A Practical Device for Measuring the Luminance Distribution

Thijs Kruisselbrink[1,2,*], Myriam Aries[1,3], Alexander Rosemann[1,2]

[1]Department of the Built Environment, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands
[2]Intelligent Lighting Institute, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands
[3]Department of Construction Engineering and Lighting Science, Jönköping University, P.O. Box 1026, 551 11, Jönköping, Sweden

Abstract

Various applications in building lighting such as automated daylight systems, dynamic lighting control systems, lighting simulations, and glare analyzes can be optimized using information on the actual luminance distributions of the surroundings. Currently, commercially available luminance distribution measurement devices are often not suitable for these kind of applications or simply too expensive for broad application. This paper describes the development of a practical and autonomous luminance distribution measurement device based on a credit card-sized single-board computer and a camera system. The luminance distribution was determined by capturing High Dynamic Range images and translating the RGB information to the CIE XYZ color space. The High Dynamic Range technology was essential to accurately capture the data needed to calculate the luminance distribution because it allows to capture luminance ranges occurring in real scenarios. The measurement results were represented in accordance with established methods in the field of daylighting. Measurements showed that the accuracy of the luminance distribution measurement device ranged from 5% to 20% (worst case) which was deemed acceptable for practical measurements and broad applications in the building realm.

Keywords: High Dynamic Range, Raspberry Pi, Measurement device, CIE XYZ, Luminance distribution, Single-board computer

1. Introduction

Lighting simulation is an efficient way for designing comfortable and sustainable lighting conditions in the built environment. However, the reliability of the simulation depends, among other things, on the quality of the input model. An important aspect of daylight simulations is the sky luminance distribution. Previous studies have shown that representation of the sky luminance distribution continues to be a challenge [1,2]. The International Commission on Illumination (CIE) developed 15 generic sky models representing sky luminance distributions for conditions varying from overcast to cloudless skies [3], based on long-term measurements using sky scanners. The usability of the expensive sky scanners is limited: it takes a few minutes to measure the hemisphere and it only allows measurements in low resolution [4,5]. The generic CIE sky-models are very suitable for comparing design decisions under different sky conditions, but they do not represent the actual luminance distribution of the

---

* Kruisselbrink TW, *E-mail address:* T.W.Kruisselbrink@tue.nl

sky for any location and the models are not sensitive to transient luminance variations in different sections of the hemisphere [6]. Due to their generic character these models create uncertainties in the lighting simulations.

More and more buildings are applied with automated daylight systems like automated Venetian blinds and dynamic solar shading. Relevant and actual luminance distributions can increase the performance of daylight systems because both the influence of the neighboring environment and the fast variations of the sky can be included in the input [7], resulting in optimized user comfort and energy performance. Currently available luminance distribution measurement methods, sky scanners and cameras with proprietary software, are not suitable for broad market penetration to support the control of automated daylight systems because of their extremely high price (cameras with proprietary software) or because they cannot handle fast variations of the sky (sky scanners) [8].

Electrical lighting and daylight influence the satisfaction and performance of the occupants. Especially daylight can cause discomfort glare, which is often translated into the Daylight Glare Index (DGI). The following factors are incorporated in the definition to the DGI [9]:

- the luminance of the glare source;
- the size of the glare source;
- the position of the glare source; and
- the luminance of the background.

These quantities are not easily measured simultaneously, with the currently available methods due to complex luminance distributions [10]. A luminance distribution measurement device is capable of measuring all required variables to calculate the DGI simultaneously.

Previous research has shown that it is possible to measure the luminance distribution with cheap commercial digital cameras using the Red-Green-Blue (RGB) information captured using High Dynamic Range (HDR) photography [2,11–13]. However, these methods require extensive post-processing and knowledge and/or assume constant correlated color temperature (CCT).

This paper describes a method for fast capturing the luminance distributions, indoors and outdoors, based on a commercially available camera. Capturing real-time luminance distributions will offer possibilities to help optimize lighting simulations, to inform building automation systems and by this, increasing the use of daylight in building interiors and the dynamic control of the electric lighting indoors, as well as to potentially carry out glare analyses on the run.

The aim of this research was to develop a practical and autonomous camera-based luminance measurement device using an inexpensive single-board computer equipped with a camera and a fisheye lens. In contrast to other measurement devices for the luminance distribution, this method was to be cheap, quick, practical and completely automated. An accuracy in the range up to ±20% was targeted, a range which was deemed appropriate for a practical measurement device [11,14]. Such a practical and autonomous device can be placed at a certain location in the building realm and provide information on the luminance distribution in real time.

2. Methods and Results

In order to build a stand-alone device a single-board computer, Raspberry Pi 2 model B, was used to control the camera, carry out the computations and communicate the results using a Wi-Fi dongle (Fig. 1). The camera functionality was accounted for by the Raspberry Pi Camera Board version 1.3 with a CMOS sensor (3.60 mm, f/2.9) with a maximum resolution of 2592 x 1944 pixels, comparable to cameras in smartphones. A miniature equisolid-angle fisheye lens, suitable for the Raspberry Pi Camera Board, with a measured angle of view of 187° (3mm, f/0.4) was used on top of the camera sensor to provide a hemispherical image. In combination with the camera board, this lens system had a focal length of 1.26 mm and provided an equisolid-angle projection with a field of view of 84% of the sky hemisphere. The code, used to automate the measurement procedure, was composed in Python 3, which is one of the programming languages supported by the Raspberry Pi.

Fig. 1. The measurement device, including a Raspberry Pi 2, Camera Board, fisheye lens, dongle, and control panel (see section 2.4)

1.1. Image Projection

Fisheye lenses have an extremely short focal length. The projection lines that do not pass through the center of the image are strongly bent, resulting in an angle of view up to 180° but with a lower resolution and large distortions at the lens' periphery [15]. Tohsing et al. suggested a straightforward method to describe the projection image of a fisheye lens by relating the elevation angle to the image radius by curve fitting [13]. The relation is described in the following equation with r_i as the image radius of the pixel, c as focal length, and ε_i as the polar angle, being the opposite of the elevation angle.

$$r_i = 2c \cdot \sin\left(\varepsilon_i / 2\right) \tag{1}$$

This equation relates every pixel to the elevation as well as to the azimuth angle. With a coefficient of determination (R^2) of 0.9989, the curve fitting equation was able to accurately determine which pixel represented what part of the photographed scene. For the maximum resolution, the camera projection as seen from sensor midpoint can be described by 2c =1796 pixels or 2c = 2.51 mm.

Two identical lenses were compared to determine the deviation percentage. There was no significant difference between the projection equations of two lenses of the same type. The two similar lenses displayed a relative difference of 0.18%.

