on a Linux machine located in the same room. It was equipped with a Core i7 920 CPU running at 2.67 GHz and had 12GB of memory.



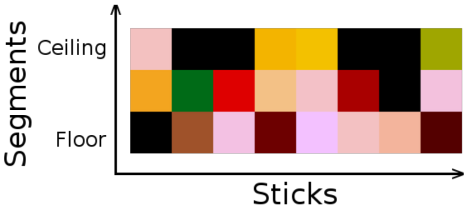
(a) Grace **(b)** Outdoor **(c)** Dots **(d)** CMY

Figure 6.1: The static environment maps we used for the transfer: The Grace HDR light probe from Debevec (a), an outdoor scene (b) captured with our Canon light probe and two hand-drawn extreme cases (c), (d).

We tested our system with the four static environment maps shown in figure 6.1. The first one (a) is a High Dynamic Range light probe taken inside the *Grace* cathedral by Debevec [PRO]. The second one (b) was captured with our Canon light probe and shows a sun-lit *Outdoor* scene without direct illumination. We also include two hand-drawn targets *Dots* (c) and *CMY* (d) as extreme cases to demonstrate the limits of our omnidirectional lighting system. Additionally, we recorded a dynamic environment map (88 frames) with the Webcam Probe by moving it through a colorful scene. Five frames are shown in table 7.2.

To measure the quality of our results we took light probe images of the produced illumination with our Canon Probe and compared them to the targets. We calculate the *result error* e_r by taking the sum of squared differences over all color channels of the available pixels P and divide it by $3|P|$. Masked pixels and pixels outside our sampling range are ignored. The *difference error* e_d is calculated the same way, but is used to compare two results.

We will show the values of our lamps in form of a small image as follows:



We split the evaluation in four parts: First, we measure how our the choice of the *segment sizes* (and thus the *number of lamps* $|L|$) influences the result error e_r. Second, we show how our nearest-neighbor approach performs for different *number of sampling directions* $|D|$ and compare it to the limit case where the neighborhood consist of a single pixel. Third, we take a look at the run-time of the optimizer and show how it is linked to the *number of lamps* and *number of sampling directions*. Fourth, we show how the *target scaling s* affects the quality of the results. Based on this analysis, we select a lamp and sampling configuration that produces good results with real-time speeds.

6.1 Segment Size

We evaluated segment sizes of 10/20/40/60/120 LEDs, which gave us 96/48/24/16/8 RGB Lamps respectively, to see if smaller segments produce better results. We also included *uniform lighting* (1 lamp) where all LEDs display the same color. Note that in the case of 8 lamps the walls and ceiling cannot be illuminated independently because each stick acts as a single lamp.

Our assumption was that increasing the number of lamps increases the quality of the result, but only up to some point because our lighting system has a limited spatial resolution.

|L|	Outdoor	Grace	Dots	CMY	Video (avg.)
1	0.052	0.086	0.118	0.088	0.075
8	0.048	0.082	0.101	0.063	0.067
16	0.047	0.078	0.101	0.063	0.063
24	0.049	0.075	0.097	0.060	0.061
48	0.049	0.075	0.097	0.059	0.061
96	0.047	0.076	0.099	0.062	0.062

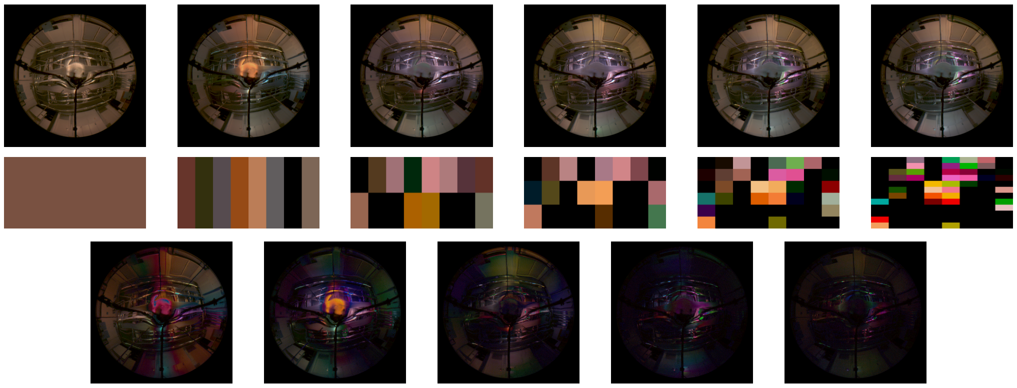
Table 6.1: Result error e_r for several lamp configurations. No downsampling was performed.

