DISSERTATION

Usability of Digital Cameras for Verifying Physically Based Rendering Systems

Ausgeführt zum Zwecke der Erlangung des akademischen Grades
einer Doktorin der technischen Wissenschaften

unter der Leitung von
Univ.Prof. Dipl.–Ing. Dr.techn. Werner Purgathofer
E186 Institut für Computergraphik und Algorithmen

eingereicht an der Technischen Universität Wien
Fakultät für Informatik

von
DI Christiane Ulbricht
Matrikelnummer: 9625056
Friedensgasse 35
A-2201, Gerasdorf bei Wien

Wien, am 27. November 2006

Abstract

Within computer graphics, the field of *physically based rendering* is concerned with those methods of image synthesis which yield results that do not only *look* real, but are also *radiometrically correct* renditions of nature, i.e. which are accurate predictions of what a real scene would look like under given lighting conditions.

In order to guarantee the correctness of the results, three stages of such a rendering system have to be verified with particular care: the light reflection models, the light transport simulation, and the perceptually based calculations used at display time.

In this thesis, the focus lies on the second step in this chain. Various approaches for verifying the implementation of a physically based rendering system have been proposed so far. However, the problem of proving that the results are correct is not fully solved yet, and no standardized methodology is available.

Using a photograph for verifying the results of a rendering system seems obvious but an image produced by a common digital camera cannot be used for this purpose directly. The sensor of a digital camera usually sees colors differently than a human observer. Several techniques have been developed to compensate for this problem. Our goal was to find and compare as many meaningful ways of using a digital camera for verifying a physically based rendering system as possible, in order to provide a practicable method for any development environment. Some of the analyzed methods were taken from the field of color management. Another method, that is based on a novel approach, was developed throughout this thesis. We did an exhaustive comparison of the usability and practicability of all the methods, focusing on required equipment and cost. We found that in general more elaborate methods give better results than low-end methods.

As some of the methods are based on XYZ color space, we considered using this space as internal color space of our rendering system, rather than doing full spectral rendering. However, we found a severe problem in using XYZ space to determine the result of interactions of light and matter, as XYZ space is not closed to component-wise multiplication of XYZ triplets. Thus, based on this analysis we recommend full spectral rendering.

This thesis also contains a comprehensive overview of related work in the field of verification of physically based rendering systems.

Kurzfassung

Physically Based Rendering, ein Teilgebiet der Computergraphik, beschäftigt sich mit jenen Methoden der Bildsynthese, deren Ziel es ist, Bilder zu berechnen, die nicht nur real *aussehen*, sondern auch eine *radiometrisch korrekte* Wiedergabe der Natur darstellen. Bilder dieser Art entsprechen einer exakten Vorhersage über das Aussehen einer Szene unter definierten Lichtverhältnissen.

Folgende drei Stufen des Renderingverfahrens müssen verifiziert werden, um die Korrektheit des Ergebnisses garantieren zu können: die Oberflächenmodelle, die Simulation der Lichtausbreitung in der Szene und die Darstellung am Bildschirm, deren Konzepte der Wahrnehmungsfähigkeit des Menschen nachempfunden sind.

Der Fokus dieser Dissertation liegt auf der zweiten der drei Stufen. In den letzten Jahrzehnten wurden viele verschiedene Ansätze präsentiert, die sich mit der Verifikation der Implementierung eines physikalisch basierten Renderingsystems befassen. Die Problemstellung ist allerdings noch nicht völlig gelöst. Es ist auch noch keine standardisierte Methode vorhanden.

Es erscheint naheliegend, zur Verifikation eines Renderingsystems ein Photo einer Szene heranzuziehen. Ein Photo einer handelsüblichen Digitalkamera kann allerdings nicht direkt zu diesem Zweck verwendet werden, da die Sensoren einer solchen Kamera Farben im allgemeinen anders wahrnehmen als ein Mensch. Es wurden veschiedenste Strategien entwickelt, um diese Abweichung zu kompensieren. Ziel dieser Dissertation ist es, möglichst viele aussagekräftige Methoden, die es erlauben, eine handelsübliche Digitalkamera zur Verifikation eines physikalisch basierten Renderingsystems zu benutzen, zu finden und zu vergleichen. Der Schwerpunkt wurde dabei auf die praktische Anwendbarkeit in unterschiedlichen Entwicklungsumgebungen gelegt. Einige der besprochenen Methoden stammen aus dem Gebiet des Color Management. Im Rahmen dieser Dissertation wurde eine weitere Methode entwickelt, die auf einem neuartigen Ansatz basiert. Die praktische Anwendbarkeit all dieser Methoden wurde verglichen, wobei besonderes Augenmerk auf benötigte Geräte und anfallende Kosten gelegt wurde. Es hat sich herausgestellt, daß die aufwendigeren Verfahren bessere Ergebnisse liefern als einfachere Methoden.

Da einige dieser Methoden auf der Verwendung des XYZ Farbraums basieren, liegt es nahe, diesen als internen Farbraum des Renderers zu verwenden, anstatt auf das aufwendigere Spektralrendering zurückzugreifen.

Im Zuge dieser Dissertation sind wir jedoch auf ein Problem bei der Berechnung der Interaktion von Licht mit einer Oberfläche gestoßen. Der XYZ Farbraum ist bezüglich der komponentenweisen Multiplikation nicht abgeschlossen, es kann daher zu ungültigen Werten kommen. Wir empfehlen deshalb die Verwendung von Spektren für die internen Berechnungen eines Renderingsystems.

Diese Dissertation beinhaltet außerdem einen ausführlichen Überblick über den Stand der Forschung auf dem Gebiet der Verifikation von physikalisch basierten Renderingsystemen.

Acknowledgements

I want to thank Werner Purgathofer for giving me the opportunity to write my PhD thesis, and for his advice and invaluable support.

I owe thanks to my mother for always encouraging me, and to my sister for her helpful comments.

Finally, I would like to thank my boyfriend for always being there for me.

Contents

List of Figures

List of Tables

1 Introduction

In recent years, a lot of work has been published in the field of photorealistic computer graphics concerning more accurate rendering algorithms, more detailed descriptions of surfaces, more realistic tone mapping algorithms and many other improvements for the process of rendering photorealistic images. Unfortunately, there have been comparatively few attempts on verifying all these algorithms in practice. In the field of photorealistic rendering, it should be common practice to compare a rendered image to measurements obtained from a real-world scene. This is currently the only practicable way to prove the correctness of the implementation of a true global illumination algorithm. However, most rendering algorithms are just verified by visual inspection of their results.

The process of verifying a photorealistic renderer can be divided into three steps [GTS+97]. First, one has to prove the correctness of the light reflection models through comparisons with measured physical experiments. One way to do this is to use a gonioreflectometer to measure the full bidirectional reflectance distribution functions (BRDFs) of all surfaces. The second step is to verify the light transport simulation, or – in other words – the actual rendering algorithm. This can be done by comparing the rendered image to a measurement of a real scene, or through analytic approaches. The final step in image synthesis generates an image from radiometric data provided by the rendering algorithm; it has to take the properties of the output device and the actual viewing conditions into account. In this stage psychophysical models are used to achieve convincing results, therefore perceptual experiments are used to evaluate how believable such an image is.

This thesis will focus on the second step: the verification of rendering algorithms. Although this step is crucial in photorealistic image synthesis, comparatively little work has been published about this topic so far. The problem is far from being solved, though. Most of the available rendering systems are not verified.

1.1 Motivation

The goal of photorealistic rendering is to generate images that look like photographs of a real scene. To verify such a rendering system, the obvious way is to compare a photograph to a synthetic image of the same scene. This thesis focuses on how a digital camera can be used in this sense and – probably even more important – what subproblems cannot be solved and therefore have to be treated differently.

1.2 Believable vs. Correct

Photorealistic image synthesis has made outstanding progress in the last decades. While the pioneers of this field tried to achieve realism with comparatively simple algorithms like raytracing and radiosity, a lot of sophisticated algorithms – such as path or photon tracing, as well as numerous hybrid techniques – have been developed since then.

Nowadays it is possible to render images that can hardly be distinguished from photographs. Consider for instance the Alias website "Is it Fake or Foto" [Ali06b]. There you can attempt to guess the origin of ten images depicting various scenes, some of them real and some of them rendered with Maya [Ali06a]. If you know what you have to look for, it is indeed possible to choose the right answers, but at first sight the renderings look believably real.

The results are convincing, but we cannot assume that the rendering algorithm is implemented correctly as far as physical accuracy is concerned. However, for many applications this is good enough. For movies, computer games and related areas it is essential that the images look appealing and – above all – believable. It is not necessary and sometimes even counterproductive that an image is a physically correct representation of the light transport in a scene.

However, for *physically based rendering* (as the field has become to be known), this is absolutely crucial. Its applications are much more restricted than the largely artistic realm of ordinary, "just believable" rendering techniques; the effort involved is far greater – e.g. because one has to use measured surface reflectance data or complicated analytical models for the scene description – and the creativity of the user is severely restricted by the constraints of physical correctness (e.g. the inability to fine-tune the lighting in the scene for artistic reasons on a per-object basis as it is common practice with current commercial systems).

1.3 Applications of Physically Based Rendering

Even though it is not to be expected that physically based rendering will ever replace the existing artistic image synthesis tools for the abovementioned reasons (although methods from this field will probably continue to make inroads as optional enhancements, such as selective raytracing of reflective objects), it still commands a small but highly viable niche market for all those who need accurate imagery of virtual scenes.

1.3.1 Virtual Prototyping

Many appearance-sensitive branches of industry, for example the car industry, have to build expensive prototypes of their products before they actually go into serial production. In the last years, a lot of effort was put into reducing the costs by using computer generated images of the products. But this can only be done if the rendering software is proven to accurately simulate the light propagation within a scene.

The "RealReflect" project [Rea06] was initiated in order to investigate and improve the process of measuring surface reflectance properties and generating high quality renderings out of the data. However, these renderings are of reduced value if it cannot be proven that the implementation of the rendering algorithm is accurate.

1.3.2 Architecture and Lighting Design

Architects and lighting designers are interested in a reliable simulation of the lighting conditions of a building. The more precise this lighting simulation is, the more accurate is the output of the design process. Beside artificial light, many natural lighting situations have to be considered, like for example direct sunlight effects, ambient skylight effects, different time conditions, and different weather conditions. This kind of calculations requires a validated physically based rendering system.

1.3.3 Medical Reconstruction

In single photon emission computed tomography (SPECT), a gamma camera is rotated around the patient to acquire images from multiple angles. Various reconstruction algorithms have been developed to implement the process of reconstructing a 3-dimensional object from these images. Bergner et al. [BDCM05] used a physically based rendering toolkit to simplify the programming of tomography algorithms in order to speed up the development and testing processes of novel algorithms. As the quality of the diagnosis relies heavily on the accuracy of the reconstruction algorithm, the correctness of the rendering toolkit again has to be guaranteed.

1.3.4 Safety Checks

Physically based rendering is also useful when it comes to safety relevant applications. In car industry, for example, the design of a car interior has to fulfill certain safety relevant issues. One of these issues is e.g. that the reflections in the windshield are to be minimized in a way that they do not distract the driver. Under daylight conditions these reflections are mainly caused by glossy materials in the upper part of a cockpit mirroring in the windshield. Under night light conditions, ambient light sources or illuminated switches can cause a similar effect. A realistic rendering of the car interior helps the designer to select suitable materials and to position the light sources in an appropriate way.

1.4 Thesis Contribution

The main contribution of this thesis is to find as many meaningful ways of using a digital camera for verifying a physically based rendering system as possible in order to provide a practicable method for any development environment. The methods were taken from the field of color management. Additionally another method was developed as part of this thesis. We compared the usability and practicability of all the methods, focusing on required equipment and cost. We also investigated the factor of improvement that methods requiring expensive equipment bring. Using a photograph for verifying the results of a rendering system seems obvious and some developers may have started to use it already, but as far as we know no exhaustive analysis apart from our work has been published, yet.

In our rendering system, we use spectral values for internal calculations. Some of the verification methods are based on XYZ color space though, which raises the question why we do not use XYZ as internal color space from the first. The reason is that we found a severe problem that can occur when interactions of light and matter are calculated in XYZ space. The details of this work are explained in [UW06].

Verification in general is quite often neglected in the development process. We composed an extensive state of the art report on the topic of "Verification of Physically Based Rendering Algorithms" (see [UWP05] and [UWP06]) in order to call attention to the importance of this field.

1.5 Thesis Outline

This thesis is organized as follows: Chapter 2 presents an overview on verification techniques, from the well-known Cornell Box to recent developments. A short introduction to the field of colorimetry is given in chapter 3. Chapters 4 and 5 describe the internal functionalities of a digital camera and a physically based rendering system as far as they are used in this thesis, respectively. The procedure of data acquisition and processing is outlined in chapter 6. Chapter 7 is dedicated to the main topic of this thesis – various approaches of using a digital camera for verifying a physically based rendering system. The question of why we rely on spectral rendering instead of using XYZ space as internal color space is answered in chapter 8. Conclusions are drawn in chapter 9 along with a discussion of future work. Appendix A explains photometric and radiometric quantities that are used throughout the thesis, whereas appendix B describes various types of measurement devices. Several instruments that were used for data acquisition are illustrated in appendix C. Appendix D gives a number of websites that contain publicly available validation data. Detailed results of the tests made in conjunction with this thesis are listed in appendix E.

2 Related Work

2.1 Visual Comparisons

A practicable way of verifying a renderer is to compare the results of the rendering process to a real scene. The very first approach in this direction was done in 1975 by Phong [Pho75]. He compared a phong-shaded sphere to a photograph of a real sphere. The two images differ apparently but he did not discuss this fact any further. As the Phong model is not a physically based global illumination algorithm, we will not go into detail. Two more representative approaches are discussed in the following sections.

2.1.1 The Original Cornell Box

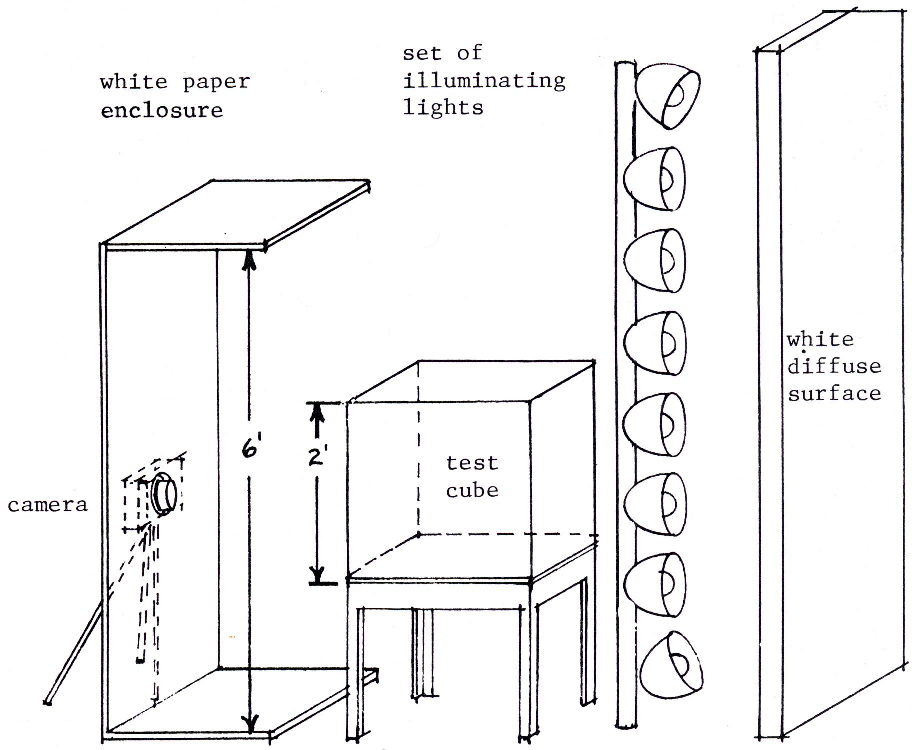


Figure 1: A schematic of the first Cornell Box setup [GTGB84].

The first approach in this direction was done in 1984 by Goral et al. [GTGB84]. They showed that the radiosity algorithm behaves similar to light propagation in a real setup of the scene. One reason for choosing this method was that the rendering

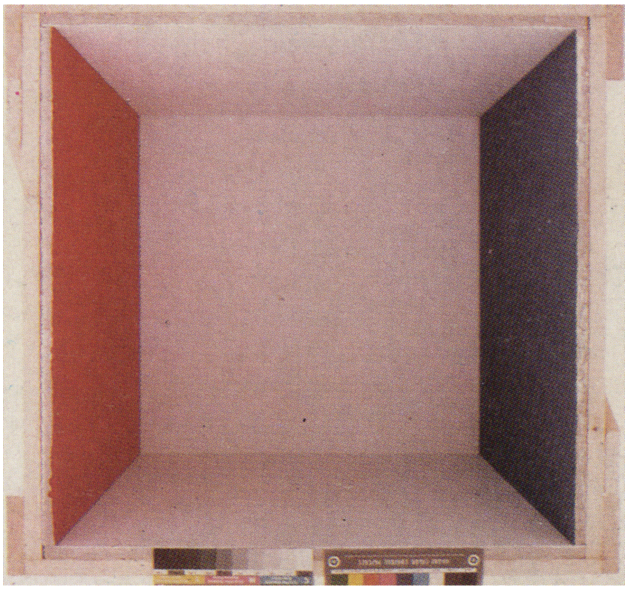


Figure 2: A photograph of the real cube [GTGB84].

equation had not been defined in the form known today [Kaj86].

The setup – also known as the Cornell Box – consisted of a cube made out of fiber board panels, colored with flat latex paints to minimize specular reflections. The panels were painted with different colors to evoke the effect of color-bleeding. The cube had just five sides to be able to take photographs of the scene and to illuminate the inside with diffuse white light. A schematic of the setup can be seen in figure 1. In order to simulate a diffuse and uniform area light source at the front side of the cube, the point light sources did not face the cube but a diffuse white surface. In front of the cube, there was another white surface that contained a small hole for the camera. Due to the multiple reflections, the lighting can therefore be considered diffuse.

Due to the lack of color measurement devices, no quantitative but only visual comparisons could be made. Hence, the result of the comparison is very superficial. It can only be said that the color-bleeding that is visible on a photograph of the cube (see figure 2) is also present on a rendered image of the scene (see figure 3). So, the structure of the light distribution can be verified but not the colors. They do not correspond to each other, e.g. the white surfaces of the photograph look much redder than the ones of the simulation. This is probably caused by using light bulbs that emit reddish light.

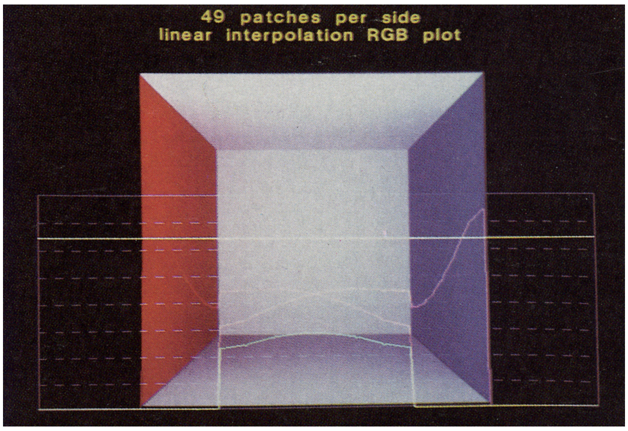


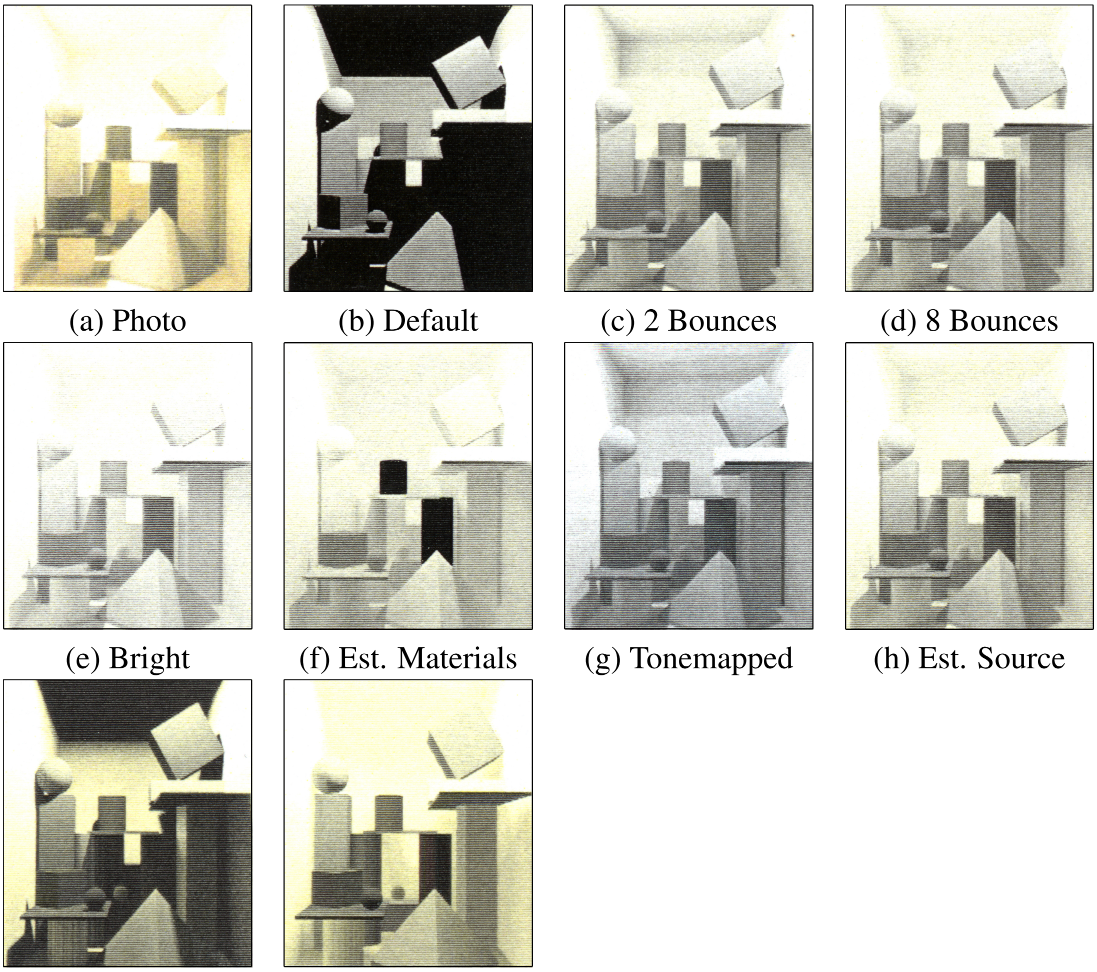
Figure 3: Radiosity with linear interpolation [GTGB84].

2.1.2 Recent developments

In 2000, McNamara et al. [MCTG00] again picked up the idea of visual comparisons. In their paper they present a way to quantify these subjective impressions. Like in the previous attempt, they also built a five sided box as a test environment. The interior of the box was painted with diffuse white paint, whereas the objects that were put inside the box were painted with different kinds of gray. The spectral reflectances of the paints were measured and converted to RGB values. Beside the box a computer monitor was placed to present simulated images of the scene. In addition, a mirror was used to facilitate alternation between the two settings.

Ten different types of images were selected to be compared to the real scene:

- A photograph (see figure 4(a)),
- three images rendered with Radiance [War94] at different quality levels (see figure 4(b), (c) and (d)),
- a brightened high quality Radiance image (see figure 4(e)),
- two Radiance simulations with controlled errors – one with estimated RGB values for the materials and one with estimated RGB values for the light source (see figure 4(f) and (g)),
- a tone-mapped Radiance image (see figure 4 (h)) and
- two images generated with RenderPark [Ren06], one using raytracing, the other using the radiosity algorithm (see figure 4(i) and (j)).



(a) Photo (b) Default (c) 2 Bounces (d) 8 Bounces

(e) Bright (f) Est. Materials (g) Tonemapped (h) Est. Source

(i) Raytraced (j) Radiosity

Figure 4: Different types of images that were used in the experiment [MCTG00].

Eighteen observers were asked to match the gray levels of the presented setting to a predefined set of samples, i.e. to judge the lightness of the objects. For this purpose, they have been trained to do this by recalling the different gray levels from memory before the actual experiment. The 11 different settings – the 10 images and the real scene – were presented in random order. Afterwards, the gray levels chosen by each participant in an image were compared with the values chosen in the real scene. The closer these values were, the closer the image was considered to be to the real scene.

In summary, the results of this study showed that high quality Radiance images (see figure 4(d) and (e)), the tone-mapped image (see figure 4(g)) and even one of the defective images (see figure 4(h)) were good representations of the actual scene. Though, the low quality Radiance simulations (see figure 4(b) and (c)), the estimated materials image (see figure 4(f)), the raytraced image (see figure 4(i)) and the radiosity image (see figure 4(j)) differ considerably from the real setting.

2.2 Experimental Measurements

2.2.1 Improved Cornell Box

One year after the first experiments in Cornell (see section 2.1.1), the radiosity algorithm was improved by projecting onto an imaginary cube instead of a sphere. This hemi-cube radiosity [CG85] was again verified using the Cornell Box approach. Now, radiometric measurements of the cube were taken and compared with the results of the algorithm.

In the context of the development of the hemi-cube radiosity algorithm, the importance of verifying the implementation of a rendering algorithm was recognized. In 1986, Meyer et al. [MRC$^+$86] investigated the procedure of experimental evaluation of rendering systems in detail. The following quotation emphasizes the necessity of experimental verification:

> “If a scientific basis for the generation of images is to be established, it is necessary to conduct experimental verification on both the component steps and the final simulation.” [MRC$^+$86]

The assembly of the Cornell Box had slightly changed since the first experiments had been made. The light bulbs and the diffuse white surfaces were replaced by a light source on top of the cube. A small rectangular opening was cut into the top panel and covered with a piece of opal glass to provide diffuse light. An incandescent flood light was mounted 15 inches above on top of a metal cone whose interior was painted white. In order to avoid interreflections with the surrounding, the box was placed on a black table and the walls were covered with black fabric. Moreover, the panels of the box could be exchanged by other panels with different colors.

In order to render an image that can be compared to the real world scene, the properties of the light source and the surfaces have to be measured. The spectral energy distribution of the light that shone through the opal glass was acquired by a method described by Imhoff [Imh83]. Moreover, the intensity and the direction of the light were needed for the calculations. They were measured by using a photometer (see appendix B.2) combined with an infrared filter. The inaccuracies of the device have been adjusted by a correction factor. Another piece of equipment – a spectrophotometer – was employed to measure the spectral reflectances of the surfaces that were painted in different colors.

For radiometric comparisons, a device is needed that accurately simulates the behavior of the human eye, like the ability to capture multiple samples at each wavelength band of light. Since such a device was not available, Meyer et al. obtained a radiometer (see appendix B.1) instead. This device was able to measure over the range of the radiometric spectrum but provided just a single reading, which is not enough to represent a whole scene. Therefore, 25 evenly distributed measurements were taken (see figure 5). The sample points were chosen that way to avoid shadows and to maximize the amount of light that hits the probe.

Three different test scenes were created to analyze the verification procedure. Two of them consisted of an empty cube – the first had only white panels, the second contained one blue panel. The third test scene was a white cube with a white box inside it. Figure 6 shows the results of the third test scene. The cube is tilted in a way that the front panel is on top and the top panel is facing the left side. The irradiation H is shown on the vertical axis. The red lines show the result calculated by the radiosity algorithm whereas the blue lines represent the actual measurements. The correspondence between the values of the computer generated image and the real scene is clearly visible. They have a root mean square difference of about four percent.

It has to be mentioned that the described method only works with diffuse envir-

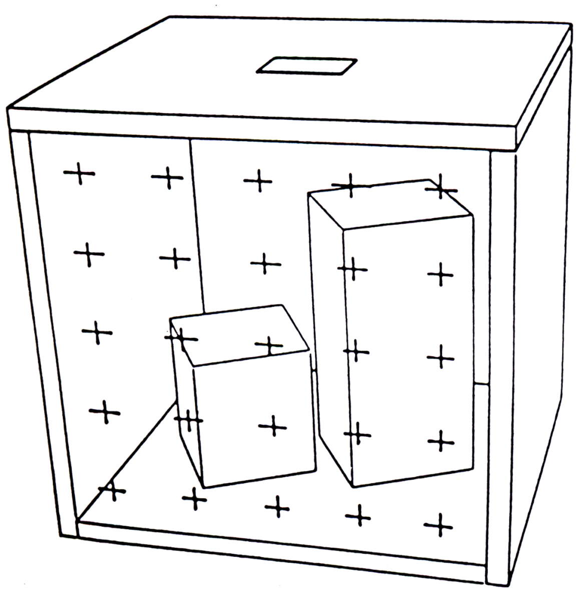


Figure 5: The positions of the 25 radiometric measurements [MRC+86].

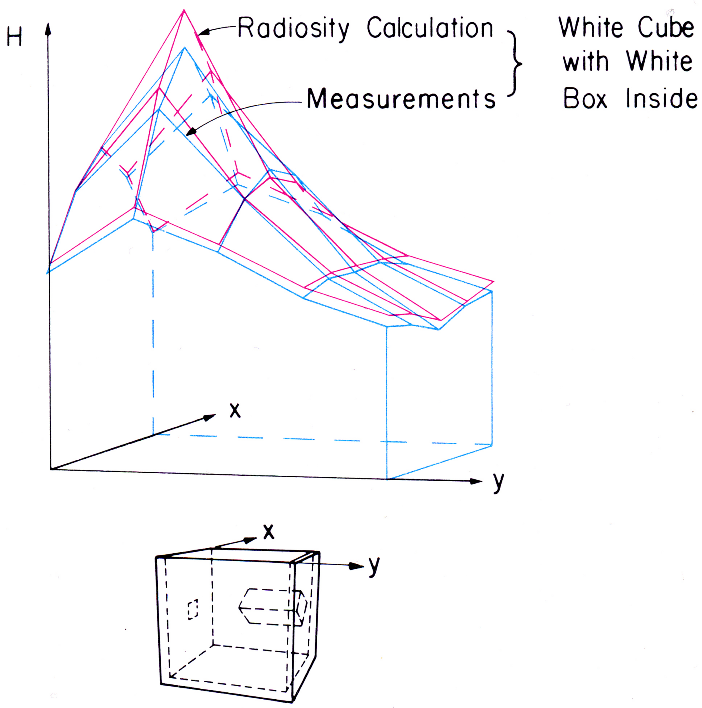


Figure 6: Results of the test scene with white panels and a white box inside it [MRC+86].

Figure 7: Points A to F were selected for measurements [TTOO90].

onments. None of the devices is able to measure specular or anisotropic surfaces. Moreover, 25 measurements are not representative for more complex scenes. But using more samples makes the procedure even more time-consuming.

2.2.2 Color Measurements

In 1990, Tagaki et al. [TTOO90] wanted to verify the results of a global illumination algorithm they developed to render realistic previews of car models. They reduced the comparisons to a few relevant feature points. A car was analyzed under different weather conditions and the most critical sections were selected (see figure 7).

A Minolta CS-100 incident color meter (see appendix B.5) was used to measure the chromaticity and the luminance of these points. According to Tagaki et al., the measured and the calculated values were almost equal. Unfortunately, the verification process is not described in detail. Therefore it is not clear how the measurements were acquired and why just six samples were taken. Figure 8 shows a photograph and a rendering of one of the car models. It can easily be seen that those images are not equal. A difference image (see figure 9) reveals that they are not geometrically equal and that there are big differences in color values on some parts of the car, in the shadows and in the background. Figure 10 points out the

Figure 8: Top: A photograph of a car. Bottom: A rendered image of this car [TTOO90].

color differences even more.

2.2.3 Scanning the Cornell Box

Because of the disadvantages mentioned in section 2.2.1, the Cornell Box setup was further enhanced. The radiometer was replaced by a scanner to verify a global illumination algorithm based on spherical harmonics [SAWG91] in 1991. Three colored filters were used to retrieve information about the spectral distribution of the light. The filters transmitted a wide range of wavelengths, whereas the algorithm calculated values for three specific monochromatic channels. Thus, only visual comparisons of the structure of the illumination were possible – similar to the very first approach (see section 2.1).

Figure 9: A difference image of photograph and rendered image (see figure 8).