To provide input for building simulations and automated operation it is not necessary to get luminance information for every individual pixel due to the overly great spatial resolution and sheer amount of data. Therefore, a subdivision was used as suggested by Tregenza [16], as shown in Fig. 2. Tregenza's subdivision provides the luminance distribution of a hemisphere in a limited amount of samples (145) while ensuring enough resolution to prevent major information losses for daylight applications. The single-board computer ran a script developed to map the Tregenza subdivision on the image sensor using the projection equation (1). This mapping algorithm had an inaccuracy of 0.1%. Subsequently, the computer determined the average luminance of each Tregenza sample by considering all pixels within it. The camera system was bound to an aspect ratio of 4:3 since the focal length was not customizable, resulting in an 84% field of view (Fig. 2).
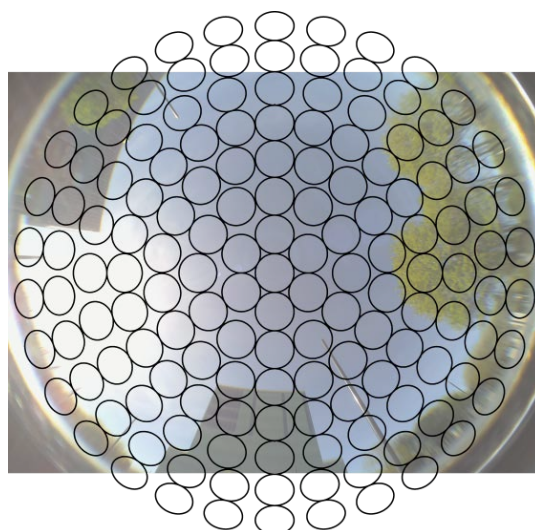
Fig. 2. Tregenza's subdivision placed over an image taken with the Raspberry Pi camera system. Due to the fixed focal length, only 84% of the hemispherical view was captured.

The applied image resolution was chosen based on the optimum between the resulting file size, the processing time, and the accuracy of the Tregenza sample mapping. The optimization between file size and accuracy of the Tregenza samples led to a resolution of 901 pixels horizontally and 676 pixels vertically, instead of 2592 and 1944 pixels respectively. With this resolution, 95% of each Tregenza sample was represented by whole pixels As a result, the projection equation (1) was scaled to this resolution.

$$r_i = 1796 \cdot \sin\left(\frac{\varepsilon_i}{2}\right) \cdot \frac{901}{2592} = 624.3 \cdot \sin\left(\frac{\varepsilon_i}{2}\right) \tag{2}$$

1.2. Input settings

Determining the luminance based on a photograph requires High Dynamic Range (HDR) imaging technology. The luminance distribution occurring in the real world can consist of luminance values in a range of 8 orders of magnitude (typically from 10^{-3} to 10^5 cd/m^2) [17]. Standard 8-bit images only capture a dynamic range of 1.6 orders of magnitude [18]. The most common method to achieve a high dynamic range is the sequential exposure change technique [19]. With this technique, simple digital cameras are used to take Low Dynamic Range (LDR) photographs with sequential exposure settings to cover the desired dynamic range. In order to keep the optical properties constant, it is recommended to only change the shutter speed [19].

A measurement setup was designed, providing constant conditions, to determine which set of exposures efficiently covered the dynamic range of the real world conditions (Fig. 3). A diffuse reflecting target (Kodak Gray Card) was illuminated with a lamp in an otherwise completely dark lab room with black interior surface. The lamp (Halogen, 220V, 650W) was dimmed by applying AC voltages in steps of 20V (within the range from 100V to 260V). In addition, the lamp was placed at multiple positions in order to achieve multiple luminance values at the target. The luminance of the target was measured with a Hagner Universal Photometer S2 and simultaneously photographed by the Raspberry Pi with shutter speeds ranging from 17,000^{-1} s to 2 s (f/2.9, ISO-100). Based on the under/over-saturation empirical equations, representing the minimum, mean, and maximum luminance, were determined describing which luminance range the different shutter speeds were able to capture. These equations allowed to generate a nine-step exposure sequence to capture High Dynamic Range images. It has previously been shown that the quality of an HDR image does not significantly increase with a higher number of exposures [19].
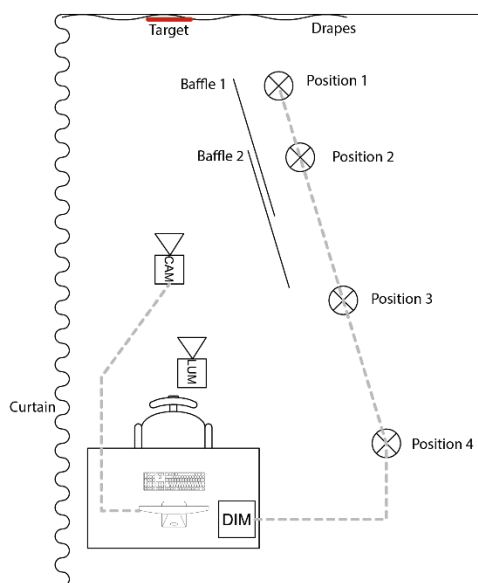
Fig. 3. Measurement setup to relate luminance to shutter speed. Images and luminance measurements were taken for the target in a black room only illuminated by a light source that was dimmed and placed at multiple positions while baffles were applied to prevent direct light entering the camera. The influence of the monitor light is negligible since only a full-screen window with a black background (terminal) was opened during the measurements.

Based on the relation between the shutter speed and the luminance range, as shown in Fig. 4, Exposure Values (EV) ranging from 5 to 19.4 EV in steps of 1.8 EV have been determined for the use by the camera system. The upper limit of 19.4 EV represents the maximum shutter speed of the camera. The exact exposure values slightly differ due to the inaccuracy of the camera device as displayed in Table 1. This sequence guaranteed that, except for the extreme values, each possible luminance value was captured by at least two exposures, with a theoretical maximum luminance of approximately 70,000 cd/m^2.