Table 6.1 shows the result error e_r for our six different lamp configurations and all five target environment maps. For the video environment map, we selected 9 frames (spaced 10 frames apart) and averaged the error. We did not perform any downsampling and used all available pixels in order to be independent from sampling artifacts.

We recorded the calibration data only once for the smallest segment size. All other configurations are derived by adding up the radiance to form larger segments. This avoids variations in the calibration data and keeps the results comparable.

The table clearly shows that a spatial illumination with multiple lamps always performs better than uniform lighting. It also reveals that a setup with 48 or 96 lamps produces only a marginal improvement over 24 lamps.

$\|L\| = 1$	$\|L\| = 8$	$\|L\| = 16$	$\|L\| = 24$	$\|L\| = 48$	$\|L\| = 96$



12.82e-05	12.76e-05	8.96e-05	3.14e-05	3.20e-05

Table 6.2: Results and lamp values for the Grace target created with different lamp configurations. Second row: absolute difference between two successive results. Third row: the difference error e_d between two successive results.

The results of the Grace target are shown in more detail in table 6.2. The top row shows the illumination of the room and the one below shows the absolute difference between two successive results. The last row contains the error e_d that corresponds to the difference images above. It is calculated exactly like e_r. We can see that a configuration with 24, 48 and 96 lamps produces almost the exact same global result. In the case of uniform lighting (1 lamp) or individually controlled sticks (8 lamps) the LEDs facing the ceiling cannot be controlled independent from those facing the walls. Because our method searches for a global solution, the ceiling it lit too bright and the walls are missing light.

If we increase the number of lamps we introduce ambiguity: Two successive lamps on a stick produce nearly the same illumination. This leads to a separation of the light color and our lamps begin to emit pure light. This does not affect the global impression, but it causes colorful highlights. The pure colors are reflected by specular objects and stand out between the uniformly lit areas. Our optimization does not "see" them because they are very small in size and get overpowered by large uniform areas when searching for a global solution.

This is another reason why we think 24 lamps is the best choice for our lighting system.

6.2 Sampling

We tried our nearest-neighbor sampling with different number of sampling directions $|D|$. For comparison, we included the limit case where a neighborhood consists of a single pixel Table 6.3 show the result error e_r for several targets. The target scaling was $s = 1.0$ in all cases.

|D|	Grace	Outdoor	Dots	CMY	Video (avg.)
33	0.070 0	0.057 2	0.091 8	0.048 8	0.056 9
63	0.072 5	0.056 4	0.093 0	0.049 9	0.057 4
128	0.070 7	0.054 9	0.092 5	0.051 4	0.058 3
291	0.072 2	0.050 5	0.094 6	0.055 1	0.060 7
449	0.073 2	0.051 3	0.094 6	0.054 9	0.060 3
all	0.075 1	0.049 1	0.097 5	0.059 6	0.061 5

Table 6.3: Result error e_r for different sampling configurations, including the limit case containing all available pixels. The lighting system was configured with 24 lamps.

We can see the regularization effect of our sampling: Because we average multiple directions, shadows and highlights are removed. The downsampled environment map does not contain as much extreme values as the original. The variance is reduced and the optimizer can find a better solution. This causes the result error to decrease with an increased neighborhood size.

We decided that a sampling with 128 direction is perfectly adequate for our system.

6.3 Run-time

The run-time of our method depends on the number of lamps $|L|$ and the number of sampling directions $|D|$ used. The following table shows the duration of the optimization process in milliseconds.

|L|	33	63	128	291	449	All
1	3.20	3.04	3.50	3.49	2.96	3.73
24	9.99	10.13	10.09	9.82	8.80	23.88
48	40.09	34.92	35.80	36.44	32.79	60.66
96	296.08	235.58	252.13	219.49	216.68	263.29

Table 6.4: Optimization run-time for the Outdoor target in milliseconds. The other targets showed the same performance.

The run-time is largely influenced by the number of lamps $|L|$ and the time complexity of our method lies in $O(|L|^2)$. Increasing the number sampling directions $|D|$ does not increase the run-time because the QP solver benefits from an overdetermined system.

We can see that the configuration with 24 lamps is fast enough for real-time ambient light transfer.

6.4 Scaling

The target scaling factor s influences the brightness of the results. It defines the relation between the exposure of the Capture Probe and the exposure of the Calibration Probe. The target environment map must be scaled in such a way, that its values are reachable by the optimization. If they are too large, our lamps cannot reproduce the desired brightness. As soon as the lamps reach their maximum intensity, the color can change.