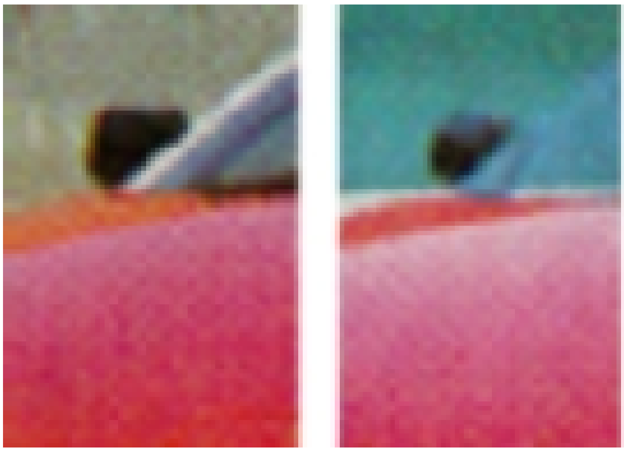


Figure 10: A comparison of a small section of the front part of the car (see figure 8).

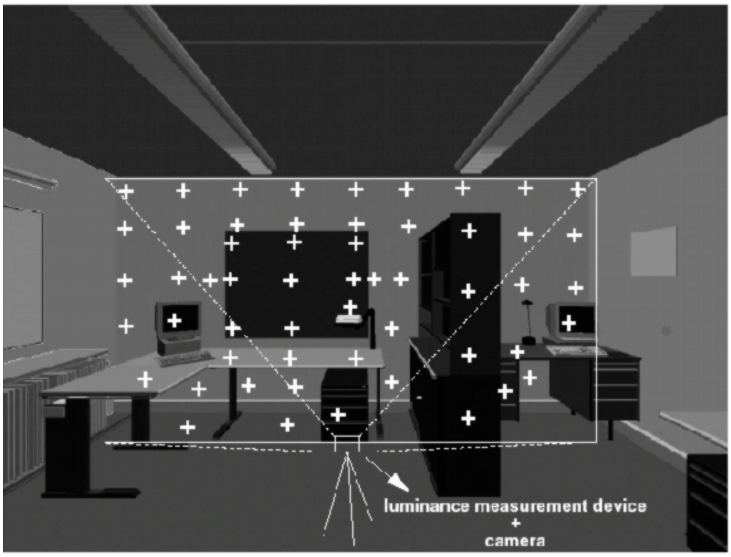


Figure 11: The sample points do not lie near edges [KP96].

2.2.4 Model of an Office Room

Karner and Prantl [KP96] developed a method to verify complex scenes. They created a model of an office room including material characteristics and used the Radiance software package for rendering. Several different approaches for comparing a rendering of this model to measurements of the real scene were made. A photometer (Minolta Spotmeter F, see appendix B.2) was used to obtain luminance values at selected points of the scene. No color measurements were taken.

The first approach consisted in just comparing the measured values to the calculated results. In order to avoid problems that arose from misalignment of the two scenes, the positions of the samples were chosen in a way that they did not lie near edges (see figure 11). For testing purposes, the model was rendered from two different viewpoints and in high and low quality. Figure 12 shows a high quality rendering, a low quality rendering and a photograph of the scene. The root mean square error lay within a margin of 18.2% to 21.8% while the average relative error lay between 44% and 59%. The low quality renderings achieved similar results as the ones that were rendered in high quality although they obviously should perform worse. The authors' explanation for this is that the eye is more sensitive to changes than to absolute values. From this follows that the combination of point to point comparisons and the root mean square error is not sufficient for quantitatively verifying a rendering system.

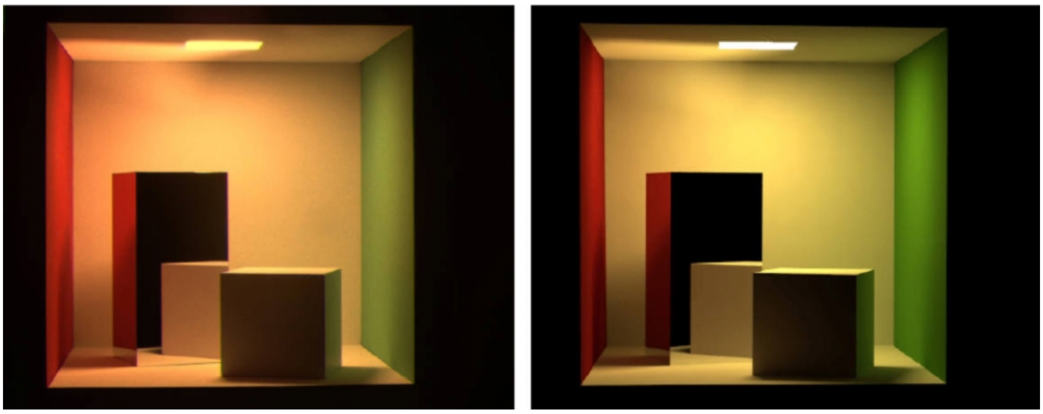
Figure 12: From top to bottom: a high quality rendering, a low quality rendering, and a photograph of the office scene [KP96].

In the second approach, the authors compared whole surface patches instead of points. They did not increase the number of measurements but took a photograph of the scene and scanned it for further processing. Hence the luminance values are known for some of the pixels of the photograph, the other values could be calculated by using a curve fitting function. The root mean square error was now between 16.4% and 18.5% and the average relative error was between 44% and 71%. Most of the errors occurred because of a misalignment of the images. Moreover it should be mentioned, that for one of the scenes the root mean square error for the low quality rendering was again lower than for the one with high quality.

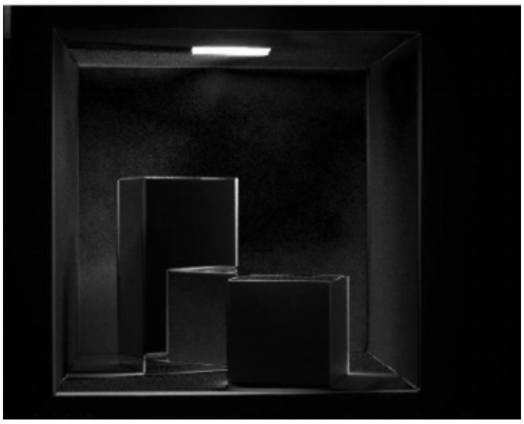
In order to cope with the problem of misalignment, the edges were excluded from the evaluation for testing purposes. This reduced the root mean square error to a value between 12.9% and 17.8% and the average relative error to a value between 32% and 52%. However, the margin of error was still very high. One reason for this was that the specifications of the light sources differed by up to 30% from the true value. Furthermore, the devices used to measure the material characteristics introduced an error in the range of 5% to 10%.

2.2.5 Charge-coupled Device (CCD)

In 1997, the Cornell Box approach was again used for verification purposes. Now a CCD camera was used for direct colorimetric comparisons of synthetic and real images. This was the first attempt where values for the whole image and not just a few samples were captured. Pattanaik et al. [PFTG97] describe the procedure of calibrating a CCD camera to reduce the error that is introduced by different forms of noise and the extraction of color values out of a monochromatic CCD. Seven narrow band filters were used to distinguish between the different ranges of wavelengths. Then, the CIE tristimulus values (X,Y,Z) were computed for each CCD element and for each pixel of the computer generated image. Figure 13(a) shows the results of those calculations. A difference in color values is clearly visible. The measured image is redder in the upper left corner whereas the computed image appears greener on the right side. A scaled difference image of the luminance values Y can be seen in figure 13(b). It can be seen that the largest errors occur at the edges of the objects and on the light source.



(a)



(b)

Figure 13: (a) Left side: measured image. Right side: computed image; (b) scaled difference image of the luminance values of the images shown in (a) [PFTG97].

2.2.6 Evaluation of Four Daylighting Software Programs

In order to help architects and lighting designers in choosing the most accurate daylighting software, Ubbelohde and Humann [UH98] did a comparative evaluation of four daylighting programs: Lumen Micro [Lum06], SuperLite [Sup06], Radiance [War94] and Lightscape [Lig06]. The comparisons were based on a model of an existing building in San Francisco. A CAD program was used to generate a standard DXF file of the architectural drawings. A LiCor 210S photosensor was used to take measurements in one of the offices of the building. The authors tried to find real skies as close to CIE standard skies as possible. In each program, overcast sky and clear sky conditions were specified for the times and dates of the real building metering.

The results of all four programs differed significantly from the on-site measurements, especially for Lumen Micro and SuperLite. The predictions were 10 to 20 times too high. The authors concluded that this is because the original DXF file could not be used with these programs and for that reason the geometry had to be simplified. Radiance and Lightscape yielded much better results. An exact match would be unlikely because of the difficulty (or impossibility) to find real sky conditions that exactly correspond to CIE standard skies.

2.2.7 Validation of Daylight Simulations using Radiance

In 1999, Mardaljevic [Mar99] finished his PhD on the validation of daylight simulation using the Radiance rendering system. In the course of his work he investigated whether Radiance yielded reliable results for daylight modeling of architectural scenes. The measurements required were accomplished within the scope of the International Daylight Measurement Programme (IDMP) organized by the CIE. These particular measurements were taken from July 1992 to July 1993. On the one hand, the sky was scanned by mounting a sky monitoring apparatus on top of a building. During daylight hours, 150 readings were made every 15 minutes. On the other hand, the illuminance inside two rooms of the same building was measured using photocells at the same time. Six photocells were positioned 0.7m above floor level along the center line of each room (see figure 14). The second room was used to test the effect of several different innovative glazings. The simulation was based on the recordings of the scan of the sky. The result of the simulation could therefore directly be compared to the values that were measured

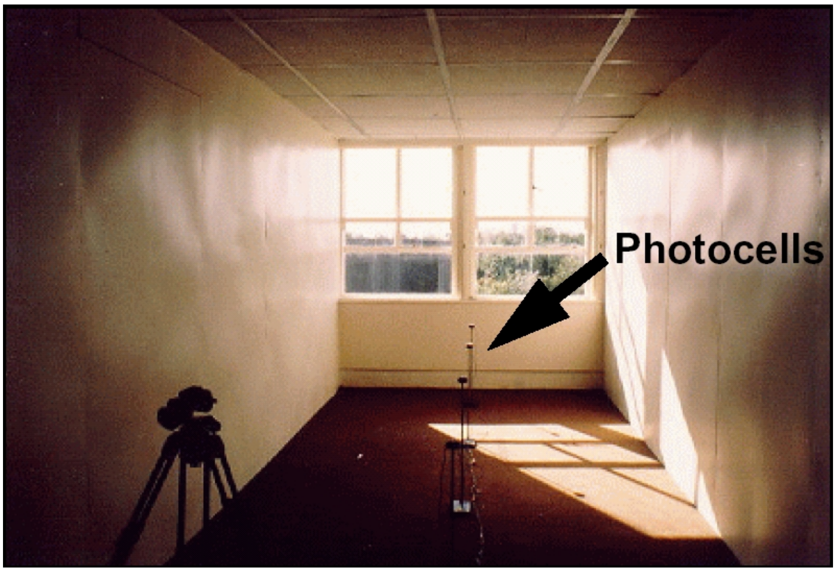


Figure 14: Six photocells were positioned along the center line of each room [Mar99].

inside the building.

In order to achieve a wide range of different sky conditions, 754 representative sky measurements were selected for further processing. The internal illuminances at the six photocells were calculated using Radiance for all of the 754 skylight configurations. Figure 15 shows one scatter plot for each photocell, each containing a comparison of the predicted and the measured illuminances. It can be seen that most of the dots lie on or near the diagonal, i.e. the measured and the calculated values are equal or nearly equal. Though, for high illuminances the accuracy decreases noticeably. Especially the first photocell, placed in front of the window, yielded a high number of over and under predictions. So, for bright clear sky conditions the prediction is less reliable than for overcast skies. Still, 63.8% of the internal illuminance predictions were within a margin of $\pm 10\%$ of the measured values.

An exhaustive analysis was done in order to find out whether the errors were related to measuring errors or misalignments in the model representation, or whether Radiance yielded inaccurate predictions. Although it was not possible to find a single cause of error, it could be shown that most likely multiple inaccuracies in the model representation were responsible for the errors rather than the Radiance program itself. Geometric misalignments as well as meteorologic phenom-

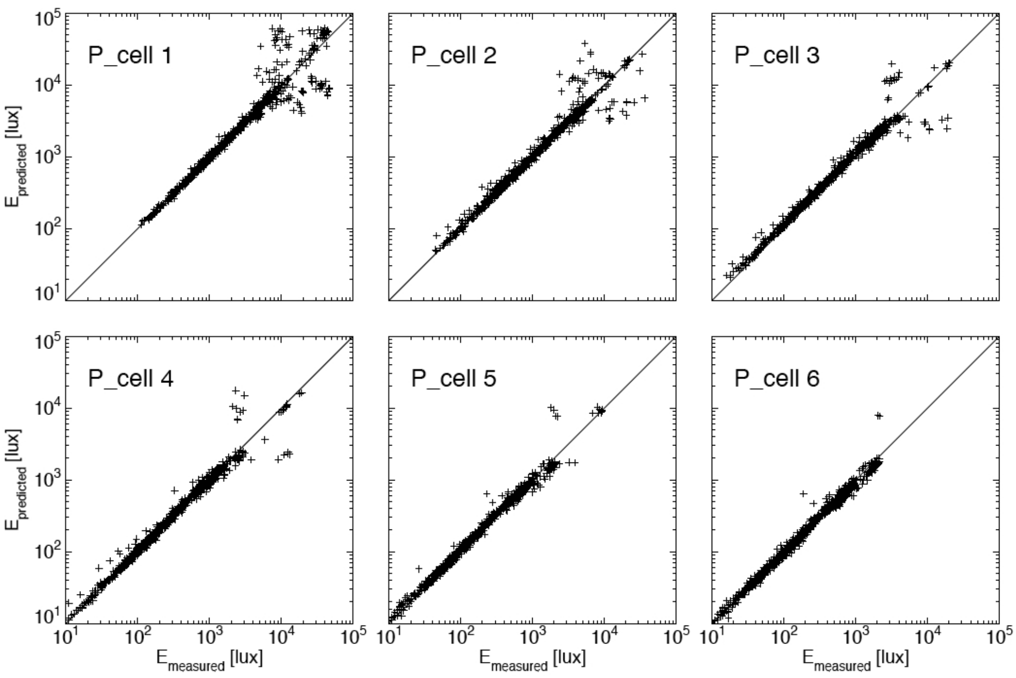


Figure 15: Six scatterplots that compare the predicted and the measured illuminances for each photocell [Mar99].

ena such as small bright clouds, fast moving patchy clouds, rain, snow or heavy showers could have biased the model. Radiance was therefore capable of reliably predicting the illuminance inside a building for a wide range of sky types – especially overcast skies – but results that were generated when the circumsolar region was visible from the point of calculation were considered to be potentially inaccurate.

The PhD of Mardaljevic also contains an evaluation of existing skylight models (e.g. the CIE models and the Perez model [PSM93]) and a description of how Radiance can be used to predict the *daylight factor*, which describes the ratio of the internal illuminance at a point to the global horizontal illuminance under overcast sky conditions and which is commonly used by architects and design consultants to evaluate the lighting situation inside a building.

2.2.8 Model of the Atrium at the University of Aizu

Two years later, Drago and Myszkowski [DM01] used a photometer (more precisely a luxmeter, see photometer in appendix B.2) to acquire data about a real

Figure 16: The left image shows a rendering of the atrium at the university of Aizu, while the right image shows a photograph of the real scene [DM01].

scene. Their goal was to provide a complete set of data which characterized a non-trivial existing environment for the test of physically based rendering engines. Unlike the Cornell Box scenes, which are of low complexity in terms of geometry and lighting, they wanted to build a scene that can be used to test the overall functionality of a rendering system. For this purpose, they created a model of the atrium at the university of Aizu based on the blueprints and on the actual scene. More than 80% of the surfaces in the atrium consisted of six materials, whose BRDFs were measured and included in the model. The reflectance properties of the remaining surfaces were estimated. For the luminaires, the goniometric diagrams were received from the manufacturer and corrected by a maintenance factor accounting for depreciation and aging. Figure 16 shows a rendering and a photograph of the atrium. The model is publicly available on the atrium webpage (see appendix D).

For the comparison, 84 sample points were chosen on the floor of the atrium (see figure 17). The illuminance values were obtained with the luxmeter and then compared to the output of a rendering engine that used the DEPT technique [VMKK00]. The calculated values and the measurements matched quite

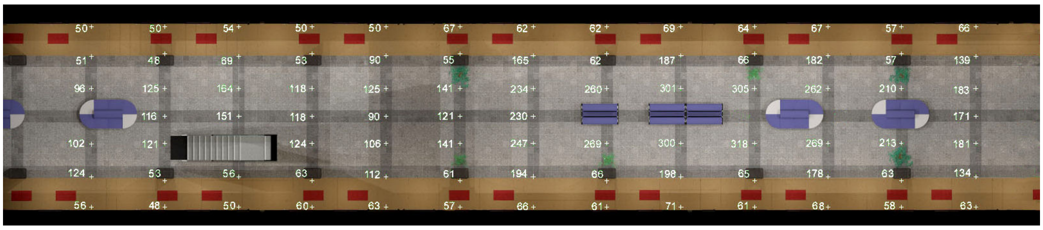


Figure 17: Illuminance values measured with a luxmeter at sampled points on the floor of the atrium [DM01].

well. For a high quality rendering, the average simulation error was 10.5%.

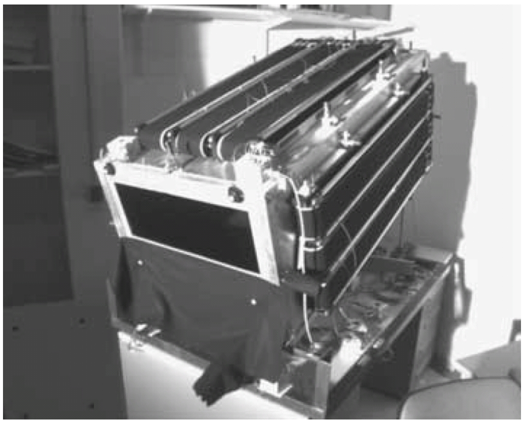
Moreover, Drago and Myszkowski did a visual comparison of a rendered image, a photograph, the real scene and a tuned rendering, i.e. the Lambertian and mirror reflection coefficients were intuitively tuned by a skilled artist to get the best match of image appearance in respect to the real scene. They asked 25 subjects to rate how similar the images were in respect to the real atrium scene. In all cases, the photograph got the highest score. The tuned image was found to be more realistic than the rendered image in terms of overall lighting and tone, which is not surprising because it was post-processed. This might be part of the explanation why industry prefers tweaking rendering parameters instead of doing physically based renderings. But it has to be mentioned that the artistic approach cannot be used if the scene does not exist in reality, i.e. when predictions of a scene have to be generated.

2.2.9 Component Case Studies

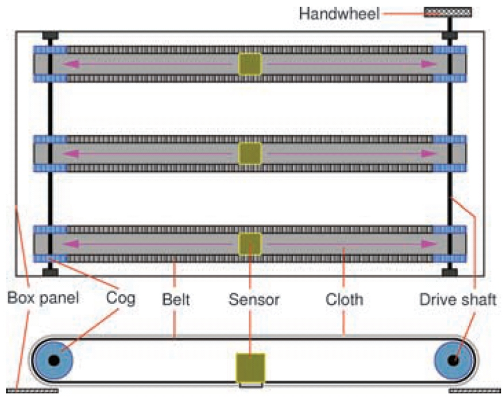
Recently, Schregle and Wienold [SW04] presented another approach for using luxmeters to verify a photorealistic rendering system, which is also the topic of Schregle's PhD thesis. Unlike Drago and Myszkowski, they focused on a setup that can be used to test each effect of global illumination separately. Therefore, they built a box similar to the Cornell box (see figure 18(a)).

On the top, the bottom, and the sides of the box, belts were mounted in order to guide the sensors. The belts were covered with the interior material of the box. They were driven in parallel by a shaft, which was operated manually. Figure 18(b) shows a schematic of the sensor guidance mechanism.

Moreover, there were four additional sensors mounted on the front face of the box to measure the direct illuminance from the light source. So, the authors were



(a)



(b)

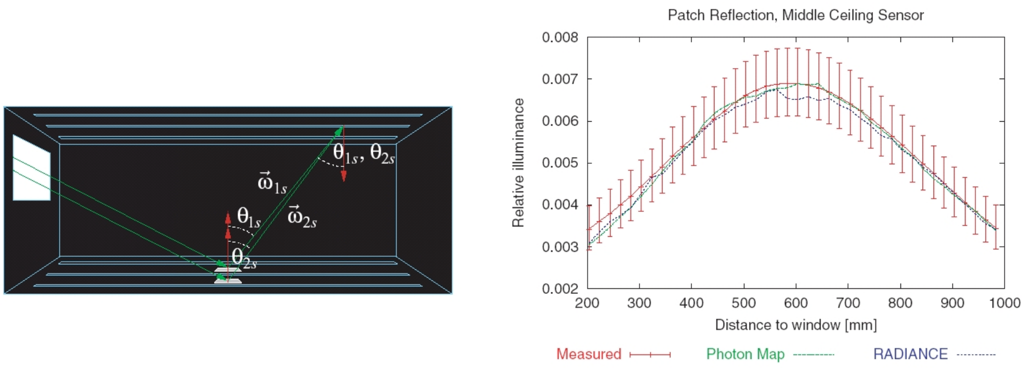
Figure 18: (a) The validation test box; (b) a schematic of the sensor guidance mechanism [SW04].

able to correct the specifications of the manufacturer by a maintenance factor. The inside of the box was covered with heavy molleton, which was nearly lambertian. Nevertheless, the discrepancy between the heavy molleton and a perfectly diffuse lambertian surface was significant. Therefore, an extensive BRDF measurement of the molleton was inevitable. In order to be able to validate caustics as well, a so-called sandblasted aluminium light shelf was attached at the open side of the box. The BRDFs of the molleton and the aluminium were obtained using a goniophotometer (see appendix B.7).

Schregle and Wienold did two different kinds of case studies. First, *component* case studies were done, where individual components of the rendering system were tested. After this, *compound* case studies were performed which were combinations of different *component* case studies. Lighting simulations were performed using photon maps and the Radiance engine to be able to compare forward and backward raytracing methods.

Four different case studies are presented in this paper: diffuse patch reflection, light shelf caustics, diffuse interreflection, and a combination of light shelf caustics and diffuse interreflection. For the first test case, two patches of light gray molleton where placed between the floor sensor tracks. The resulting illuminance was measured on the ceiling. To validate the accuracy of their measurements, Schregle and Wienold analytically calculated the theoretical result of the diffuse reflection. A schematic of the inside of the box can be seen in figure 19a.

A comparison of the measured values and the results of the Radiance and



(a) (b)

Figure 19: (a) A schematic of the diffuse patch reflection test case; (b) Comparison of the measured illuminance with the results of the photon map and Radiance [SW04].

photon map calculations is shown in figure 19b. The vertical bars in the measured data indicate the sensor tolerance. It can be seen that the curves fit quite well. The average deviation for Radiance is 3%, while it is 2% for the photon map.

For the case study involving caustics, the aluminium light shelf was mounted on the outside of the window in order to create a caustic directed toward the ceiling. The accurate simulation of this effect was possible with both rendering algorithms, though the calculations of Radiance were slightly more noisy than the ones of the photon map algorithm. This is also reflected in the average deviations, which are 7% for Radiance and only 2% for the photon map.

The third case study that is described in the paper is a generalization of the diffuse patch reflection case and therefore a compound case study. The whole interior of the box was covered with light gray molleton. The BRDF data had to be corrected because the molleton parts were from different consignments. After this adjustment, the average deviations were 1% for Radiance and 2% for photon map.

Similar results were achieved for the fourth test case, a combination of the light shelf caustic and the diffuse interreflection. Due to the correction of the BRDF data, deviations averaged 1% and 2%, for Radiance and the photon map respectively.

Both algorithms performed well and the relative deviations were consistently within the estimated error margins. Nonetheless, this setup can only be used for point samples but not for other purposes as for instance for verifying the exact shape of the caustic.

2.3 Analytical Tests

Arbitrary images that were created by using a stochastic global illumination algorithm are often just verified by the programmer because the result includes random variables and can therefore not easily be compared with a reference solution. This leads to the fact that sometimes errors in the implementation are explained by random noise or other artifacts and are not further investigated. In order to avoid this kind of misinterpretation, another way of testing the correctness of the implementation of global illumination algorithms is to create scenes where the solution can be determined analytically.

2.3.1 Scenes With a Constant Solution

In 2001, Szirmay-Kalos et al. [SKKA01] extended previous attempts in order to be able to test Monte Carlo global illumination algorithms and to use arbitrary BRDF models and light sources. Two different types of test scenes were described in this paper: scenes with constant radiance and scenes that represent the internal surface of a sphere. Scenes with constant radiance could contain arbitrary geometry with the restriction that it had to be a closed environment. As the radiance was constant everywhere and in every direction, the incoming radiance was also constant. To be able to use non-diffuse rendering algorithms, two approaches were presented. First, the BRDFs were given and the light source that was needed to provide constant radiance had to be calculated. In the second approach, the light sources were left unchanged and the BRDFs were determined in a way that the radiance was again constant.

In scenes that represent the internal surface of a sphere, the material models and the light sources could be defined more generally. Again Szirmay-Kalos et al. presented two different versions of this test scene. In one case, all the surfaces and light sources were supposed to be diffuse, in the other case, the surfaces were perfect mirrors.

In all cases, the fully converged solution was completely white. This allows the programmer to verify the basic implementation of the global illumination algorithm, because the correct solution of the calculation is already known. If the result of the rendering process contains areas that are not white, the programmer can be sure that the implementation is not correct.

2.3.2 A Representative Set of Test Cases

Another attempt on defining a set of representative test cases with analytical solutions was done by Maamari and Fontoynont [MF03] in 2003. They created six test cases where the solution is known and therefore can be compared to the results of a simulation. The tests were done with a radiosity solution only, but the authors state that the set of test cases will be expanded by scenes that include specular surfaces as well. This work is part of the main objectives of the CIE Technical Committee 3.33 [CIE06a], that concentrates on "Test Cases to Assess the Accuracy of Lighting Computer Programs".

The first test case consists of a square room with either an opening at the roof or at one of the walls. The interior surfaces are black, i.e. they have a reflectance of 0%. A skylight model for cloudy sky and a sun at 90° elevation is used as light source. The total direct luminous flux arriving at the opening surface has to be equal to the flux reaching the interior surfaces. The radiosity algorithm produced a systematic error of about -16% for the roof opening and +13% for the wall openings.

To simulate windows, a second test case was defined where a perfectly specular glass material is positioned at the top of the opening. Several simulations were done where the incident angle of the light varied from 0° to 90° in 10° steps. However, the software that was used did not take into account the transmittance variation of incoming light with incidence angle. Therefore the internal illuminances were over-estimated.

In order to describe the intensity distribution of luminaires, the most commonly used file formats are the IESNA format and the Eulumdat format. To test the quality of the interpolation of the intensity, a third test case was created that consists of a horizontal surface with a point light source at 3m height from the surface center. It could be shown that the radiosity algorithm solved this test case correctly because the error was below 0.2% compared with the analytical solution.

The shape factor formulae, which are used to define the fraction of luminous flux leaving a perfectly diffuse surface that reaches an elementary surface, were verified in the fourth test case. The geometry is again a square room as in the first two test cases, but instead of the opening there is a diffuse area light source at the center of the ceiling. Again, the error was very low (below 1.3%) and it could be proved that the software used was able to calculate this effect correctly.

The fifth test case dealt with the reflectance of the room surfaces. The surfaces of a square room are defined as uniform diffusers. A point light source at the center of the ceiling is used for illumination. The errors were within a negligible margin except for the 0 and 0.9 to 1 reflectance values. The authors assumed that these errors came from an epsilon value being affected at these extreme values, but also point out that these values rarely exist in reality.

The last test case was specified to verify the calculation of the daylight factor, which is commonly used to evaluate daylighting inside buildings. The geometry is the same as in the fourth test case. For the analytical solution, a module developed and validated by Dumortier and Van Roy [Sod06] at the LASH laboratory of the ENTPE–France was used. Unfortunately, in this case the software yielded errors of over 13%.

The work of Maamari and Fontoynont shows that defining a set of test cases where the solution can be derived analytically is a promising approach to validate global illumination rendering engines. Having a reliable benchmark would help to classify existing software.

2.4 Combined Approaches

2.4.1 Comparison of several Rendering Systems

In 1996, Khodulev and Kopylov [KK96] did a comparison of three physically based rendering systems: Lightscape Visualization System (LVS [Lig06], since at that time Lightscape had been taken over by Autodesk), Specter System [Int06] and Radiance [War94]. Amongst other criteria they investigated the physical accuracy of the global illumination simulation. They combined different approaches to verification – analytical tests (see section 2.3) and visual comparisons (see section 2.1) – to benefit from the advantages of each approach.

Three different types of test scenes were used. Firstly, they defined a test case where the result could be calculated analytically. It consisted of a diffuse white cube with a point light source in its center. Six sample points on the inside of the cube were chosen for comparisons. In this way, the absolute simulation accuracy of each system could be estimated.

Secondly, one of the LVS example scenes was rendered with each of the systems, and the results were compared to each other at three sample points. Khodulev and Kopylov found out that two of the systems (Specter and Radiance) yielded quite similar results whereas the result of the third system (LVS) was significantly different from the two others. They deduced that the first two systems were more accurate than the third one. As no reference measurements or analytical results of this scene were available, the validity of this comparison is questionable.

Thirdly, they did a visual comparison of various complex test scenes. Although visual comparisons are not very accurate, some basic knowledge can nevertheless be gathered. For example, the specular reflections in the test images were considerably different. Furthermore, it could be seen whether a system had problems with e.g. very thin triangles or special surfaces. With visual comparisons it is possible to discover noticable weaknesses of the systems, but not to determine the physical accuracy of the resulting images.

The result of this comparison was that concerning physical accuracy Radiance performed best, followed by Specter and LVS. The use of different verification methods enhanced the quality of this study. The physical accuracy of the systems was evaluated using analytical tests while more general problems were detected by visually comparing the calculated images.

2.4.2 A Multistage Validation Procedure

Myszkowski and Kunii [MK00] defined a multistage validation procedure in order to validate an enhanced radiosity algorithm. It consisted of three stages: analytical tests, comparison with measurements results and visual comparisons.

The authors used two different test scenes for analytical comparisons: the interior of a diffuse empty cube and the interior of a diffuse empty sphere with one or two orthogonal mirrors inside. The total lighting in a scene can be divided into the direct and the indirect illumination. For direct illumination, the theoretically derived result and the simulation matched exactly. For indirect illumination, the simulated results diverged slightly from the correct solution. This is because the energy transfer for the chosen sample points is either over- or underestimated due to approximations introduced into the form factor estimates. An additional source of error was the triangulation of the sphere, which led to differences between the segments of the sphere and its actual radius.

For experimental measurements, the authors again created two different test scenes. Each of the scenes consisted of an empty room with four light sources mounted on stands and directed toward the ceiling. The only difference was the height of the rooms. Unfortunately, the authors do not describe the measurement procedure in detail, as for example what kind of measurement device they used.

Since what really matters in practice is how a rendering is perceived by a human observer, the synthetic images were also compared to photographs of a real scene. Therefore, the authors did a case study where they chose an atrium of the University of Aizu as reference scene. In section 2.2.8 this case study is described in more detail.

By combining three validation approaches and doing such exhaustive comparisons, the authors wanted to compensate for the disadvantages of each approach. Furthermore they wanted to make the global illumination community aware of the fact that there is no standardized approach available, although so many different light transport simulation algorithms have been developed in the last decades. As their method of combining different approaches covers diverse aspects of validation it could serve as a basis for the development of a standardized validation method.

3 Colorimetry

Colorimetry [WS82, Sto03, GM02, GJB80, Gre99, Gla95, Sha04, JMF94] is the science of measuring, describing, and evaluating colors. It is based on empirical studies of humans matching colors. This chapter gives a short overview of how color can be specified and how color distances can be measured.