Table 1. Exposure sequence with nine exposures values that were conducted by the Raspberry Pi camera system to make an accurate High Dynamic Range image for each possible condition. (EV = exposure value)

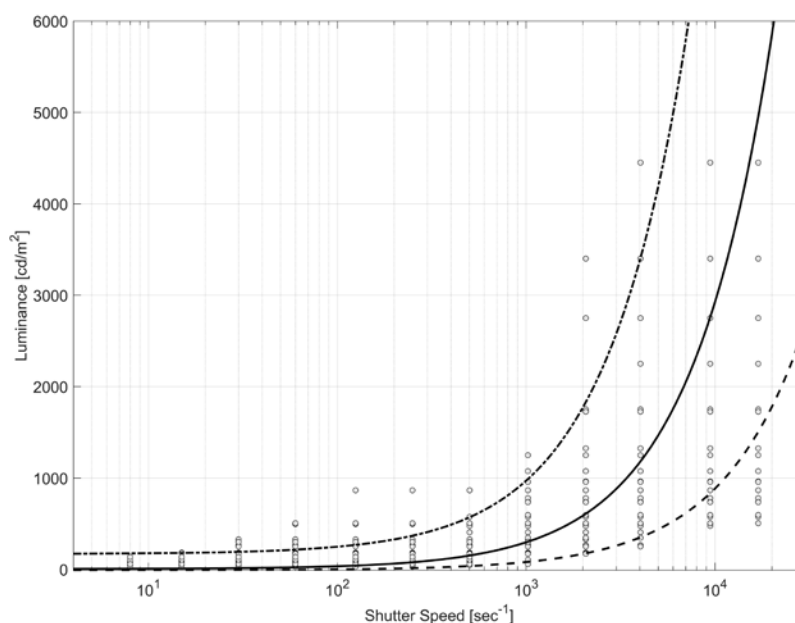| Exposure	Shutter Speed [µs]	EV
1	250,000	5.07
2	76,923	6.77
3	21,739	8.60
4	6,211	10.40
5	1,779	12.21
6	507	14.02
7	130	15.98
8	36	17.83
9	12	19.42

Fig. 4. The relation between luminance and shutter speed. In a measurement, the shutter speed of the Raspberry Pi camera system was related to the luminance based on the saturation of the images (o). The luminance range captured by the camera system was approximated with curve fitted equations for the minimum luminance (dashed line), median luminance (solid line) and maximum luminance (dash-dot line).

Tests with the exposure sequence showed that a number of exposures were always under- or over-saturated. For high luminance values, exposures 1 and 2 turned out to be always completely over-saturated, while for low luminance values, exposures 8 and 9 were always completely under-saturated. Therefore, the exposure sequence was further optimized by leaving out the first or last two exposures depending on the conditions. This way, the quality of the HDR images increased and the influence of transient processes was limited. The most applicable sequence was determined by conducting a base of the exposure sequence (exposure 3-7) and subsequently assessing the 7th exposure on the level of saturation. When an area of exposure 7 was (almost) saturated exposure 8 and 9 were conducted instead of exposure 1 and 2 (Fig. 5).

Photographs taken according to the determined exposure sequence were transformed into a single HDR image by the command-line HDR builder for the Raspberry Pi (HDRgen), originally developed by Ward [20]. This process uses the OpenEXR (.exr) format with RGB encoding and a depth of 96 bits, providing a sufficient dynamic range (76 orders of magnitude). The resulting files were smaller than for other established formats (e.g., HDR and TIFF), it had a relative step size, the relative difference between adjacent values, of 0.1% and was easy to read using the OpenCV library (Version 2) for Python [18,21]. The HDR builder was able to approximate the specific camera response curve using radiometric self-calibration [9,11,22]. The camera-specific response curve was approximated in accordance with the method described by Reinhard et al. [18], by determining the camera response curve for three scenes and averaging the results into one final response curve that was used for all luminance measurements. The response curve is camera-specific. Measurements with another Raspberry Pi camera board showed that the differences between response curves of two similar camera boards were limited to a maximum absolute difference of 2% and a maximum relative difference, for very low exposures, of 60%, and an average relative difference of 12%. The larger differences were mainly present in the darkest 30%.
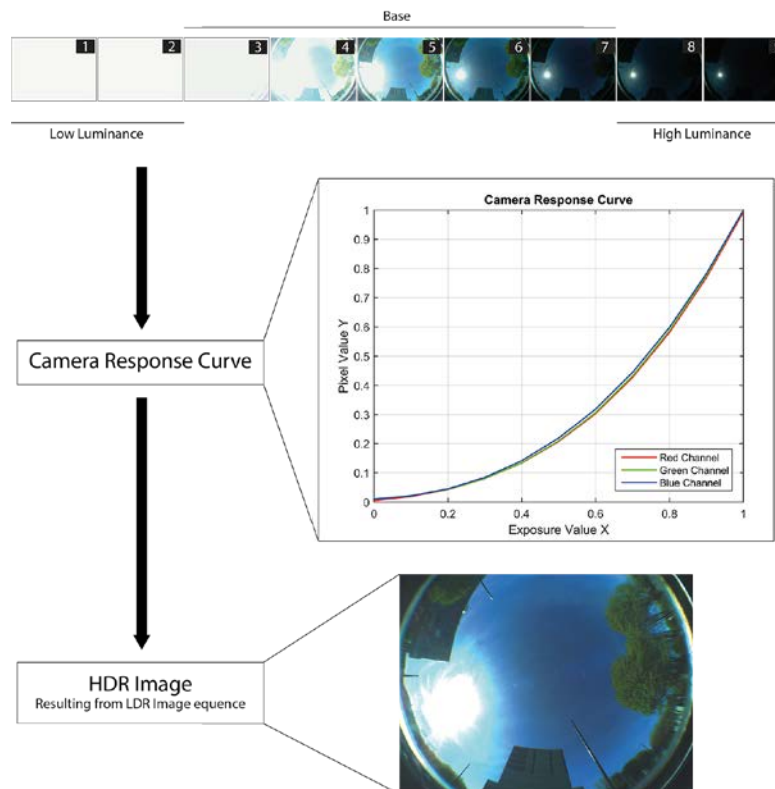
Fig. 5. Formation of High Dynamic Range image. Two aspects were needed to form an HDR image: An image sequence and a camera response curve. The image sequence consisted of two parts: The base, this was always captured; and depending on the light intensity images 1 and 2 or images 8 and 9 were also captured.

1.3. Luminance Calculation

The luminance was determined based on the floating point RGB values of an HDR image. In order to determine the luminance, the RGB color space was converted to the XYZ color space. An important property of the CIE XYZ color space is that the color matching function $\bar{y}(\lambda)$ is equal to $V(\lambda)$, the sensitivity curve of the human eye for photopic vision, meaning that the Y channel indicates the incident radiation weighted by the sensitivity curve of the human eye [5], or, in other words, the luminance. The translation of RGB values to the Y tristimulus value was done according to the protocol as described by Inanici [11]. By applying a conversion matrix depending on the primaries and the white point, the RGB tristimulus values could be turned into equivalent XYZ tristimulus values. The primaries are stored in the EXIF data, while the white point, depending on the CCT, can be extracted from tables [18], or calculated according to three equations as described by Schanda [23].