The following tables 6.5 and 6.6 show the influence of the scaling factor on the result.

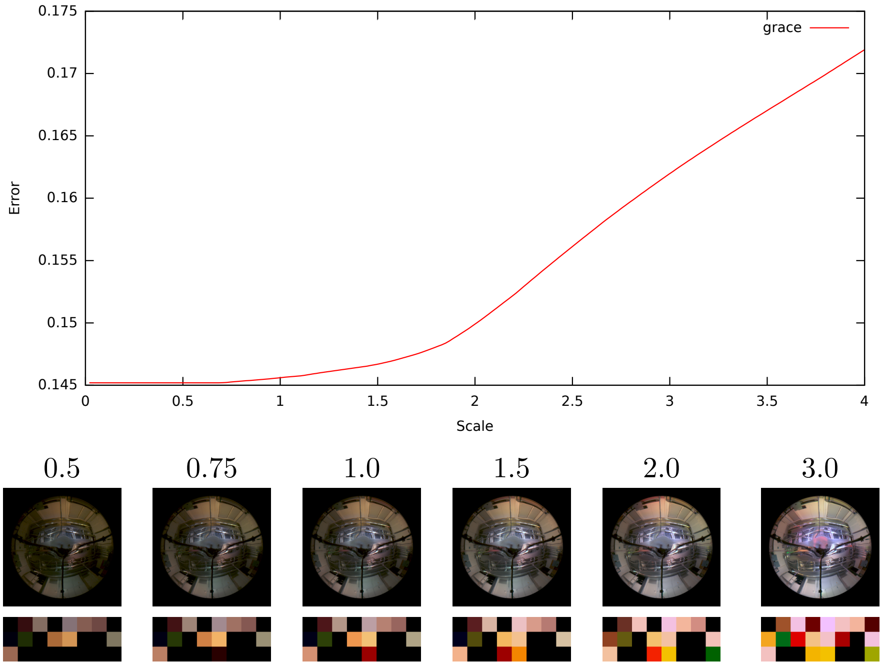


Table 6.5: Optimization error e_o, results and lamp values for different scale factors s using the Grace target.

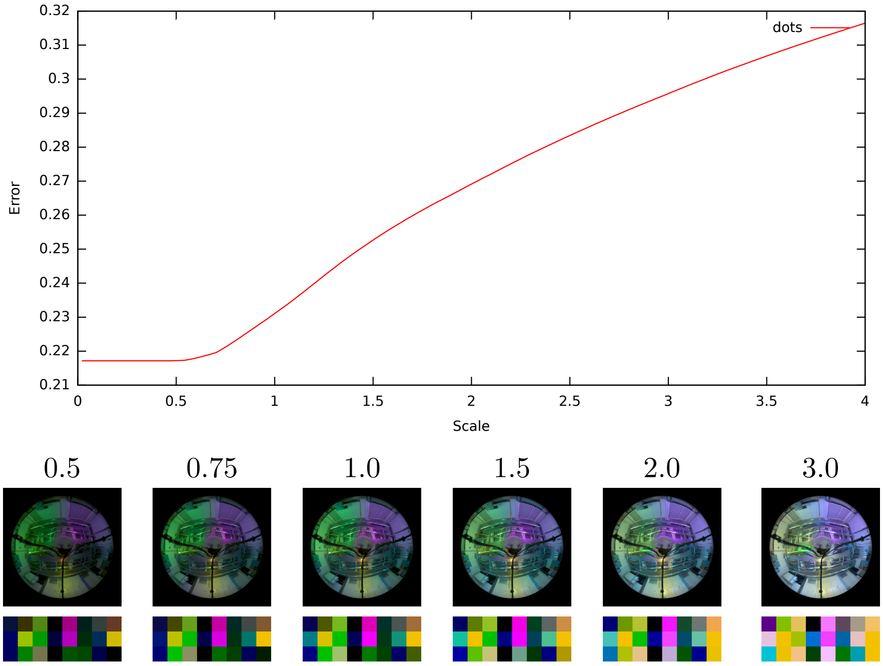


Table 6.6: Optimization error e_o, results and lamp values for different scale factors s using the Dots target.

Both plots show how the optimization error e_o varies with the scaling factor s. The curve can be divided in three segments with different characteristics: For small values s, the error is constant. A minimal number of lamps are turned on and their brightness increases in a linear fashion. If we increase s, the optimizer turns on more lamps and the error rises continuously. For large values we can observe a shift in the color of the result. Some color channels have reached their maximum intensity and our system is no longer able to reproduce ambient light with the desired brightness..