3.1 Color Spaces

3.1.1 CIE XYZ Color Space

In 1931, the Commission Internationale de l'Eclairage (CIE) standardized a set of color-matching functions that can be used to convert spectral data to CIE XYZ color space [WS82]. These color-matching functions are the result of color-matching experiments. Three primary lights, which can vary in intensity, are chosen. The observers then adjust these primary colors to create colors that match a set of monochromatic colors of the spectrum. The CIE defined a transformation to such kind of color-matching functions to make them more convenient to use. These transformed functions are known as the CIE color-matching functions or the CIE standard observer. The CIE tristimulus values X, Y, and Z are calculated by multiplying the stimulus times the color-matching functions and integrating the result, as shown in the following formulae:

$$X = k \sum_{\lambda=\lambda_a}^{\lambda_b} \beta(\lambda) S(\lambda) \bar{x}(\lambda) \Delta\lambda$$
$$Y = k \sum_{\lambda=\lambda_a}^{\lambda_b} \beta(\lambda) S(\lambda) \bar{y}(\lambda) \Delta\lambda$$
$$Z = k \sum_{\lambda=\lambda_a}^{\lambda_b} \beta(\lambda) S(\lambda) \bar{z}(\lambda) \Delta\lambda$$

where λ is the wavelength, β is the reflectance spectrum, S is the spectrum of the light source and $\bar{x}(\lambda)$, $\bar{y}(\lambda)$ and $\bar{z}(\lambda)$ are the color matching functions. The constant k is equal to $638lmW^{-1}$ if tristimulus values are to be provided in units of luminance [GJB80].

In XYZ color space, two colors with different coordinates have different appearance. Though, equal distances in XYZ space do not represent equal shifts in color appearance, i.e. XYZ space is not perceptually uniform.

In order to be able to define a color independently from its brightness, the tristimulus values can be translated into a projection on a two-dimensional plane. The xy coordinates on this plane are called chromaticity coordinates and are calculated by:

$$
\begin{aligned}
x &= \frac{X}{X+Y+Z} \\
y &= \frac{Y}{X+Y+Z} \\
z &= \frac{Z}{X+Y+Z}
\end{aligned}
$$

where $x + y + z = 1$. A color can therefore be uniquely specified by giving the xy chromaticity coordinates together with the tristimulus value Y. A plot of the chromaticity coordinates of all tristimulus values can be seen in figure 20.

3.1.2 CIE $L^*a^*b^*$ Color Space

CIE $L^*a^*b^*$ values [WS82] are a non-linear transformation of the CIE tristimulus values that create an approximately perceptually uniform color space. This simplifies the computation of color differences. The * after each letter is added in order to distinguish it from the Hunter Lab color space. L^* stands for the lightness of a color, whereas a^* corresponds to the redness and greenness and b^* corresponds to the yellowness and blueness of a color. The transformation is described by the following formulae:

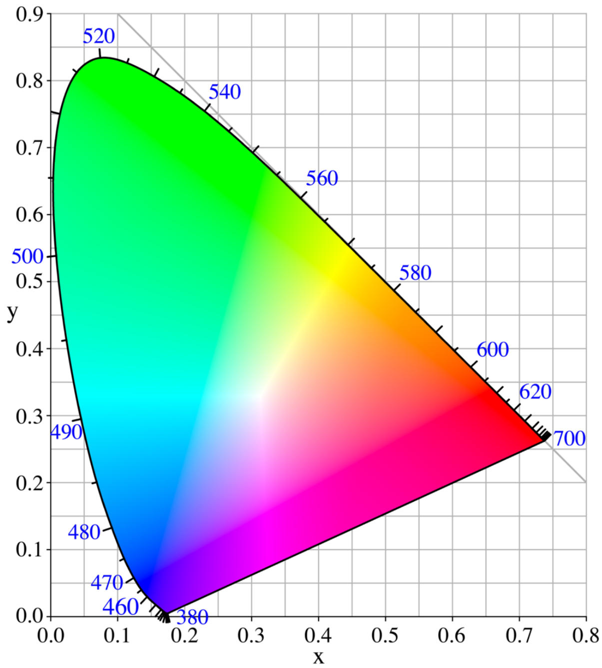


Figure 20: The CIE 1931 chromaticity diagram showing the wavelengths of light and the xy coordinates [cie06b].

$$
\begin{aligned}
L^* &= 116\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - 16 \\
a^* &= 500\left[\left(\frac{X}{X_n}\right)^{\frac{1}{3}} - \left(\frac{Y}{Y_n}\right)^{\frac{1}{3}}\right] \\
b^* &= 200\left[\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - \left(\frac{Z}{Z_n}\right)^{\frac{1}{3}}\right]
\end{aligned}
$$

where X_n, Y_n and Z_n are the tristimulus values of the reference white. For small values of $\frac{V}{V_n}, V \in \{X, Y, Z\}$, the normal formulae are replaced by the following modified formulae:

$$
\begin{aligned}
L_m^* &= 903.3\left(\frac{Y}{Y_n}\right) && \text{for } \frac{Y}{Y_n} \leq 0.008856 \\
a_m^* &= 500\left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right] \\
b_m^* &= 200\left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right]
\end{aligned}
$$

where

$$
\begin{aligned}
f\left(\frac{X}{X_n}\right) &= \left(\frac{X}{X_n}\right)^{\frac{1}{3}} && \text{for } \frac{X}{X_n} > 0.008856 \\
f\left(\frac{X}{X_n}\right) &= 7.787\left(\frac{X}{X_n}\right) + \frac{16}{116} && \text{for } \frac{X}{X_n} \leq 0.008856 \\
f\left(\frac{Y}{Y_n}\right) &= \left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} && \text{for } \frac{Y}{Y_n} > 0.008856 \\
f\left(\frac{Y}{Y_n}\right) &= 7.787\left(\frac{Y}{Y_n}\right) + \frac{16}{116} && \text{for } \frac{Y}{Y_n} \leq 0.008856 \\
f\left(\frac{Z}{Z_n}\right) &= \left(\frac{Z}{Z_n}\right)^{\frac{1}{3}} && \text{for } \frac{Z}{Z_n} > 0.008856 \\
f\left(\frac{Z}{Z_n}\right) &= 7.787\left(\frac{Z}{Z_n}\right) + \frac{16}{116} && \text{for } \frac{Z}{Z_n} \leq 0.008856.
\end{aligned}
$$

3.1.3 CIE $L^*u^*v^*$ Color Space

CIE $L^*u^*v^*$ color space is similar to the $L^*a^*b^*$ color space as it is also a non-linear transformation of the CIE XYZ color space and approximately perceptually uniform. They have the same lightness axis L^*, but the other two components are computed differently, as can be seen in the following formulae:

$$
\begin{aligned}
L^* &= 116\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - 16 \\
u^* &= 13L^*\left(u' - u'_n\right) \\
v^* &= 13L^*\left(v' - v'_n\right)
\end{aligned}
$$

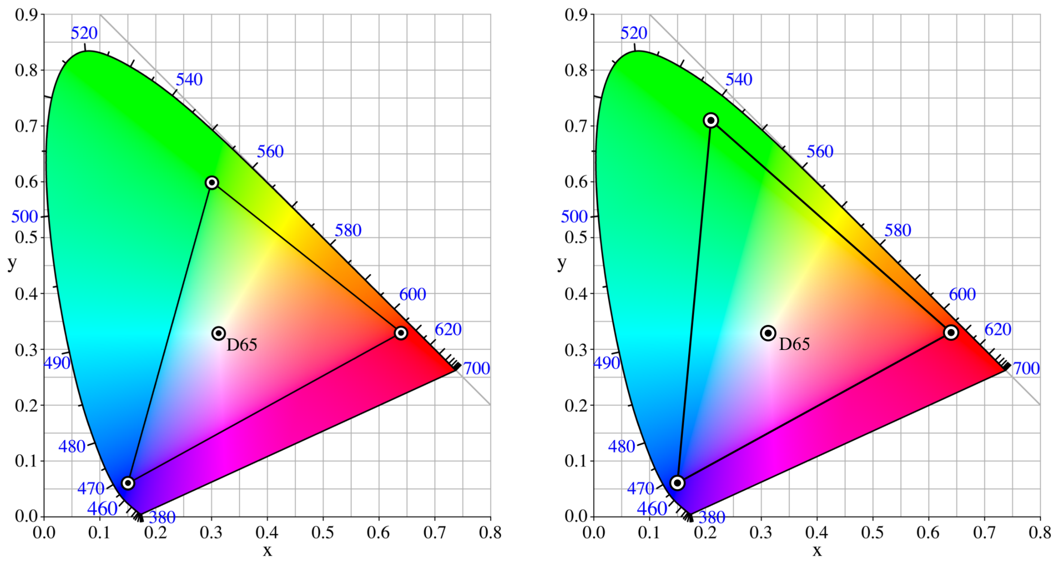
where X_n, Y_n and Z_n are the values of the reference white and u', u'_n, v' and v'_n are calculated from:

$$
\begin{aligned}
u' &= \frac{4X}{X + 15Y + 3Z} \\
u'_n &= \frac{4X_n}{X_n + 15Y_n + 3Z_n} \\
v' &= \frac{9Y}{X + 15Y + 3Z} \\
v'_n &= \frac{9Y_n}{X_n + 15Y_n + 3Z_n} \,.
\end{aligned}
$$

If $\frac{Y}{Y_n}$ is less than 0.008856 the following formula is used to calculate L^*_m:

$$
L^*_m = 903.3\left(\frac{Y}{Y_n}\right) \qquad \text{for } \frac{Y}{Y_n} \leq 0.008856.
$$

At the time the CIE was evaluating color difference spaces both $L^*a^*b^*$ and $L^*u^*v^*$ color spaces were proposed. As they were equally valid the CIE decided to formalize them both. $L^*a^*b^*$ color space is more commonly used for surface colors like paints, textiles and prints, whereas $L^*u^*v^*$ is normally used for self-luminous color displays.



(a) (b)

Figure 21: Chromaticity diagrams showing the primaries and the white point of (a) sRGB [srg06] and (b) Adobe RGB [ado06].

3.1.4 RGB Color Spaces

RGB color spaces are commonly used in computer graphics and digital photography. They form subspaces of the CIE XYZ color space and therefore do not contain all visible colors. RGB spaces are specified by defining a white point and three primary colors. RGB values itself do not identify absolute colors unless they are assigned to a certain RGB color space.

Common RGB color spaces that have been standardized are sRGB and Adobe RGB. The location of the primaries and the white point of these RGB color spaces are shown in figure 21. It can be seen that Adobe RGB has a significantly larger gamut than sRGB, improving primarily in cyan-greens. The CIE xy chromaticity coordinates for sRGB are (0.64, 0.33) for red, (0.30, 0.60) for green and (0.15, 0.06) for blue. The red and green vertices for Adobe RGB are very similar, but the values of the green vertex differ significantly. Red is at (0.64, 0.34), green is at (0.21, 0.71) and blue is at (0.15, 0.06) [GM02]. The Adobe RGB color space nonetheless encompasses only roughly 50% of all visible colors.

3.2 Color Difference Formulae

If a color space is approximately perceptually uniform, color differences can be specified as Euclidian distances in this space. The color difference between two colors in $L^*a^*b^*$ color space is thus defined as follows:

$$\Delta E^*_{ab} = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2}$$

$L^*a^*b^*$ differences [WS82] are noted as ΔE^*_{ab}. A difference of 1 is called a "just noticeable difference" (JND). Depending on the application, values of $\Delta E^*_{ab} = 4$ to $\Delta E^*_{ab} = 8$ are acceptable [Sha04].

Color difference in $L^*u^*v^*$ color space is defined similarly to ΔE^*_{ab}:

$$\Delta E^*_{uv} = \sqrt{(\Delta L^*)^2 + (\Delta u^*)^2 + (\Delta v^*)^2}$$

In color imaging, $L^*a^*b^*$ difference is commonly used, therefore we will focus on this metric. Several other color difference metrics have been proposed in the last years. They try to better adjust to the still remaining non-uniformity of $L^*a^*b^*$ color space by e.g. introducing different weightings for lightness and chroma. Since in the literature that is related to this thesis mainly ΔE^*_{ab} is used for comparisons, though, we will use this metric for reasons of compatibility and comparability.

4 Digital Cameras

Pattanaik et al. [PFTG97] (see section 2.2.5) showed that it is possible to derive colorimetric values of an unknown target by taking multiple measurements through narrow band filters. But obtaining a scientific grade CCD sensor and the corresponding band filters is very costly, what we think is the reason why this method is not used very often. So, our goal was to find a cheaper solution that yields comparable results within a quantified error range. Thus, we started to investigate whether a prosumer-grade digital camera can be used for this purpose.

Comparing a rendering to a photograph seems to be a simple task, because both contain RGB values – a difference image can easily be created. But the problem with this approach is that it is far from being an accurate comparison. Too many sources of error are affecting the result. On the one hand, a lot of processing is done by the camera, like demosaicing and white balance. On the other hand, we want to verify a spectral rendering system, i.e. our rendering system is able to handle more than three (RGB) sample points of the electromagnetic spectrum. So, a lot of information gets lost if the spectral values are resampled to RGB space.

If we want to use a prosumer-grade digital camera as a device to verify a physically based renderer, we have to refrain from seeing the camera as a device to make nice images of a scene, but we have to start using it as a color measurement device. This cannot be done by just using the final image a camera provides, because the camera software does a lot of post-processing to the original CCD data, like demosaicing and several image enhancement algorithms. In the next section, we will discuss the various forms of output one can get from a digital camera.

4.1 Various Forms of Output

Inside a digital camera, a light-sensitive CCD image sensor is used to capture the light that passes through the lens. In order to be able to create color images, several color filters are put in front of the CCD. The most common technique is to use a Bayer pattern, which consists of red, green and blue filters that are arranged on a grid (see figure 22). Twice as many green elements as red or blue are used to mimic the human eye's greater resolving power with green light. For a 4 Megapixel camera e.g., this means that we get 2 Million sensor values for green

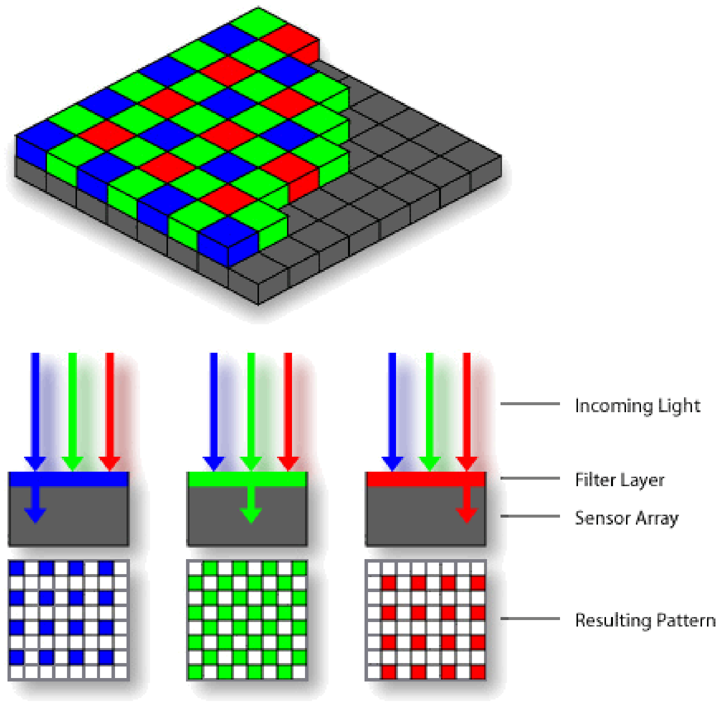


Figure 22: A Bayer filter mosaic is a color filter array for arranging RGB color filters on a square grid of photosensors [bay06].

and 1 Million for both red and blue filters. To recalculate a full 4 Megapixel RGB image a color interpolation called demosaicing has to be done. This algorithm estimates the missing RGB values by using information from the surrounding pixels. After this step, various image enhancement techniques like white balance, sharpening, contrast and brightness adjustments, noise reduction etc. are applied to the interpolated sensor data. The final image is usually either saved in JPEG or TIFF file format – JPEG format is more common because of the smaller file size due to compression. Figure 24 illustrates the individual steps of image processing done by camera software.

Most prosumer-grade cameras provide two forms of output data: the RAW image data [SG06] and the final image (see figure 23). The RAW image data contains the actual values of each CCD sensor element. The final image contains the enhanced data and is usually converted to 8-bit and compressed with lossy compression algorithms. If we want to use the final image for verification purposes we have to have that in mind.

Most camera manufacturers have created their very own raw file format – the

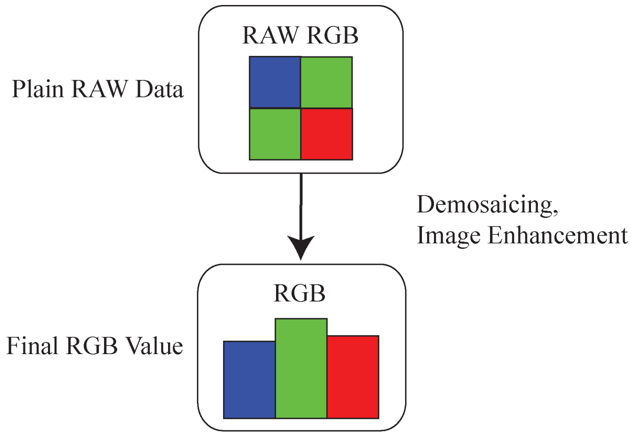


Figure 23: The RAW image data is converted to the final RGB value.

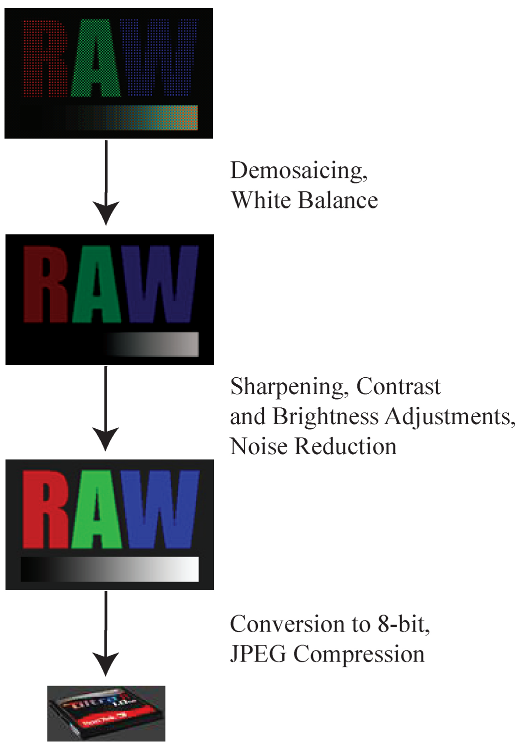


Figure 24: The individual steps of color processing done by common digital cameras [raw06b].

specifications of these formats are usually not publicly available. However, some RAW converters, e.g. dcraw [raw06a], also provide raw CCD data as output.

The errors that are introduced by the camera software can be avoided by using the raw CCD data. Unfortunately, this data cannot be compared to a rendered RGB image directly. It just contains information about how much light passed the filter but there is no information about how this relates to an XYZ or sRGB value. The reason for this is that most digital camera spectral sensitivities are not linear transformations of average human visual systems's spectral sensitivities. Therefore, the camera and a standardized human observer do not "see" color identically. Thus, a camera may generate two different sets of RGB values at two color samples, while to a human observer they look the same. This is the most severe problem we have to deal with when using a prosumer-grade digital camera for color measurements. Because of the different spectral sensitivities it is not possible to get an exact match between colors seen by a camera and a human observer. This thesis will discuss how a prosumer-grade camera can nevertheless be used for verification purposes.

5 Physically Based Rendering Systems

Our goal is to verify the result of the light transport simulation of a physically based rendering system. Therefore we take a closer look at the type of data we have after this step. We get a spectral value for each of the pixels in the image. This is represented for one pixel in the first row of the graph shown in figure 25. The next step in the rendering pipeline is to multiply the spectrum by the color matching functions to gain XYZ values. Gamut mapping and tone mapping are done in XYZ color space to shift the color value into an appropriate RGB subspace (e.g. sRGB or AdobeRGB, see section 3.1.4). After that, we multiply by a transformation matrix to gain RGB values. With each step we lose information. Therefore, we want to do our comparisons as early in the pipeline as possible. Apart from that there is another important fact that has a major influence on the decision: whether we want to

- take the human observer into account, i.e. make a qualitative statement on how different an image looks like compared to a real scene (i.e. with tone mapping step) or whether we want to
- purely verify the calculations (i.e. without tone mapping step).

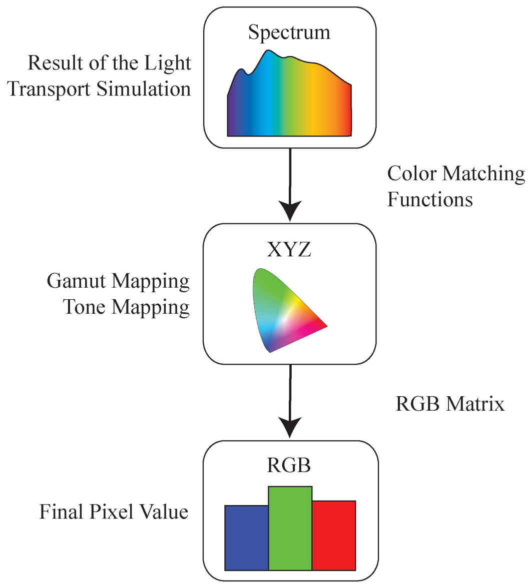


Figure 25: Rendering steps after the completion of the light transport simulation.

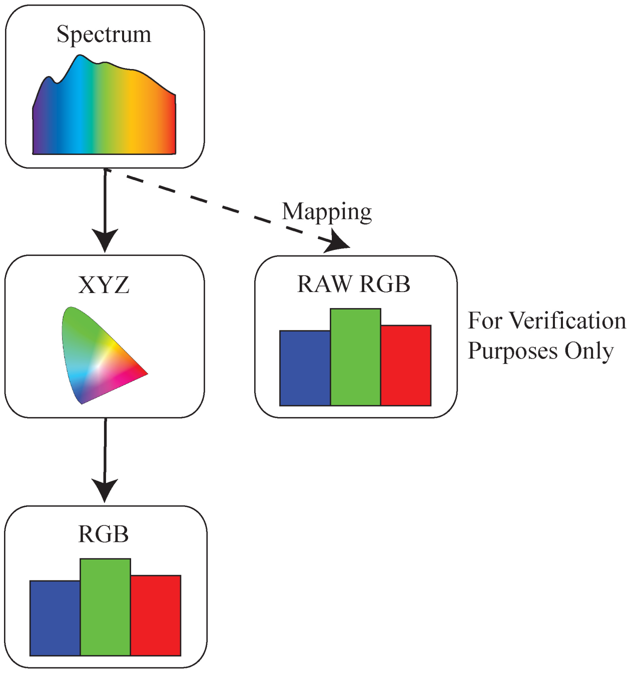


Figure 26: We can define a mapping from spectral values to RGB values.

In the first case, we use XYZ values because they take the human observer into account. To get the actual difference between image and real scene we convert both the rendering result and the photograph first to XYZ color space and then to $L^*a^*b^*$ color space and calculate the color difference ΔE^*_{ab} for each pixel. If ΔE^*_{ab} is within a certain error range we can consider two colors as visually equal. However, doing those conversions brings up a severe problem: the different spectral sensitivities of a human observer (XYZ space) and a (non-colorimetric, i.e. common) digital camera. For the rendering, the original spectrum is multiplied with the spectra of the color matching functions. For the photograph, the original spectrum is multiplied with the spectra of the CCD sensors, which are different from the spectra of the color matching functions. Two different original spectra may cause the same XYZ value but different RGB values. Therefore, two colors that are a metameric pair to a human observer will in general look different to a digital camera. Thus, in this case a certain amount of error is introduced by converting to XYZ space.

In the second case, we do not convert to XYZ space but to the camera's RAW RGB space directly. Therefore, we have to define a mapping from spectral values to the RAW RGB values of the camera (see figure 26). For details on the mapping see section 7.4. By avoiding XYZ space we also avoid the error that is introduced by converting to XYZ space. In case of a difference, we cannot quantify the actual ΔE^*_{ab} color difference between both values, though.

6 Data Acquisition and Processing

The following sections give a detailed description of the procedure of data acquisition – both images and spectral data – and various post-processing steps. Not all mentioned steps are necessary for each method described in section 7, but we want to give a complete overview of data acquisition.

6.1 Photographs

6.1.1 Position and Orientation of the Camera

Each of the characterization methods described in section 7 is based on using a photograph of a color chart (or a similar selection of colored patches) as a reference image to deduce information about the behavior of the digital camera. To optimize the quality of the image, some constraints have to be set. We use a Canon EOS 20D SLR camera with a Canon Zoom Lens EF 17-40mm lens. The camera mode was set to manual to be able to control parameters like aperture or exposure. Apart from that, we chose the settings shown in table 1. As color chart we use a GretagMacbeth ColorChecker SG (see appendix C.3).

Parameter	Setting	Comment
Quality	RAW + large fine (3504 x 2336 pixels)	
ISO	100	Noise reduction
White balance	Custom	
Color space	Adobe RGB	Larger than sRGB
Processing Parameters	Parameter 2	All parameters are set to 0
Metering mode	Evaluative metering	
Drive mode	Self-timer operation	
AF point	Automatic selection	To avoid camera shake

Table 1: Camera settings that were chosen for taking the photographs.

We took the photographs inside a light cabinet (see appendix C.1 for a description) to get a controllable and uniform illumination of the color chart. Lighting conditions were adjusted to be adequate for our purpose (see section 6.2). In order to specify the position of the camera and the color chart, we use the lower left corner of the front side of the light cabinet as reference point, i.e. as origin of our coordinate system. We took the sensor plane marking as reference

Figure 27: The sensor plane marking on the camera was used as camera reference point.

point on the camera (see figure 27). Based on the results of the tests we made with the light cabinet (see section 6.2), we chose the values seen in figure 28 for camera and chart positions. The size of the GretagMacbeth ColorChecker SG is $21.59 \times 27.94cm$. The camera is aiming at the center of the color chart.

A color chart should usually be measured under either a (45/0)-condition or (0/45)-condition [WS82], i.e. either illuminated at an angle of $45 \pm 5°$ and viewed at $0 \pm 5°$ or vice versa with respect to the center of the color chart. In our setup, we are using a (0/45)-condition. We observed that the glossy reflection is still very high at $45°$, though. Therefore we are using an angle of $48°$ – which is still within $\pm 5°$ – because the glossy reflection decreases noticeably with increasing angle (see figure 29).

6.1.2 Representative RGB Values

Figure 30 (a) shows a photograph of the ColorCheckerSG after correcting for distortion. As some of the methods described in section 7 are based on comparing the RGB values of the patches to measured values manually, we have to extract RGB values for all of the patches out of this photograph. Taking the RGB value of only one random pixel of a patch is too imprecise, as we could accidently pick a dead pixel (a defective pixel that remains unlit), a hot pixel (a pixel that stays on a solid color, also known as stuck pixel) or a non-representative pixel (e.g. the maximum value of all pixels in this patch). The first two cases are well known problems in

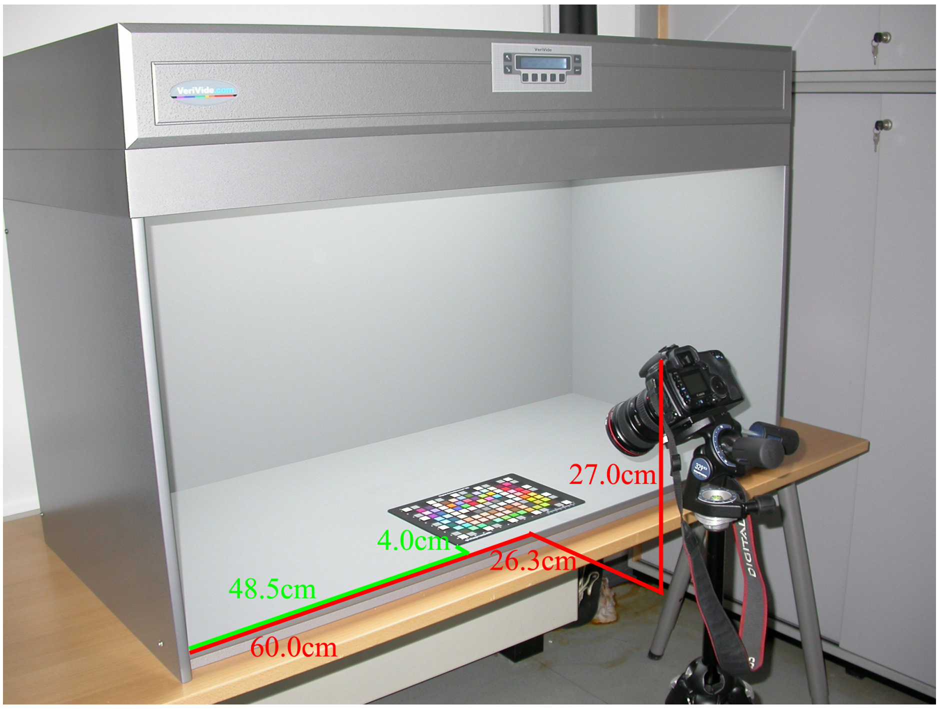
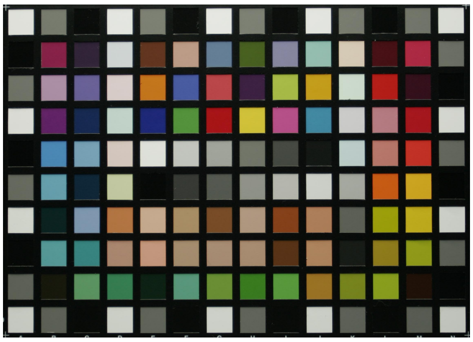
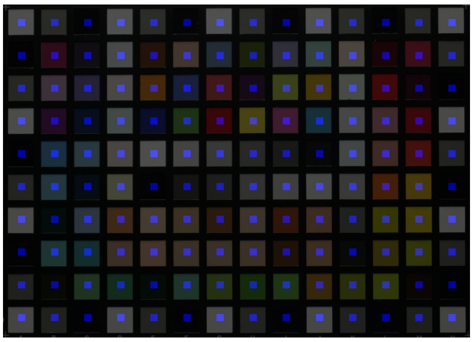


Figure 28: The position of the camera and the color chart with the lower left corner of the front side of the light cabinet as reference point.

Figure 29: The angle of the camera was set to $48°$.



(a)

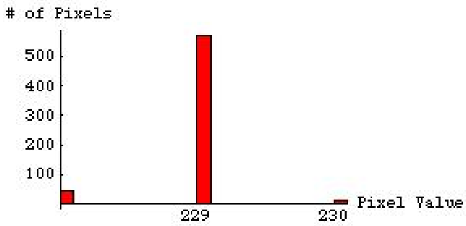


(b)

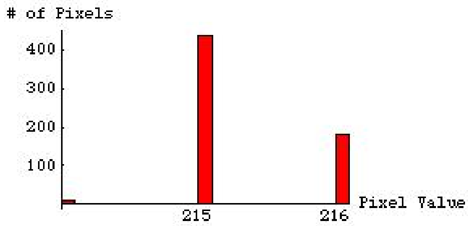
Figure 30: (a) ColorCheckerSG after correcting for distortion; (b) the subareas of the patches that were used for averaging (blue squares).

digital photography and can be solved easily. Software for identifying such pixels is available as freeware (e.g. Dead/Hot Pixel Test [dea06]). Pixels that are labeled as dead or hot pixels are excluded from further calculations. With the third case we have to deal separately, though. We analyzed the deviation of pixel values of several patches, in order to find a method for picking the most representative pixel value. As we cannot pick only one pixel of a patch, we have to take a region of pixels into account. The areas we chose for selecting a representative pixel value can be seen in figure 30 (b). The size of the area depends on the size of the image. For the image seen in figure 30 each area covers 990 (33x30) pixel. We excluded the border of each patch to avoid inaccuracies due to noise and interpolation at the edges.