All variables of the conversion matrix except the CCT were constant. The exact CCT for each condition was not determined since it is an extensive process. Most studies developing a luminance distribution measurement device assumed a constant CCT [5,11,13,24], mostly illuminant D65, to determine the white point. Such an approach results typically in significant luminance errors, as the assumption of a constant CCT (i.e., constant white point) can cause deviations up to 17.9% in the conversion matrix for CCTs far from 6,504 K, the CCT of illuminant D65. This methodological error comes on top of uncertainties caused by noise etc. Alternatively, the luminance distribution measurement device can do the measurements in accordance with three reference CCTs, each with its own conversion matrix, to limit this methodological error (Fig. 6). Next to CIE standard illuminant D65, CCT references of 3,000 K and 14,000 K were used, reducing the maximum methodological error from 17.9%

to 5.4%. The CCT of 3,000 K was suitable for luminance measurements indoors (warm white), illuminant D65 for overcast skies (daylight white), and the CCT of 14,000 K for clear blue skies. The switching point between 3,000 K and D65 was at a CCT of 6,000 K and the switching point between D65 and 14,000 K was at a CCT of 8,600 K. When taking measurements, the most suitable reference CCT was selected by the user (see section 2.4).
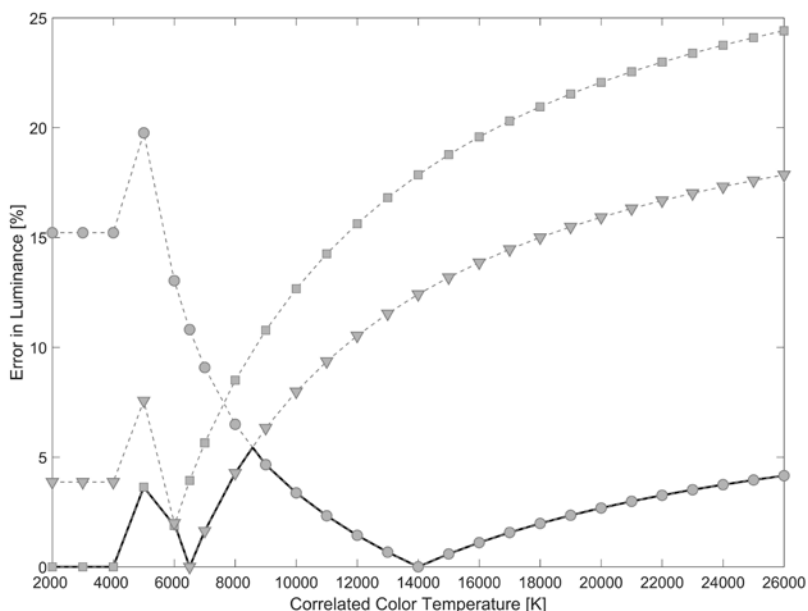

Fig. 6. Deviation from luminance caused by constant CCTs. The conversion matrix to calculate the XYZ color space is dependent on the CCT, the figure illustrates the deviation that occurs when a reference CCT is used. CCT =3,000K (□), CCT = D65 (▽), CCT = 14,000K (o) and the three reference CCT combined (black) with switching points 6,014 K and 8,571 K.

The primaries, obtained from the HDR files' EXIF data, and the calculated white points, led to the color space conversion matrices as displayed in Table 2. The luminance was calculated by extracting the CIE Y tristimulus value, leading to a simple equation (L), with calibration factor k and primaries R, G, and B.

Table 2. Variables of conversion matrices to translate RGB to XYZ for reference CCTs 3,000 K, 6,504 (D65) and 14,000 K. In contrast to the primaries the white points were dependent on the CCTs, resulting in three conversion matrices.

Reference CCT	3,000 K		6,504 K (D65)		14,000 K	
R Primary (x;y)			0.64; 0.33			
G Primary (x;y)			0.3; 0.6			
B Primary (x;y)			0.15; 0.06			
White Point (x;y)	0.3300; 0.3454		0.3127; 0.3291		0.2637; 0.2732	
Conversion Matrix	$\begin{bmatrix} 0.4497 & 0.3536 & 0.1521 \\ 0.2319 & 0.7073 & 0.0608 \\ 0.0211 & 0.1179 & 0.8008 \end{bmatrix}$		$\begin{bmatrix} 0.4121 & 0.3577 & 0.1804 \\ 0.2125 & 0.7154 & 0.0721 \\ 0.0193 & 0.1192 & 0.9499 \end{bmatrix}$		$\begin{bmatrix} 0.3075 & 0.3615 & 0.2963 \\ 0.1585 & 0.7230 & 0.1185 \\ 0.0144 & 0.1205 & 1.5603 \end{bmatrix}$	

For the three ranges of CCT used in this study the luminance is calculated according to equations (3-5).

$$L_{3,000 \ K} \ = \ k \ \cdot (0.2319 \cdot R + 0.7073 \cdot G + 0.0608 \cdot B)$$
(3)
$$L_{D65} \ = \ k \ \cdot (0.2125 \cdot R + 0.7125 \cdot G + 0.0721 \cdot B) \tag{4}$$
$$L_{14,000 \ K} \ = \ k \ \cdot (0.1585 \cdot R + 0.7230 \cdot G + 0.1185 \cdot B) \tag{5}$$

Determining the CIE Y tristimulus value accurately for all pixels requires accounting for the vignetting effect. The vignetting effect of a lens refers to light fall-off at the periphery of the lens [2,5,25]. Especially fisheye lenses exhibit noticeable light fall-off, visible by the gradual darkening towards the corners of the image. In literature, it is noted that some fisheye lenses can exhibit 73% light fall-off at the periphery of the lens [26]. The vignetting effect is a non-linear radial effect along the image radius of the lens and is often approximated by a polynomial function. It has a radial symmetric character, whereby the polynomial function can be used to determine the vignetting effect for all pixels of an image [26–28]. Even with the limiting aspect ratio, the used fisheye lens was considered radial symmetric, despite the fact that the complete projection was not captured by the sensor. Therefore, the vignetting correction, the reciprocal of the vignetting effect, was approximated by an empirical equation along the image radius.