7 Results

This chapter we present our final results. We configured the lighting system with 24 Lamps and used 128 sampling directions and used a target scaling factor $s = 1.0$. The setup was exactly the same as described in the evaluation section 6.

7.1 Static Scenes

We transferred four static environment maps, shown in table 7.1. The first row shows the value of the lamps, the run-time Δt in milliseconds, the optimization error e_o and the result error e_r. The second row displays the *target*, which is the ambient light we want to reproduce, and the third row shows the result our method produces. The fourth row shows the *simulated* result which we get by evaluating the linear combination of the *raw* light probe images we acquired in the calibration step. The last two rows show the *downsampled* light probe image of the target (fifth row) and of the simulated result (last row).

We can clearly see the effectiveness of a spatial illumination. The global impression of the Grace environment map is reproduced faithfully: The ceiling is lit brightly with a white color and the walls are lit in a red color with lower intensity. The Outdoor scene is mostly blue due to the large area of the sky, but shows a tendency towards a green color at the top, and a gray color at the bottom. We can see that the dark corner in the CMY scene is not reproduced correctly due to the inter-reflections of the light inside the room. In the results of the Dots scene, we can see that the blue background color near the green and magenta dots is reproduced correctly, but becomes more white near the yellow dot. Our lighting system is not able to reproduce the high spatial resolution present in this scene.

If we compare the simulated result with the result, we can see the limitations of our lighting system: The red channel of the results is missing light due to the interdependency between the color channels we explained in subsection 4.1.1.

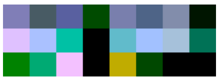
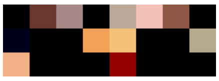
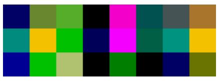
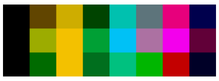
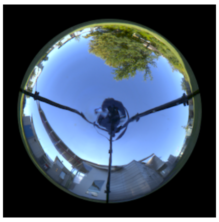
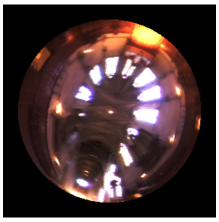
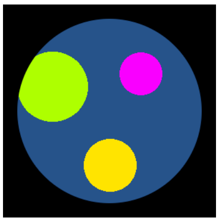
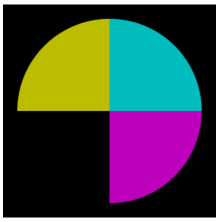
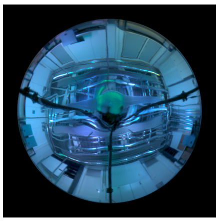
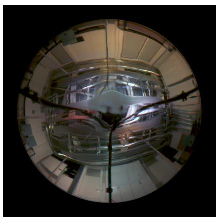
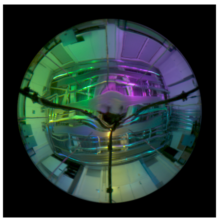
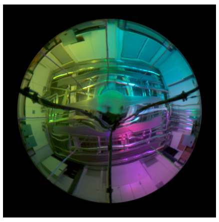
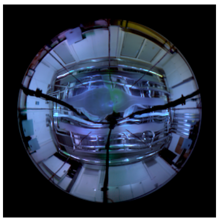
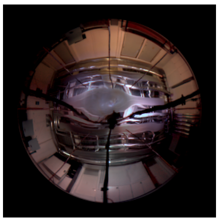
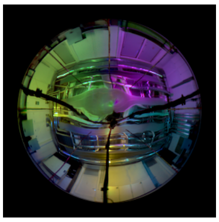
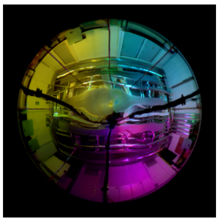
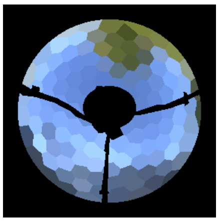
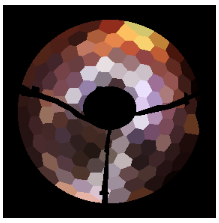
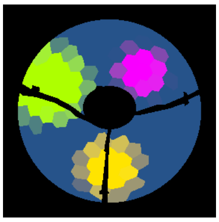
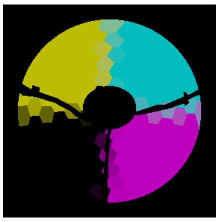
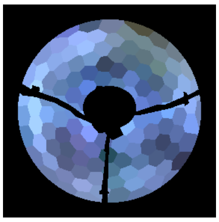
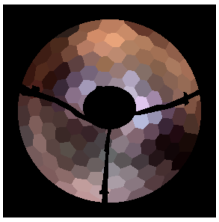
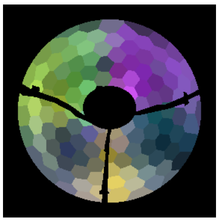
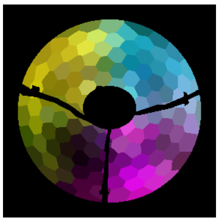
	Outdoor $e_r = 0.055$ $e_o = 0.254$ $\Delta t = 7.897\text{ms}$	Grace $e_r = 0.070$ $e_o = 0.077$ $\Delta t = 7.813\text{ms}$	Dots $e_r = 0.092$ $e_o = 0.181$ $\Delta t = 7.871\text{ms}$	CMY $e_r = 0.051$ $e_o = 0.063$ $\Delta t = 7.045\text{ms}$
				