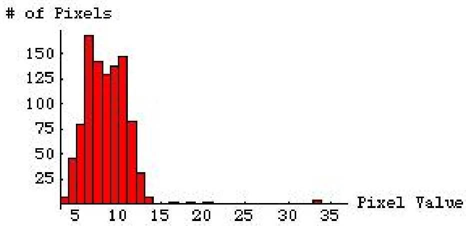
Figure 31 shows histograms of exemplary distributions of pixel values of one channel of a patch. In the first histogram (see figure 31(a)) the values are hardly effected by noise and it is therefore obvious that 229 is a representative value for this kind of distribution. The median (229) and the mean value (228.95) are very close in this case. For the distribution seen in figure 31(b) the representative value is more ambiguous. A considerable amount of pixels has the value 216, which indicates that the value we are searching for lies in between 215 and 216. The median of 215 and the mean value of 215.28 differ noticeably. Here, the mean value seems to better represent the relevance of the particular pixel values. Figure 31(c) shows a distribution were this is not the case, though. We can see that there is a huge amount of noise present and that there are even some severe outliers. These outliers were caused by a light dust particle on a dark patch. The



(a)



(b)



(c)

Figure 31: Histogram of several exemplary distributions of pixel values in a patch.

mean value (8.11) is considerably influenced by these outliers. In this case, the median (8) has to be favored. This shows us that neither the median nor the mean value yields reliable results in every possible case. Therefore we have to adjust the selection process to better account for the kind of data we have to deal with.

The mean value yields good results as long as it is not affected by outliers. In order to eliminate the influence of outliers, we use the following strategy to select pixel values. First, we calculate the median of all pixel values. Then, we determine the mean value of all pixel values, that lie within an interval of ± 5 of the median, assuming that the average noise of a channel is not greater than ± 5 of the ideal value. For the three distributions that we presented, the result of this method is (a) 228.95, (b) 215.28 and (c) 7.91, respectively. For (a) and (b) the result has not changed because there were no outliers present in these cases but for the third case (c) we can see a difference. The value is lower than the mean value and the median which indicates that they were biased by the outliers. Applying this method to all three channels of all patches will therefore give us representative RGB values for each patch.

We use the publicly available lcms library [lcm06] for color conversion to $L^*a^*b^*$ color space. The ICC profile for Adobe RGB color space conversions is available on MAC systems. We chose D50 as light source because it is commonly used in color management software.

6.1.3 Image Averaging

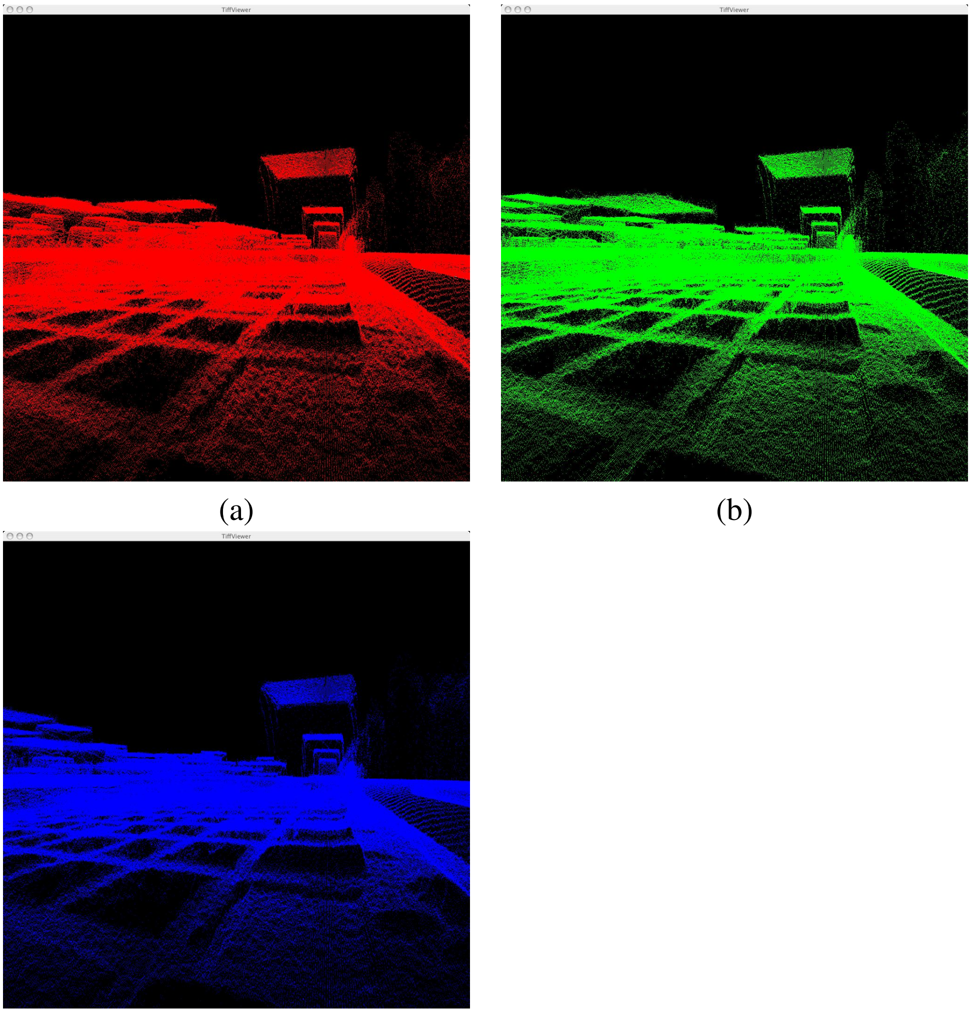
In order to further enhance the quality of our photographs we use image averaging to reduce random noise. Therefore, we take multiple images at the exact same settings and then average the pixel values. Random fluctuations above and below actual image data will gradually even out as we average more and more images. The magnitude of noise fluctuation drops by the square root of the number of images averaged. This is based on the law of large numbers [Wei99], that implies that the average of a large number of independent measurements of a random quantity tends toward the theoretical average of that quantity.

We took 25 images of a color chart in order to investigate the amount of benefit we get from averaging more or less images and compared the following 6 test cases:

1. no averaging at all, i.e. the first image only
2. averaging the first 2 images
3. averaging the first 4 images
4. averaging the first 8 images
5. averaging the first 16 images
6. averaging all 25 images

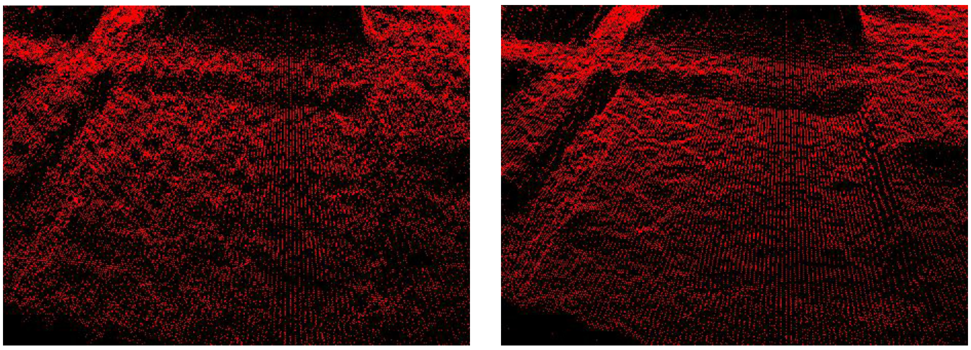
We implemented a 3D viewer for TIFF images in order to be able to visually inspect our photographs. The red, green and blue channels of an image can be viewed as height field. Figure 32 shows the red, green and blue channel of a photograph of a color chart. In the front one can see the low values of a black patch for all three channels. A closeup view of the red channel (see figure 33(a)) shows that there is no systematic error present in the data but that the variance is caused by noise. After averging 25 images (see figure 33(b)), the area appears smoother than without any averaging.

Figure 34 shows three plots that emphasize this observation. We calculated the standard deviation for the pixel values of the subareas we use for finding the



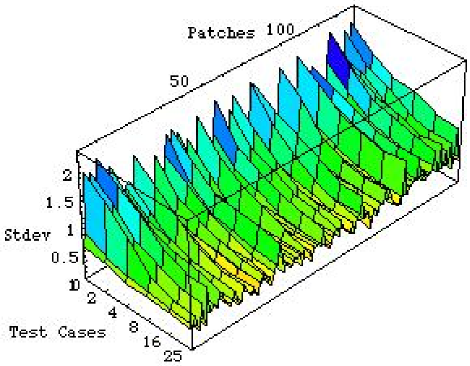
(a) (b) (c)

Figure 32: (a) Red, (b) green and (c) blue channel of a photograph of the color chart (without image averaging).

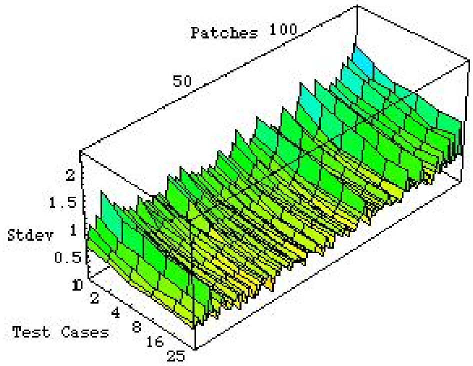


(a) (b)

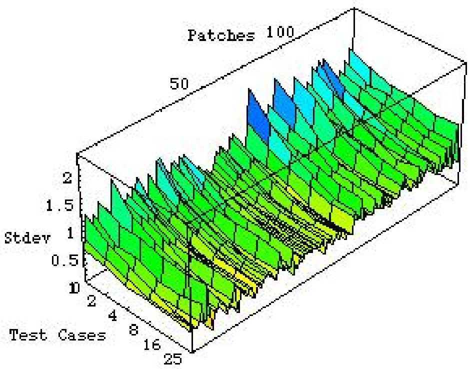
Figure 33: A closeup view of the red channel of (a) a single image and (b) an averaged image.



(a)



(b)



(c)

Figure 34: Standard deviations of all 140 patches for taking only one photograph and averaging either 2, 4, 8, 16 or 25 photographs. The results for the (a) red, (b) green and (c) blue channel are shown separately.

representative pixel value. We did these for all 6 test cases and treated the R, G, and B channel separately. A higher standard deviation indicates that there is more noise in the image than in one with a lower standard deviation. We can clearly see that the standard deviation tends to decrease with an increasing number of images averaged.

As one can see in table 2, the median and the mean values of the standard deviations for each test case is decreasing as expected, showing that there is random noise present in a photograph that can be reduced by averaging multiple images.

	1	2	4	8	16	25
R Median	1.08	0.85	0.70	0.62	0.56	0.55
G Median	0.95	0.77	0.66	0.59	0.55	0.54
B Median	1.18	0.92	0.75	0.65	0.59	0.56
R Mean Value	1.21	0.97	0.78	0.68	0.61	0.58
G Mean Value	0.97	0.81	0.68	0.62	0.58	0.57
B Mean Value	1.19	0.95	0.78	0.68	0.62	0.59

Table 2: Median and mean values of standard deviations of all 140 patches for all six test cases, for R, G, and B channel, respectively.

6.1.4 RAW image

The Canon EOS 20D offers RAW image support. In addition to the further processed JPEG image, it also provides a file of type CR2, a manufacturer specific format. In order to obtain the raw RGB values we use the publicly available software dcraw [raw06a].

6.2 Light Cabinet

For an accurate mapping of RGB values to another color space (see section 7), it is crucial that the illumination of the color chart is as evenly distributed as possible while taking the photograph. For example, on a GretagMacbeth ColorChecker SG, several equally white patches are placed in the border region. If the light intensity on one side differs from the light intensity of the other side, this will result in different RGB values for the same kind of white patch. Therefore, we have to ensure that the difference in illumination distribution is minimized.

In order to find the optimal position for placing the color chart, we measured the light distrubution inside the light cabinet (see appendix C.1). We defined 80 measurement spots on the bottom of the cabinet (see figure 35). They were only placed in the left part of the cabinet because we expect the cabinet to be symmetric. As the photographs would be taken at the front side of the cabinet, we measured the luminance that was reflected from the spots to the upper part of the front side. For example, the measurement device was placed in the upper left corner of the front side when the leftmost column was measured. For the second column, the device was moved a bit to the right until it was aligned with this column. The rest of the front side was covered with cardboard to avoid external light. A Konica Minolta LS-110 luminance meter (see appendix C.2) was used for taking the

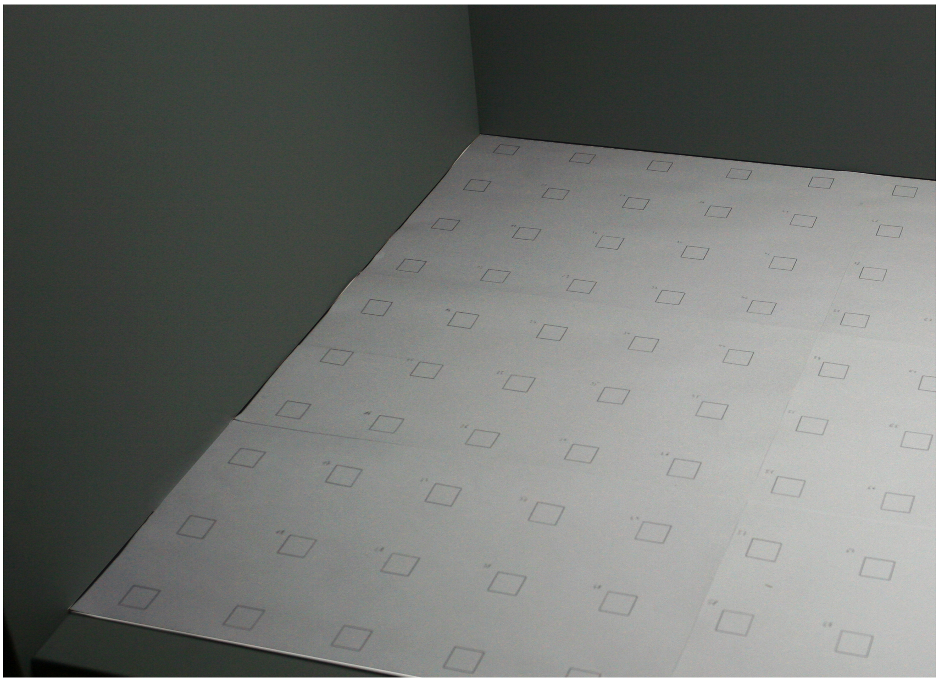
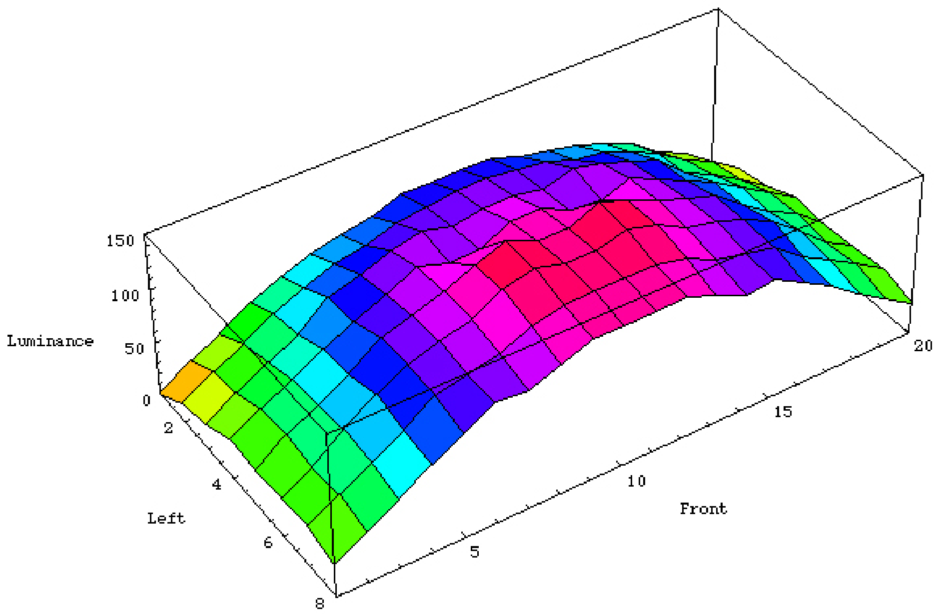


Figure 35: Eighty measurement spots were definied on the bottom of the light cabinet.

measurements. We tested three lightsources: D65, 840 and CWF. Lightsource 840 is also known as TL84 or CIE Illuminant F11; lightsource CWF is also known as CIE Illuminant F2. All lightsources were tested at several dimmer settings.

For each set of measurement points we subtracted the smallest value from all of the samples to get the absolute difference in luminance distribution. Figure 36 shows a Mathematica [mat06] plot of the results of D65 at 100% and a photograph of the light cabinet made at approximately the same view point as the plot. Different colors denote different height values. The color is selected according to the height of the center of a square and runs through red, yellow, green, cyan, blue, magenta and back to red. It can be seen that for these settings the brightest area is in the middle of the front part of the light cabinet. The maximum difference in luminance over the whole area is 146.70 cd/m^2, which is the difference between the minimum luminance of 173.20 cd/m^2 and the maximum luminance of 319.90 cd/m^2.

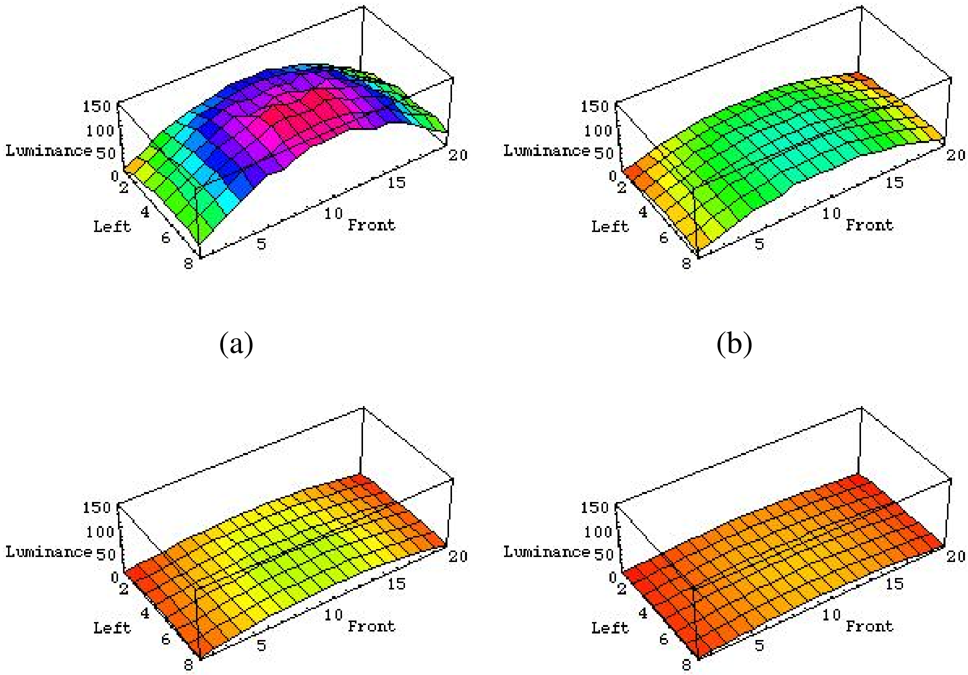
Figure 37 shows the luminance difference distribution of D65 for dimmer settings 100%, 50%, 25% and 15%. We can see that the absolute difference in luminance decreases noticeably with decreasing light intensity. Figure 38 shows



(a)

(b)

Figure 36: (a) Luminance difference distribution of D65 at 100%, (b) a photograph of the light cabinet made at approximately the same view point as the plot of the luminance difference distribution.



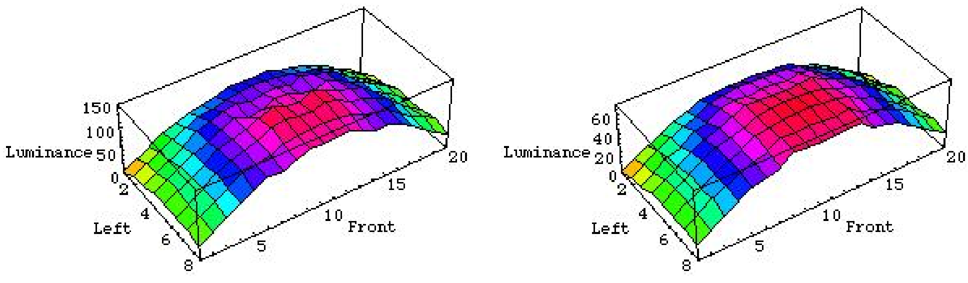
(a) (b)

(c) (d)

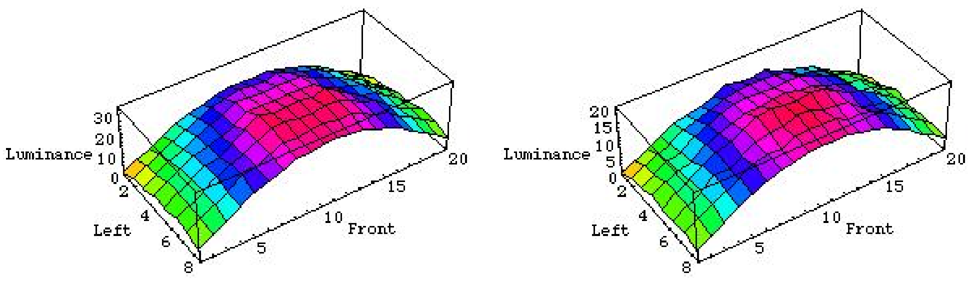
Figure 37: (a) Luminance difference distribution of D65 at 100%, (b) 50%, (c) 25% and (d) 15% with equal scaling on the luminance axes.

the same values, but the luminance axes of each plot are scaled to the maximum luminance value of the particular plot (for example, the maximum difference in luminance for 15% is 19.75 cd/m^2). This indicates that the shape of luminance difference distribution for different dimmer settings is quite similar with respect to the maximum luminance of a setting.

We then wanted to find out, whether the amount of decrease in luminance is directly proportional to the dimmer setting. Therefore, we multiplied all values of a lower dimmer setting by the reciprocal factor that was used as dimmer setting, e.g. we multiplied by 2 for dimmer setting equal to 50%. Figure 39 (a) shows the result of this multiplication, while (b) shows the difference to the values measured at 100% light intensity. The maximum absolute error is 17.92. Figures 40 and 41 show corresponding plots for 25% and 15%. The maximum absolute error is 24.58 for 25% and 25.07 for 15%, respectively. The values of 25% and 15% indicate that there is a difference in luminance distribution for different dimmer settings, that is not only related to the dimming factor.

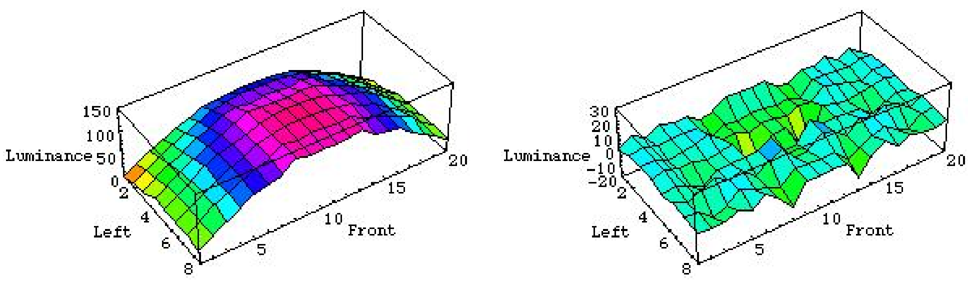


(a) (b)



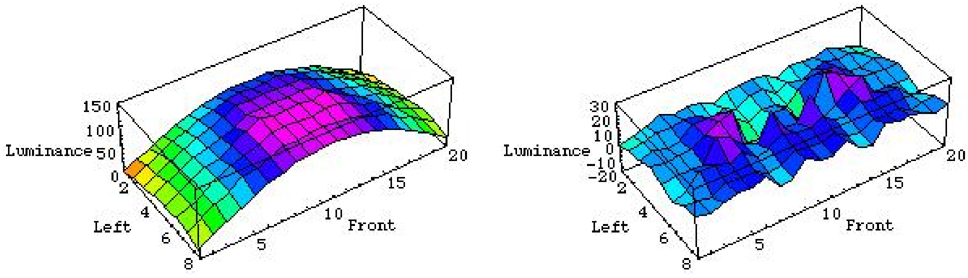
(c) (d)

Figure 38: (a) Luminance difference distribution of D65 at 100%, (b) 50%, (c) 25% and (d) 15%, but unlike the plots in figure 37 the luminance axes of each plot are scaled to the maximum luminance value of the particular plot.



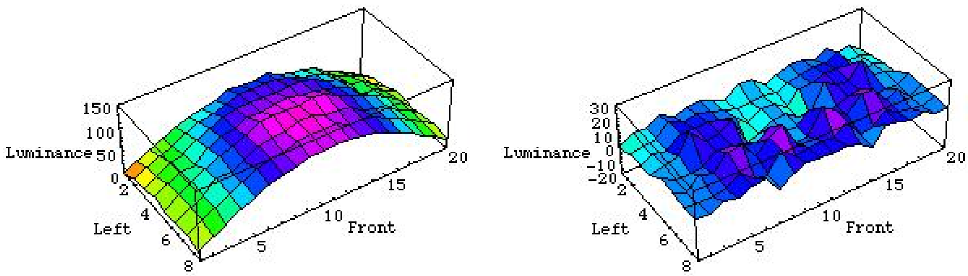
(a) (b)

Figure 39: (a) Luminance difference distribution of D65 at 50%, scaled by a factor of 2, (b) the difference of 100% light intensity (figure 38(a)) and of the values shown in (a).



(a) (b)

Figure 40: (a) Luminance difference distribution of D65 at 25%, scaled by a factor of 4, (b) the difference of 100% light intensity (figure 38(a)) and of the values shown in (a).



(a) (b)

Figure 41: (a) Luminance difference distribution of D65 at 15%, scaled by a factor of $100/15$, (b) the difference of 100% light intensity (figure 38(a)) and of the values shown in (a).

i \ j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	26.16	24.57	23.32	19.84	16.31	14.74	13.69	11.59	11.59	13.69	14.74	16.31	19.84	23.32	24.57	26.16
1	25.12	23.67	22.08	18.92	16.05	14.10	12.97	13.37	13.37	12.97	14.10	16.05	18.92	22.08	23.67	25.12
2	25.53	23.76	21.06	17.64	14.77	12.51	11.19	12.93	12.93	11.19	12.51	14.77	17.64	21.06	23.76	25.53
3	24.76	22.90	19.73	16.26	14.02	11.61	9.52	10.06	10.06	9.52	11.61	14.02	16.26	19.73	22.90	24.76
4	25.63	23.80	19.67	16.35	14.17	11.43	9.12	**6.28**	**6.28**	9.12	11.43	14.17	16.35	19.67	23.80	25.63

Table 3: Standard deviation of all 80 submatrices for D65 and dimmer setting 100%.

In order to find the optimal position for placing the color chart, we were searching for a subarea of the size of a color chart, where the standard deviation of the difference in luminance distribution is minimal. The GretagMacbeth Color Checker SG has 4x5 white patches in the boarder area. As we chose our measurement points according to the distance between two of those white patches, we have to find the 4x5 submatrix in our 8x20 matrix of measurements, that has the smallest standard deviation. The number of possible candidates is 80, as we have 5 submatrices in vertical direction and 16 submatrices in horizontal direction. We simply calculate the standard deviation for all possible submatrices and choose the one with the smallest result. Tables 3, 4, 5 and 6 show the results of these calculations; the smallest results are emphasized. For the submatrices with the most even luminance difference distribution the standard deviations are 6.28 for dimmer setting 100%, 1.61 for 50%, 1.12 for 25% and 0.96 for 15%. Figure 42 shows Mathematica plots of the same results. These results clearly show that for D65 the best position for the color chart is in the middle part of the front region.

Our results for lightsource 840 at dimmer settings 100% and 50% are shown in figure 43 (a) and (b). Again we scaled the values of 50% by 2 (see figure 43 (c)) and visualized the difference to the values of 100% (d). Figure 44 shows the according plots for lightsource CWF. Standard deviation for dimmer settings 100% and 50% are shown in figures 45 (840) and 46 (CWF). The maximum values are 40.39 cd/m^2 (840) and 40.30 cd/m^2 (CWF), which is considerably higher than 26.16 cd/m^2 for D65. The lowest values for 840 are also in the middle part

i \ j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13.15	12.40	11.77	11.18	9.88	8.80	7.90	7.92	7.92	7.90	8.80	9.88	11.18	11.77	12.40	13.15
1	13.16	12.13	10.96	10.00	8.44	7.09	5.75	5.60	5.60	5.75	7.09	8.44	10.00	10.96	12.13	13.16
2	13.00	11.72	10.44	9.21	7.27	6.00	4.43	4.23	4.23	4.43	6.00	7.27	9.21	10.44	11.72	13.00
3	12.57	11.26	9.62	8.55	6.84	5.26	3.22	**1.61**	**1.61**	3.22	5.26	6.84	8.55	9.62	11.26	12.57
4	12.85	11.30	9.52	8.64	6.91	5.61	3.84	1.98	1.98	3.84	5.61	6.91	8.64	9.52	11.30	12.85

Table 4: Standard deviation of all 80 submatrices for D65 and dimmer setting 50%.

i \ j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	5.79	5.30	4.90	4.63	4.25	3.74	3.31	3.05	3.05	3.31	3.74	4.25	4.63	4.90	5.30	5.79
1	5.55	4.97	4.67	4.71	4.23	3.48	2.79	2.31	2.31	2.79	3.48	4.23	4.71	4.67	4.97	5.55
2	5.55	4.98	4.64	4.68	4.14	3.29	2.30	1.71	1.71	2.30	3.29	4.14	4.68	4.64	4.98	5.55
3	5.49	4.90	4.61	4.53	3.88	3.04	1.89	**1.12**	**1.12**	1.89	3.04	3.88	4.53	4.61	4.90	5.49
4	5.74	5.21	4.91	4.59	3.67	2.77	1.67	1.15	1.15	1.67	2.77	3.67	4.59	4.91	5.21	5.74

Table 5: Standard deviation of all 80 submatrices for D65 and dimmer setting 25%.

i \ j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	3.31	3.23	3.12	3.04	2.54	2.32	2.10	1.77	1.77	2.10	2.32	2.54	3.04	3.12	3.23	3.31
1	3.31	3.15	3.07	3.07	2.50	2.15	1.80	1.59	1.59	1.80	2.15	2.50	3.07	3.07	3.15	3.31
2	3.37	3.04	2.94	2.82	2.23	1.84	1.38	1.25	1.25	1.38	1.84	2.23	2.82	2.94	3.04	3.37
3	3.34	3.02	2.87	2.58	2.12	1.68	1.18	**0.96**	**0.96**	1.18	1.68	2.12	2.58	2.87	3.02	3.34
4	3.55	3.13	2.92	2.42	2.15	1.67	1.17	**0.96**	**0.96**	1.17	1.67	2.15	2.42	2.92	3.13	3.55

Table 6: Standard deviation of all 80 submatrices for D65 and dimmer setting 15%.