The vignetting effect was determined in an Ulbricht's sphere (Ø 700 mm). According to theory, such integrating spheres create a uniform luminance distribution over its inner surface (±1%) [29]. The vignetting effect was determined for every tenth pixel along the image diameter by dividing the luminance, determined with the Raspberry Pi, with the maximum luminance, which was the luminance close to the zenith. The vignetting correction was measured along the diameter of the image. The radial symmetry of the lens allowed to determine a function along the image radius. This process was repeated multiple times to limit measurement uncertainties and achieve accurate results, since the vignetting effect displayed differences up to 20% under 'constant' conditions. In contrast to previous research [11,26,30], the vignetting filter was not described by a polynomial function. Curve fitting to an exponential function showed the best match. Robust fitting to a second-degree exponential function resulted in the function as described in Fig. 7, with $R^2 = 0.9093$. In order to extract an applicable function outliers were neglected: The outliers at a distance of 240 pixels from the image center were caused by an irregularity of the sphere. Some outliers were exhibited at a distance of 450 pixels, the very last pixel of the image, due to darkening caused by the image border. Fig. 7 shows that the luminance at the lens' periphery was 56% (1/1.8) of the luminance in the lens' center in the case no vignetting filter was applied. Application of the approximated function accounting for this vignetting effect limited the maximum vignetting effect to 14% and the average vignetting to 2.5%. The vignetting effect could not be eliminated completely. Nevertheless, the reduction of the vignetting effect increased the measurement accuracy close to the periphery significantly. With this equation as derived from Fig. 7, a post-process correction filter was defined, containing a vignetting correction factor for each individual pixel.
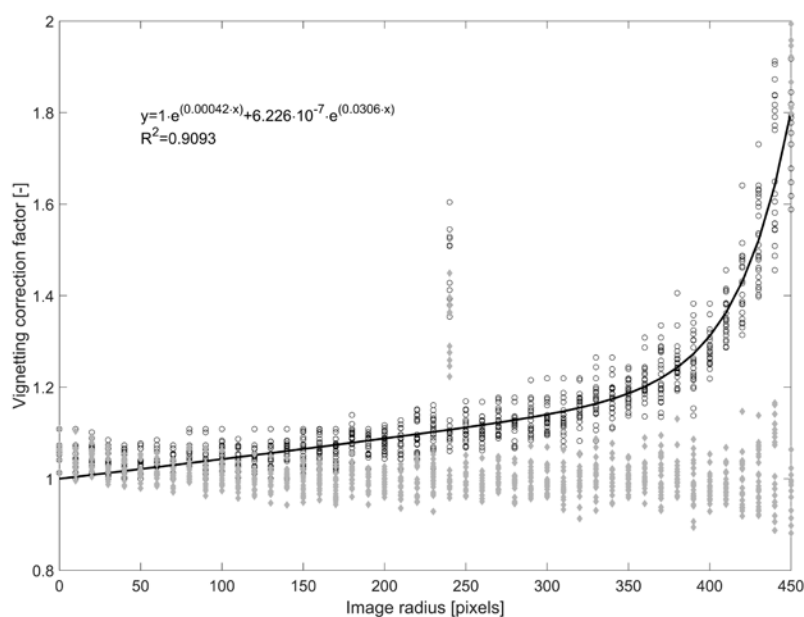
Fig. 7. Determination and effect of vignetting correction. The black dots represent the correction factors to account for the vignetting effect that was measured in the Ulbricht Sphere, resulting in an approximated curve fitted equation representing the vignetting filter (solid line). When the images were corrected with this fitted equation the vignetting effect was minimized (gray diamonds).

In a last step, a photometric calibration was required to accurately extract the luminance from the HDR image. This linear calibration factor k related the CIE Y tristimulus to the real photometric quantity luminance and brought the luminance to the correct order of magnitude. The calibration factor was determined for a gray (ρ = 0.18) and a white (ρ = 0.18) sample of the Kodak Gray Cards under various conditions the measurement device is to cover. The samples were placed in front of the camera and were measured with the Hagner Universal Photometer S2 while the CIE Y tristimulus value was calculated with the Raspberry Pi. This calibration process was repeated multiple times to avoid a calibration factor based on a coincidental measurement. The final calibration factor was the average of all measurements. The calibration measurements showed that the calibration factor depends on the exposure sequence. The absolute Y tristimulus values differed for two HDR images of the exact same scene with different exposure sequences. Therefore, the two exposure sequences got separate calibration factors.

1.4. Processing

The entire process was automated using a Python script. The code is structured as shown in Fig. 8. The process used an infinite loop, which guaranteed continuous measurements until interrupted by the user. The code only asks for interaction regarding the CCT, but choses the default setting of CIE standard illuminant D65 if there is no user input. The user is able to switch between the reference CCTs at any time. The luminance measurement is started by capturing the most suitable image sequence. For scenes with too high luminance values, resulting in over-saturated exposures, the process is aborted and will retry after a time delay. If the exposure sequence is captured successfully, it is formed to an HDR image. Based on the reference CCT the tristimulus value Y is extracted from the HDR image. Consequently, the calibration factor and the vignetting correction are applied. Once completed, the luminance of each individual pixel is known. The results are represented in a more useful manner by averaging over Tregenza's subdivision. Finally, the results are uploaded to a server, allowing access to the measurement results from an external computer. The results are presented as a list with the average luminance for each Tregenza sample and a tone mapped HDR image.

The process from taking the pictures to uploading the results takes approximately 35 s. The duration of each individual task is shown in Table 3. The loop is restarted every 5 minutes, indicating that the process is roughly 4.5 minutes on hold. The measurement frequency can be increased by changing the delay parameter but cannot be shorter that the total processing time.

Table 3. Processing time, in seconds, of separate processes in the Python script, to calculate the luminance, executed on a Raspberry Pi 2.

Process	Processing Time [s]
Preparations	2.1
Capturing LDR Images (Dark/Bright) *	8.2/8.0
Forming HDR Image	5.6
Calculating Luminance**	2.1
Uploading Results***	15.9
Total Processing time (Dark/Bright)	33.7/33.5
Idle (Dark/Bright)	266.3/266.5
Total Loop Time	300.0

* Suitable image sequence for dark or bright conditions.
** Including vignetting correction and calibration.
*** Uploading results and uploading and formation of a tone-mapped HDR image.