Target				
Result				
Simulation				
Target (sampled)				
Simulation (sampled)				

Table 7.1: Resulting illumination and the result error e_r for static targets (24 lamps, 128 sampling directions, scaling $s = 1.0$).

7.2 Dynamic Scene

We captured a dynamic environment map with our Webcam Probe and observed the behavior of our optimizer over time. The table 7.2 shows the result for five frames. We also plotted the optimization error e_o and the duration of the optimization over all 88 frames, shown in figure 7.1. The results are smooth: the values of the lamps change gradually from frame to frame and do not jump or jitter. The run-time is almost constant for all frames. The calculation of the first frame takes longer because the CPU cache has not yet been filled.

We are now confident that our method can be used for real-time Ambient Light Transfer.

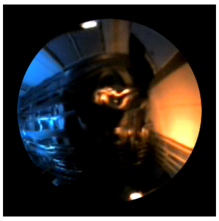
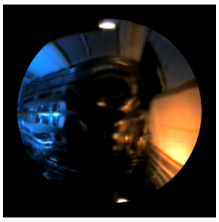
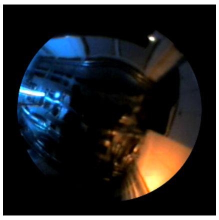
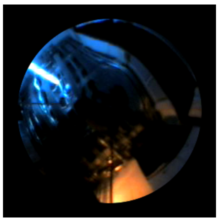
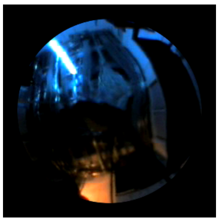
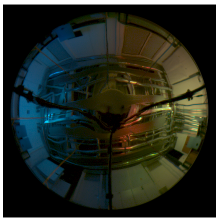
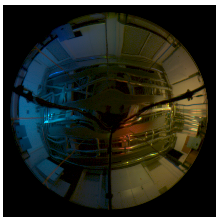
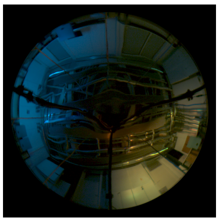
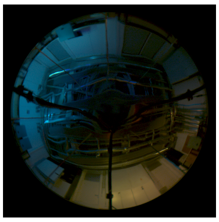
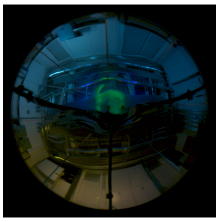
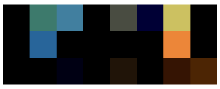
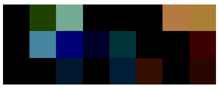
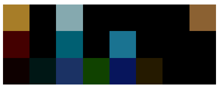
Frame 0	Frame 22	Frame 44	Frame 66	Frame 88
		Target		
				
		Result		
				
				
$e_r = 0.0745$	$e_r = 0.0714$	$e_r = 0.0510$	$e_r = 0.0434$	$e_r = 0.0536$

Table 7.2: Results for 5 frames of the dynamic environment map. The first row shows the target and the second row the result.