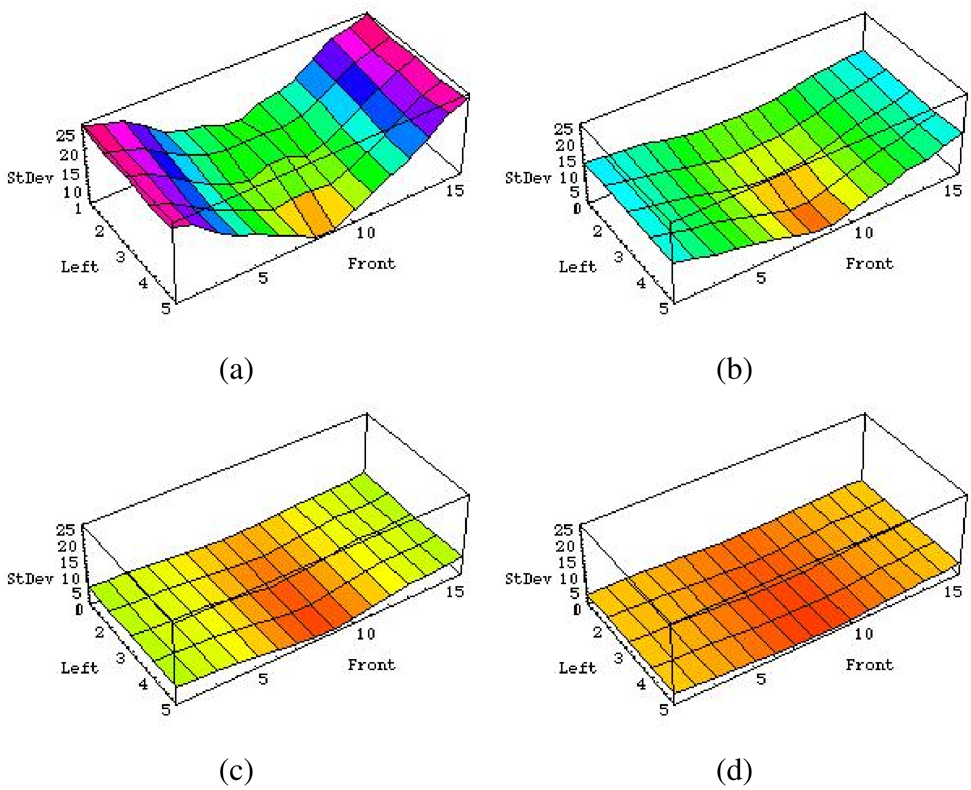
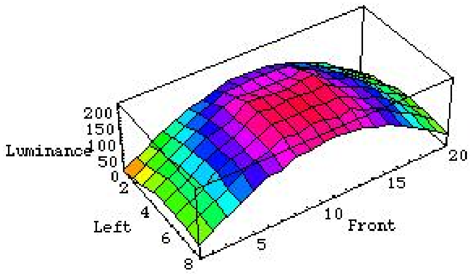
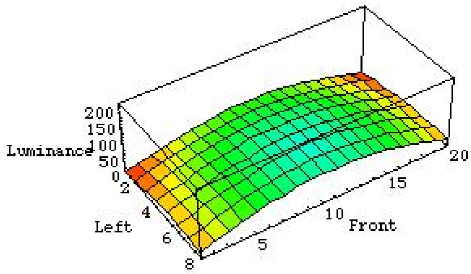


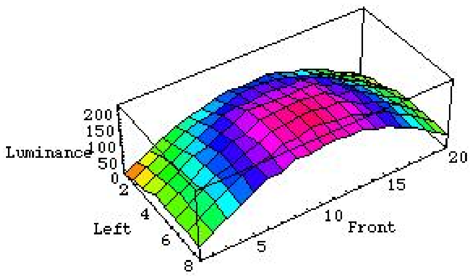
Figure 42: Standard deviation of all 80 submatrices for dimmer settings (a) 100%, (b) 50%, (c) 25% and (d) 15% and lightsource D65.



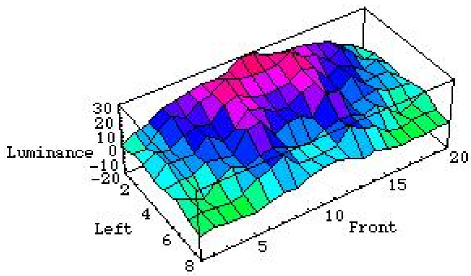
(a)



(b)



(c)



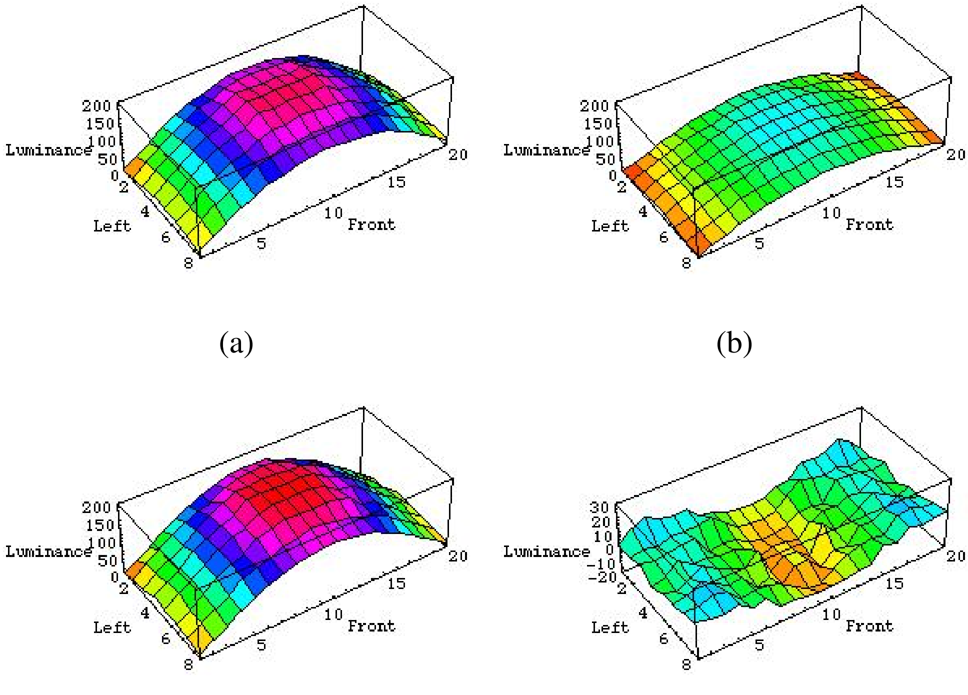
(d)

Figure 43: (a) Luminance difference distribution of 840 at 100% and (b) 50%; (c) luminance difference distribution of 840 at 50%, scaled by a factor of 2, (d) the difference of the values shown in (a) and (c).

of the front region (like for D65), whereas for CWF they lie in the center of the light cabinet.

We did not test all dimmer settings for lightsources 840 and CWF, because we were mainly interested in the type of luminance difference distribution. If it would have been significantly better than for D65, we might have considered to use one of these lightsources, but there is a more severe reason why we chose D65. CIE recommends D65 as illuminant for color matching appraisal. It is specified for applications where there is a need to maintain color consistency and quality. Lightsources 840 and CWF are known as European and American "point of sale" illuminant with good to moderate color rendering but they are not explicitly recommended for color measurements. A closer look on the lightsource spectra (see figure 47) shows that the spectrum of D65 is much more evenly distributed than those of 840 and CWF.

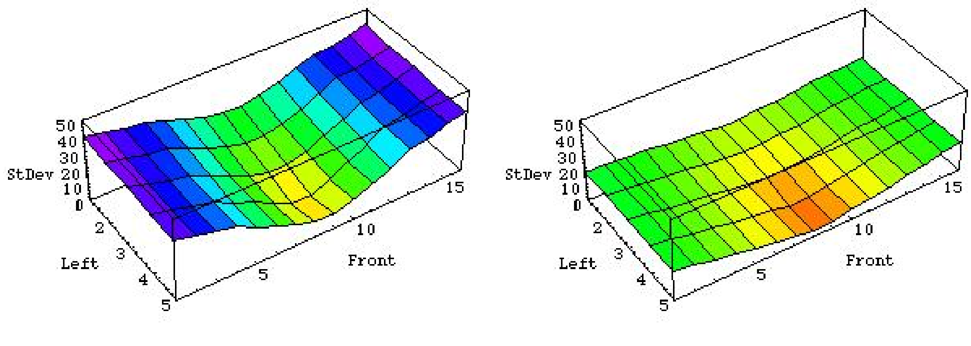
We tested three additional positions of the color chart where we thought that



(a) (b)

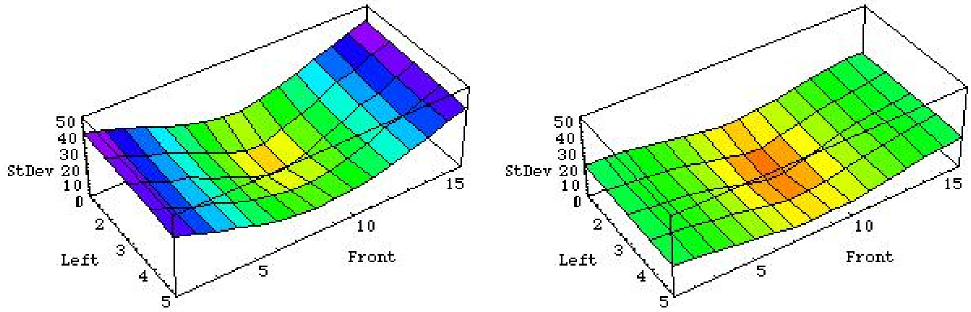
(c) (d)

Figure 44: (a) Luminance difference distribution of CWF at 100% and (b) 50%; (c) luminance difference distribution of CWF at 50%, scaled by a factor of 2, (d) the difference of the values shown in (a) and (c).



(a) (b)

Figure 45: Standard deviation of all 80 submatrices for dimmer settings (a) 100% and (b) 50% and lightsource 840.



(a) (b)

Figure 46: Standard deviation of all 80 submatrices for dimmer settings (a) 100% and (b) 50% and lightsource CWF.

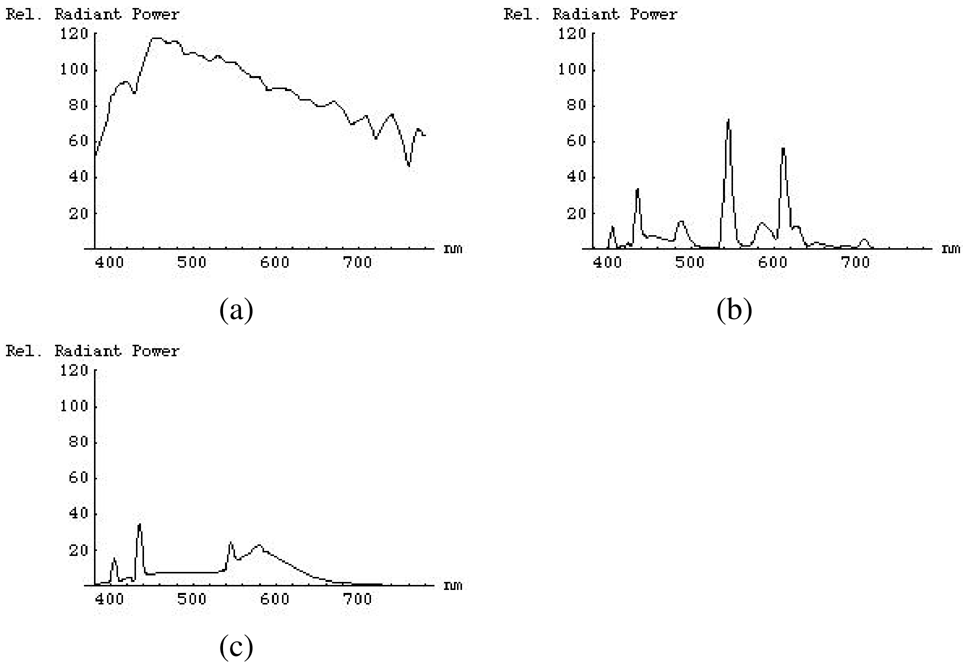
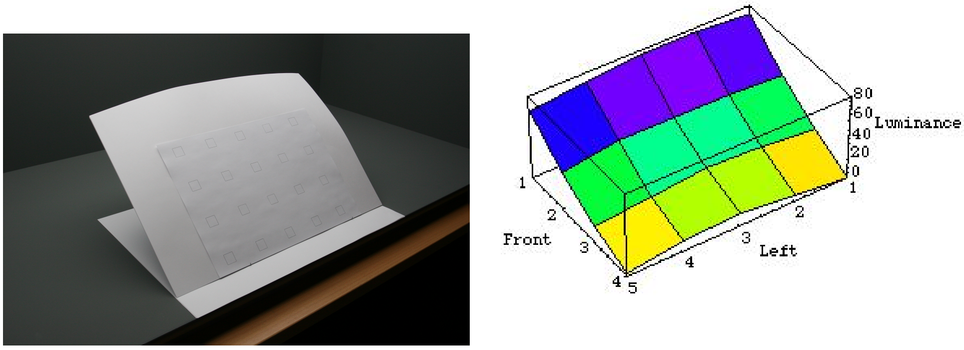


Figure 47: Lightsource spectra of illuminants (a) D65, (b) 840 and (c) CWF.



(a) (b)

Figure 48: (a) The measurement points at $45°$; (b) absolute difference in luminance for the 20 measurement points, accordingly.

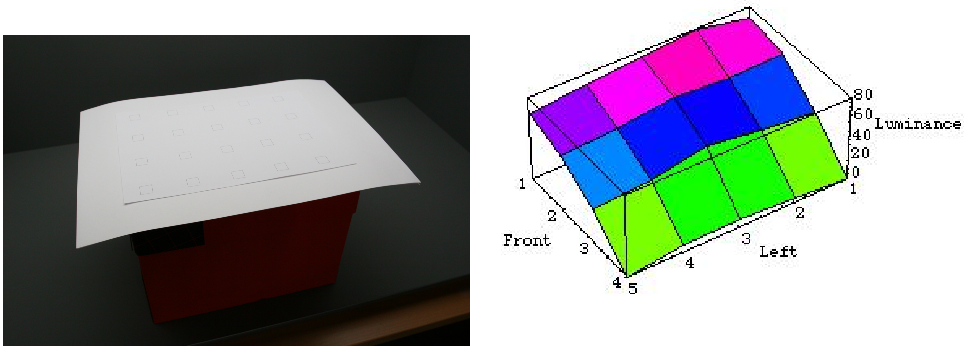
the luminance difference distribution might be more even:

- we rotated it by $45°$,
- we placed it at a height of $24.5cm$ and
- we put it inside a black box.

Figures 48, 49 and 50 show (a) a photo of the measurement setup and (b) the relative error in luminance, respectively. During the measurement process the light cabinet again was covered with cardboard. The standard deviations are 26.97, 27.93 and 7.39, respectively. All three tests did not bring the desired improvement, as even the setup including the black box has a higher standard deviation than the plain light cabinet at 100% (standard deviation: 6.28).

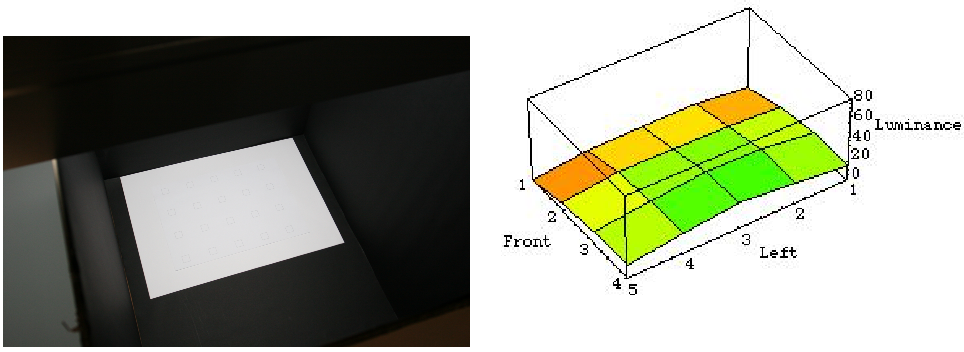
In order to decide which dimmer setting to use, we took one photograph each at dimmer settings 100%, 50%, 25% and 15%. The camera settings used are shown in table 7. As the exposure time was twice as long when the light intensity was halved, the images theoretically should be equal, or at least very similar. We used two methods to judge the quality of the images:

- We opened the images in GretagMacbeth ProfileMaker "Camera" Tab, because in one of our methods (see section 7.2) we use this software. ProfileMaker checks, if the image can be used for generating an ICC profile (e.g. it analyzes the luminance difference distribution on the ColorChecker). We



(a) (b)

Figure 49: (a) The measurement points placed at a height of $24.5cm$; (b) absolute difference in luminance for the 20 measurement points, accordingly.



(a) (b)

Figure 50: (a) The measurement points put inside a black box; (b) absolute difference in luminance for the 20 measurement points, accordingly.

	100%	50%	25%	15%
Aperture	4.0	4.0	4.0	4.0
Exposure time	1/30	1/15	1/8	1/4

Table 7: Camera settings at 100%, 50%, 25% and 15%.

	100%	50%	25%	15%
R	14.06	14.53	15.57	19.35
G	13.61	14.56	14.86	17.79
B	14.16	14.77	15.62	18.62

Table 8: Interval length of R, G and B values of dark border patches at 100%, 50%, 25% and 15%.

assume, that if an image passes this test, it can be considered suitable for characterization purposes.

- We did an analysis of the pixel values themselves.

ProfileMaker rated the image at 100% as being of good enough quality but for the other images it displayed the following error message: "The dark border patches are not even illuminated across the chart." The white and dark patches that lie on the border of the GretagMacbeth ColorChecker SG as well as the gray ramps in the middle are used by ProfileMaker to judge the quality of the photograph in respect to usage for ICC profile generation. As the error message is evoked by the dark border patches, we further analyzed those values. Table 8 shows the interval length (i.e. the difference of the maximum and the minimum value) of R, G and B channel for all 14 dark border patches. We can clearly see that the interval length increases with decreasing light intensity caused by noise due to longer exposure times.

As concluding result of our light cabinet measurements we can say that for our purpose the best position for the color chart is the middle part of the front region of the light cabinet without further adjustments. The optimal setting for the light source is D65 with dimmer set to 100%.

6.3 Spectral Measurements

We use a Gretag Macbeth Color Checker SG (see appendix C.3), as it is especially designed for creating ICC profiles for digital cameras. We measured the spectrum of each patch using a spectrophotometer – a Gretag Macbeth Spectrolino (see appendix C.4). The measurement range is from $380nm$ to $730nm$ by steps of $10nm$. Comparable devices cost about 770,- to 1300,- Euros, depending on the software included and additional functionality. Though, we assume that a similar device is available at each facility that develops a physically based rendering engine, as it is necessary to obtain the spectra of surfaces and light sources that are required as input data to the rendering process. To speed up the process of measuring a chart, we obtained a GretagMacbeth SpectroScan T (see appendix C.5) for automatic measurements. Each spectrum was converted to XYZ space and then to $L^*a^*b^*$ color space, using D50 as light source.

7 Various Approaches of Using a Digital Camera for Verification Purposes

7.1 Using the Camera's Color Space

Prosumer-grade cameras – like the Canon Eos 20D – usually offer the possibility of specifying the color space of the RGB values of the output image. As the camera manufacturer knows all the details of the used components, the color reproduction should be quite accurate. In order to find out the grade of correctness of the reproduction, we took an image of a color chart and compared the color values of the photograph to the actual color values of the chart.

We first obtained the color values of the color chart (see section 6.3). Then, we took a photograph of the color chart as described in section 6.1. The comparison was performed in $L^*a^*b^*$ color space. Therefore, the colors of both the color chart and the image were converted to $L^*a^*b^*$ color space.

We can now choose between several possible forms of output of our digital camera, as section 4.1 explains in more detail. We cannot use RAW data for this approach though, because it is not converted to a color space, yet. We will therefore investigate the following test cases:

1. an 8-bit TIFF image generated by converting the RAW image to 8-bit TIFF using a RAW converter,
2. a JPEG image as provided by the camera and
3. a 16-bit TIFF image, also generated with a RAW converter.

The Canon Eos 20D has a 12-bit CCD sensor. Hence, converting to 8-bit means a loss of information. We include the 8-bit case to simulate cameras that only provide 8-bit data. The JPEG image contains all the image enhancement that is automatically done by the camera. It is therefore probably the most inaccurate image, but we include it for comparison and as an example for cameras that do not provide RAW format. The first and the third test case only differ in the number of bits of the TIFF image.

For each test case, we averaged the RGB values of each patch as described in section 6.1. This gives us 140 RGB values per test case. As the camera's

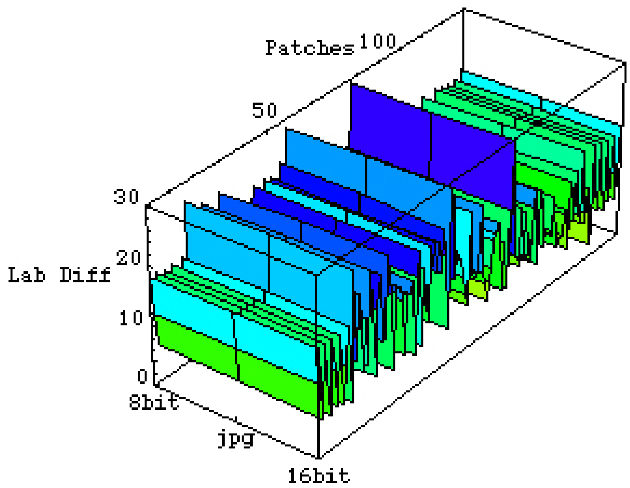


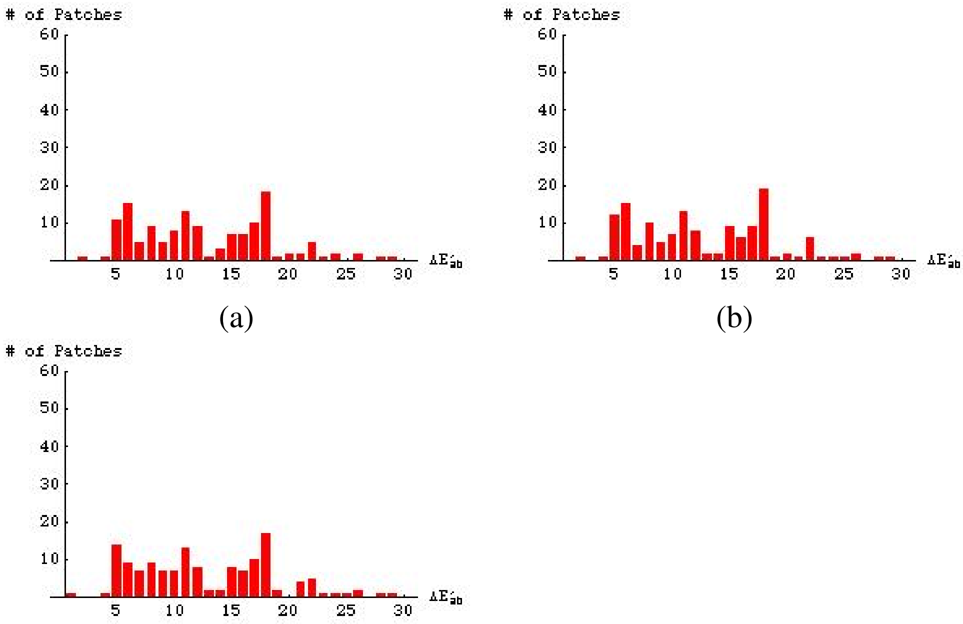
Figure 51: $L^*a^*b^*$ differences for the three test cases (8 bit, jpg and 16 bit images) for all 140 color patches.

	8 bit	jpg	16 bit
Min	2.77	2.28	1.85
Max	29.21	29.26	29.39
Median	12.32	12.32	12.34
Mean	13.30	13.31	13.33
Stdev	5.87	5.94	5.91

Table 9: Minimum and maximum, median, mean values, and standard deviations of $L^*a^*b^*$ difference values for using the camera's color space.

color space was set to Adobe RGB, we use the appropriate ICC profile for converting RGB values to $L^*a^*b^*$ values. Color space conversions are done using lcms library [lcm06]. Now, we calculate the $L^*a^*b^*$ difference between the values measured with the spectrophotometer and the values calculated from each of the photographs. As figure 51 shows, the results are very similar for the three test cases. Minimum and maximum, median and mean values, and the standard deviations are shown in table 9. The values of all 140 patches can be found in appendix E.1. The distribution of the $L^*a^*b^*$ differences is plotted in form of a histogram in figure 52 for all three test cases. We can see that most $L^*a^*b^*$ differences lie between 5 and 19 ΔE^*_{ab}.

In order to find out which of the colors tend to cause high $L^*a^*b^*$ differences,



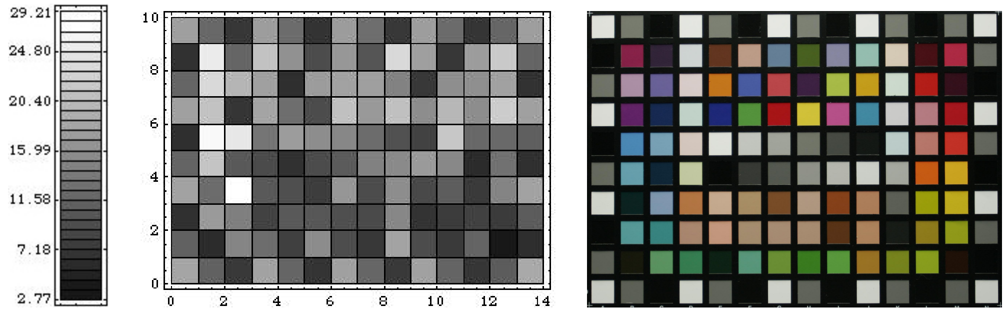
(a) (b) (c)

Figure 52: Histograms of $L^*a^*b^*$ differences for using the camera's color space for test cases (a) 8bit, (b) jpg and (c) 16bit.

we made a density plot of all values of test case one, ordered in the same way as on the color chart (see figure 53). We can see that $L^*a^*b^*$ differences are generally low for dark patches and greenish patches like the ones in the bottom right corner. Purple, red, and light blue patches seem to be a problem for color measurement.

As we found a tendency of certain colors to be reproduced less accurate than others, we wanted to investigate, whether we can confirm this assumption. We therefore compared measured L^*, a^*, and b^* values to the ones we calculated by using the Adobe RGB ICC profile. Figure 54 shows the according plots. As L^* being the lightness of a color we can say that for high L^* values there is a tendency to underestimation. For a^* values, that correspond to the redness and greenness of a color we can see no clear color shift, but there is a clear overestimation for b^*, corresponding to the yellowness and blueness of a color.

To conclude, we can say that the mean and maximum values of all test cases are quite high and can therefore not be considered as accurate measurements.



(a) (b)

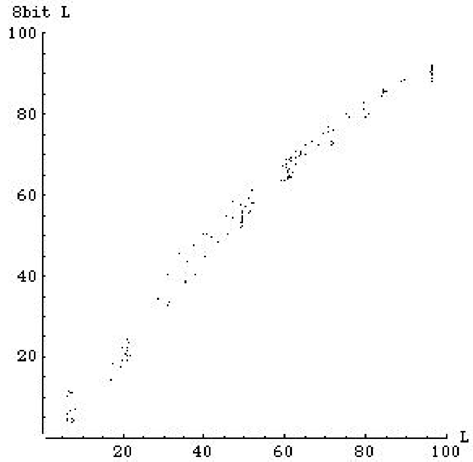
Figure 53: (a) Density plot of the $L^*a^*b^*$ differences for using the camera's color space for test case 1 (8bit). Dark areas mean low $L^*a^*b^*$ differences, while bright areas mean high $L^*a^*b^*$ differences; (b) a photograph of the test chart.

7.2 Using an ICC Profile

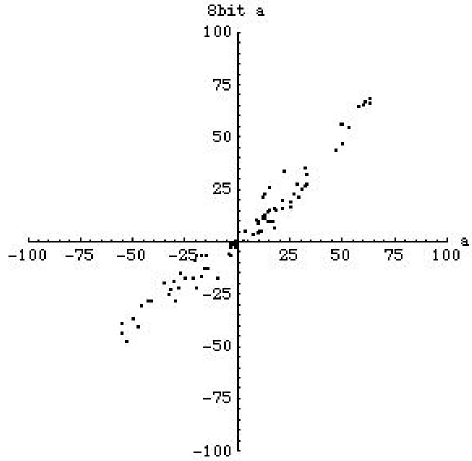
Instead of using the built-in color management we now generated an ICC profile for our camera. We assumed that the built-in software is based on the average of fluctuations due to the manufacturing process. Therefore, we supposed that actually measuring the behavior of our camera would give more accurate results. ICC profiles for digital cameras can be generated without the use of a spectrophotometer, as long as the software supports the test target that is used. Publicly available profiling software often performs worse than commercial software [FS02]. We used Gretag Macbeth Profile Maker 5.0.5 (around 2000,- Euros).

We created three ICC profiles for our camera based on the three test cases described in section 7.1. For each test case, we used the profile together with the lcms library to convert the average RGB values of the patches to $L^*a^*b^*$ color space. The histograms of the three test cases (see figure 55) show, that the result is considerably better than with only using the camera's color space, as the maximum values are much lower when an ICC profile is used. Median and mean values – that can be found in table 10, together with the maximum and minimum values and the standard deviations – are only slightly lower, though. Detailed results for all 140 patches are listed in appendix E.2.

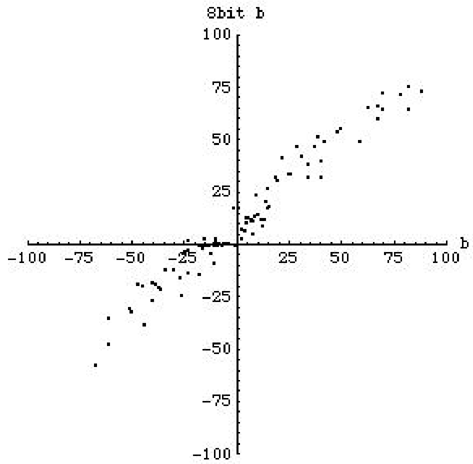
The density plot (see figure 56) for test case 1 shows that dark patches cause fewer problems than bright patches. There is no clear tendency to a color as in the previous approach, though. The scatter plots for L^*, a^*, and b^* (see figure 57(c)) show that the tendency to underestimation of L^* and to a shift of color for b^* is also present in this approach.



(a)



(b)

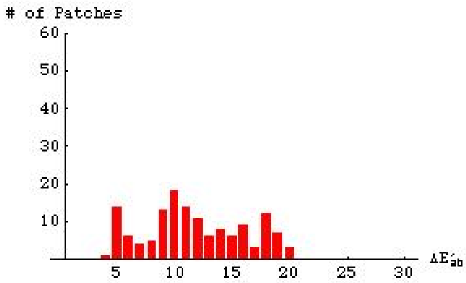


(c)

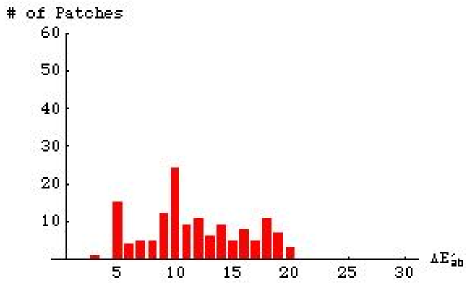
Figure 54: Scatter plots of (a) L^*, (b) a^*, and (c) b^* values for using the camera's color space for test case 1 (8bit), comparing measured values to reproduced values.

	8 bit	jpg	16 bit
Min	4.05	3.82	3.65
Max	20.96	20.95	20.69
Median	11.24	11.30	11.21
Mean	12.25	12.23	12.20
Stdev	4.33	4.35	4.33

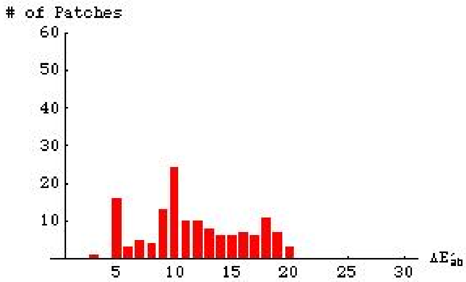
Table 10: Minimum and maximum, median, mean values and standard deviations of $L^*a^*b^*$ difference values for using an ICC profile.



(a)

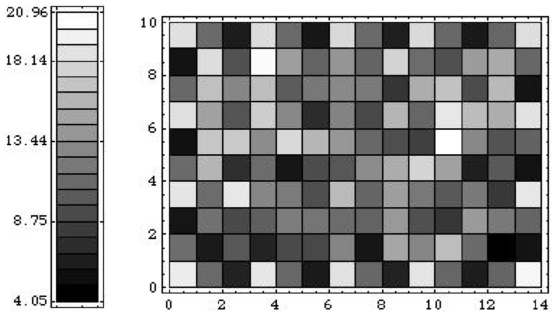


(b)

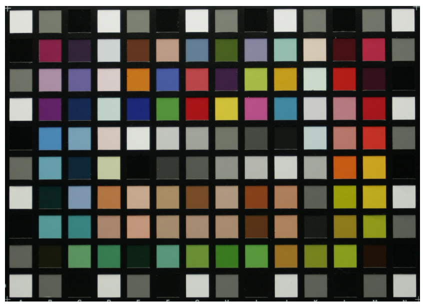


(c)

Figure 55: Histograms of $L^*a^*b^*$ differences for using an ICC profile for test cases (a) 8bit, (b) jpg and (c) 16bit.



(a)



(b)

Figure 56: (a) Density plot of the $L^*a^*b^*$ differences for using an ICC profile for test case 1 (8bit). Dark areas mean low $L^*a^*b^*$ differences, while bright areas mean high $L^*a^*b^*$ differences; (b) a photograph of the test chart.