1.5. Accuracy

The accuracy of the measurement device was measured according to the method described by Inanici [11]. Two gray Kodak cards were used next to an uncalibrated gray scale and an uncalibrated color scale. Except for one gray card, all color scales were placed in the center of the image. The remaining gray card was placed close to the periphery of the image to address the potential gradient in accuracy along the radius due to the vignetting effect (see Fig. 9 on the far right). The luminance was measured with the Hagner Universal Photometer S2 as well as determined by the Raspberry Pi. Based on a CCT measurement taken with the Konica Minolta illuminance spectrometer CL-500A, the most suitable reference CCT was determined and used. The accuracy was indicated by relating the physical measurement to the measurement results of the Raspberry Pi. This process was repeated for multiple scenes under different indoor and outdoor conditions.
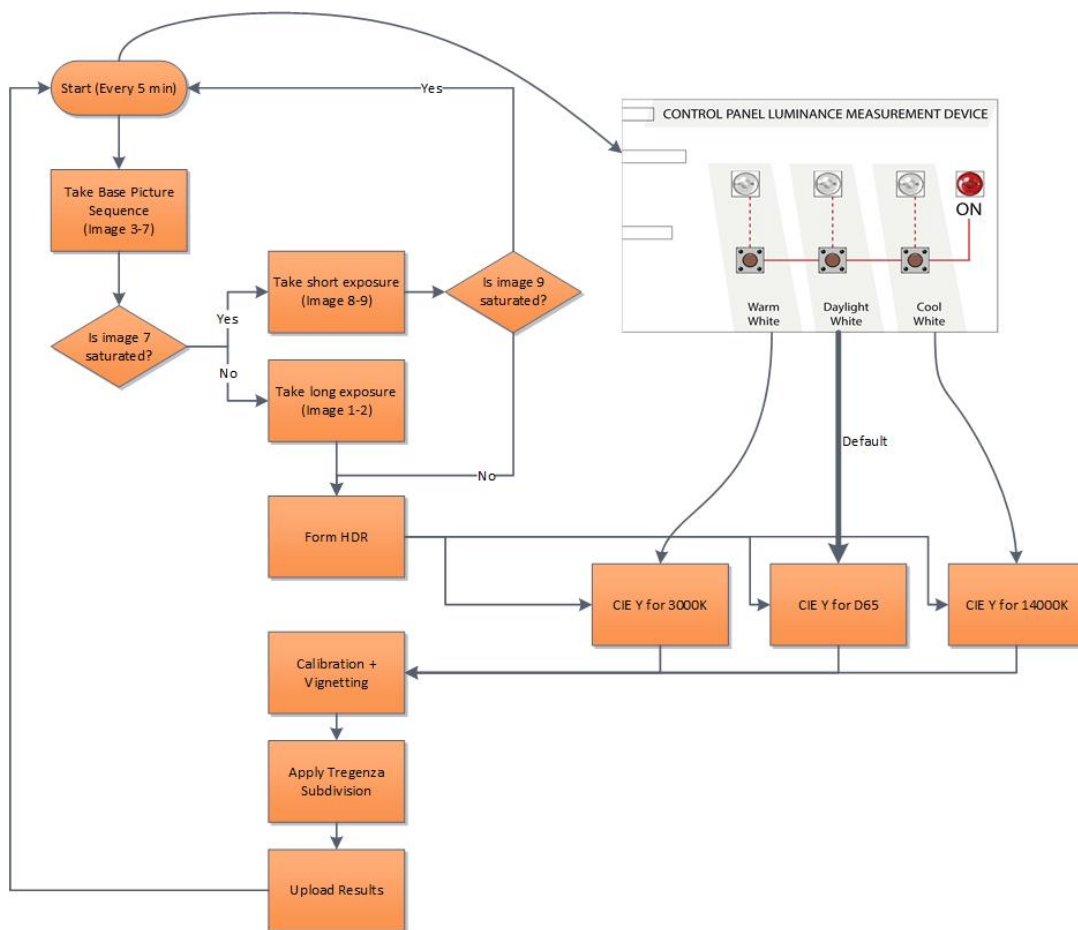
Fig. 8. Flowchart representing the automated luminance distribution measurement. The straight arrows represent the flow, the blocks the processes, the diamond decisions, and the curved arrows represent user input that is acquired at the start of the process and used later in the measurement process.

Fig. 9. Setup for the accuracy measurements. Gray and colored targets were placed in the center of the image and one gray target was placed at the border. The accuracy was determined by comparing luminance measurements with the calculated luminance, this was repeated for multiple conditions.

A selection of the accuracy results is shown in Fig. 10 and 11. Other accuracy measurements showed similar results for different luminance ranges. The measurements had an average error of 10.1% for a range of 3 to 18,000 cd/m^2. The average errors for the gray and colored targets were 8.0% and 12.5% respectively. The accuracy measurements also showed that the device did not work accurately enough for very high luminance values (e.g. sun or reflections of the sun) due to saturation of the shortest exposure, leading to errors in the HDR assembly

and, subsequently, to false results. The exact luminance that saturated the shortest exposure could not be determined but was assumed to be in the range between 18,000 and 70,000 cd/m^2. The lower end of this range represents the highest luminance measured during the tests; its maximum the greatest calculated luminance for the shortest exposure (device limitation).
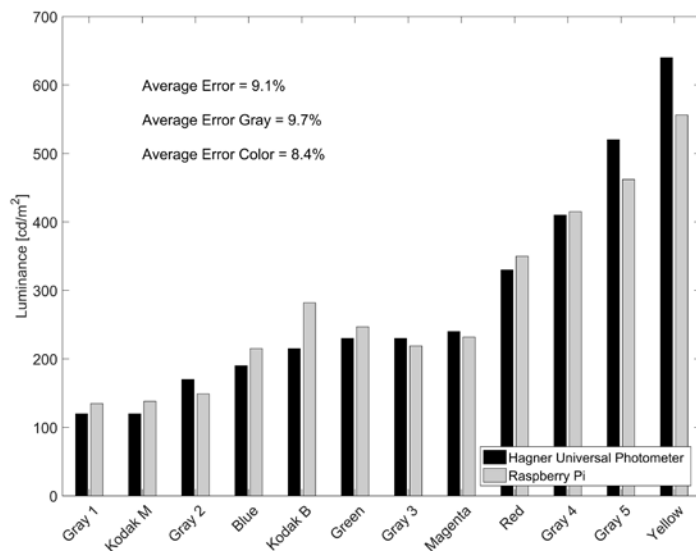

Fig. 10. The measured accuracy for an indoor condition with a CCT of 6,370K. The black bars represent the luminance measured with the Hagner Universal Photometer while the gray bars represent the luminance determined by the Raspberry Pi. (M = middle, B = border)
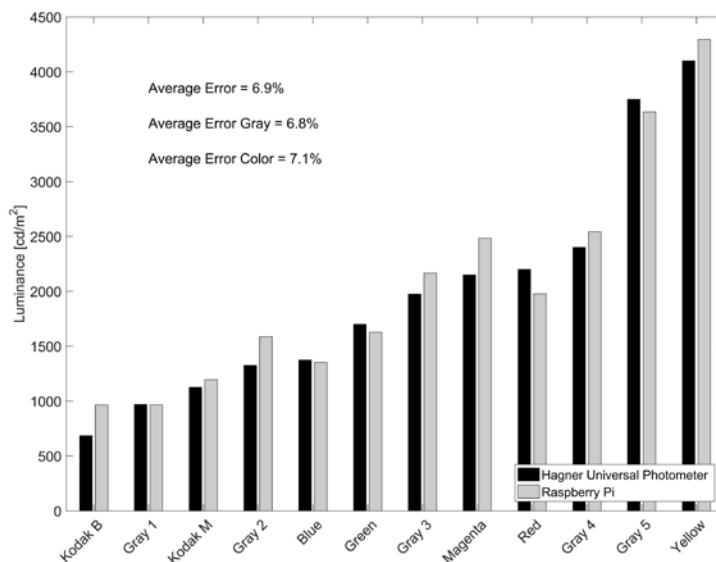

Fig. 11. The measured accuracy for an outdoor condition with a CCT of 6,170K. The black bars represent the luminance measured with the Hagner Universal Photometer while the gray bars represent the luminance determined by the Raspberry Pi. (M = middle, B = border)

It was expected that close to the periphery of the sensor the error would increase because the vignetting correction could not completely account for the vignetting effect. The results supported this hypothesis (Fig. 10 and 11), the errors close to the border (Kodak B) of the sensor were significantly higher than at the center of the

image (Kodak M); for the Kodak gray cards, it displayed an average inaccuracy of 27% compared to 8% in the center. It is assumed that this error applies to the last 75 pixels along the radius because in this region the impact of the vignetting effect became significant. For the other pixels, the vignetting effect was much smaller and therefore had a lower impact on the overall measurement accuracy.