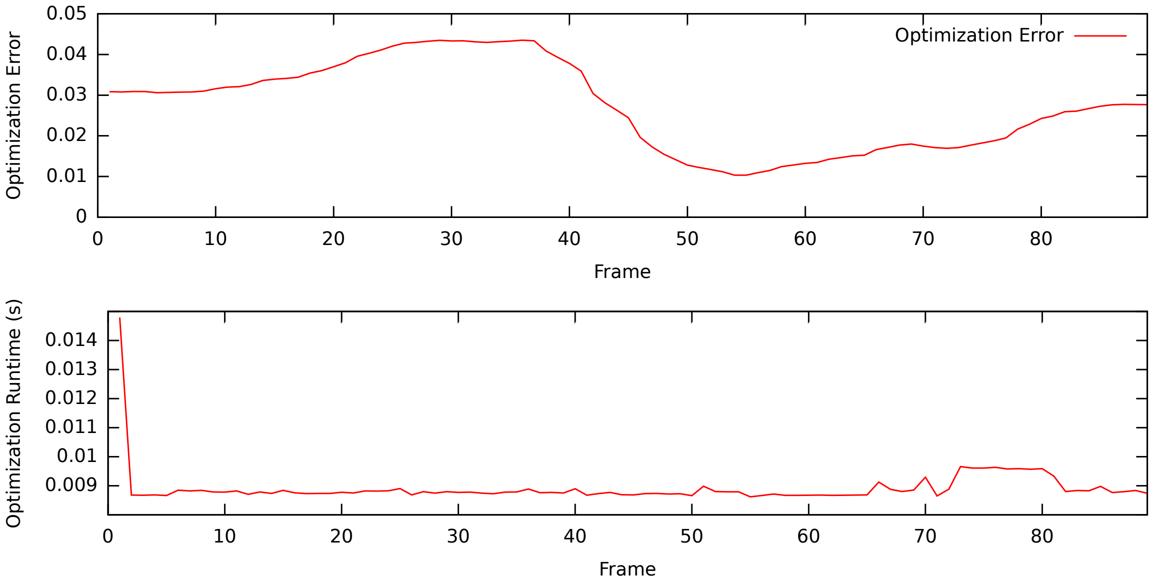


Figure 7.1: Optimization error e_o and run-time for all 88 frames of the dynamic environment map.

8 Future Work

We have shown that our lighting system can illuminate a room omnidirectional and reproduce the global impression of environment maps. We presumed our LEDs are linear in the input data and mentioned in section 4.1.1 that this is not the case. The color channels showed an interdependency and the output of the red channel depends on the output of the blue and green channel. An additional calibration could alleviate this effect. We could for example measure the light color of the LEDs for all possible combinations of input data and derive a model that linearizes the response.

We have shown our method works with indirect lighting and we can imagine an extension to direct illumination. This not only requires a lighting system capable of direct illumination, for example a room-sized light stage [DWT+02], but also further investigation regarding the light probe. If we want to capture direct light in addition to the indirect light, we must use a camera with a much higher dynamic range. For the Calibration Probe, we can take multiple images with different exposures and reconstruct an image with a high dynamic range (HDR) [DM08]. For the Capture Probe, which has to record in real-time, we could use an expensive HDR video camera. A simpler solution was proposed by Waese [Wae09] and uses a special prism with shaded surfaces to record multiple exposures of the light probe object with a single image.

9 Conclusion

In this paper we have presented a system that is able to capture an environment map in one place and recreate it in a room with controllable lamps. We have shown that our method is fast enough for real-time transfer and reproduces dynamic environment maps reliably. Our approach is general and works independently from the layout of the room and placement of the lighting system. We believe LEDs are the future of room lighting and expect to see a rise in the application of digital computer-controlled illuminants.

A Appendix

Part	Size	Amount	Price per piece
Aluminum L-profile 25x25x3 mm	200 cm	8	8.96 Eur
Aluminum L-profile 25x25x2 mm	75 cm	8	2.61 Eur
Aluminum L-profile 15x10x2 mm	40 cm	8	1.39 Eur
Baseplate, 22mm pressboard	45x45 cm	8	3.75 Eur
Strip of pinewood 13x10 mm	40 cm	8	1.50 Eur
Strip of pinewood 13x10 mm	50 cm	8	1.50 Eur
Steel bracket	10x10 cm	8	2.00 Eur
Misc. nuts and bolts	-	-	45 Eur
Powersupply 5V, 30A	-	4	115 Eur
WS2812 LED-Strip 60 LEDs/m	4 m	8	45 Eur
Teensy 3.0 [TEE]	-	1	19 Eur
Copper wire 6 mm^2	4.5 m	8	4.50 Eur
Copper wire 1.5 mm^2	4.5 m	8	1.30 Eur
Misc. cable terminals (55 pcs.)	-	8	4.11 Eur
		total	1134.56 Eur

Table A.1: The bill of material for our lighting system.