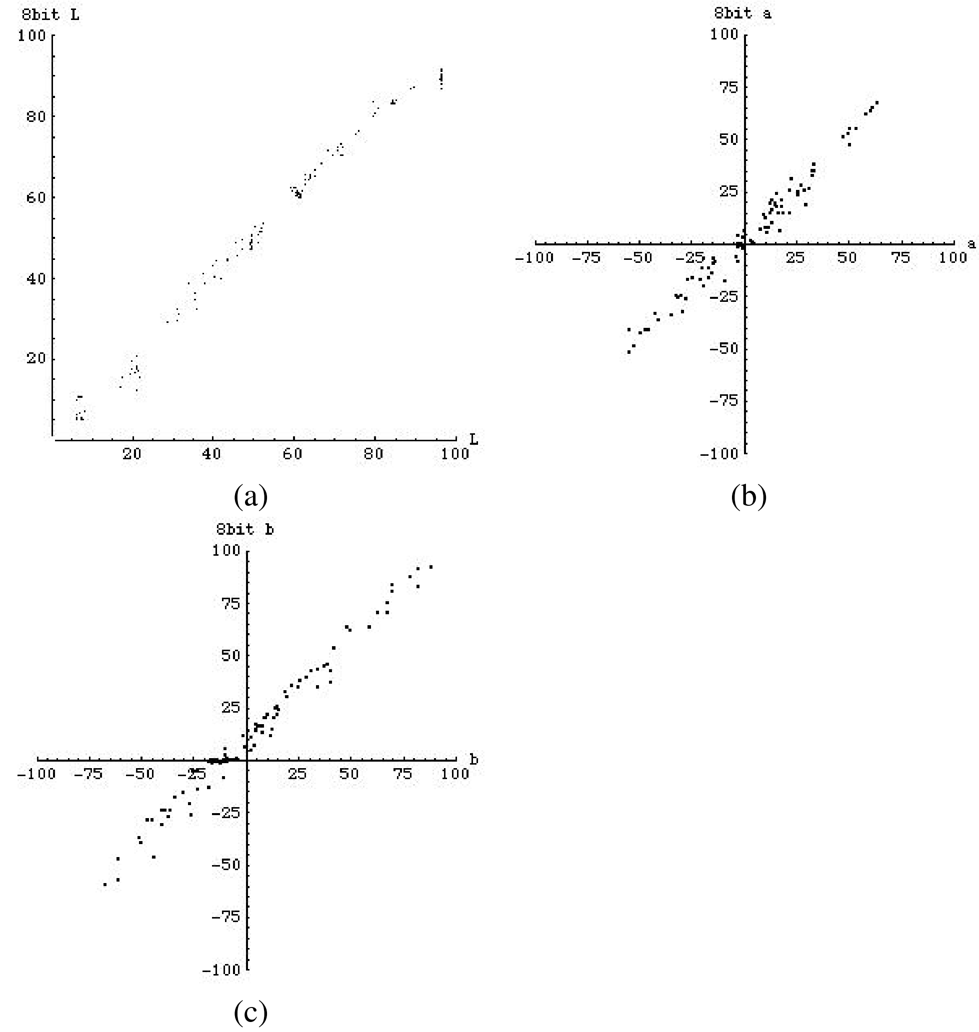


Figure 57: Scatter plots of (a) L^*, (b) a^*, and (c) b^* values for using an ICC profile for test case 1 (8bit), comparing measured values to reproduced values.

Using an ICC profile obviously improves the quality of camera characterization compared to relying on the camera's color space only. Although the mean ΔE^*_{ab} value is only slightly lower, the maximum value could be reduced by around $10\Delta E^*_{ab}$.

7.3 Implementing an Adapted Mapping

In order to be able to convert the RGB values of a photograph to XYZ space one has to find a transformation that provides a best agreement between these RGB values and the measured XYZ values. ICC profiles internally use either look-up-tables or matrices to implement this transformation. We implemented a mapping by ourselves for two reasons: first, this mapping possibly yields better results because we can better address particular properties of our data and secondly, because we also wanted to provide an approach that does not rely on expensive profiling software, but gives more accurate results than using the camera's color space.

Izadan and Nobbs [Iza06] did a comparison between iteration and regression methods using either XYZ or $L^*a^*b^*$ color space. Basically, all methods yield comparable results. They conclude that regression method with $L^*a^*b^*$ approach will lead to the best results.

We based our calculations on polynomial regression with least squares fitting described by Hong et al. [HLR00], who have shown that a higher order polynomial is able to produce satisfactory characterization accuracy.

Let $R \in \mathbb{R}^{n \times m}$ denote a matrix of RGB vectors and $L \in \mathbb{R}^{n \times 3}$ the corresponding matrix of $L^*a^*b^*$ vectors, the mapping from RGB to $L^*a^*b^*$ can be represented by

$$L = RM, \tag{1}$$

where the matrix $M \in \mathbb{R}^{m \times 3}$ is derived by the following equation:

$$M = (R^T R)^{-1} R^T L. \tag{2}$$

Here, n is the number of samples and m is the number of terms of the polynomials. The number and the grade of these terms define the type of the mapping. As our image is gamma corrected and therefore not linear, a higher order polynomial will most probably give better results than a linear polynomial. On the other hand, for higher order polynomials the solution may start oscillating, i.e. the polynomial has adapted to the training data too accurately and is therefore not usable for general data anymore [JHP+06]. In order to find the polynomial that gives the best results, we tested the following coefficients:

1. r, g, b
2. r, g, b, rgb, 1
3. r, g, b, rg, rb, gb
4. r, g, b, rg, rb, gb, rgb, 1
5. r, g, b, rg, rb, gb, r^2, g^2, b^2
6. r, g, b, rg, rb, gb, r^2, g^2, b^2, rgb, 1
7. r, g, b, rg, rb, gb, r^2, g^2, b^2, r^3, g^3, b^3, 1
8. r, g, b, rg, rb, gb, r^2, g^2, b^2, r^2g^2, r^2b^2, g^2b^2, $r^2g^2b^2$, r^3, g^3, b^3, r^3g^3, r^3b^3, g^3b^3, 1
9. r, g, b, rg, rb, gb, r^2, g^2, b^2, r^3, g^3, b^3, r^4, g^4, b^4, 1
10. r, g, b, rg, rb, gb, rgb, r^2, g^2, b^2, r^2g^2, r^2b^2, g^2b^2, $r^2g^2b^2$, r^3 g^3, b^3, r^3g^3, r^3b^3, g^3b^3, $r^3g^3b^3$, r^4, g^4, b^4, r^4g^4, r^4b^4, g^4b^4, $r^4g^4b^4$, 1
11. r, g, b, rg, rb, gb, r^2, g^2, b^2, r^3, g^3, b^3, r^4, g^4, b^4, r^5, g^5, b^5, 1

As the results were quite similar for the 3 test cases described in sections 7.1 and 7.2, we only analyze test case 1 – the 8bit TIFF image – from now on. Figure 58 shows histograms of the $L^*a^*b^*$ differences between the measured $L^*a^*b^*$ values and the results of using the 4 linear polynomials, whereas figure 59 shows the according histograms for the higher order polynomials. For the linear polynomials, we see that adding more terms clearly reduces the maximum error. The same effect can be observed when adding terms of higher order.

Further analyses of the $L^*a^*b^*$ differences give us the results presented in tables 11 (linear polynomials) and 12 (higher order polynomials). Adding either linear terms or terms of higher order improves the mapping noticeably. While the

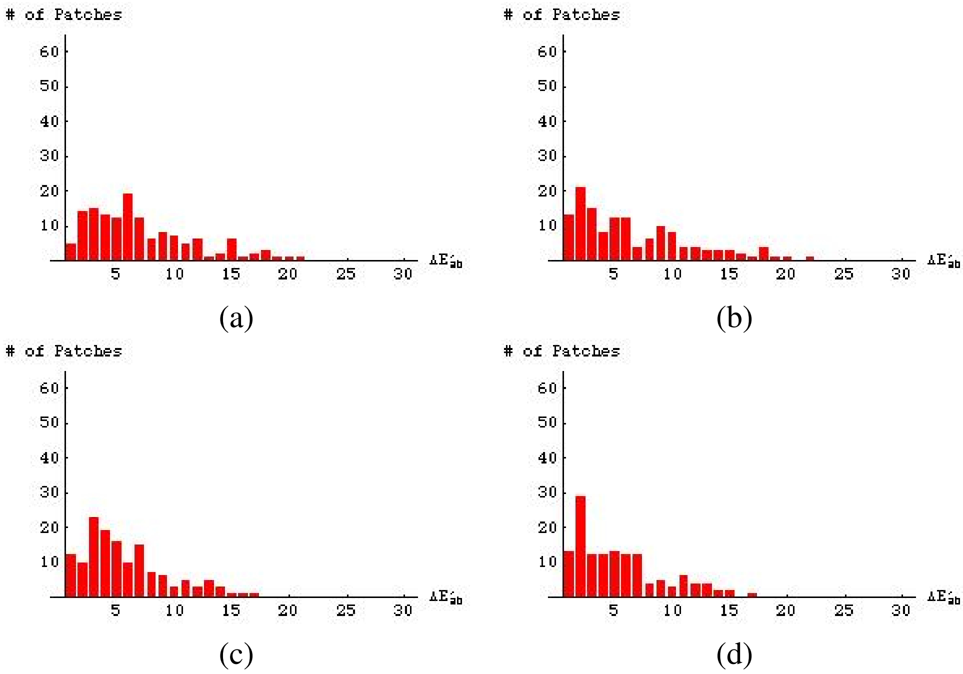


Figure 58: Histogram of $L^*a^*b^*$ differences between the measured $L^*a^*b^*$ values and the results of using the linear polynomials (a) 1, (b) 2, (c) 3 and (d) 4.

linear polynomials still give quite high maximum values, the third order polynomials already yield acceptable results, i.e. the maximum value is close to $\Delta E^*_{ab} = 8$ (see section 3.2 for details). The actual improvement for each polynomial can be better seen in figure 60, where the results listed in tables 11 and 12 are presented in a plot. Polynomials 1 (red) to 11 (blue) are sorted from left to right. The result seems to converge to values close to that of polynomial 10. Detailed results for all 140 patches can be found in appendices E.3 (linear polynomials) and E.4 (higher order polynomials).

	1	2	3	4
Min	1.44	0.55	1.70	0.41
Max	21.90	22.33	17.81	17.42
Median	6.64	5.80	5.30	4.92
Mean	7.63	6.99	6.26	5.68
Stdev	4.60	5.01	3.64	3.86

Table 11: Minimum and maximum, median, mean values, and standard deviations of $L^*a^*b^*$ difference values for the adapted mapping for linear polynomials (1-4), respectively.

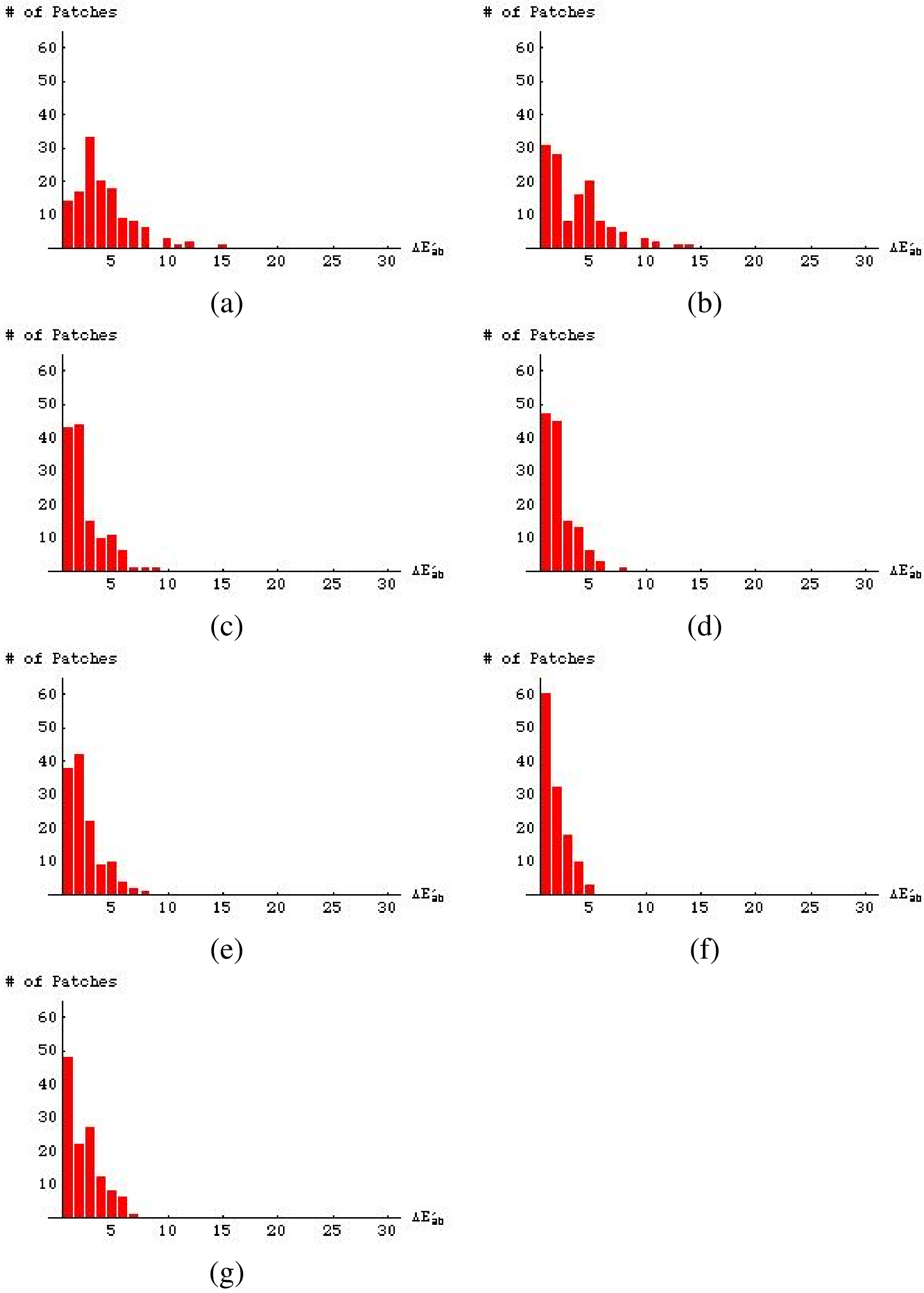


Figure 59: Histogram of $L^*a^*b^*$ differences between the measured $L^*a^*b^*$ values and the results of using the higher order polynomials (a) 5, (b) 6, (c) 7, (d) 8, (e) 9, (f) 10 and (g) 11.

	5	6	7	8	9	10	11
Min	0.50	0.49	0.13	0.26	0.32	0.13	0.08
Max	15.84	14.97	9.67	8.92	8.73	5.67	7.68
Median	3.87	3.00	2.42	2.28	2.42	1.86	2.14
Mean	4.50	3.93	2.89	2.56	2.79	2.13	2.64
Stdev	2.65	2.81	1.69	1.45	1.65	1.16	1.64

Table 12: Minimum and maximum, median, mean values, and standard deviations of $L^*a^*b^*$ differences for the adapted mapping for higher order polynomials (5-11), respectively.

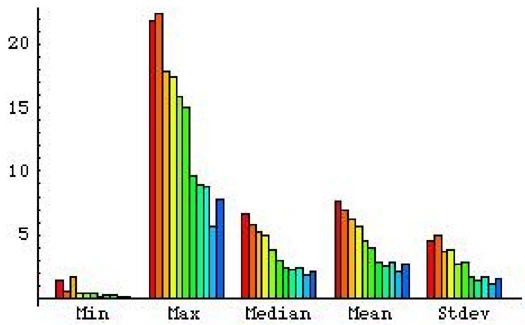


Figure 60: Comparison of minimum and maximum, median, mean values, and standard deviations of $L^*a^*b^*$ differences for polynomials 1 to 11.

Using an adapted mapping clearly improves the result, as the ΔE^*_{ab} values are generally lower than for using an ICC profile or just the color space of the camera. By adding fewer terms we avoid that the polynomial starts oscillating. This gives us a general mapping that can also be used for different targets or materials. Adding more terms enhances the result for a specific test target. Therefore, we propose to generate two different mappings: one using a polynomial of degree two or three to get a general mapping that can be used for arbitrary images, and a second one using a polynomial of degree four. This second mapping should only be used for a specific test scene: the color chart itself. For this, we model the scene we set up for taking the photograph (see section 6.1.1) and render an image of that scene. Now we can compare this scene to a very accurate measurement by applying the mapping to the RGB values of the photograph. If our renderer produced an image that is practically identical to these measurements, we can assume that our renderer is correct (as far as the included materials and light sources are concerned).

7.4 Treating the Camera as a Black Box

The basic goal of color management is that of reproducing color as it is seen by a human observer, like for example taking a photograph and displaying it on a screen in a way that the colors match the original scene. Our problem is different to that of color management, though. As stated in section 1, we primarily want to verify the light transport simulation but not necessarily the subsequent steps of image synthesis (like e.g. the tone mapping step). First, we have to prove that our physically based rendering algorithm is implemented correctly. Only after that, we can verify the tone mapping and gamut mapping algorithms. Otherwise we would analyze two different steps in the rendering pipeline at once and – if errors would occur – we could not say whether the tone mapping step is the cause of error or the rendering algorithm itself is not performing correctly. One possible solution to this problem is not to convert to XYZ or $L^*a^*b^*$ color space at all, but to do the comparison in RGB space.

After light hits the CCD, various processing steps are done in the camera. Some of them can be simulated – like e.g. the interaction of light and CCD elements – but others, like e.g. color correction, cannot. Manufacturers usually do not publish details about the internal processes of their cameras. We do not have to care about steps like color correction if we use the RAW file format. But we still have to know the camera's spectral sensitivities and probably some details about the behaviour of the CCD element itself to be able to convert a rendering to a RAW file that is identical to a RAW photo of the same scene.

If we lack information about the camera's spectral sensitivities, the following method can be used. As we do not know exactly what is happening inside the camera, we treat it as a black box. The input data of our black box is light, i.e. light spectra. The output data is of type RGB, i.e. the RGB values of the final image. What we want to find now is a mapping that simulates what happens inside the black box for one pixel.

As described in section 5, the result of the light transport simulation consists of one spectrum per pixel. If we apply the mapping to each of the spectra we get an RGB value for each pixel. This procedure is illustrated by the dotted line in figure 26. For verification purposes, this RGB value is now directly comparable to the RAW RGB value of the correspondent pixel of an image produced by a camera. A difference between these two values can be caused either by an error in the light transport simulation or by inaccuracies of the mapping. We therefore have to determine the amount of error caused by the mapping in order to be able to better interpret a difference between these two RGB values.

We seek a mapping μ, that models the response of the CCD elements for each of the channels R, G, and B. For the following, the vector (R', G', B') denotes the simulated values yielded by the mapping μ, whereas (R, G, B) denotes the measured values of the CCD elements. Furthermore, x_i denotes the $i-th$ spectral sample. Therefore, μ performs a mapping from $\mathbb{R}^m$ to $\mathbb{R}^3$, where m is the number of spectral samples. We define

$$\mu : (x_1 \ldots x_m) \longrightarrow (R', G', B')$$

and

$$\begin{aligned} R' &= \sum_{i=1}^{m} r_i^2 x_i, \\ G' &= \sum_{i=1}^{m} g_i^2 x_i, \text{ and} \\ B' &= \sum_{i=1}^{m} b_i^2 x_i, \end{aligned}$$

where r_i, g_i, and b_i denote the coefficients we are searching for. The coefficients are squared to avoid negative values as they represent the spectral sensitivities of a camera. We measure the quality of μ by

$$\begin{aligned} \Delta R &= \sum_{j=1}^{n} \left(R'_j - R_j\right)^2, \\ \Delta G &= \sum_{j=1}^{n} \left(G'_j - G_j\right)^2, \text{ and} \\ \Delta B &= \sum_{j=1}^{n} \left(B'_j - B_j\right)^2, \end{aligned}$$

where (R'_j, G'_j, B'_j) denotes the simulated values for patch j, (R_j, G_j, B_j) denotes

	R	G	B
Min	0.81 (0.02%)	0.36 (0.009%)	0.15 (0.004%)
Max	193.93 (4.74%)	295.46 (7.22%)	167.69 (4.09%)
Median	57.44 (1.40%)	51.70 (1.26%)	46.30 (1.13%)
Mean	65.57 (1.60%)	63.98 (1.56%)	52.99 (1.29%)
Stdev	45.24 (1.10%)	51.31 (1.25%)	40.77 (1.00%)

Table 13: Minimum and maximum, median, mean values, and standard deviations of the residuals of R, G, and B values with respect to the values of the RAW image. The RGB values lie in the range of 0 to 4095. Relative errors are given in brackets.

the measured values for patch j, and n is the number of patches used in our tests. We obtain the optimal coefficients for r_i, g_i and b_i by minimizing ΔR, ΔG, and ΔB. We do this by using non-linear regression.

The spectra we use in this case are the spectra of the $n = 140$ patches of the ColorChecker SG. Those were measured with a GretagMacbeth Spectrolino (see section 6.3).

In order to avoid inaccuracies due to image enhancement we use the RAW image for gaining the RGB values (see section 6.1.4). Again, we search for the most representative RGB value in a subarea of each patch. As the Canon EOS 20D is a 12bit camera, the RGB values are in the range of 0 to 4095.

The results of the mapping cannot be compared to methods that are operating in $L^*a^*b^*$ space. Our results are part of the camera's RAW RGB space and can therefore not easily be converted to XYZ and then $L^*a^*b^*$ space. We would have to find a mapping from camera's RAW RGB to XYZ that again will introduce some error. Therefore, we analyze the difference of calculated RGB values to the ones that were actually produced by a camera. In order to test the quality of our mapping, we use the 140 spectra (the same that were used to generate the mapping) as input value and compare the result to the RGB values of the camera. Absolute values for minimum, maximum, mean value, median, and standard deviation of this comparison can be found in table 13. The maximum error of 7.22% occurs in the green channel. The mean value of 1.56% is much lower though so we can assume that there are only a few outliers. The same tendency can be observed for the other two channels. Moreover we can see that the blue channel causes the least amount of error in general. The residuals for R, G, and B channels for all 140 patches can be found in appendix E.5.

This method has two main advantages: We do not need any knowledge about the internal data of a camera and we avoid the errors that are caused by the different spectral sensitivities of a camera and a human observer. The camera is used as a pure measurement instrument, without any image enhancement. We can directly compare the camera's RAW RGB values to the results of applying the described mapping to the spectral values of a rendered image.

7.5 Other Approaches

In this section we list other possible ways of using a digital camera for verification purposes. For each method, we point out why we could not test it or why it cannot be used in practice.

7.5.1 Imitation of a Multi-spectral Camera

We wanted to find out whether it is possible to imitate the approach of Pattanaik et al. [PFTG97] (see section 2.2.5) by using a prosumer-grade camera instead of a scientific grade CCD sensor with narrow band filters, i.e. a multi-spectral camera. Pattanaik et al. used such a camera to reconstruct XYZ values for each pixel of an image. Multi-spectral cameras usually contain seven or more narrow band filters, in order to take a sufficient number of samples of the spectrum. We wanted to investigate, whether a prosumer-grade camera with less filters can also be used in this way. Some prosumer-grade cameras – like the Nikon Coolpix 4500 – do not use red, green, and blue filters for the Bayer pattern (see section 4), but cyan, yellow, green, and magenta filters. This gives us four filters instead of seven. Unfortunately, the spectral curves of these filters differ too much from those of narrow band filters, as can be seen in figure 61. Therefore, they cannot be used to reproduce the approach of Pattanaik et al.

7.5.2 Converting RGB Values to Spectra

Jetsu et al. [JHP$^+$06] use the least-squares regression method to convert RGB values to spectra, instead of converting to XYZ or $L^*a^*b^*$ values (see section 7.3). The main advantage of this version is that spectra are light source independent, i.e. that it is possible to calculate any needed color information using arbitrary light sources. Compared to the CIE $L^*a^*b^*$ approach, it yielded similar or even slightly worse results.

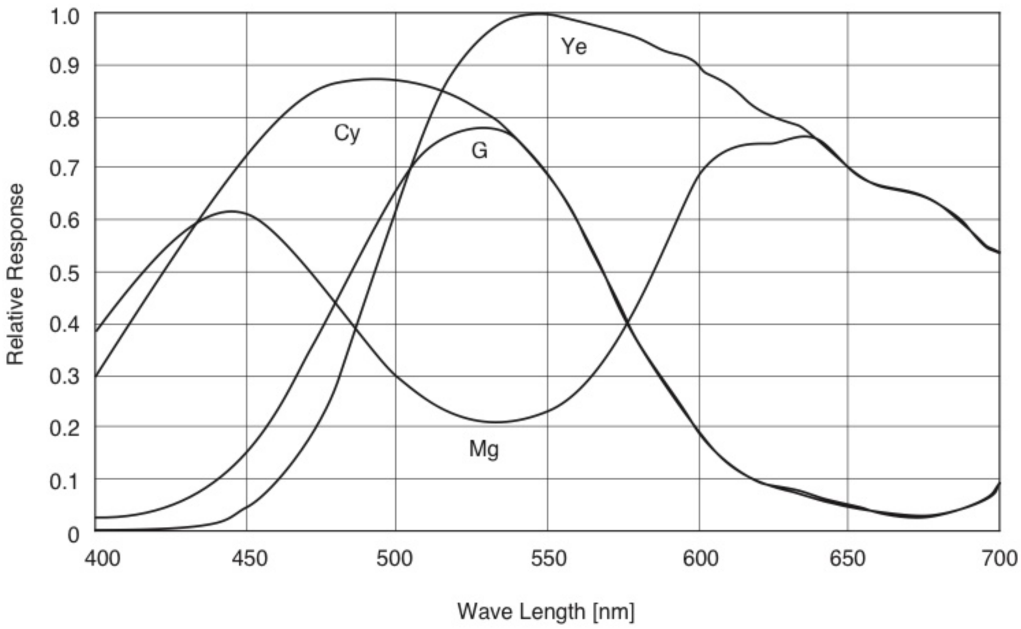


Figure 61: Example of spectral sensitivity curves of a CYGM CCD sensor.

The main problem of this approach is that of metamerism. A single RGB value can be caused by an infinitely large number of spectra. The mapping is of the type $\mathbb{R}^3 \rightarrow \mathbb{R}^n$, n being the number of samples of a spectrum, usually $n > 3$. The set of equations is under-determined, therefore we get a non-trivial solution space. Let W be a transformation matrix that gives us the point in this space that has the smallest Frobenius norm. Let S_i be the spectrum that caused the RGB value and let S_j be the reconstructed spectrum. There is no way to tell whether S_j is equal to the original spectrum S_i, or not. It is possible, that if $S_i \neq S_j$, S_j is a metamere of S_i as far as the camera is concerned. Most likely S_i and S_j are not a metameric pair to a human observer due to the different spectral sensitivities, though.

It is obvious, that although this method gives acceptable results when e.g. samples of the Munsell Book of Colors are used for the training set and the validation set, it will normally fail on measuring arbitrary scenes due to metamerism.

7.5.3 Simulating the CCD Sensor

The method presented in section 7.4 tries to find a mapping that converts light spectra to RGB values. No knowledge about the internal properties of the camera is needed. If this knowledge is available though, the following method may give better results:

One can measure the spectral sensitivity curves of the camera using e.g. a monochromatic light source and a spectroradiometer. Then, instead of multiplying the spectrum of each pixel of the rendering image with the CIE color matching functions to get XYZ values, one can multiply it with the spectral sensitivity curves of the camera to get RGB values. In doing so, one avoids the most severe problem in using a camera for color validation, that is the different sensitivity curves of camera and human observer.

We could not test this method as we do not have the required instruments to measure the spectral sensitivity curves of our camera.

8 A Problem with the Use of XYZ Color Space for Photorealistic Rendering Computations

8.1 Introduction

CIE XYZ color space is sometimes recommended in literature for photorealistic rendering computations because it allows one to simultaneously handle all perceivable colors at much lower computational cost than by using spectral rendering.

In this chapter we show that XYZ is actually a poor choice for such computations, since it is not closed under the component-wise multiplication which is a typical operation in a rendering system. We also discuss why this rather obvious fact has not received much attention yet, and give pointers to alternative solutions.

Even though the theoretical desirability of using full spectral representations for light and reflectance values in photorealistic rendering computations has been known for a very long time [HG83], the overwhelming majority of rendering applications performs those computations which model the interaction of light with matter by using color values.

Color values are numerical correlates of human perception, which in turn is an exceedingly complex phenomenon. If one considers this, it is actually somewhat surprising that color values can be used so well to model light-matter interactions; the main reason why they work is because an RGB triplet basically amounts to a very crude spectral sampling of the visual range.

8.1.1 RGB Rendering

Even though a large number of other color models – such as CIE XYZ or $L^*a^*b^*$ – exist, the standard choice for the color space to perform rendering computations in is usually an RGB space; the question of which specific RGB space is suited best for this purpose has been investigated by Ward and Eydelberg-Vileshin [WEV02].

The major disadvantage of RGB color spaces is of course that none of them contains all visible colors; the obvious problem that arises from this fact is that not all colors that can be found in reality can be represented by positive RGB triplets. This in turn can lead to problems when modelling scenes which contain large numbers of very saturated colors and lights.

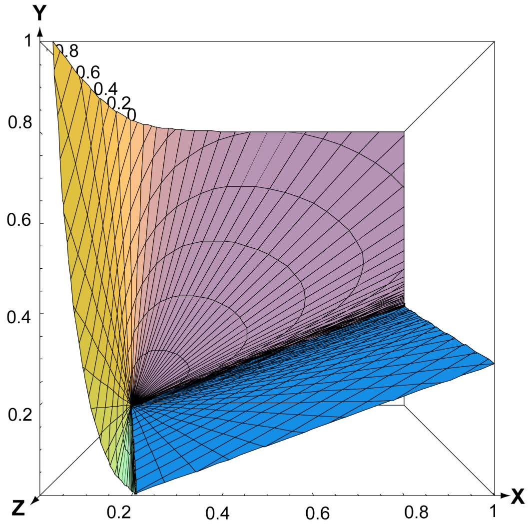


Figure 62: A Mathematica generated plot of the domain of valid XYZ values in the first octant.

8.1.2 CIE XYZ Rendering

A seemingly logical alternative that has been infrequently used in practice is to perform those computations directly in XYZ space [GJB80][Gla95][WS00], thereby eliminating any gamut restrictions from the rendering process. Figure 62 shows a three-dimensional representation of XYZ space, which forms a closed subspace of the first octant ($X, Y, Z \geq 0$). The parameter lines follow different contours for the curved mantle section and the magenta plane, since these were done using different Mathematica plots which were joined together afterwards. A better visualization of this limit surface is shown in figure 63. This figure shows the same shape as figure 62. The geometrical data for the limit surface was exported from Mathematica to 3D Studio Max [3dS06] and rendered using false-color shading, transparency and cutting planes which reveal the chromaticity-diagram shaped cross-section of the subspace.

The key argument in favour of using XYZ space for rendering calculations is that it assigns positive values to all colors that can be perceived by humans; this property is crucial to ensure a meaningful component-wise multiplication between individual color values. Nevertheless, Ward et al. [WEV02] came to the conclusion that a carefully chosen RGB space yielded better results than using XYZ space. Unfortunately, the reasons for this behavior were not analyzed further.

It is known and understood that multiplication between light and reflectance

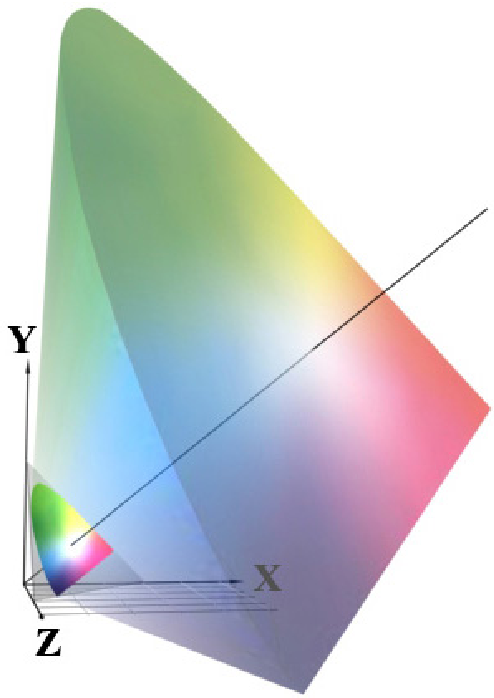


Figure 63: A more intuitive representation of the subspace of valid XYZ values in the first octant (produced by Andrea Weidlich).

values in color space is just an approximation of reality; the error incurred by this approximation has been studied by Borges [Bor91]. He compared the result of a multiplication in XYZ space to a reference solution obtained by multiplying two spectra and then converting the result to XYZ space.