2. Discussion

The established exposure sequence had two variations to minimize the number of saturated exposures. It was developed in such a way that the entire range of possible luminance values was captured. The accuracy of the HDR image, and hence the accuracy of the luminance measurement device, can be improved by basing the exposure sequence on the current lighting situation. Moreover, it turned out that the shortest exposure possible was not able to capture the luminance of the sun and its direct reflections. The luminance of the sun was several orders of magnitude greater than the maximum luminance that could be captured with this exposure sequence. The maximum measurable luminance is currently limited to 18,000 cd/m^2 because no higher luminance had been measured during the accuracy measurements. The actual maximum luminance might be higher. With the chosen measurement setup it was not possible to reach higher luminance values on the targets (color scales). These targets were required to assure that the same luminance was measured by the Hagner Universal Photometer and the Raspberry Pi. The exposure sequence was translated into an HDR image using the HDR-builder developed by Ward [20]. The settings were assumed constant for all situations, which means that for some conditions the settings were not optimal.

The camera response curve was approximated with the HDR-builder, and therefore it was not the exact camera response curve. The maximum relative difference between the camera response curves of two comparable cameras was 60% with an average relative difference of 12%. Therefore, the applied camera response curve cannot be applied for other Raspberry Pi camera boards without consideration.

Additionally, two identical camera lenses were compared. The maximum relative difference was 0.18%, therefore the lenses were found to be equal. The developed code can be used to measure the luminance distribution, with an acceptable accuracy, using another lens of the same type.

The luminance calculation was based on the similarity between the tristimulus value Y and the sensitivity curve of the human eye for photopic vision (V(λ)). Therefore, the measurement device can only be applied to situations where photopic vision occurs, thus for luminance values greater than 3 cd/m^2 [31].

To calculate the luminance from an RGB HDR image the CCT is required. A default CCT of 6500 K (D65) is a good solution when the main light source is daylight. However, for CCTs far from 6500 K (i.e., blue sky) this might lead to methodological errors up to 18%. In this study, three reference CCT's were applied to limit this error to approximately 5%. A downside of this is the required user intervention. This intervention will result in some uncertainties, possibly increasing the inaccuracy. However, the maximum methodological error will never exceed 18%.

It seems that in some other studies the vignetting filter was based on a single measurement, resulting in extremely good fits [11,26]. This research showed that the vignetting correction needs to be based on multiple measurements because the vignetting effect displayed differences up to 20% under 'constant' conditions. The vignetting effect close to the periphery was only limited to 14% by fitting to data achieved by multiple measurements. The accuracy of most of the image was improved by slightly compromising the accuracy close to its periphery. This is motivated by the fact that most information is extracted from the center part of the image, and not from its boundaries. Apparently, the conditions were not entirely constant during these measurements, but due to the application of an optimized vignetting filter the usability of the camera-lens system with limited capabilities was improved. Nevertheless, it is still reasonable to perform multiple measurements in all cases of vignetting effect measurements to achieve an optimal overall vignetting correction.

The calibration factor was determined for white and gray targets to further increase the overall measurement accuracy. The exposure sequences both had their own calibration factor because different exposure sequences of the exact same scene showed that the absolute Y tristimulus values differed.

The time needed to perform the entire process was 33.7 s, meaning that it is possible to perform nearly 2 measurements within a minute. For this study, measurements were performed every 5 min. The actual time

necessary to take the LDR images was the time that the device was vulnerable to transient conditions. This was approximately 8 s, compared to 3 min required for a sky scanner [32–34], and 1-2 min for HDR camera system measurement [8].

The accuracy was determined with the Hagner Universal Photometer S2, which itself has an accuracy of ±5%. This means that the actual accuracy of the luminance distribution measurement device could deviate ±5%. By taking the inaccuracy of the Hagner Universal Photometer into account the average accuracy of the developed device was in the range of: 5.1% - 15.1%, 3% - 13.0% and 7.5% - 17.5% for respectively all targets, gray targets, and colored targets. Even in the worst case scenario, this falls within the range of accuracies, ±5% to ±20%, found in other similar studies using more sophisticated devices [11,14]. Thereby, this device performs along a similar trend, higher accuracy for gray targets compared to colored targets, as found by Inanici [11].

3. Conclusions And Recommendations

3.1. Conclusions

The luminance distribution was determined based on the similarity of the CIE color matching function $\bar{y}(\lambda)$ and the sensitivity curve of the human eye, including some corrections. The CIE Y tristimulus value channel was achieved by translating the RGB information of the High Dynamic Range (HDR) image to the CIE XYZ color space. This was done using three conversion matrices each representing an illuminant with a particular CCT. The High Dynamic Range technology was essential to accurately capture the luminance distribution because it is, in contrast to standard 8-bit images, able to capture the entire dynamic range that occurs in real scenarios. Finally, the luminance distribution was represented according to Tregenza's subdivision.

The process of determining the luminance distribution was conducted using a Raspberry Pi (with camera board) as a single-board computer, which was able to perform all calculations automatically. The device can operate autonomously.   The best performance was acquired when the user selected the suitable reference CCT at the start of the measurement and changing this when the conditions had changed. The results were automatically digitalized and uploaded to a server. The accuracy of the device falls within an acceptable range with an average accuracy ranging from 5.1% to 15.1% and an average accuracy range of 3.0% to 13.0% and 7.5% to 17.5% for respectively gray and colored targets. All of this was achieved with low costs components.

The device in its current form is tested within a limited performance range. Reliable results could only be guaranteed within a luminance range from 3 to 18,000 cd/m^2. Measurements showed that for luminance values somewhere over 18,000 cd/m^2 the results became unreliable due to saturation of the shortest exposure. Therefore, the measurement range was limited to 3-18,000 cd/m^2, the actual maximum luminance limit might be higher.

3.2. Recommendations

The device can potentially be further optimized by applying some additional improvements that are subject to further research.

In this study, two different exposure sequences were used, being the optimal sequence for a limited set of conditions. To improve the quality of the HDR image it is recommended to determine the exposure sequence specifically for each condition. This prevents saturated images, meaning that all nine exposures are evenly distributed within the occurring luminance range.

The current device was limited to a maximum luminance of 18,000 cd/m^F. This was because high luminance values lead to application of saturation of the shortest exposure time. The shutter speed cannot be further shortened but a neutral density filter is a possibility [30]. This way the current dynamic range can be shifted towards longer exposures and the shorter exposures can be used to capture higher luminance values. Disadvantages are that for darker conditions the exposure time becomes significantly higher, whereby the influence of transient processes increases. This can potentially be accounted for by adding an extra camera to the measurement device.