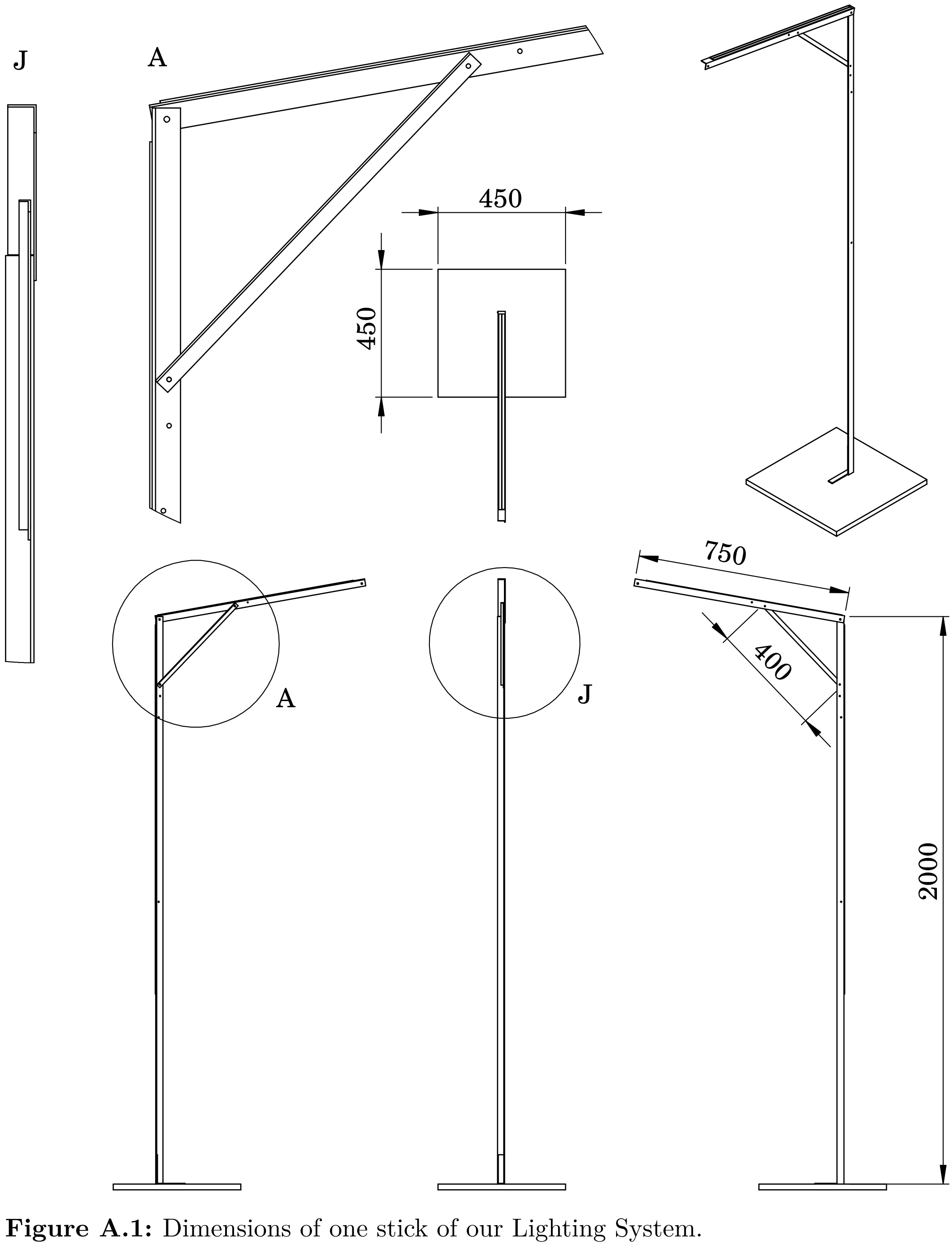


Figure A.1: Dimensions of one stick of our Lighting System.