Since the potential error bounds were within an acceptable range, the author recommended to use XYZ space instead of spectral rendering because of the lower computational cost and lower storage requirements. However, he did not investigate the error bounds of an RGB space with carefully chosen primaries, which still might be smaller than those of XYZ, especially in the light of Ward's work [WEV02].

8.2 A Potential Problem with the Use of XYZ Color Space

In practice, renderers usually still resort to using an RGB space for their computations, not at least because colorimetric accuracy is usually not a prime concern for such applications at the moment. Therefore actual rendering systems – as opposed to scientific proof of concept implementations – have rarely used XYZ as their internal color space.

This might account for the fact that one fundamental problem of using XYZ coordinates for computations that model the interaction of light with matter has apparently not received any attention in the computer graphics community so far.

In color space the interaction of light and matter is approximated through the component-wise multiplication of the tristimulus values of the light and the surface reflectance. In any given RGB space, this operation is not problematical since such an RGB space by definition occupies the entire first octant of the three dimensional coordinate system established by the R, G, and B basis vectors; therefore the RGB space is closed with respect to the component-wise multiplication of RGB triplets (i.e. a multiplication of two valid RGB values will always yield a valid, entirely positive RGB triplet as result).

In contrast to this, not all positive XYZ triplets are valid color values; the gamut of perceivable colors is only a subset of the first octant. This means that a component-wise multiplication of valid XYZ values cannot be guaranteed to yield a meaningful color value as result, even though all its components will of course be positive.

In particular, multiplications of similar, highly saturated colors almost always generate colors which are outside the range of valid XYZ triplets.

Figure 64 shows the space that is created by "squaring" (i.e. multiplying by itself component-wise) the XYZ triplet of each point that lies on the border of the subspace of valid XYZ values. All results of component-wise multiplication in XYZ space lie within this "squared" space, which turns out to be considerably larger than the gamut of valid XYZ values. A better visualization of the relationship between these two surfaces is shown in figure 65. This is again a 3D Studio Max rendering of the same data-set that is shown in figure 64, and it clearly shows that the gray subspace is outside the range of valid XYZ triplets almost everywhere.

8.3 Severity of the Problem in Practice

There is one mitigating circumstance which leads to this problem not being as grave as it might seem at first glance for real rendering systems, and which might also account for the fact that it has apparently not been discussed in computer graphics literature so far.

Meaningful interactions of light and matter usually only occur between val-

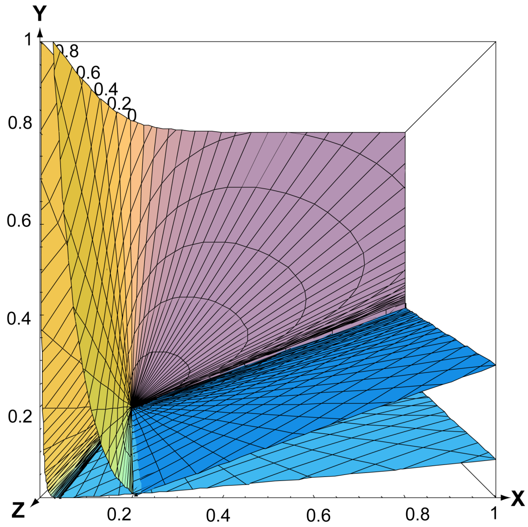


Figure 64: A Mathematica generated plot of the domain of valid XYZ values with the larger surface of possible result values from component-wise multiplication superimposed over it.

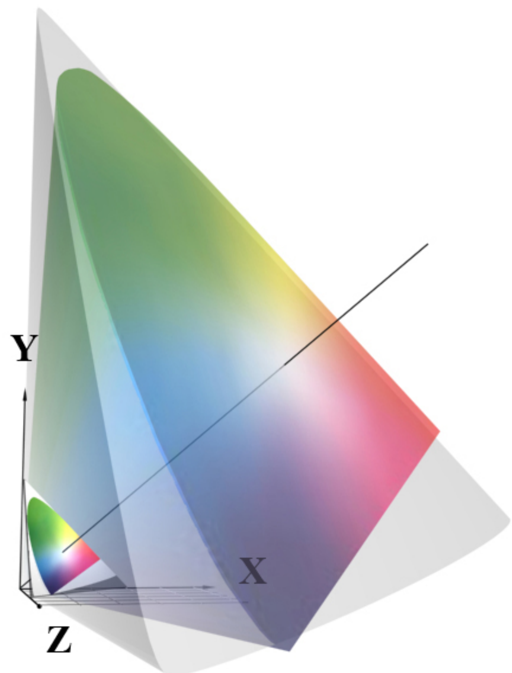


Figure 65: A visualization of both the gamut of valid XYZ values (in false-color) and the gamut of triplets that can be generated by component-wise multiplication of these valid XYZ values (gray). The slight overlapping on the edges occurs because of different clipping planes (produced by Andrea Weidlich).

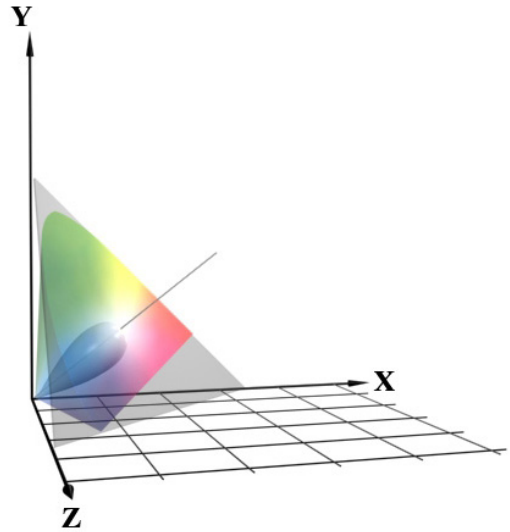


Figure 66: Surface reflectance values form a comparatively small subset (gray volume) of all possible XYZ colors (produced by Andrea Weidlich).

ues which describe a light – which can be arbitrarily saturated – and surface reflectance values, which form a comparatively small subset of all possible XYZ colors [Wys62]; see figure 66 for a qualitative sketch. While the shape of the subset is not correctly represented in this plot it still shows its key feature as far as the topic of this paper is concerned: even its most saturated colors are located at comparatively large distances from the gamut boundary, which means that combinations of surface colors and light values are very unlikely to produce invalid results.

Since the interactions which potentially produce invalid results are those where both operands are highly saturated colors (i.e. those which are already near the boundary of the solid of valid XYZ colors) this means that realistic light and surface interactions are not likely to produce the kind of problem we are trying to illustrate here.

However, the theoretical weakness of XYZ space as a color space for rendering calculations remains, especially since some kinds of light and matter interactions (such as calculations involving transparency, diffraction effects or dispersion) can lead to situations where invalid colors are easily produced.

9 Conclusions and Future Work

9.1 Using a Digital Camera for Verification Purposes

In physically based rendering, it is crucial to be able to rely on the results of rendering systems. In section 2, we have discussed several different approaches for verifying physically based rendering sytems; none of them is what one could consider a truly workable, robust solution suitable for widespread use. While the sophistication of the published techniques has grown considerably over the years, even the latest contributions still have weaknesses. One major drawback is that many of them require expensive, non-standard measurement devices. For this reason, we focused on using prosumer-grade digital cameras. These devices can be easily obtained and are much cheaper than e.g. multispectral imaging devices.

As each laboratory has different premises, we analyzed the quality of several approaches to using a digital camera for verification purposes. On the one hand, we examined standard and non-standard methods from the field of color management. On the other hand, we also developed a novel approach, that is especially designed for verifying rendering systems. Depending on whether particular devices or software packages are available or not, one can choose the appropriate method from table 14.

	Devices	Software	Time	Accuracy
Camera's color space	None	Color management library	Little	Insufficient
ICC profile	None	Profiling software, color management library	Little	Insufficient
Adapted mapping	Colorimeter / None	None	Medium	Medium to good
Black box	Spectrophotometer	None	Medium	Medium to good

Table 14: Qualitative comparison of four approaches (analyzed in section 7) on using a digital camera for verification purposes.

If no measurement device is available, one of the first two methods (described in chapters 7.1 and 7.2) can be used, although the results have to be considered as bad compared to other methods. Still, those methods are standard methods in the field of color management and therefore easy to use and affordable. And even if a serious amount of error is present, it is still better to compare rendering results to biased measurements than do no comparison (or no color management) at all. Outstanding differences between measured and calculated values can still be considered as inaccuracy or error of the rendering process.

The third method (see section 7.3) can be used if a measurement device capable of measuring $L^*a^*b^*$ values is available. Only the $L^*a^*b^*$ values of the color chart used are necessary. If these values are provided by the manufacturer or can be obtained otherwise, no measurement device is needed at all (referring to the "None"-entry in the "Devices" column in table 14). The advantage of this method over the previous two is that by selecting an appropriate polynomial one can adjust the mapping to better fit the requirements. Higher order polynomials tend to oscillate and can therefore usually not be used to define a general mapping. In our case, though, we can use this kind of polynomials to get a very accurate mapping for one specific test scene.

The fourth method (see section 7.4) differs from the other three as it does not operate in $L^*a^*b^*$ color space. Therefore it is not directly comparable to the other methods. It avoids the effect of eye-camera metamerism, i.e. that the spectral sensitivities of the eye and a digital camera are different and that they therefore see colors differently. Knowledge about the interior of the camera is not needed. We just apply the generated mapping to the spectral values of the resulting image of the light transport simulation and can then directly compare the RGB values to the RAW RGB values of the photograph. Instead of trying to simulate a human observer with our camera, we use it as measurement device only.

For both the third and the fourth approach some time is needed in order to implement the required software. These methods generally yield better results than the first two methods. Depending on the amount of accuracy needed and the available time and equipment, a developer can choose the appropriate procedure in order to get the best possible verification result.

9.2 Using XYZ Color Space as Internal Color Space

Based on the observations we made in chapter 8, it can only be concluded that XYZ space should never be used as the internal color space for rendering computations. While it does offer the benefit of being able to represent all possible colors, the fact that color multiplications are no longer an operation which can be guaranteed to yield meaningful results pretty much removes any incentive to use it.

As outlined by Ward et al. [WEV02], one should always use a suitably chosen RGB space instead or consider full spectral rendering if colorimetric accuracy is of prime concern.

9.3 Future Work

At this time, it is necessary to define what an ideal validation procedure should look like. For comparisons, it would be good to have one or several complex test scenes where for each pixel of a rendered image the physically correct spectral distribution is known. Moreover, the acquisition of these scenes has to be practicable and affordable. The problem of the methods developed so far is that none of them is capable of providing such scenes. Analytically derived light distribution solutions are exact, but the complexity of the scene is limited. If we measure the reflected light with an adequate device, the scene can be arbitrarily complex, but there will always be some kind of error that is introduced by the measurement method itself. The amount of this error is often hard to determine. If we compare our image to such ambiguous values, an exact match will be most unlikely. It is hard to say whether this mismatch is caused by an error in our physically based rendering system or by the inaccuracy of our measurements. Therefore the most promising solution is probably a combination of different approaches (see section 2.4) to compensate for the disadvantages of each approach and thus enhance the quality of our verification (for analytical tests one could use the set of test cases that is provided by the CIE Technical Committee 3.33 [CIE06a]).

The fact that no standard procedures for this fundamental problem of computer graphics exist also raises the question what open research challenges remain – apart from the possible use of improved measurement devices. The following list is probably incomplete, but should give an idea of how far-spread the problems are.

- Differences in scene geometry between real and synthetic images – mainly caused by misalignment and inaccurate measurements – are a source of error. Its exact influence on the verification process has not been characterized in sufficient detail. The approach of Karner and Prantl [KP96] to simply skip the problematic areas of the images under consideration might be an improvement over not acknowledging the problem at all, but is something that should definitely be improved.

- Previous work does not explicitly discuss the issue of new techniques in computer graphics like scenes with measured Bi-directional Texture Functions (BTFs) or lightfields (e.g. near field photometry measurements of light sources). As BTFs, for example, are usually generated out of a small sample part of a surface and then extrapolated to larger areas, an exact match with the original surface will not be possible. Therefore one has to find a method to determine whether a generated surface has an appearance which is reasonably similar to the original surface.

- Whole classes of physical effects like e.g. polarization and fluorescence have never been verified in practice yet. Since the problems for which physically based renderers are being used – e.g. glare prediction in the automotive industry, which needs polarization support if it is to be highly accurate – very often depend on just these capabilities, this is a grave omission. It is aggravated by the fact that the measurement procedures published in literature so far are usually not capable of characterizing such effects at all.

- HDR technology could be used to overcome the limited dynamic range of conventional CCD sensors. This is especially useful for capturing outdoor scenes and scenes that contain light sources that are directly visible from the view point.

- Is it possible to introduce a ranking scheme for physically based rendering systems? What are the criteria for such a ranking scheme? Although the focus of this thesis mainly is on the acquisition of measurements of real scenes, these questions should not be disregarded.

As the capabilities of physically based rendering engines continue to grow and appearance–sensitive industries increasingly rely on physically based rendering technologies, one can confidently expect that these problems will become active research areas in the near future, and that the field of image synthesis system verification will become more active as the demands of customers for accuracy guarantees in commercial rendering solutions grow.

As a concluding remark it can be said that the groundwork in this field has been well laid, but that the task of developing robust, practicable solutions is still mostly before us.

A Radiometric and Photometric Quantities

Photometric quantities consider only the visible part of the electromagnetic spectrum and are weighted by the visibility factor V(λ), which is the photopic sensitivity curve of the human eye. The quantities of the radiometric measurements correspond to photometric quantities, but consider the total amount of radiation with the *watt* replacing the *lumen*. Table 15 lists radiometric and corresponding photometric quantities.

Radiometric Quantities	Photometric Quantities
Radiant Flux is the total radiant power that is either emitted by a source or received by a surface. The unit is *watt* $[W]$.	**Luminous Flux** is a measure of the energy emitted by a light source or received by a surface, weighted by the visibility factor V(λ). The SI unit of Luminous flux is the *lumen* $[lm]$ or *candela* $\cdot$ *steradian* $[cd \cdot sr]$. One lumen is defined as the amount of light that falls on a unit spherical area at unit distance from a light source of one candela.
Radiant Intensity is the radiant power of a source emitted in a certain direction. The unit is *watts per steradian* $[W \cdot sr^{-1}]$.	**Luminous intensity** is a measure of luminous flux emitted by a light source in a particular direction. The SI unit of luminous intensity is *candela* $[cd]$.
Irradiance is a measure of the total radiant flux incident on a surface. The unit is *watts per square meter* $[W \cdot m^{-2}]$.	**Illuminance** is the total luminous flux incident per unit area. It refers to the amount of incident light. The unit is *lux* $[lx]$ or *lumen per square meter* $[lm \cdot m^{-2}]$.
Radiance is the total radiant intensity emitted or reflected from a certain location on an emitting or reflecting surface in a particular direction. The unit is *watts per steradian per square meter* $[W \cdot sr^{-1} \cdot m^{-2}]$.	**Luminance** is the measure of luminous flux emitted from, or reflected by a surface. The unit is *candela per square meter* $[cd \cdot m^{-2}]$.

Table 15: Radiometric and photometric quantities.

B Measurement Devices - Definitions

B.1 Radiometer

Radiometer is a general term for a device that is used to measure the intensity of radiant energy. A photometer is a special type of radiometer.

B.2 Photometer

In the broadest sense, a photometer is any instrument used to measure light intensity. A photometer must respond to light as the CIE standard observer. Luminance and illuminance meters are the most common photometers.

B.2.1 Luminance Meter

A luminance meter is an instrument for measuring luminance, i.e. the amount of light that is reflected by a surface in a given direction.

B.2.2 Illuminance Meter

An illuminance meter is used to measure the visible energy falling upon an object's surface.

B.3 Spectroradiometer

A spectroradiometer measures absolute light intensity as a function of the wavelength of light.

B.4 Spectrophotometer

A spectrophotometer measures relative reflectance as a function of the wavelength of light.

B.5 Colorimeter

A colorimeter is used for the measurement of colored light. It uses three filters whose spectral sensitivities are matched to the CIE color matching functions.

B.6 Goniophotometer

A goniophotometer is a device for measuring the directional pattern of light distribution from a source.

B.7 Gonioreflectometer

A gonioreflectometer is a device to measure the reflectance properties of a material and is therefore commonly used to determine BRDFs. It consists of a light source illuminating the material to be measured and a sensor that captures light reflected from that material. Both the sensor and the light source are moved around the material in order to measure the anisotropic properties of the material.

C Color Lab

This appendix presents some of the instruments that are available in the institute's color lab and that were used within the scope of this thesis.

C.1 Light Cabinet

The Verivide Color Assessment Cabinet (see figure 67) is of $120cm$ width and has 5 light sources. We chose light sources D65, TL84, CWF, A and UV. Light source D65 has a correlated color temperature of 6500K and simulates daylight. It is specified for applications where there is a need to maintain color consistency and accurate color matching. Light source TL84 (also known as 840 or CIE Illuminant F11) has a correlated color temperature of 4000K and provides good color rendering. It is often chosen as European "Point of Sale" illuminant. Light source CWF (also known as CIE Illuminant F2) has the same color temperature as TL84 but only moderate color rendering. It is used as American "Point of Sale" illuminant. CIE illuminant A has a correlated color temperature of 2856K and corresponds to a typical tungsten filament lighting. Ultra-Violet Blacklight is commonly used to detect the presence of optical brightening agents or fluorescent dyes. As our rendering engine is capable of handling fluorescent materials we plan to use it for verifying these calculations. The light cabinet also has a dimmer and a diffusor.

Figure 67: Verivide color assessment cabinet with five light sources.

C.2 Luminance Meter

The Konica Minolta Luminance Meter LS-100 (see figure 68) is a compact, lightweight meter for measuring the luminance of light sources or reflective surfaces. The luminance unit can be set to either cd/m^2 or fL. The measuring range is from 0.001 to $299,900cd/m^2$ (0.001 to 87,530fL).

Figure 68: A Konica Minolta LS-110 luminance meter.

C.3 Color Chart

The GretagMacbeth Digital ColorChecker Semi Gloss (SG, see figure 69) has 140 patches in total, including 24 patches from the original ColorChecker, 17 step gray scale and 14 skin tone colors. Its size is $21.59 \times 27.94cm$. It is specifically designed for digital photography, e.g. to use the chart with camera profiling software to create an ICC profile of a camera.

Figure 69: The Digital ColorChecker SG can be used for creating an ICC profile of a camera.

C.4 Spectrophotometer

The GretagMacbeth Spectrolino (see figure 70) is a handheld tethered spectrophotometer with a high level of accuracy at low cost. Its spectral range is from $380nm$ to $730nm$ with a physical resolution of $10nm$. It is capable of measuring reflection, emission and transmission (with SpectroScan T, see appendix C.5).

Figure 70: The GretagMacbeth Spectrolino spectrophotometer is capable of measuring reflection, emission and transmission.

C.5 Spectroscan T

The Spectrolino (see appendix C.4) can be mounted on the GretagMacbeth Spectroscan T automatic table for fast, automatic measurement of color charts. Different from the GretagMacbeth Spectroscan the GretagMacbeth Spectroscan T also provides the possibility of measuring transmission.

Figure 71: The GretagMacbeth Spectroscan T allows automatic measurement of color charts.

D Web Resources of Validation Data

For some of the projects described in section 2, the data that was acquired during the studies was made publicly available. This gives other researchers the chance to use this data for comparison and further research. The following list enumerates these projects and the corresponding web addresses, all last accessed in November, 2006.

- The Cornell Box: `http://www.graphics.cornell.edu/cbox/`

  Specifications of the geometry and material properties of the box used in the approach of Pattanaik et al. [PFTG97] are provided together with the result images .

- Aizu atrium: `http://www.mpi-sb.mpg.de/resources/atrium/`

  Drago and Myszkowski [DM01] contribute the complete scene of the atrium of Aizu, including model, goniometric diagrams, textures, and BRDF measurements.

E Detailed Results

E.1 Using the Camera's Color Space

	8 bit	jpg	16 bit
1	18.25	18.25	18.18
2	11.91	11.88	11.96
3	6.36	6.06	6.00
4	18.02	18.01	17.93
5	12.37	12.31	12.39
6	5.90	5.82	5.80
7	17.88	17.87	17.81
8	12.55	12.52	12.56
9	6.40	6.25	6.16
10	17.95	17.95	17.88
11	12.28	12.31	12.29
12	6.15	5.93	5.77
13	11.75	11.75	11.81
14	18.09	18.10	17.98
15	5.60	5.35	5.32
16	26.91	26.79	26.63
17	11.53	11.56	11.37
18	22.16	22.16	22.16
19	16.59	16.61	16.10
20	8.69	8.63	8.83
21	17.31	17.45	17.54
22	9.73	9.55	9.37
23	24.80	24.84	24.60
24	18.20	18.32	18.44
25	6.38	6.19	6.63
26	17.84	17.85	17.52
27	22.99	22.72	22.88
28	11.55	11.53	11.49

	8 bit	jpg	16 bit
29	11.58	11.52	11.63
30	24.82	25.20	25.21
31	20.66	20.84	21.19
32	18.27	18.26	18.34
33	5.40	5.21	5.29
34	17.80	18.05	18.18
35	18.55	18.51	18.43
36	18.18	18.26	18.13
37	14.91	15.22	15.72
38	6.47	6.52	6.59
39	16.26	16.28	16.45
40	16.40	16.40	16.25
41	18.48	18.60	18.01
42	5.78	5.56	5.43
43	18.25	18.25	18.17
44	22.16	22.25	22.28
45	6.09	6.21	5.99
46	18.76	18.82	18.91
47	12.56	12.69	12.86
48	8.70	9.05	9.31
49	21.95	22.13	21.67
50	17.55	17.82	18.02
51	21.91	21.81	21.78
52	16.11	16.29	16.42
53	20.95	20.98	21.20
54	16.19	15.98	16.01
55	23.34	23.70	23.04
56	18.34	18.33	18.24
57	5.38	5.29	5.40
58	28.44	28.78	28.77
59	26.40	26.47	26.65

	8 bit	jpg	16 bit
60	13.41	13.32	13.34
61	17.85	17.85	17.81
62	16.14	16.13	16.19
63	15.49	15.51	15.59
64	12.35	12.33	12.48
65	9.25	9.23	9.28
66	7.57	7.38	7.30
67	22.78	22.77	22.65
68	11.87	11.65	11.71
69	12.04	12.35	11.81
70	10.91	10.88	10.99
71	11.45	11.46	11.47
72	22.40	22.51	22.59
73	10.24	10.70	10.86
74	8.62	8.63	8.51
75	5.59	5.26	5.32
76	8.74	8.75	8.70
77	10.44	10.46	10.43
78	14.75	14.76	14.87
79	15.82	15.83	15.88
80	17.63	17.63	17.61
81	15.54	15.60	15.66
82	4.96	5.07	5.33
83	12.99	13.02	13.22
84	5.57	5.39	5.36
85	18.51	18.51	18.42
86	12.39	12.72	12.51
87	29.21	29.26	29.39
88	10.16	10.08	9.72
89	9.08	8.55	9.05
90	7.01	7.00	7.26

	8 bit	jpg	16 bit
91	17.21	17.11	17.13
92	8.72	8.69	8.65
93	16.09	15.90	15.90
94	8.35	8.22	8.32
95	10.68	10.73	10.75
96	6.62	6.73	7.00
97	15.11	15.48	15.64
98	18.58	18.56	18.47
99	5.49	5.21	5.27
100	17.58	17.68	17.90
101	14.60	14.65	14.71
102	7.55	7.48	7.67
103	10.17	9.95	10.25
104	9.98	9.72	9.95
105	9.02	8.99	9.23
106	8.48	8.24	8.48
107	15.41	15.40	15.05
108	5.84	5.80	5.90
109	6.88	6.82	6.68
110	7.43	7.18	7.13
111	6.82	6.92	7.29
112	10.59	10.64	10.53
113	11.13	11.05	11.11
114	5.06	4.60	4.01
115	15.28	15.68	15.71
116	12.50	12.79	12.92
117	7.40	7.81	8.18
118	15.98	16.17	16.51
119	8.85	8.81	8.90
120	6.48	6.84	7.14
121	18.68	18.93	19.29

	8 bit	jpg	16 bit
122	8.76	8.59	8.54
123	6.67	6.47	6.40
124	11.46	11.83	12.17
125	2.77	2.28	1.85
126	5.32	5.20	5.10
127	18.85	18.86	18.77
128	11.15	11.14	11.19
129	6.55	6.64	6.86
130	18.53	18.53	18.45
131	11.62	11.63	11.67
132	6.00	6.07	6.32
133	18.27	18.26	18.19
134	11.73	11.73	11.78
135	6.59	6.69	7.02
136	18.42	18.40	18.32
137	11.43	11.48	11.37
138	6.25	6.34	6.60
139	10.61	10.68	10.43
140	19.32	19.32	19.21

Table 16: $L^*a^*b^*$ differences for all 140 patches of the three test cases for using the camera's color space. The patches are consecutively numbered from left to right, starting at the upper left corner of the color chart. For details on the used method see section 7.1.

E.2 Using an ICC Profile

	8 bit	jpg	16 bit
1	18.86	18.86	18.79
2	11.07	10.99	10.95
3	5.94	5.78	5.76
4	18.58	18.58	18.50
5	11.07	10.93	10.89
6	5.59	5.64	5.66
7	18.43	18.41	18.37
8	11.06	10.96	10.89
9	6.04	6.02	5.99
10	18.50	18.51	18.44
11	10.97	10.96	10.78
12	5.75	5.66	5.55
13	10.81	10.77	10.71
14	18.73	18.74	18.62
15	5.33	5.19	5.16
16	18.68	18.55	18.39
17	9.42	9.34	9.14
18	20.65	20.64	20.56
19	14.52	14.56	14.19
20	10.64	10.46	10.67
21	13.95	13.92	13.93
22	10.66	10.70	10.54
23	18.01	17.95	17.28
24	11.21	11.30	11.30
25	9.40	9.05	9.65
26	14.30	14.21	14.19
27	15.29	14.99	15.35
28	10.89	10.83	10.68
29	10.94	10.83	10.86
30	16.79	17.42	17.06

	8 bit	jpg	16 bit
31	12.90	12.87	13.20
32	16.67	16.66	16.69
33	10.19	10.07	10.23
34	12.03	12.24	11.85
35	13.14	13.15	13.16
36	12.16	12.01	11.78
37	7.59	7.59	7.66
38	15.50	15.73	15.68
39	16.90	16.97	17.15
40	9.10	8.94	9.20
41	16.31	16.42	16.04
42	5.39	5.31	5.22
43	18.88	18.87	18.80
44	14.73	14.69	14.72
45	9.60	9.68	9.72
46	17.64	17.83	17.96
47	13.04	12.91	12.82
48	7.00	7.10	7.20
49	12.75	12.81	12.73
50	8.83	8.61	8.90
51	15.97	16.05	16.00
52	10.80	10.77	10.59
53	19.31	19.35	19.65
54	16.50	16.50	16.43
55	15.54	15.79	15.51
56	19.00	18.99	18.90
57	5.21	5.22	5.33
58	17.19	17.42	17.07
59	17.35	17.23	17.30
60	13.34	13.20	13.21
61	18.42	18.42	18.38

	8 bit	jpg	16 bit
62	16.10	16.08	16.09
63	14.05	14.02	13.99
64	10.97	10.90	10.94
65	9.23	9.12	9.04
66	8.24	8.13	8.08
67	20.96	20.95	20.69
68	13.10	13.06	13.07
69	9.54	9.82	9.55
70	10.54	10.48	10.51
71	11.10	11.05	10.97
72	16.07	15.98	15.98
73	7.21	7.63	7.75
74	11.24	11.31	11.26
75	5.55	5.32	5.29
76	9.13	9.09	8.98
77	9.80	9.74	9.60
78	13.30	13.29	13.25
79	15.41	15.42	15.41
80	18.09	18.08	18.05
81	14.71	14.78	14.76
82	6.11	5.97	5.91
83	9.99	9.96	9.90
84	5.29	5.24	5.26
85	19.21	19.21	19.12
86	11.23	11.59	11.15
87	19.36	19.19	19.21
88	12.93	12.97	12.55
89	12.18	11.86	12.18
90	9.25	9.24	9.57
91	16.35	16.32	16.48
92	10.75	10.73	10.68

	8 bit	jpg	16 bit
93	14.53	14.37	14.37
94	11.99	12.00	12.05
95	9.97	9.98	9.88
96	12.06	12.31	12.08
97	7.78	7.78	7.71
98	19.32	19.30	19.21
99	5.45	5.29	5.25
100	11.55	11.66	11.63
101	8.89	8.86	8.89
102	10.35	10.36	10.43
103	12.31	12.22	12.54
104	11.81	11.50	11.93
105	11.18	11.17	11.14
106	10.61	10.31	10.42
107	14.19	14.15	13.89
108	9.38	9.34	9.50
109	7.65	7.60	7.45
110	14.25	14.24	14.13
111	12.05	12.24	12.43
112	10.75	10.78	10.58
113	11.28	11.14	11.11
114	5.79	5.52	5.22
115	9.92	10.21	10.39
116	6.43	6.40	6.41
117	8.95	9.37	9.47
118	8.83	8.81	9.33
119	12.87	12.69	12.70
120	5.56	5.54	5.55
121	15.08	15.08	15.09
122	12.94	13.01	12.95
123	16.56	16.60	16.72

	8 bit	jpg	16 bit
124	11.08	10.89	11.17
125	4.05	3.82	3.65
126	5.32	5.31	5.19
127	19.69	19.69	19.61
128	11.04	10.98	10.95
129	6.35	6.35	6.36
130	19.25	19.25	19.18
131	10.95	10.92	10.86
132	5.78	5.74	5.77
133	18.95	18.95	18.88
134	10.87	10.82	10.75
135	6.25	6.20	6.30
136	19.14	19.11	19.04
137	10.81	10.82	10.59
138	5.98	5.95	5.97
139	10.75	10.80	10.44
140	20.30	20.30	20.21

Table 17: $L^*a^*b^*$ differences for all 140 patches of the three test cases for using an ICC profile. The patches are consecutively numbered from left to right, starting at the upper left corner of the color chart. For details on the used method see section 7.2.

E.3 Adaptive Mapping – Linear Polynomials

	1	2	3	4
1	2.12	3.00	0.77	2.73
2	1.92	1.76	2.40	0.55
3	1.86	3.89	2.56	3.73
4	0.98	2.08	1.24	1.78
5	2.14	2.13	2.63	0.76
6	2.12	3.68	1.54	3.46
7	0.55	1.74	1.73	1.43
8	2.29	2.31	2.71	1.12
9	1.73	3.76	2.14	3.57
10	0.85	1.96	1.35	1.66
11	1.95	1.92	2.43	0.58
12	1.77	3.86	2.58	3.63
13	1.73	1.70	2.31	0.50
14	2.21	3.02	1.00	2.76
15	2.50	3.80	1.23	3.63
16	11.93	11.80	11.44	5.39
17	9.11	7.07	4.86	8.00
18	6.30	6.88	6.53	4.86
19	5.95	5.15	3.91	7.20
20	7.24	7.89	6.51	6.09
21	5.74	5.31	5.40	2.87
22	4.94	11.92	12.13	5.65
23	10.83	9.85	9.93	3.50
24	14.85	10.63	10.61	8.88
25	8.53	7.65	7.76	5.72
26	3.49	8.02	6.25	4.75
27	15.79	12.14	11.97	4.31
28	1.92	1.91	2.46	1.14
29	1.93	1.98	2.54	1.11
30	10.28	8.26	7.96	2.67

	1	2	3	4
31	9.10	3.61	4.26	5.96
32	8.23	7.15	7.10	5.75
33	10.17	8.89	8.97	2.87
34	14.87	6.41	6.52	11.16
35	9.70	7.55	7.45	5.24
36	13.41	9.89	8.86	10.56
37	10.25	13.04	12.84	5.66
38	17.94	12.45	12.36	7.15
39	12.54	11.03	11.34	10.01
40	18.20	9.16	9.22	3.57
41	10.57	12.46	10.41	8.99
42	1.77	3.36	2.25	3.21
43	2.27	3.12	0.97	2.86
44	5.91	7.69	6.31	2.33
45	7.57	8.37	6.90	5.90
46	9.76	7.44	7.73	6.74
47	22.33	6.30	6.60	4.71
48	10.71	5.03	4.86	4.91
49	16.04	5.21	5.89	3.69
50	11.55	13.63	13.81	6.09
51	6.99	9.52	9.66	8.63
52	4.68	6.89	5.77	6.58
53	6.70	7.33	7.03	5.10
54	6.85	4.82	3.50	2.69
55	15.54	5.52	5.15	3.52
56	2.88	3.64	1.63	3.39
57	2.78	3.83	0.55	3.62
58	14.35	15.36	15.11	8.72
59	8.24	9.06	9.10	4.87
60	7.28	5.93	5.93	5.33
61	0.98	2.01	1.12	1.71

	1	2	3	4
62	3.63	3.77	2.81	3.42
63	4.75	4.89	3.24	4.36
64	2.02	1.89	2.36	0.69
65	2.53	3.41	3.11	2.90
66	2.80	4.92	2.52	4.86
67	8.15	7.17	7.07	5.98
68	6.73	5.40	4.20	3.21
69	12.22	8.43	7.99	3.75
70	2.09	2.07	2.36	1.98
71	2.31	2.30	2.77	1.67
72	5.73	5.07	5.49	3.38
73	9.09	16.17	15.20	12.51
74	13.16	14.16	14.44	7.30
75	3.60	4.34	0.89	3.94
76	3.05	4.24	3.19	3.81
77	2.09	2.45	2.80	1.76
78	3.74	3.75	2.82	2.94
79	3.99	4.11	2.74	3.70
80	2.10	2.57	2.38	2.21
81	4.10	4.27	2.71	3.77
82	10.23	8.85	8.84	4.45
83	19.38	4.93	4.99	7.64
84	2.02	3.37	1.53	3.16
85	3.37	4.08	2.21	3.83
86	15.61	17.81	17.42	15.84
87	12.04	13.07	13.07	5.37
88	6.85	4.44	4.41	4.77
89	9.21	7.10	6.26	6.08
90	5.65	6.60	5.57	4.42
91	8.62	7.84	7.76	7.01
92	5.87	6.07	4.61	4.34

	1	2	3	4
93	6.25	4.11	4.05	8.69
94	4.02	2.97	1.93	0.91
95	1.59	1.81	2.20	1.40
96	11.21	7.97	8.05	2.89
97	11.04	5.99	6.87	3.36
98	4.06	4.68	3.10	4.46
99	3.56	4.10	1.15	3.73
100	12.88	7.24	7.70	4.68
101	16.04	11.83	11.18	1.84
102	5.45	6.13	5.03	4.46
103	10.65	8.52	7.61	6.44
104	3.57	3.23	1.56	2.23
105	4.55	4.90	3.52	3.45
106	5.20	5.62	4.27	3.88
107	5.62	4.83	4.45	7.43
108	6.11	6.30	5.41	3.52
109	2.29	4.07	1.95	3.77
110	9.97	9.68	9.97	4.01
111	3.48	3.54	3.28	0.61
112	3.08	3.15	3.14	3.14
113	3.11	3.12	3.30	2.85
114	4.76	6.33	6.26	5.60
115	18.85	14.90	14.43	7.77
116	18.68	13.25	12.26	2.61
117	13.06	11.06	11.24	10.97
118	18.29	13.36	13.02	6.16
119	7.20	4.02	3.82	5.06
120	9.01	5.04	5.01	8.12
121	20.44	14.88	13.99	4.71
122	8.68	7.58	7.86	4.56
123	9.26	10.08	10.47	6.05

	1	2	3	4
124	5.03	7.19	6.68	2.10
125	9.41	10.55	11.20	12.32
126	2.53	3.36	0.41	3.06
127	5.02	5.55	4.32	5.35
128	2.67	2.69	2.98	2.32
129	6.25	5.96	5.30	5.51
130	3.66	4.33	2.59	4.09
131	1.94	1.95	2.52	1.05
132	5.69	5.28	4.98	4.79
133	3.03	3.74	1.83	3.49
134	1.82	1.81	2.43	0.65
135	6.52	5.96	5.82	5.50
136	3.41	4.08	2.32	3.85
137	1.92	1.99	2.51	1.26
138	6.26	5.76	5.60	5.31
139	2.99	3.07	3.08	3.07
140	6.53	6.94	6.18	6.77

Table 18: $L^*a^*b^*$ differences for all 140 patches for linear polynomials when using the adaptive mapping. The patches are consecutively numbered from left to right, starting at the upper left corner of the color chart. For details on the used method see section 7.3.

E.4 Adaptive Mapping – Higher Order Polynomials

	5	6	7	8	9	10	11
1	2.73	0.49	0.13	0.74	0.72	0.15	0.31
2	0.55	1.06	1.77	1.27	1.31	1.28	1.18
3	3.73	2.48	2.96	2.23	1.92	1.68	1.40
4	1.78	1.14	1.57	2.46	2.55	1.25	1.88
5	0.76	1.20	1.98	1.74	1.84	2.01	1.70
6	3.46	1.52	1.97	1.35	1.15	0.89	0.84
7	1.43	1.69	2.21	3.23	3.34	1.56	2.50
8	1.12	1.34	2.05	1.91	2.02	2.23	1.87
9	3.57	2.08	2.42	1.87	1.62	1.56	1.31
10	1.66	1.27	1.74	2.69	2.78	1.29	2.04
11	0.58	0.97	1.76	1.54	1.63	1.79	1.49
12	3.63	2.45	2.95	2.20	1.88	1.72	1.37
13	0.50	0.91	1.70	1.24	1.26	1.23	1.10
14	2.76	0.74	0.69	0.26	0.34	0.58	0.42
15	3.63	1.22	1.69	0.86	0.62	0.63	0.30
16	5.39	5.66	3.42	3.54	2.90	1.65	3.57
17	8.00	6.61	1.33	1.94	2.22	2.38	2.04
18	4.86	4.56	2.13	2.57	2.46	1.85	2.41
19	7.20	7.18	3.82	2.81	2.66	1.71	0.54
20	6.09	4.75	5.20	4.95	4.91	4.40	4.12
21	2.87	2.53	2.62	2.63	2.14	1.95	3.08
22	5.65	4.40	5.38	4.30	5.72	2.14	5.32
23	3.50	2.80	2.79	2.71	2.32	3.29	2.33
24	8.88	7.70	2.58	1.95	1.79	2.13	2.73
25	5.72	5.75	1.19	0.95	2.05	1.98	0.25
26	4.75	5.09	6.41	5.28	6.90	2.36	6.96
27	4.31	4.74	2.36	0.90	2.68	1.24	2.74
28	1.14	1.29	1.90	1.35	1.29	1.05	1.10
29	1.11	1.34	1.99	1.45	1.40	1.21	1.23
30	2.67	0.88	3.51	1.84	3.53	1.86	3.62

	5	6	7	8	9	10	11
31	5.96	5.29	1.13	2.38	2.06	0.57	2.26
32	5.75	5.77	1.63	1.50	1.74	1.81	1.54
33	2.87	2.77	2.05	2.87	2.77	1.05	2.94
34	11.16	11.02	6.92	5.72	7.21	3.94	7.68
35	5.24	5.45	2.75	1.97	2.97	2.29	3.33
36	10.56	10.12	4.54	4.10	4.78	2.40	4.81
37	5.66	4.53	4.47	2.99	4.00	2.28	3.49
38	7.15	7.04	6.77	4.93	7.19	4.07	6.55
39	10.01	10.27	5.06	4.53	5.39	5.67	5.42
40	3.57	4.17	3.63	2.32	3.44	1.01	3.98
41	8.99	8.02	3.52	3.62	3.83	2.17	3.24
42	3.21	1.98	2.36	1.78	1.46	1.78	1.26
43	2.86	0.71	0.41	0.33	0.32	0.28	0.08
44	2.33	2.20	3.06	4.57	2.65	2.51	0.77
45	5.90	3.57	2.70	2.87	3.18	2.82	2.69
46	6.74	6.71	1.10	1.36	0.92	1.38	1.09
47	4.71	5.82	6.41	4.22	5.76	1.39	5.58
48	4.91	5.62	3.35	3.41	3.60	3.51	4.02
49	3.69	4.47	2.73	1.58	2.44	2.43	1.79
50	6.09	5.12	5.11	3.01	5.44	0.80	6.18
51	8.63	8.61	4.98	0.79	5.19	1.55	5.48
52	6.58	6.37	4.01	4.48	3.58	3.47	3.57
53	5.10	4.88	2.79	3.28	2.91	1.85	3.35
54	2.69	1.75	0.35	1.45	0.50	2.27	0.48
55	3.52	3.37	2.69	2.12	2.66	1.99	3.78
56	3.39	1.47	1.24	0.85	0.84	0.72	0.95
57	3.62	0.72	1.04	0.32	0.35	0.41	0.45
58	8.72	8.63	3.90	2.51	3.86	2.44	4.09
59	4.87	4.31	4.20	3.72	3.37	2.26	4.00
60	5.33	5.40	1.28	1.81	2.30	2.26	1.63
61	1.71	1.04	1.49	2.38	2.46	1.19	1.79

	5	6	7	8	9	10	11
62	3.42	2.52	2.51	2.56	2.88	1.02	2.37
63	4.36	2.65	2.80	2.58	2.98	2.75	3.17
64	0.69	0.95	1.67	1.49	1.60	1.79	1.46
65	2.90	1.34	1.64	2.23	1.70	0.94	1.48
66	4.86	2.58	1.95	3.13	3.07	4.31	3.88
67	5.98	5.83	1.61	2.73	2.40	2.41	1.84
68	3.21	2.86	1.92	1.96	1.28	2.13	1.46
69	3.75	3.32	2.73	3.49	3.16	2.61	3.17
70	1.98	1.70	1.96	1.41	1.22	0.80	1.04
71	1.67	1.75	2.23	1.66	1.56	1.20	1.35
72	3.38	3.43	1.82	1.10	1.01	0.71	1.23
73	12.51	11.07	5.41	4.45	4.82	1.26	4.37
74	7.30	6.98	3.94	2.28	3.50	1.39	3.21
75	3.94	1.26	1.56	1.42	1.62	1.36	1.66
76	3.81	1.67	1.62	2.62	2.02	1.51	1.34
77	1.76	1.05	1.64	1.82	1.65	1.48	1.73
78	2.94	1.86	2.54	2.14	2.59	3.52	3.11
79	3.70	2.34	2.32	2.36	2.84	0.57	2.43
80	2.21	2.11	2.00	1.94	2.03	2.21	1.86
81	3.77	2.15	2.40	2.38	2.96	1.66	2.84
82	4.45	4.19	2.36	2.75	2.39	1.96	3.08
83	7.64	7.87	3.74	3.57	2.90	2.75	2.76
84	3.16	1.32	1.63	1.12	0.86	1.14	0.69
85	3.83	2.09	1.91	1.63	1.64	1.30	1.67
86	15.84	14.97	9.67	8.92	8.73	4.08	6.41
87	5.37	5.16	1.23	1.09	0.51	1.59	1.42
88	4.77	5.35	2.12	1.45	1.43	0.44	0.53
89	6.08	5.71	2.93	2.54	3.33	2.63	3.79
90	4.42	3.18	4.00	3.84	3.57	3.40	3.20
91	7.01	6.69	2.24	2.28	1.83	3.75	2.14
92	4.34	2.78	1.75	1.83	1.19	1.57	0.75

	5	6	7	8	9	10	11
93	8.69	8.40	2.42	3.36	2.53	4.49	2.67
94	0.91	1.34	1.54	1.15	1.34	1.19	1.28
95	1.40	0.77	1.29	1.15	0.93	0.73	0.70
96	2.89	3.02	4.33	2.15	5.64	3.55	6.06
97	3.36	3.99	3.10	2.60	3.20	3.38	3.03
98	4.46	2.98	2.92	2.70	2.77	2.07	2.65
99	3.73	1.37	1.58	1.49	1.67	1.39	1.71
100	4.68	4.75	5.06	3.40	4.45	3.81	3.64
101	1.84	1.75	4.14	2.19	4.85	2.31	4.36
102	4.46	3.43	4.08	3.74	3.83	3.36	3.28
103	6.44	6.17	0.93	1.18	1.17	1.91	2.02
104	2.23	1.16	0.61	1.11	1.22	1.71	1.52
105	3.45	2.03	2.21	2.17	2.02	1.82	1.23
106	3.88	2.46	2.52	2.43	2.20	2.00	1.48
107	7.43	7.36	2.87	2.31	2.06	2.98	2.10
108	3.52	2.78	3.85	3.28	3.31	3.14	3.11
109	3.77	1.23	0.57	2.06	1.79	3.00	2.14
110	4.01	4.02	1.21	1.55	1.54	1.62	1.60
111	0.61	0.78	2.97	2.10	3.08	2.10	3.54
112	3.14	2.70	2.77	2.30	2.05	1.55	1.79
113	2.85	2.63	2.84	2.30	2.11	1.58	1.85
114	5.60	5.85	3.82	4.34	3.96	4.09	3.38
115	7.77	7.19	5.70	5.63	5.41	4.11	4.68
116	2.61	2.17	1.84	1.14	2.85	1.29	3.57
117	10.97	10.91	7.16	6.22	6.25	3.32	5.92
118	6.16	5.44	3.66	2.75	3.35	2.11	1.83
119	5.06	5.41	2.10	2.71	2.00	3.90	1.72
120	8.12	8.16	2.40	1.06	2.33	1.89	3.10
121	4.71	4.37	5.14	4.43	4.93	2.31	5.16
122	4.56	4.86	1.36	1.54	1.77	2.80	1.66
123	6.05	6.10	4.14	2.52	4.27	3.76	4.63

	5	6	7	8	9	10	11
124	2.10	2.14	2.96	2.28	3.12	1.48	3.00
125	12.32	13.55	8.05	6.64	6.12	3.77	5.63
126	3.06	0.52	0.50	0.45	0.53	0.13	0.39
127	5.35	4.23	4.26	4.22	4.34	3.40	4.08
128	2.32	2.19	2.51	1.95	1.78	1.30	1.55
129	5.51	5.26	5.62	5.21	5.17	4.65	4.75
130	4.09	2.47	2.35	2.08	2.12	1.61	2.08
131	1.05	1.31	1.96	1.42	1.38	1.20	1.21
132	4.79	4.87	5.32	4.80	4.76	4.21	4.29
133	3.49	1.70	1.53	1.24	1.23	0.96	1.29
134	0.65	1.07	1.83	1.34	1.36	1.30	1.20
135	5.50	5.70	6.16	5.65	5.62	5.09	5.18
136	3.85	2.18	2.09	1.74	1.79	1.31	1.76
137	1.26	1.38	1.99	1.44	1.36	1.11	1.18
138	5.31	5.50	5.95	5.41	5.35	4.71	4.82
139	3.07	2.65	2.75	2.27	2.02	1.55	1.79
140	6.77	6.09	6.28	6.43	6.63	5.36	6.17

Table 19: $L^*a^*b^*$ differences for all 140 patches for higher order polynomials when using the adaptive mapping. The patches are consecutively numbered from left to right, starting at the upper left corner of the color chart. For details on the used method see section 7.3.

E.5 Treating the Camera as a Black Box

	R	G	B
1	-62.00	-13.77	-1.37
2	47.27	7.05	-0.15
3	124.57	126.82	118.12
4	-26.69	62.20	44.63
5	53.17	19.91	7.56
6	121.82	121.93	111.66
7	-13.02	91.41	69.61
8	56.30	28.36	14.79
9	119.17	113.61	106.26
10	-18.24	79.25	66.31
11	50.58	14.43	4.28
12	119.13	114.29	106.78
13	48.16	6.32	2.03
14	-63.04	-11.91	16.23
15	124.05	125.66	116.29
16	-57.17	-86.93	-40.63
17	9.35	-38.76	-19.79
18	7.97	100.73	98.31
19	107.53	90.09	69.92
20	-1.53	49.41	-25.19
21	4.25	-20.15	4.16
22	112.78	65.30	60.79
23	-6.49	-10.60	30.71
24	38.88	89.19	32.47
25	-13.93	-30.84	-67.56
26	80.24	48.42	46.30
27	-13.80	-77.00	-56.54
28	39.88	-9.25	-10.25
29	45.45	2.11	-2.91
30	-33.37	-22.21	12.76

	R	G	B
31	-38.94	-25.49	29.26
32	-9.58	54.47	34.83
33	100.41	97.02	40.54
34	-7.36	23.02	87.18
35	56.65	-8.01	-14.64
36	-2.19	-21.09	22.82
37	150.12	155.67	82.61
38	119.35	72.94	13.56
39	-47.64	-15.57	-61.38
40	57.72	59.38	51.91
41	5.11	34.67	54.89
42	127.46	130.60	123.22
43	-66.63	-24.99	-5.58
44	-54.61	-45.44	31.11
45	85.47	90.73	92.14
46	-47.96	-53.99	-61.02
47	89.05	111.03	141.68
48	129.85	141.84	98.77
49	71.04	72.81	67.98
50	158.21	157.72	53.56
51	-37.98	-74.44	-16.78
52	49.63	96.51	86.05
53	6.01	108.27	104.64
54	10.00	-34.56	-20.01
55	29.97	19.39	22.84
56	-77.60	-43.33	-2.88
57	131.05	138.20	129.21
58	35.07	105.09	126.30
59	7.25	46.16	56.95
60	-67.53	-85.79	-90.30
61	-26.96	61.10	49.22

	R	G	B
62	18.92	61.77	45.70
63	25.58	13.78	-4.32
64	61.20	33.00	20.63
65	78.36	48.07	38.61
66	107.23	94.33	85.07
67	-66.93	-85.97	-46.31
68	8.87	-59.64	-53.41
69	35.25	4.72	-4.13
70	34.87	-18.33	-17.76
71	37.19	-17.11	-20.05
72	14.68	25.49	33.47
73	108.53	136.17	105.75
74	10.09	-0.54	-45.96
75	133.65	145.14	131.89
76	77.88	43.57	34.22
77	66.70	29.43	20.11
78	25.57	-10.64	-16.48
79	31.27	57.50	49.22
80	-0.81	63.64	64.19
81	19.71	22.01	17.93
82	78.01	81.80	34.58
83	62.53	13.71	-35.75
84	124.38	124.42	115.76
85	-100.81	-103.66	-63.73
86	115.44	135.35	109.42
87	3.93	34.73	69.82
88	33.66	-13.27	-60.12
89	6.41	24.66	-60.60
90	51.33	55.18	4.77
91	83.67	84.14	56.00
92	27.45	34.69	-22.64

	R	G	B
93	103.86	109.28	55.06
94	31.38	28.25	-48.41
95	39.82	-16.39	-19.43
96	118.78	74.76	39.84
97	97.51	39.65	20.36
98	-130.76	-155.42	-79.11
99	126.78	130.93	118.32
100	29.39	27.01	-5.54
101	72.52	103.03	53.22
102	-1.01	-7.92	-63.04
103	3.92	31.79	-64.95
104	20.49	33.33	4.02
105	20.38	6.31	-21.98
106	24.22	29.83	-25.33
107	87.22	66.29	48.23
108	21.52	25.77	-59.88
109	102.81	87.59	77.88
110	83.76	4.31	8.32
111	97.55	-7.70	20.71
112	28.93	-33.60	-24.79
113	34.81	-20.58	-17.93
114	102.29	81.92	71.11
115	68.54	39.66	6.23
116	96.05	67.06	55.78
117	120.50	119.56	106.85
118	63.74	65.24	25.72
119	95.72	-12.77	17.19
120	132.05	42.90	80.47
121	131.45	83.01	85.16
122	63.70	35.80	-14.21
123	93.64	0.36	15.72

	R	G	B
124	105.28	-31.34	18.45
125	125.55	134.71	117.68
126	126.55	130.89	120.40
127	-157.73	-222.38	-139.47
128	35.36	-19.04	-17.10
129	134.97	149.82	133.22
130	-113.30	-122.97	-74.57
131	36.66	-14.75	-18.35
132	134.39	148.32	131.89
133	-87.85	-70.37	-30.17
134	44.93	2.62	-2.81
135	141.36	162.52	145.33
136	-110.21	-116.09	-60.72
137	42.06	-3.51	-3.99
138	137.66	155.27	138.18
139	31.90	-25.50	-17.38
140	-193.93	-295.46	-167.69

Table 20: Residuals for all 140 patches for channels R, G and B when treating the camera as a black box. The patches are consecutively numbered from left to right, starting at the upper left corner of the color chart. For details on the used method see section 7.4.

References

[3dS06] 3D Studio Max: www.autodesk.com/3dsmax, November 2006.

[ado06] Wikipedia: Adobe RGB: http://en.wikipedia.org/wiki/Image:CIExy1931_AdobeRGB.png, November 2006.

[Ali06a] Alias: http://www.alias.com/, November 2006.

[Ali06b] Alias: Is it Fake or Foto?, http://www.alias.com/eng/etc/fakeorfoto/, November 2006.

[bay06] Wikipedia: Bayer Pattern Filtration: http://en.wikipedia.org/wiki/Image:BayerPatternFiltration.png, November 2006.

[BDCM05] Steven Bergner, Eric Dagenais, Anna Celler, and Torsten Möller. Using the Physics-based Rendering Toolkit for Medical Reconstruction. In *IEEE Nuclear Science Symposium Conference Record*, volume 5, pages 3022–3026, 2005.

[Bor91] Carlos F. Borges. Trichromatic approximation for computer graphics illumination models. In *SIGGRAPH '91: Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, pages 101–104, New York, NY, USA, 1991. ACM Press.

[CG85] Michael Cohen and Donald P. Greenberg. The Hemi-Cube: A Radiosity Solution for Complex Environments. In *Computer Graphics (ACM SIGGRAPH '85 Proceedings)*, volume 19, pages 31–40, August 1985.

[CIE06a] CIE TC 3-33: Test Cases to Assess the Accuracy of Lighting Computer Programs http://www.cie.co.at/publ/abst/171-06.html, November 2006.

[cie06b] Wikipedia: CIE xy 1931: http://en.wikipedia.org/wiki/Image:CIExy1931.png, November 2006.

[dea06] Dead/Hot Pixel Test: `http://www.foto-freeware.de/deadpixeltest.php`, November 2006.

[DM01] Frédéric Drago and Karol Myszkowski. Validation Proposal for Global Illumination and Rendering Techniques. *Computers & Graphics*, 25(3):511–518, 2001.

[FS02] Paul D. Fleming and Abhay Sharma. Color Management and ICC Profiles; Cant Live Without It So Learn To Live With It! *Gravure Magazine*, August 2002.

[GJB80] Franc Grum and C. James Bartleson. *Optical Radiation Measurements*, volume 2. Academic Press, Inc., 1980.

[Gla95] Andrew S. Glassner. *Principles of Digital Image Synthesis*, volume 1. Morgan Kaufmann Publishers, Inc., 1995.

[GM02] Phil Green and Lindsay MacDonald. *Colour Engineering – Achieving Device Independent Colour*. John Wiley & Sons, Inc., 2002.

[Gre99] Phil Green. *Understanding Digital Color*. Graphic Arts Technical Foundation, 1999.

[GTGB84] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modelling the Interaction of Light Between Diffuse Surfaces. In *Computer Graphics (SIGGRAPH '84 Proceedings)*, volume 18, pages 213–222, July 1984.

[GTS+97] Donald P. Greenberg, Kenneth E. Torrance, Shirley Shirley, James Arvo, James A. Ferwerda, Sumanta Pattanaik, Eric P. F. Lafortune, Walter Walter, Sing-Choong Foo, and Ben Trumbore. A Framework for Realistic Image Synthesis. In Turner Whitted, editor, *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, pages 477–494. ACM SIGGRAPH, Addison Wesley, August 1997. ISBN 0-89791-896-7.

[HG83] R. A. Hall and D. P. Greenberg. A Testbed for Realistic Image Synthesis. *IEEE Computer Graphics and Applications*, 3:10–20, 1983.

[HLR00] Guowei Hong, M. Ronnier Luo, and Peter A. Rhodes. A Study of Digital Camera Colorimetric Characterization Based on Polynomial Modeling. *Color Research & Application*, 26(1):76–84, 2000.

[Imh83] E. A. Imhoff. *Raman Scattering and Luminescence in Polyacetylene During the Cis-trans Isomerization*. Ph.D. Dissertation, Cornell Univ., Ithaca, N.Y., Physics Dept., May 1983.

[Int06] Integra: `http://www.integra.co.jp/eng/`, November 2006.

[Iza06] Izadan, H. and Nobbs, J.H. Input Device Characterisation: A Comparison between Iteration and Regression Methods Using either XYZ or L*a*b*. In *Proceedings of the IS&T's Third Conference on Colour in Graphics, Imaging, and Vision*, pages 158–162, 2006.

[JHP+06] Tuija Jetsu, Ville Heikkinen, Jussi Parkkinen, MarkkuHauta-Kasari, Birgitta Martinkauppi, Seong Deok Lee, Hyun Wook Ok, and Chang Yeong Kim. Color Calibration of Digital Camera Using Polynomial Transformation. In *Proceedings of the IS&T's Third Conference on Colour in Graphics, Imaging, and Vision*, pages 163–166, 2006.

[JMF94] Richard Jackson, Lindsay MacDonald, and Ken Freeman. *Computer Generated Color – A Practical Guide to Presentation and Display*. John Wiley & Sons, Inc., 1994.

[Kaj86] James T. Kajiya. The Rendering Equation. In *Proceedings of SIGGRAPH*, pages 143–150. ACM Press, 1986.

[KK96] A. Khodulev and E. Kopylov. Physically Accurate Lighting Simulation in Computeer Graphics Software. In *6. International Conference on Computer Graphics and Visualization*, St. Petersburg, Russia, 1996.

[KP96] Konrad F. Karner and Manfred Prantl. A Concept for Evaluating the Accuracy of Computer-Generated Images. In *Proceedings of the Twelfth Spring Conference on Computer Graphics (SCCG' 96)*, Comenius University, Bratslavia, Slovakia, 1996.

[lcm06] Little CMS library: `http://www.littlecms.com/`, November 2006.

[Lig06] Lightscape: `http://www.lightscape.com/`, November 2006.

[Lum06] Lumen Micro: `http://www.lighting-technologies.com/`, November 2006.

[Mar99] John Mardaljevic. *Daylight Simulation: Validation, Sky Models and Daylight Coefficients*. Ph.d. dissertation, De Montfort University, Leicester, Institute of Energy and Sustainable Development, Dec 1999.

[mat06] Mathematica: `http://www.wolfram.com/`, November 2006.

[MCTG00] Ann McNamara, Alan Chalmers, Tom Troscianko, and Iain Gilchrist. Comparing Real and Synthetic Scenes using Human Judgements of Lightness. In B. Piroche and H. Rushmeier, editors, *Proceedings of the Eurographics Workshop*. Eurographics, June 2000.

[MF03] F. Maamari and M. Fontoynont. Analytical Tests for Investigating the Accuracy of Lighting Programs. *Lighting Research and Technology*, 35(3):225–242, 2003.

[MK00] Karol Myszkowski and Tosiyasu L. Kunii. A Case Study Towards Validation of Global Illumination Algorithms: Progressive Hierarchical Radiosity with Clustering. *The Visual Computer*, 16(5):271–288, 2000.

[MRC+86] Gary W. Meyer, Holly E. Rushmeier, Michael F. Cohen, Donald P. Greenberg, and Kenneth E. Torrance. An Experimental Evaluation of Computer Graphics Imagery. *ACM Trans. Graph.*, 5(1):30–50, 1986.

[PFTG97] Sumanta N. Pattanaik, James A. Ferwerda, Kenneth E. Torrance, and Donald P. Greenberg. Validation of Global Illumination Solutions Through CCD Camera Measurements. In *Proceedings of the Fifth*

Color Imaging Conference, pages 250–253. Society for Imaging Science and Technology, November 1997.

[Pho75] Bui Tuong Phong. Illumination for Computer Generated Pictures. *Graphics and Image Processing*, 18(6):311–317, 1975.

[PSM93] R. Perez, R. Seals, and J. Michalsky. All-Weather Model for Sky Luminance Distribution - Preliminary Configuration and Validation. *Solar Energy*, 50(3):235–245, 1993.

[raw06a] Dcraw: `http://www.cybercom.net/~dcoffin/dcraw/`, November 2006.

[raw06b] Raw File Format Tutorial: `http://www.cambridgeincolour.com/tutorials/RAW-file-format.htm`, November 2006.

[Rea06] RealReflect Project `http://www.realreflect.org/`, November 2006.

[Ren06] RenderPark: A Test-Bed System for Global Illumination `http://www.renderpark.be/`, November 2006.

[SAWG91] Francois X. Sillion, James A. Arvo, Westin Westin, and Donald P. Greenberg. A Global Illumination Solution for General Reflectance Distributions. In Thomas W. Sederberg, editor, *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics (SIGGRAPH '91)*, pages 187–196, Las Vegas, Nevada, USA, July 1991. ACM Press.

[SG06] Uwe Steinmüller and Jürgen Gulbins. *Die Kunst der RAW-Konvertierung – RAW-Files bearbeiten mit führenden RAW-Konvertern*. dpunkt.verlag, 2006.

[Sha04] Abhay Sharma. *Understanding Color Management*. Thomson Delmar Learning, 2004.

[SKKA01] L. Szirmay-Kalos, L. Kovás, and Ali Mohamed Abbas. Testing Monte-Carlo Global Illumination Methods with Analytically Computable Scenes. In *Winter School of Computer Graphics Conf.*, 2001.

[Sod06] SODA Daylighting Resource: `http://www.soda-is.com/eng/about/`, November 2006.

[srg06] Wikipedia: sRGB: `http://en.wikipedia.org/wiki/Image:CIExy1931_sRGB.png`, November 2006.

[Sto03] Maureen C. Stone. *A Field Guide to Digital Color*. A K Peters, Ltd., 2003.

[Sup06] SuperLite: `http://eetd.lbl.gov/`, November 2006.

[SW04] Roland Schregle and Jan Wienold. Physical Validation of Global Illumination Methods: Measurement and Error Analysis. *Computer Graphics Forum*, 23(4):761–781, 2004.

[TTOO90] Atsushi Takagi, Hitoshi Takaoka, Tetsuya Oshima, and Yoshinori Ogata. Accurate Rendering Technique Based on Colorimetric Conception. In *Computer Graphics (SIGGRAPH Proceedings) 24*, pages 263–272. ACM Press, 1990.

[UH98] S. Ubbelohde and C. Humann. Comparative Evaluation of Four Daylighting Software Programs. In *ACEEE Summer Study on Energy Efficiency in Buildings Proceedings*. American Council for an Energy-Efficient Economy, 1998.

[UW06] Christiane Ulbricht and Alexander Wilkie. A Problem with the Use of XYZ Colour Space for Photorealistic Rendering Computations. In *Third European Conference on Colour in Graphics, Imaging, and Vision*, pages 435–437, 2006.

[UWP05] Christiane Ulbricht, Alexander Wilkie, and Werner Purgathofer. Verification of Physically Based Rendering Algorithms. In *State of the Art Reports, Eurographics 05*, pages 95–112, 2005.

[UWP06] Christiane Ulbricht, Alexander Wilkie, and Werner Purgathofer. Verification of Physically Based Rendering Algorithms. *Computer Graphics Forum*, 25(2):237–255, June 2006.

[VMKK00] V. Volevich, K. Myszkowski, A. Khodulev, and EA. Kopylov. Using the Visible Differences Predictor to Improve Performance of Progressive Global Illumination Computations. *Transactions on Graphics*, 19(2):122–161, 2000.

[War94] Greg Ward. The RADIANCE Lighting Simulation and Rendering System. In *Proceedings of SIGGRAPH*, pages 459–472. ACM Press, 1994.

[Wei99] Eric W. Weisstein. *CRC Concise Encyclopedia of Mathematics*. Chapman& Hall/CRC, 1999.

[WEV02] Greg Ward and Elena Eydelberg-Vileshin. Picture Perfect RGB Rendering Using Spectral Prefiltering and Sharp Color Primaries. In *Thirteenth Eurographics Workshop on Rendering*, 2002.

[WS82] Günther Wyszecki and W.S. Stiles. *Color Science - Concepts and Methods, Quantitative Data and Formulae, Second Edition*. John Wiley & Sons, Inc., 1982.

[WS00] Günther Wyszecki and W. S. Styles. *Color Science*. John Wiley & Sons, Inc., 2000.

[Wys62] Günther Wyszecki. *Farbsysteme*. Musterschmidt Verlag, 1962.