The fixed focal length limited the capture of a full hemispherical view. There are two methods available to overcome this. A camera with an adjustable focal length can be used to fit the entire hemispherical view on the image sensor. However, in the case of the Raspberry Pi, this means that the fisheye lens cannot be placed in front

of the camera as was done here. Another option could be using two cameras which are rotated 90° compared to each other. This way the entire hemispherical view is captured and fisheye lenses can still be applied. It is recommended to test both suggestions, and compare it with the original set-up.

The usability of the measurement device can be expanded via the connection to networks, i.e., communication and interaction using an SSH protocol. An automated start-up of the luminance distribution calculation could be incorporated in the code. The next step in providing a useful representation could be a false color representation instead of Tregenza's subdivision. A false color representation can provide an intuitive and quick understanding of the measurement results.

Acknowledgements

References

[1] Spasojević, B., & Mahdavi, A. (2005, August). Sky luminance mapping for computational daylight modeling. In Proceedings of the 9th international Building Performance Simulation Association Conference, Montreal–Canada (Vol. 1163).
[2] Inanici, M. (2010). Evalution of high dynamic range image-based sky models in lighting simulation. Leukos, 7(2), 69-84.
[3] CIE. (2014). CIE Standard General Sky Guide. Vienna, Austria.
[4] Kobav, M. B., & Dumortier, D. (2007). Use of a digital camera as a sky luminance scanner. 26th Session of the CIE.
[5] Wüller, D., & Gabele, H. (2007, January). The usage of digital cameras as luminance meters. In Proc. SPIE (Vol. 6502, p. 65020U).
[6] Spasojević, B., & Mahdavi, A. (2007). Calibrated sky luminance maps for advanced daylight simulation applications. In Proceedings of the 10th International Building Performance Simulation Association Conference and Exhibition (BS2007), Beijing, China (pp. 1205-1210).
[7] Aries, M. B. C., & Zonneveldt, L. (2011). Daylight variations in a moderate climate as input for lighting controls.
[8] Chiou, Y. S., & Huang, P. C. (2015). An HDRi-based data acquisition system for the exterior luminous environment in the daylight simulation model. Solar Energy, 111, 104-117.
[9] Mead, A. R., & Mosalam, K. M. (2016). Ubiquitous luminance sensing using the Raspberry Pi and Camera Module system. Lighting Research & Technology, 1477153516649229.
[10] Bellia, L., Cesarano, A., Iuliano, G. F., & Spada, G. (2009, May). HDR luminance mapping analysis system for visual comfort evaluation. In Instrumentation and Measurement Technology Conference, 2009. I2MTC'09. IEEE (pp. 957-961). IEEE.
[11] Inanici, M. N. (2006). Evaluation of high dynamic range photography as a luminance data acquisition system. Lighting Research & Technology, 38(2), 123-134.
[12] Sarkar, A., & Mistrick, R. G. (2006). A novel lighting control system integrating high dynamic range imaging and DALI. Leukos, 2(4), 307-322.
[13] Tohsing, K., Schrempf, M., Riechelmann, S., Schilke, H., & Seckmeyer, G. (2013). Measuring high-resolution sky luminance distributions with a CCD camera. Applied optics, 52(8), 1564-1573.
[14] Moeck, M. (2007). Accuracy of luminance maps obtained from high dynamic range images. Leukos, 4(2), 99-112.
[15] Schneider, D., Schwalbe, E., & Maas, H. G. (2009). Validation of geometric models for fisheye lenses. ISPRS Journal of Photogrammetry and Remote Sensing, 64(3), 259-266.
[16] Tregenza, P. R. (1987). Subdivision of the sky hemisphere for luminance measurements. Lighting Research & Technology, 19(1), 13-14.
[17] Moeck, M., & Anaokar, S. (2006). Illuminance analysis from high dynamic range images. Leukos, 2(3), 211-228.
[18] Reinhard, E., Heidrich, W., Debevec, P., Pattanaik, S., Ward, G., & Myszkowski, K. (2010). High dynamic range imaging: acquisition, display, and image-based lighting. Morgan Kaufmann.
[19] Cai, H., & Chung, T. M. (2011). Improving the quality of high dynamic range images. Lighting Research and Technology, 43(1), 87-102.
[20] Ward, G. (2005). Photosphere, Anyhere Software.
[21] Holzer, B. (2006). High dynamic range image formats. Institute for Computer Graphics and Algorithms, TU Wien.

[22] Mitsunaga, T., & Nayar, S. K. (1999). Radiometric self calibration. In Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on. (Vol. 1, pp. 374-380). IEEE.

[23] Schanda, J. (Ed.). (2007). Colorimetry: understanding the CIE system. John Wiley & Sons.

[24] Roy, G. G., Hayman, S., & Julian, W. (1998). Sky modeling from digital imagery. ARC Project A, 89530177.

[25] Cai, H. (2013). High dynamic range photogrammetry for synchronous luminance and geometry measurement. Lighting Research & Technology, 45(2), 230-257.

[26] Cauwerts, C., PhD, M. B., & Deneyer, A. (2012). Comparison of the vignetting effects of two identical fisheye lenses. LEUKOS, 8(3), 181-203.

[27] Inanici, M.N., & Viswanathan, K. (2013). Hdrscope : High Dynamic Range Image Processing Toolkit for Per-Pixel Lighting Analysis and Hdrscope : Lighting Analysis. 13th Conference of International Building Performance Simulation Association, Chambéry, France, 3400-3407.

[28] Moore, T., Graves, H., Perry, M. J., & Carter, D. J. (2000). Approximate field measurement of surface luminance using a digital camera. International Journal of Lighting Research and Technology, 32(1), 1-11.

[29] Ulbricht, R. (1920). Das Kugelphotometer. Berlin Und Munchen: Verlag Oldenburg.

[30] Stumpfel, J., Tchou, C., Jones, A., Hawkins, T., Wenger, A., & Debevec, P. (2004, November). Direct HDR capture of the sun and sky. In Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa (pp. 145-149). ACM.

[31] Baer, R., Seifert, D., & Barfuß, M. (2016). Beleuchtungstechnik. (4th editio). Berlin: HUSS-MEDIEN GmbH.

[32] Coutelier, B., & Dumortier, D. (2002).Luminance calibration of the Nikon Coolpix 990 digital camera. Application to glare evaluation. AIVC and EPIC Conference, Lyon, France.

[33] Kobav, M. B., Bizjak, G., & Dumortier, D. (2013). Characterization of sky scanner measurements based on CIE and ISO standard CIE S 011/2003. Lighting Research & Technology, 45(4), 504-512.

[34] Ineichen, P., & Molineaux, B. (1993, August). Characterisation and comparison of two sky scanners: PRC Krochmann and EKO instruments. In First draft, IEA Task XVII expert meeting.