Bibliography

[ADW04] F. Anrys, P. Dutré, Y. Willems. Image based lighting design. In *the 4th IASTED International Conference on Visualization, Imaging, and Image Processing*. 2004. (Cited on page 15)

[Beg05] P. D. I. S. Begemann. A scientific study on the effects of Ambilight in flat-panel displays. web, 2005. URL http://www.embedded.com/design/embedded/4012996/1/A-scientific-study-on-the-effects-of-Ambilight-in-flat-panel-displays. (Cited on page 16)

[BN76] J. F. Blinn, M. E. Newell. Texture and reflection in computer generated images. *Communications of the ACM*, 19(10):542–547, 1976. (Cited on page 11)

[CCH07] B. Choudhury, S. Chandran, J. Herder. A survey of image-based relighting techniques. *Journal of Virtual Reality and Broadcasting*, 4(7), 2007. (Cited on pages 12 and 13)

[CVX] URL http://cvxopt.org/. (Cited on page 28)

[Deb98] P. Debevec. Rendering synthetic objects into real scenes: Birging traditional and image-based graphics with global illumiation and high dynamic reage photograph. In *SIGGRAPH'98 Conf. Proc.*, pp. 189–198. 1998. (Cited on page 11)

[DM08] P. E. Debevec, J. Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes*, p. 31. ACM, 2008. (Cited on page 41)

[DWT+02] P. Debevec, A. Wenger, C. Tchou, A. Gardner, J. Waese, T. Hawkins. *A lighting reproduction approach to live-action compositing*, volume 21. ACM, 2002. (Cited on pages 13 and 41)

[EGK+13] E. V. Ellis, E. W. Gonzalez, D. A. Kratzer, D. L. McEachron, G. Yeutter. Autotuning Daylight with LEDs: Sustainable Lighting for Health and Wellbeing. *THE VISIBILITY OF RESEARCH*, p. 465, 2013. (Cited on pages 9 and 10)

[FBS05] M. Fuchs, V. Blanz, H.-P. Seidel. Bayesian relighting. In *Proceedings of the Sixteenth Eurographics conference on Rendering Techniques*, pp. 157–164. Eurographics Association, 2005. (Cited on page 15)

[FM99] J. Fliege, U. Maier. The distribution of points on the sphere and corresponding cubature formulae. *IMA Journal of Numerical Analysis*, 19(2):317–334, 1999. URL http://www.mathematik.uni-dortmund.de/lsx/research/projects/fliege/nodes/nodes.html. (Cited on page 26)

[GTSH05] A. Ghosh, M. Trentacoste, H. Seetzen, W. Heidrich. Real illumination from virtual environments. In *ACM SIGGRAPH 2005 Sketches*, p. 41. ACM, 2005. (Cited on pages 15 and 17)

[GWN$^+$03] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, et al. blue-c: a spatially immersive display and 3D video portal for telepresence. In *ACM Transactions on Graphics (TOG)*, volume 22, pp. 819–827. ACM, 2003. (Cited on page 7)

[Hae] P. Haeberli. URL http://www.graficaobscura.com/synth/index.html. (Cited on page 12)

[HUE] URL http://www.meethue.com/. (Cited on page 16)

[JBOW13] B. R. Jones, H. Benko, E. Ofek, A. D. Wilson. IllumiRoom: peripheral projected illusions for interactive experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 869–878. ACM, 2013. (Cited on page 16)

[MLP04] W. Matusik, M. Loper, H. Pfister. Progressively-Refined Reflectance Functions from natural Illumination. In *Rendering Techniques*, pp. 299–308. 2004. (Cited on page 15)

[MSP02] S. Muthu, F. J. Schuurmans, M. D. Pashley. Red, green, and blue LEDs for white light illumination. *Selected Topics in Quantum Electronics, IEEE Journal of*, 8(2):333–338, 2002. (Cited on page 10)

[OCT] URL http://www.pjrc.com/teensy/td_libs_OctoWS2811.html. (Cited on page 21)

[PFS] URL http://www.mpi-inf.mpg.de/resources/hdr/calibration/pfs.html. (Cited on page 24)

[PRO] URL http://www.pauldebevec.com/Probes/. (Cited on page 31)

[RBS99] M. A. Robertson, S. Borman, R. L. Stevenson. Dynamic range improvement through multiple exposures. In *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*, volume 3, pp. 159–163. IEEE, 1999. (Cited on page 24)

[SK97] E. B. Saff, A. B. Kuijlaars. Distributing many points on a sphere. *The Mathematical Intelligencer*, 19(1):5–11, 1997. (Cited on page 26)

[TEE] URL http://www.pjrc.com/store/teensy3.html. (Cited on pages 20 and 45)

[Wae09] J. Waese. A real time high dynamic range light probe. *People*, 2010, 2009. (Cited on page 41)

[WS2] URL http://www.world-semi.com/en/. (Cited on page 19)

All links were last followed on Sep 27, 2013.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift