Radiometric Compensation of Global Illumination Effects with Projector-Camera Systems

by

Gordon Wetzstein

Submitted to the Department of Media System Science

in partial fulfillment of the requirements for the degree of

Diplom-Mediensystemwissenschaftler

at the

BAUHAUS-UNIVERSITÄT WEIMAR

Thesis Supervisors	Jun.-Prof. Dr.-Ing. Oliver Bimber Bauhaus-Universität Weimar
	Prof. Dr. rer. nat. habil. Klaus Gürlebeck Bauhaus-Universität Weimar Second Supervisor

Date of Submission	06/28/2006
Date of Defence	07/19/2006

To Gerhard

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Gordon Wetzstein, Weimar 06/28/2006

ABSTRACT

Projector-based displays have been evolving tremendously in the last decade. Reduced costs and increasing capabilities have let to a widespread use for home entertainment and scientific visualization. The rapid development is continuing - techniques that allow seamless projection onto complex everyday environments such as textured walls, window curtains or bookshelfs have recently been proposed. Although cameras enable a completely automatic calibration of the systems, all previously described techniques rely on a precise mapping between projector and camera pixels. Global illumination effects such as reflections, refractions, scattering, dispersion etc. are completely ignored since only direct illumination is taken into account. We propose a novel method that applies the light transport matrix for performing an image-based radiometric compensation which accounts for all possible lighting effects. For practical application the matrix is decomposed into clusters of mutually influencing projector and camera pixels. The compensation is modeled as a linear equation system that can be solved separately for each cluster. For interactive compensation rates this model is adapted to enable an efficient implementation on programmable graphics hardware. Applying the light transport matrix's pseudo-inverse allows to separate the compensation into a computational expensive preprocessing step (computing the pseudo-inverse) and an on-line matrix-vector multiplication. The generalized mathematical foundation for radiometric compensation with projector-camera systems is validated with several experiments. We show that it is possible to project corrected imagery onto complex surfaces such as an inter-reflecting statuette and glass. The overall sharpness of defocused projections is increased as well. Using the proposed optimization for GPUs, real-time framerates are achieved.

ACKNOWLEDGMENTS

First of all I would like to thank my supervisor, Oliver Bimber, for his excellent work. He gave suggestions whenever I lost myself into some detail, but allowed me to develop my own ideas. I would also like to thank my second supervisor, Klaus Gürlebeck, for helping me out with detailed background knowledge in numerics, listening to my ideas and checking the equations.

My parents and grandparents helped me in the last six years, by supporting me with love, groceries and also financially. I would not be what I am without them.

Thanks to Anselm Grundhöfer, Sebastian Knödel and Sven Banisch for numerous discussion on this work, life, the universe and everything. I probably would have died of starvation without my flatmates Frauke, Alex, Manuel, Caro, Jule and Sascha who supported me with cooking, cleaning and a good time. Thanks to Kristin Kropidlowski for proofreading the thesis.

When I applied for the Ph.D. programs I got a lot of help from Charles Wüthrich, Bernd Schalbe, Kristian Hildebrand and Christian Lessig.

I had a great time in Weimar and do not want to miss it. It would have not been the same without a lot of people, so thanks to my former roommates Peter, Konstantin, Martin, Christoph, Rebecca, Sarah and Przm, the kickboxing club, all guys from my favorite place to work, hang out and play kicker (table soccer): the m18 and my labmates Saskia, Stefanie, Tobi, Alex, Daniel and Daniel.

”All problems in computer graphics can be solved with a matrix inversion.”

James Blinn, computer graphics pioneer

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

The rapid development of video projection technology in the last years led to an astonishing increase of capabilities such as higher spatial resolution and dynamic range as well as reduced costs. Widespread availability and the fact that projectors can display images that are much larger than the devices themselves made them a mass-market product. Nowadays, many households are equipped with projectors used for home-entertainment. However, such displays are not only used for private, but also for scientific purposes. Most of the old-fashioned large-screen projector-based visualization setups using big, heavy and expensive CRT projectors have been replaced by flexible, inexpensive and easy configurable multi-projector systems using common LCD or DLP devices.

The trend of employing flexible multi-projector systems continues today. Recently, numerous approaches that enable a seamless projection onto everyday surfaces have been proposed [46, 4, 6, 23, 17]. These compensate for geometrical distortions and color modulation of projected imagery. Eventually, the generated image appears as if projected on a planar white canvas, although this is physically not present.

Applications for radiometric compensation with projector-camera systems do not only include home-entertainment by allowing correct projections on wallpapers, window curtains and bookshelfs. Inexpensive and flexible multi-projector are used for scientific visualization by enabling the projection onto special canvases (dome- or L-shaped surfaces). Ad-hoc visualization is utilized at construction sites where possible interior and other architectural content can be displayed spatially aligned with the future location. Other applications include cultural heritage related projections. Compensation techniques have been used to overlay pictorial artwork with projected interactive multi-media content [4]. The possibility of seamlessly integrating projections into existing environments make it an interesting tool to be used in historical site, e.g. castles or ancient water reservoirs [9].

All proposed radiometric compensation techniques assume a simple geometric relation between cameras and projectors that can be automatically derived using structured light range scanning. This results in a precise mapping between camera and projector pixels. In physical setups, the light projected by a single display pixel often bounces back and forth several times, before it eventually reaches the imaging sensor. Due to reflections, refractions, scattering and other global illumination effects, it may contribute to several spatially distant portions of the camera image. Assuming a direct mapping usually only considers the contribution with the highest intensity, thus, discarding all global illumination effects.

The entire contribution of projected light to a camera is described by the forward light transport. This has recently been used for environment matting [71, 50], computational photography techniques [63, 62, 20], relighting [43, 14] and other applications. Capturing the light transport with a projector-camera system requires advanced structured illumination techniques supported by computer-controllable point light sources, lasers or video projectors.

We propose a novel, image-based approach to radiometric compensation that accounts for all possible global illumination effects. Conventional schemes for light transport acquisition are employed to capture these effects with a projector-camera system. The goal of the compensation is to find an illumination pattern that, when projected on the scene, results in a desired image from the camera's perspective. We model the compensation as a set of linear equations that can be solved with respect to the projector pattern. The general mathematical foundation is validated using several experiments.

Many applications require real-time compensation rates for displaying interactive multi-media content or videos. Due to the size of the equation systems resulting from the acquired light transport, it is not always possible to provide an interactive compensation. However, this can be achieved by reformulating the problem so that it is suitable to be implemented on programmable graphics hardware. Depending on the complexity of the scene and included global illumination effects, real-time framerates can be achieved.

The thesis is structured as follows: chapter 3 starts with introducing basic radiometry and photometry related terminology and quantities. Furthermore, the light transport between a projector and a camera is discussed as well as advanced techniques to accurately capture it with high dynamic range imaging. The last part of chapter 3 presents data structures and mathematical constructs that can be used to synthesize images from the camera's point of view under novel projector illumination (relighting). Using the introduced mathematical foundations of forward light transport, a dual image (scene from the projector's point of view illuminated by the original camera) can be created.

A novel radiometric compensation approach is presented in chapter 4. This is based on the fundamental light transport techniques introduced in chapter 3 and takes all global illumination effects into account. Due to the enormous size of the light transport matrix, clustering and decomposition methods are discussed and employed to compute a compensation image. This is projected onto the scene and results in a corrected image from the camera's point of view. The approach is evaluated using several examples setups including different global illumination effects such as reflections, refractions, defocus and scattering.

A modified version of the radiometric compensation, which runs at interactive framerates, is introduced in chapter 5. A reformulation of the problem along with optimized data structures for an implementation on programmable graphics hardware is discussed and compared to the method presented in chapter 4. An overview of related work is given in chapter 2. Chapter 6 summarizes the proposed techniques and illustrates the workflow of the compensation. Results are discussed, concluded and compared to previous approaches. Finally, suggestions for future work are given.

2 RELATED WORK

The research in camera-based projection systems has recently gained a lot of interest in the computer graphics and vision community. Traditionally, multi-projector configurations are employed to create large-scale high-resolution displays on planar uniformly colored surfaces. However, in the last years many new applications such as projector-based augmentation, radiometric compensation, focus correction and relighting have been proposed. This section gives an overview of projector-camera related work.

2.1 Multi-Projector Systems with Planar Canvases

The challenge in multi-projector systems is a geometrical alignment on the one hand and a photometric correction on the other. Geometric registration refers to aligning all displays so that the output images appears seamless and geometrically consistent, regardless of possible color variations. A photometric compensation is used to provide (perceptually) uniform colors and intensities over the entire display.

Early approaches to projector-based display systems such as the CAVE [13] relied on manual geometric registration. Recently, many projection systems have been enhanced with one or multiple cameras to allow for automatic geometric registration and photometric correction [41, 40, 38, 37, 39, 56, 12, 67, 54, 55, 36]. This allows easy setup, flexible configuration and reduced costs for large-scale projection screens. A good overview of camera-based calibration techniques can be found in [68].

Geometric correction on planar surfaces can be performed using simple homographies. A homography describes a transformation between two-dimensional spaces such as projector, camera or display space and is usually represented by a 3x3 matrix. Hence, a transformation of points in one space to another can be carried out with a matrix multiplication. The homography matrix for two spaces can be calculated when several (at least four) corresponding points in both coordinate systems are well known. Often cameras are used to automatically determine corresponding features using projected patterns and serve as a reference space for the transformations between different projector or screen spaces [12, 67].

In this way it is possible to create a geometrically consistent image by defining a common screen space and a transformation to each projector, for instance via a camera. A desired image that is given in screen space can be transformed into the individual projector spaces and displayed. This creates a single image and can be integrated directly into the graphics pipeline. However, overlapping projections and the surface-projector form factors result in varying intensities even when projecting a uniform color. The color range that can be generated with a projector, called gamut, also varies between projectors. Another aspect is the non-linear mapping of intensities displayed by projectors, which is similar to the gamma of conventional CRT displays, but slightly more complex. Each of these issues has to be addressed to create a photometrical seamless image.

Chrominance matching is often ignored since human vision is much more sensitive to intensity variations and the chrominance difference between projectors of the same manufacturer is assumed to be rather small. If required, it can be performed by measuring the color gamut of each projector using a spectroradiometer (which is usually way more expensive than a projector) and map all the device colors to a common (i.e. device independent) color space. The union of this set of color spaces can be displayed and the colors of an original image given in that common space are transformed to the device spaces.

An approach to creating consistent intensities over the entire screen was proposed by Majumder and Stevens [36]. The maximum luminance of the display is measured by a camera. In order to adjust the intensity, a scaling factor is calculated that adjusts every pixel to fit the lowest achievable measured intensity of the display in the camera image. The scaling is evenly distributed over all projectors contributing to that camera pixel and a luminance attenuation map (LAM) is generated for every projector. This contains the per-pixel intensity attenuation that can be multiplied with displayed output.

The intensity transfer function (ITS) of a projector describes its non-linear mapping of input intensities. This has been shown to be spatially invariant [54] but different for each color channel. An ITS can be acquired using spectroradiometers, luminance meters or a camera and high dynamic range imaging (HDRI) as described by Raji et al. [54]. It is used to linearize the projection by applying its inverse, which can be efficiently implemented with programmable graphics hardware and a simple look-up table for each pixel intensity and color.

In combination geometric predistortion using acquired homographies, intensity linearization using the inverse ITS, multiplication with the LAM and, if necessary, gamut matching can be applied to create perceptually seamless multi-projector systems. Except for the gamut matching (cameras have a limited gamut as well) all of the required parameters can be acquired with a camera and HDRI, which allows a completely automatic calibration. The decreased costs for projectors as well as simple and automatic calibration and registration techniques enable to easily build reconfigurable and inexpensive multi-projector displays.

2.2 Correcting Non-Planar Geometric Distortion

The standard approach for displaying images on geometrically more complex surfaces is to project a set of uniformly spaced features with a projector, capture this with a camera and estimate a mapping between corresponding features in both spaces. These features may, for instance, be binary encoded and can easily be triangulated, thereby creating a mesh that can be transformed to either space. Given a desired image in camera space, the textured mesh is warped to projector space and projected. Depending on the distance of the projected features this method may only give a rough approximation of the precise mapping, which

may not be sufficient for complex surfaces. Note that this kind of mapping from projector to camera and vice versa also includes the radial lens distortion of both devices. If the camera's lens distortion is significant, i.e. when using fish eye lenses, this leads to noticeable distortions of the projected imagery.

Techniques that estimate two-dimensional mappings between projector and camera spaces are usually referred to as image-based approaches. In contrast to these are algorithms that require a full three-dimensional description of the scene. An example for this is "the office of the future" as presented by Raskar et al. [57]. Projector-camera systems are used to extract depth and reflectance information of physical objects via projected structured light. Image-based modeling and reconstruction techniques are employed to generate a digital model of the entire environment, which is augmented with the projectors. Therefore, a precise registration of the projectors and the acquired 3D model is necessary. The main drawback of geometry-based registration methods is that the intrinsic projector and camera parameters have to be modeled accurately, which can be quite difficult.

Another approach that requires a given geometric description of the scene was proposed by Raskar et al. [58]. Projectors, called shader lamps, are employed for augmenting white diffuse objects with artificial textures. The desired appearance of the scene can be generated with projective texture mapping for transforming a desired appearance of the scene into the projector spaces. A camera is not essential for this approach.

2.3 Radiometric Compensation of Textured Canvases

Conventional projection displays, as described in the last sections, employ geometric registration and photometric correction methods that allow consistent projections onto uniformly colored (i.e. white) canvases. If the underlying surface has a varying reflectance, radiometric compensation approaches as presented in [46, 23, 4, 6, 17] can be applied to minimize the artifacts induced by light modulation between projection and surface.

Nayar et al. [46] used a planar textured surface as canvas and compensated for it's albedo using a projector-camera system. A geometric mapping between both devices is estimated by capturing 1024 square, uniformly spaced patches. These were projected in a binary coding style within 10 images. A precise mapping between camera and projector pixels is described by piecewise second-order polynomials. The radiometric compensation model is based on the assumption $C = VP$, where C is a camera pixel, P the corresponding projector pixel and V a color mixing matrix that takes the surface's albedo and the spectral differences between camera and projector into account. A compensation image P that has to be projected to create a desired image C is given by $P = V^{-1}C$. The necessary data has been reported to be 500 MB for a camera with a resolution of 640x480 and a projector resolution of 800x600. Acquiring the color mixing matrices for each camera pixel and the projector ITF required to capture 260 images.

A similar approach was presented by Grossberg et al. [23] who employed a projector-camera configuration to perform a radiometric compensation of geometrically complex surfaces. A similar registration as described in [46] is performed. The camera must be placed near the projector, because the resolution of the correspondences is quite coarse for only 1024 projected patches. A compensation image is generated with an enhanced version of the algorithm described in [46]. The inverse projector ITF is stored separately from a single color mixing matrix V per camera pixel using 16 half-precision (16 bit) floating point values. The amount of required data is reported to be reduced to 7.7 MB, which allows a real-time compensation.

Both described compensation approaches rely on an off-line calibration, in which all parameters for the on-line compensation are acquired. Fuji et al. [17] introduced an adapting real-time algorithm for radiometric compensation of dynamic environments. The projector and camera are coaxial by mounting a mirror beam splitter in between and adjusting the viewing frustra to be equal. The radiometric model is based on the previous ones, but also takes the environmental light contribution F and the varying surface reflectance A into account: $C = A(VP + F)$. A is in this case a 3x3 diagonal matrix. A dynamic compensation at time t is based on the assumptions that V remains constant, P is the known projection, the changes in environment light are negligible ($F^t \approx F^0$) and C^t is captured by the camera. Using these assumption the changes of the captured scene reflectance is recovered using the error between captured and desired image. The compensation is performed in real-time using the reflectance computed in the last time step.

Augmentation techniques to superimpose pictorial artwork with projected imagery were proposed by Bimber et al. [4]. A transparent film that reflects a portion of incident light while transmitting the remaining portion was mounted in front of picture. This special film material allows to project onto surfaces, where the underlying color pigments reflect only few light. Manual geometric registration between projection and the picture is necessary to align the images. A digital representation of the artwork is assumed to be present, otherwise it can be scanned. The radiometric compensation is performed in real-time on a per-projector pixel basis with programmable graphics hardware. Incident environment light as well as the light bouncing between the picture and the film are considered in the compensation. A camera is not required.

An approach that corrects complex geometry of unknown surfaces and compensates for radiometric artifacts was proposed by Bimber et al. [6]. Projected structured light patterns [53, 27, 61] are used to estimate a direct mapping between all projector and camera pixels. This results in a much higher resolution of the generated look-up tables compared to the binary patterns used by Nayar et al. [46] and allows to place the camera at arbitrary locations. The mapping allows accessing a captured floodlight image of the camera on a per-projector pixel basis. This floodlight image contains color values, which are directly used for a real-time radiometric compensating. In contrast to the color mixing matrix this does not take the spectral differences between camera and projector into account. Using this image-based technique, a

compensation image can be created for arbitrary located stationary cameras and projectors.

The geometric mapping has been enhanced to also support moving viewers [9]. In a preprocessing step, direct pixel correspondences between projectors and a set of cameras are estimated using projected structured light. Weights are assigned to each source camera based on the position of a tracked user. In order to perform a real-time compensation image-based rendering (IBR) methods are employed. Based on the weights, the geometric warping is adapted by modulating the projector-to-camera mappings between the calibrated source cameras. Color values are interpolated as well. This technique allows view-dependent stereoscopic projections onto everyday environments.

All of the discussed compensation methods try to create a correct image for a camera. A content-dependent optimization of radiometric compensation for human perception was described by Ashdown et al. [3]. The compensation itself is separated from a modification of the original image. This modification is a five-stage process that adjusts luminance and chrominance of the image so that a conventional radiometric compensation of the adjusted image leads to less visible artifacts within the projection. The five stages consist of a transformation of the original image into a perceptually uniform space, a chrominance fitting to the projector's gamut, a luminance range computation based on the calculated chrominance, a fitting to the computed luminance range and the compensation itself. Grundhöfer and Bimber [25] presented a real-time technique for a content-dependent radiometric compensation.

An approach to compensate specular reflections with projector-camera systems is presented by Park et al. [49]. Multiple projectors are employed to minimize specular reflections on a registered canvas with well-known geometry. It is assumed that if reflections occur, these are mostly due to a single projector. Hence, its contribution to the projection can be reduced while other displays generate more light in this area to compensate for a loss of brightness. Specular reflections are estimated using the viewer's position, the projector's incident angle and the surface normal. This technique is capable of compensating a specific type of global illumination assuming a well-known surface geometry and registration.

2.4 Focus Related Projector-Camera Techniques

Another interesting aspect of projections is focus/defocus. Levoy et al. [33] proposed a technique that uses multiple virtual cameras (represented by a single camera and an array of mirrors) for simulating a camera with a wide aperture. This is used to blur out objects, which are not located on the focal plane. In an experiment it has been shown that it is possible to see a telephone calling card through murky water. The camera can be replaced by a projector that displays binary patterns for lighting objects at the focal plane. Partly occluded objects can be selectively illuminated in this way.

Bimber et al. [5] employed multiple overlapping projectors with different focal planes that create a seam-

less image with minimal defocus. Therefore, a camera is used to measure the defocus of each projector pixel of every projector. A composition of all available displays that results in minimal defocus can be created by blending the individual projector contributions.

The overall sharpness of an image projected by a single projector was enhanced by Nayar et al. [70]. The depth of an arbitrary complex scene is estimated with depth-from-defocus methods. This is used to create a modified image that, when projected onto the scene, appears to be more sharp than the corresponding unmodified image. Refocused projected imagery appears not as blurry as if projected unmodified.

2.5 Forward Light Transport, BRDF Acquisition and Relighting

Several approaches have been proposed to capture forward light transport [63, 14, 22]. This implicitly takes all global illumination effects such as (subsurface) scattering, reflections, refractions, dispersion, diffraction etc. into account. The acquired light transport is used for different applications such as BRDF (bidirectional reflection distribution function) acquisition, relighting or environment matting. Depending on the application different sorts of light sources can be used to measure the forward light transport. The following paragraphs introduce several techniques that are representative for approaches to various problems related to light transport. Due to the huge amount of work that has been done, covering all related work is not emphasized here.

An example of BRDF acquisition was described by Goesele et al. [22]. A laser pointer illuminated an object at different positions and from different angles. The resulting impulse response was measured with a high dynamic range camera. All acquired data was resampled and interpolated for missing locations, which allowed to efficiently synthesize the object under novel illumination and from variable viewpoints [32]. Goesele's technique requires a well-known and registered geometry of the scanned objects. Peers et al. [51] presented a novel method to capture heterogeneous subsurface scattering of objects with an unknown geometry using a projector-camera system. The acquired transport matrix is factorized and can be applied to arbitrary objects for relighting them with the captured BSSRDF (bidirectional surface scattering reflectance distribution function).

While lasers represent very bright displays, they can only synthesize fixed wavelengths at a small region. Light sources, e.g. LEDs, exist that can be computer controlled and arranged to create different colors. These sample the reflected light of human faces under various illumination conditions [14]. The light sources are mounted on a dome-like spherical gantry called the light stage. Image-based relighting approaches are employed to display the acquired face under arbitrary distant illumination, for example using environment maps [10].

The most flexible displays, compared to laser pointers and distant light sources, are video projectors since

they are spatially variant, have a large gamut and are completely controllable. Masselus et al. [43] described a projector-based relighting approach. A scene is scanned by assuming only direct illumination. Multiple uniformly spaced square patches are displayed with a projector and captured with a camera. The projector can be moved to different locations, thereby illuminating the object from various positions. The resulting outgoing light field allows to synthesize images of the scene under arbitrary new illumination conditions.

Another projector-based approach is dual photography as proposed by Sen et al. [63]. The 4D light transport between a projector and a camera is captured and allowed to interchange the camera and projector. This is done by representing the light transport as a matrix, transposing it and multiplying with an illumination pattern for creating a dual image (picture of the scene from the projector's point of view, illuminated from the camera). A detailed discussion on dual photography can be found in section 3.4.3. Recently, Garg et al. [20] captured the full 8D reflectance field and introduced hierarchical tensors as the underlying data structure.

Environment matting refers to capturing mattes and reflectance properties of (transparent, refractive etc.) objects and placing them in a new environment. Zongker et al. [71] presented an approach to capture complex refraction and reflection properties of objects. These were acquired by a camera-monitor setup and used to rerender the scene with novel backgrounds. Even though the captured environment mattes are only approximations of the true reflectivity, convincing results were synthesized and compared to photographs of a similar setting.

Except for the radiometric compensation methods, relatively few work has been done on relighting real scenes. Debevec et al. used various version of the light stage [15, 66] to simulate artificial incident illumination on human actors in real-time. These were filmed and superimposed on a virtual background. The illumination conditions of this background served as a reference for the physical light synthesis.

In order to create a perceptually more seamless illumination between a virtual and the real environment, Gosh et al. [21] combined computer controlled light sources with image-based lighting techniques. The displayed content of conventional or high dynamic range displays was analyzed for actively controlling the real illumination.

Consistent illumination within optical see-through augmented environments was explored by Bimber et al. [7]. A projector-camera configuration enabled the acquisition of diffuse reflectance of a real object and could later be used to create consistent illumination effects such as shadows, shading and reflections between real and virtual objects.

2.6 Inverse Illumination

A compensation of scattering within immersive and semi-immersive projection screens of a known and registered geometry using a reverse radiosity approach has been proposed by Bimber et al. [8]. The screen space is subdivided into discrete patches. Form factors for these are computed based on the surface material. These are utilized to simulate global light interaction between the patches depending on a projected image. Real-time framerates were achieved with a radiosity implementation on programmable graphics hardware.

While the form factors for the reverse radiosity approach are precomputed using a given geometry, Seitz et al. [62] proposed a technique that automatically determines global illumination effects using a camera and a laser pointer. Latter is mounted on a movable gantry and illuminates a set of predefined points on the scene's surface. This is captured by a camera using high dynamic range imaging. The centroid of each directly illuminated spot in every camera image is determined. Sampled luminance values of all centroids in one of the camera image represent an intensity scatter function (ISF). The combination of all ISFs forms an ISF matrix that is the basis for computing inter-reflection cancellation operators, which are applied to cancel out indirect illumination of photographs taken from the camera's point of view. Note that the ISF matrix only considers the outgoing light field.

Light fields have previously been used for image-based rendering (IBR) [34] and represent a 4D function describing light propagation in three-dimensional space. The outgoing light field of a scene defines reflected light on every point of a bounding surface for every possible direction. Combining the outgoing light field with an incoming light field (its equivalent for all incoming light) allows to describe the full 8D reflectance function [14] of an object or scene. Other work for recovering diffuse and specular material properties from photographs has been done by Yu et al. [69]. However, the geometry of the scene, which can be quite complex, is assumed to be known.

3 Light Transport and Acquisition

This chapter is to define basic terminology and discuss fundamental knowledge of light, its digital representation and capture. General concepts of light transport and effects resulting from the interaction of radiant energy and objects are introduced. The presented approaches focus on information that can be derived of a scene using computer controllable display devices in combination with digital cameras. Specifically, approaches are discussed, that exceed simple digital photography by acquiring irradiance values that reflect physical quantities rather than non-linear mappings in form of 8 bits pixel brightness. Conventional digital photography is often not capable of capturing scenes that contain a high dynamic range[1] properly. Methods that allow handling such scenes are explained along with their accelerated implementation on programmable graphics hardware.

The combination of high dynamic range imaging and sophisticated structured illumination techniques allows capturing the full light transport between projectors and cameras. Illumination patterns and light transport representation are discussed as well as the application to dual photography, a method that enables to interchange a camera and a projector. Using dual photography, an image from the projector's point of view, illuminated from the camera's position can be synthesized. The captured light transport matrix is fundamental for all radiometric compensation techniques described in later chapters.

3.1 Radiometric and Photometric Terminology

Research in the fields of radiometry and photometry is all about measuring light. Radiometric quantities define units for the power of electromagnetic radiation. Photometry measures light in terms of its perceived brightness to human vision. This section is to discuss the difference between both sciences as well as to introduce their basic qualifiers.

Light is energy, thus measured in joules. Radiant power (a.k.a. radiant flux) P_e defines the amount of energy over time. Irradiance E_e and radiant exitance M_e on the other side measure incident flux on a surface and the power leaving an area respectively. In order to describe the amount of radiant flux in a certain direction (radiant intensity), a solid angle $d\omega$ has to be taken into account. This angle is given in steradians sr, a unitless quantifier defining area on the unit sphere. However, one of the most commonly used quantities in radiometry is radiance L_e. It measures energy per time and direction as well as per area. Table 3.1(a) gives an overview of the most important radiometric quantities along with their units.

The science of photometry has quantifiers that are quite similar to the radiometric ones (see table 3.1(b)). They are differentiated by the subscripts e for radiometric values and v for photometric quantities. Pho-

[1] The dynamic range of a scene is the ratio of the brightest and the dimmest pixel value. Alternative concepts such as local dynamic range exit - these describe the amount of measurable nuances within that range.

(a)

Quantity	Unit	Definition
Radiant energy (Q_e)	J (joule)	Q_e
Radiant power (P_e)	$J\,s^{-1} = W$ (watt)	$P_e = \frac{dQ_e}{dt}$
Radiant exitance (M_e)	$W\,m^{-2}$	$M_e = \frac{dP_e}{dA_e}$
Irradiance (E_e)	$W\,m^{-2}$	$E_e = \frac{dP_e}{dA_e}$
Radiant intensity (I_e)	$W\,sr^{-1}$	$I_e = \frac{dP_e}{d\omega}$
Radiance (L_e)	$W\,m^{-2}\,sr^{-1}$	$L_e = \frac{d^2 P_e}{d\,A \cos\theta\, d\omega}$

(b)

Quantity	Unit
Luminous power (P_v)	lm (lumen)
Luminous energy (Q_v)	$lm\,s$
Luminous exitance (M_v)	$lm\,m^{-2}$
Illuminance (E_v)	$lm\,m^{-2}$
Luminous intensity (I_v)	$lm\,sr^{-1} = cd$ (candela)
Luminance (L_v)	$cd\,m^{-2} = nit$

Table 3.1: Radiometric (a) and photometric (b) quantities as defined in Reinhard et al. [59]. The cosine term of L_e is the angle between the surface normal and the angle of incidence.

tometry differs to radiometry by weighting spectral distribution of radiant energy with a function. This has been defined by the Commission Internationale de l'Eclairage (CIE) and represents the spectral sensitivity of human photoreceptors. It is usually referred to as the photopic luminous efficiency curve or photopic luminous function V_λ (as illustrated in figure 3.1). Radiance can be converted to luminance using V_λ

$$L_v = \int L_e(\lambda)\,V(\lambda)\,d\lambda. \tag{3.1}$$

Since V_λ is based on the human visual system (HVS) it only takes the visible light spectrum (wavelengths between 380nm and 830nm) into account. In physical terms, one lumen is defined by a radiometric reference value. This is 1/683 watt of radiant power at a frequency of $540x10^{12}$ Hz (about 555nm).

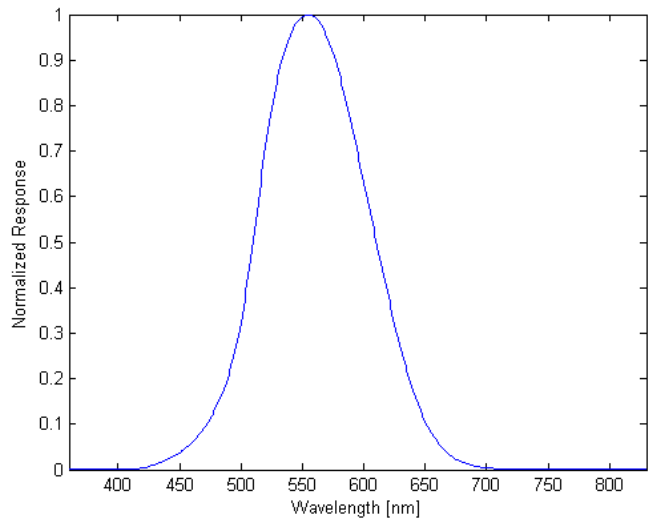

Figure 3.1: The photopic luminous efficiency curve as defined by CIE. The plot is based on data from CVRL [1].

3.2 Light Transport - From Source to Sensor

All visual perception results from light reaching a sensor that is capable of converting incident radiance into a representation that can be further processed, stored, transmitted and/or perceived. Light is produced by transforming some form of energy into radiant energy. Emitting sources are for instance the sun, light bulbs or a candle flame. Matter interacts with light, leading to a modulation that results in effects such as reflection or refraction of light, scattering or absorption. Eventually, it reaches a sensor, i.e. a human eye, where incident radiance is converted to electrical signals that carry information to the brain, where the perceived image is generated.

Radiant energy emitted by light sources has a specific spectral distribution. Max Planck invented a model for this based on the emission of a black body radiator. This is a theoretical material that absorbs all incident radiance without reflecting or transmitting light. As the body is heated, it starts to transform heat into radiance with a continuous spectrum. Thus, the emitted spectrum of a black body radiator at temperature τ Kelvin is referred to as color temperature τ. Table 3.2 shows the temperatures of several standard light sources and illumination conditions. Although such a body does not exist, many metals behave in a similar way. Artificial light sources can be characterized by their color temperature (often specified by the manufacturer), however, this is only an approximation of the source's actual distribution since it is not necessarily continuous depending on the emitting material. Radiant power, given in Watts, on the other hand quantifies the intensity of light. It is defined as the integral over the power of the radiated light's spectrum $P_e = \int P_{e,\lambda} d\lambda$. The combination of color temperature and radiant power gives a good idea of light characteristics.

Scene	T (in °K)
Candle flame	1850
Sunrise / sunset	2000
Tungsten (TV / film)	3200
Summer sunlight at noon	5400
CIE A (incandescent)	2854
CIE B (direct sunlight)	4874
CIE C (overcast sky)	6774
CIE D50 (noon skylight)	5000
CIE D65 (average daylight)	6504
CIE E (equal energy)	5500
CIE F2 (office fluorescent)	4150

Table 3.2: Color temperatures of common light sources and scene types. Courtesy: Reinhard et al. [59].

Video projectors are special light sources in being spatially variant. Despite the blacklevel, each projector pixel's radiation can be controlled separately. Various techniques for modulating the pixels exist, including controllable moving micro mirrors for intensity modulation with color separation filters (DLP technology) or liquid crystal elements (LCD projectors). Detailed aspects of these and other projection technologies are discussed in Stupp and Brennesholtz [64].

Irradiance is a physical quantity that measures incident radiant power per unit area on a surface. The radiance transfer between a point light source and a planar surface is given by

$$E_e = \frac{cos\ \alpha}{d^2} P_e, \tag{3.2}$$

where the fraction is the form factor including the angle between surface normal and incident light direction α and the distance of a point on the surface and the light source d. Interaction effects of light and some object may include reflection, refraction, scattering, polarization, dispersion and diffraction. The modulated light can bounce several times back and forth to other surfaces until it eventually reaches a sensor.

Physical laws exist for most of the light effects given a proper description of the scene (3D geometry and material properties). The mathematical model describing the interaction of a surface and incident light is called bidirectional reflection distribution function (BRDF) as introduced by Nicodemus et al. [48].

A digital camera is a sensor that can be used for sampling scene radiance. Among the essential parts of a camera are the light sensor elements that measure irradiance and a lens system for focusing light rays on the sensor. Typical sensors are charged coupled devices (CCDs) and complementary metal oxide

semiconductors (CMOS). Most digital cameras generate pictures with three color channels (red, green and blue), the sensor elements on the other hand are usually sensitive to the entire light spectrum. Thus spatially fixed color bandpass filters are mounted in front of the sensors to separate incident illumination into multiple components. In this way each sensor element measures one channel only, leading to an image that actually contains single color samples. The color values have to be interpolated in order to generate a picture with three channels per pixel. Depending on the interpolation method color artifacts may occur.

The lens system of a camera is a collection of spherical glass or plastic lenses and apertures. Latter control the light rays that pass through the system and actually reach the sensor elements while the lenses redirect and bundle rays. Lenses induce certain effects in the captured image such as radial image distortion, depth of field and vignetting (decreasing intensity with increasing distance to the optical axis). Kolb et al. [29] proposed a physically-based camera model for computer graphics that is capable of simulating the effects induced by the lens system.

The spectral distribution of irradiance on a sensor depends on the spectrum of the emitted light and its modulation due to material properties of objects in the scene. This means that a white object reflects incident light uniformly, i.e. when illuminated with red light it reflects only red light. While the human visual system easily adapts to different illumination conditions, a digital camera does not. In order to generate pictures where objects appear in their true colors, cameras can apply an additional scaling to the individual color channels called white balancing. Often predefined settings for different lighting situations (e.g. sunlight, cloudy sky, halogen light bulb etc.) based on the light source's color temperature can be chosen from the camera settings. Such color processing may be of interest for capturing visually pleasant pictures, but it leads to a color distortion of the physically present spectral distribution of radiance on digital sensors.

Other effects of digital imaging include blooming (the charge of a light sensor element is transmitted to neighboring elements), analog-to-digital conversion quantization and different types of noise.

3.3 High Dynamic Range Imaging

High dynamic range imaging is a field that recently gained more and more interest in the computer graphics and image processing research community. Common digital imaging devices usually encode captured real-world irradiance values in 8 bits per channel. This limits the dynamic range of the content to 256:1. Photographing a white sheet of paper contains very low dynamic range (LDR), while capturing a scene containing the sun, dark shadows and moderately bright spots contains a very high dynamic range (HDR). A photographer can adapt aperture, shutter speed and other settings to capture a HDR scene properly, however due to the limited range of the device either darker or brighter parts are saturated. Irradiance

values that are higher that the maximum pixel value are clamped to 255, low radiances are clamped to a brightness of 0.

A straightforward approach to encode high dynamic range content into a single image is to combine several LDR photographs captured with different exposures. Varying the exposure is typically done by changing the integration time or shutter speed. While it is also possible to vary the aperture, this is usually avoided due to depth of field variations, vignetting and other optical effects that may occur. If a camera is not capable of changing the shutter speed, neutral density filters can be used as well. At least one or two of these input images should contain unsaturated information about a pixel location, allowing irradiance to be recovered and stored. These recovered values are stored as radiance or irradiance maps[2] and usually encoded as 16 or 32 bit floating point values, although alternative representations exist (e.g. RGBE [65]).

As radiant energy hits a sensor it is converted to a representation that can be stored or transmitted by the sensor. Examples include photochemical reactions on films in analog cameras or collected charge of charged coupled devices (CCD). It is quite common to induce non-linear mappings in the imaging pipeline that transforms captured irradiances to digital images. The pipeline may include scanning, printing, analog-to-digital conversion and many other procedures. Digital cameras using CCD elements often apply such mappings to mimic an analog film response, adapt resulting pixel intensities to the human visual system (HVS) and/or a specific display. Sometimes simply to map a greater range of physical intensity values into the 8 bit space of conventional digital images.

In order to combine multiple LDR exposures correctly the effect of non-linear mappings has to be neutralized. This is performed by estimating the response curve that is used to transform physical quantities. Then, the image device's inverse response curve is applied directly to the pixel brightness values, resulting in a linearization of the images. However, the inverse response has to be estimated before it can be applied, since it is usually unknown.

Numerous techniques have been described that compute approximations of film response and other mappings within the image acquisition pipeline [16, 60, 44, 24, 47, 42]. An overview of different approaches can be found in [28].

All approaches are based on the assumption that pixel values Z_i are the product of the irradiances E_i and the exposure time Δt transformed by the camera response function f:

$$Z_{ij} = f\left(E_i \Delta t_j\right), \tag{3.3}$$

[2]The difference between radiance and irradiance was discussed in section 3.1. Cameras always capture irradiance values. However, Debevec [16] states that most modern lenses compensate for intensity variations on the sensor surface. Thus, captured irradiance is proportional to scene radiance and both terms will be used to refer to HDR images.

where i indicates pixels and j different exposure times. Some cameras allow to deactivate the camera response mapping, so that f can be discarded. If this is not possible several photographs of a static scene[3] from a fixed viewpoint with different integration times (shutter speeds, Δt_j) can be taken and used to estimate a camera response curve. For detailed discussions on response curve recovery the reader is referred to [16, 60, 44].

3.3.1 Radiance Map Recovery

Knowledge of the camera response curve allows computing relative irradiance values that are linear proportional to absolute physical quantities. Based on the defined observation models, arbitrary exposures, not only the ones used for the response recovery, can be combined. Weights are applied to weaken the effect of more uncertain pixel values near the extremes. This is necessary to compensate for saturated regions and noise, which is more likely to occur in darker parts of the image.

According to Debevec's model [16] the logarithm of the exposure E_i can be computed as follows:

$$\ln E_i = \frac{\sum_{j=1}^{P} w(Z_{ij})\,(g(Z_{ij}) - \ln \Delta t_j)}{\sum_{j=1}^{P} w(Z_{ij})}, \tag{3.4}$$

where Z_{ij} is a pixel value that results from exposure and incident irradiance transformed by the camera response f with $Z_{ij} = f\,(E_i \Delta t_j)$. P is the number of input images, g the natural logarithm of the recovered inverse camera response and Δt_j the shutter speed of exposure j. A simple hat function w is used to weight pixel intensities

$$w(z) = \begin{cases} z - Z_{\min} & \text{for } z \leq 0.5\,(Z_{\min} + Z_{\max}) \\ Z_{\max} - z & \text{for } z > 0.5\,(Z_{\min} + Z_{\max}) \end{cases}. \tag{3.5}$$

A different model was proposed by Robertson et al. [60]. It is also based on equation 3.3, but differs to Debevec's approach by taking quantization noise into account. It is assumed that longer exposures result in more reliable measurements of true irradiance values, which is indicated by the t_i term in the numerator

$$E_i = \frac{\sum_{j=1}^{P} w\,(Z_{ij})\, t_j\, f^{-1}\,(Z_{ij})}{\sum_{j=1}^{P} w\,(Z_{ij})\, t_j^2} \tag{3.6}$$

Note that the original notation of the authors is changed for consistency. In this equation f^{-1} is used instead of g with $g = lnf^{-1}$. Robertson et al. also use a different, Gaussian-like weighting function, which is scaled and shifted so that $w(0) = w(255) = 0$ and $w(127.5) = 1.0$:

$$w\,(Z_{ij}) = e^{-W \frac{\left(Z_{ij} - 127.5\right)^2}{127.5^2}}. \tag{3.7}$$

[3] The illumination conditions of the scene are assumed to be constant during the acquisition time.

Figure 3.2: A HDR scene composed of 13 different exposures with one-f/stop increments starting at 0.125 ms. Approximately three orders of magnitude are encoded in the irradiance map. The lower right image shows a tonemapped representation of the scene.

Figure 3.2 shows an HDR scene that was computed by combining 13 LDR input images. The source images were captured with shutter speeds of 0.125 ms, 0.25 ms, 0.5 ms, 1.0 ms, 2.0 ms, 4.0 ms, 8.0ms, 16.0 ms, 32.0 ms, 64.0 ms, 128.0 ms, 256 ms and 512.0 ms. The lower right image shows a tonemapped version of the irradiance map. This was done by taking the natural logarithm of the values v and scaling the values $v \in \{v_{min}, ..., v_{max}\}$ between v_{min} and v_{max}. A dynamic range with approximate three orders of magnitude is encoded in the map as indicated by the red squares with relative RGB irradiance values, indicated from left, of 0.006, 0.73 and 20.08 respectively.

3.3.2 GPU Accelerated Implementation

The performance of graphics processing units (GPUs) has evolved tremendously during the last decade. One of Nvidia's current top models, the GeForce 7800 GTX, has 302 million transistors. Compared to Intels Pentium 4 6xx CPUs with 169 million transistors, this is almost twice as much[4]. GPUs are designed to operate highly parallel, thus optimized for single-instruction-multiple-data (SIMD) tasks. Due to their high potential, many research groups have been concerned with implementing general purpose computation on GPUs (GPGPU) such as numerical algorithms [11, 30]. Another active field of research is image processing on programmable graphics hardware [18, 19]. Due to their parallel nature, most image processing techniques are well suited for being ported to the GPU.

Generating a high dynamic range map is an example of image processing that allows processing multiple fragments in parallel. All it requires is a set of LDR input images and their corresponding exposure values, the camera response function as well as the precomputed weighting function (both can be encoded as textures). These are passed into the shader as uniform parameters. Figure 3.3 shows different weighting functions and their representations as textures. Since the shader-code depends on the number of input images, it is dynamically synthesized using the standard C++ string structure.

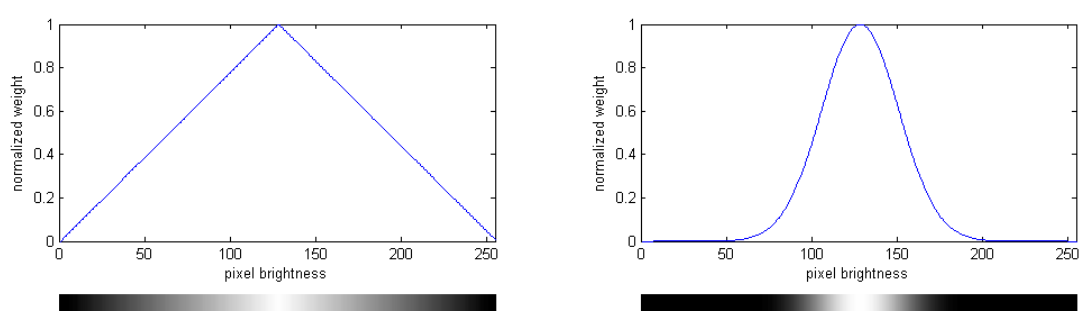

Figure 3.3: Weighting functions with corresponding textures: (a) shows a simple hat function as used by Debevec while (b) is a Gaussian-like distribution function.

[4]The amount of transistors is not necessarily linear proportional to overall performance.

While an implementation on programmable graphics hardware yields a huge performance boost it also has its limitations. Nvidia's GeForce FX and GeForce 6800 families support up to 16 textures per fragment shader. If the number of LDR input images exceeds the amount of available texture units, alternative approaches have to be used. This can, for instance, be done by encoding multiple images into one texture. The maximal size of textures on current Nvidia graphics adapters is 4096x4096, which limits the amount of images that fit in one texture. A different method would be to use a hierarchical multi-pass rendering algorithm. Intermediate results can be stored in floating point offscreen buffers (e.g. framebuffer objects) and combined successively. The number of passes depends on the amount of textures that are used. However, it should be noted that the number of required exposures for most applications (especially when multiple images are encoded in a single texture) does not exceed the amount of available texture units.

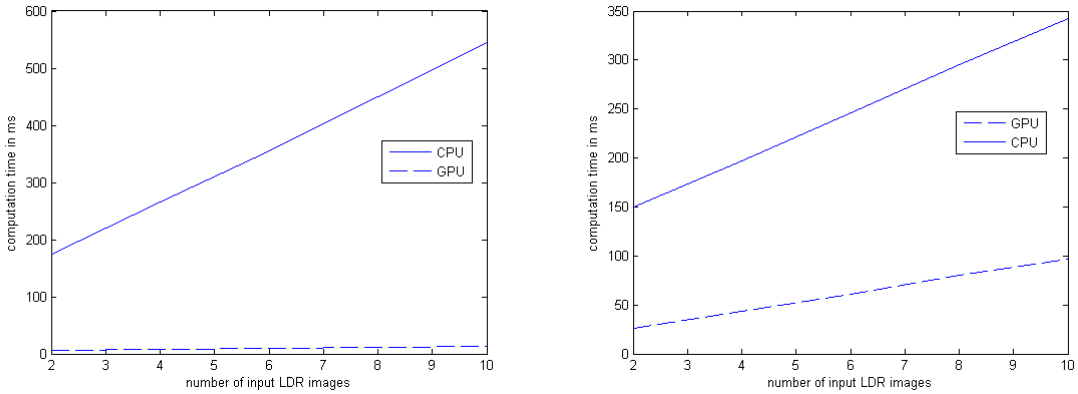

Figure 3.4: Performance comparison of (a) HDR computation on the GPU and CPU and (b) LDR image averaging. The data was measured on an Intel Pentium 4 3.0 GHz with a Nvidia GeForce 6800 GT.

Debevec and Malik [16] use single-precision floating point values for computing the radiance map. The simple GPU implementation described above utilized 16 bit floating point values and framebuffer objects to hold the resulting map on the GPU, which is downloaded on the CPU again. Using 16 bits allows to encode approximately 10 orders of magnitude in radiance and is compatible to Industrial Light and Magic's OpenEXR format [35] to store files to harddisks. Figure 3.4 (a) shows a comparison of two different HDR algorithms and implementations on the GPU and CPU.

3.4 Light Transport Revisited

Light transport determines how light propagates in three-dimensional space. On its way to a sensor it is modulated by matter. A straightforward description of light transport along with general radiance distribution equations are introduced in the next section. Techniques for an efficient acquisition of the light transport matrix as well as applications are presented as well.

3.4.1 Light Fields and the Rendering Equation

The most widely used scene description for computer graphics are explicit models. Geometry is defined by a mesh contain vertices and edges that form polygons. Materials and light sources are defined by several attributes, such as diffuse and specular emitance or reflectance. Rendering an image from a specific viewpoint requires to compute light distribution within the scene and the final contribution to each camera pixel.

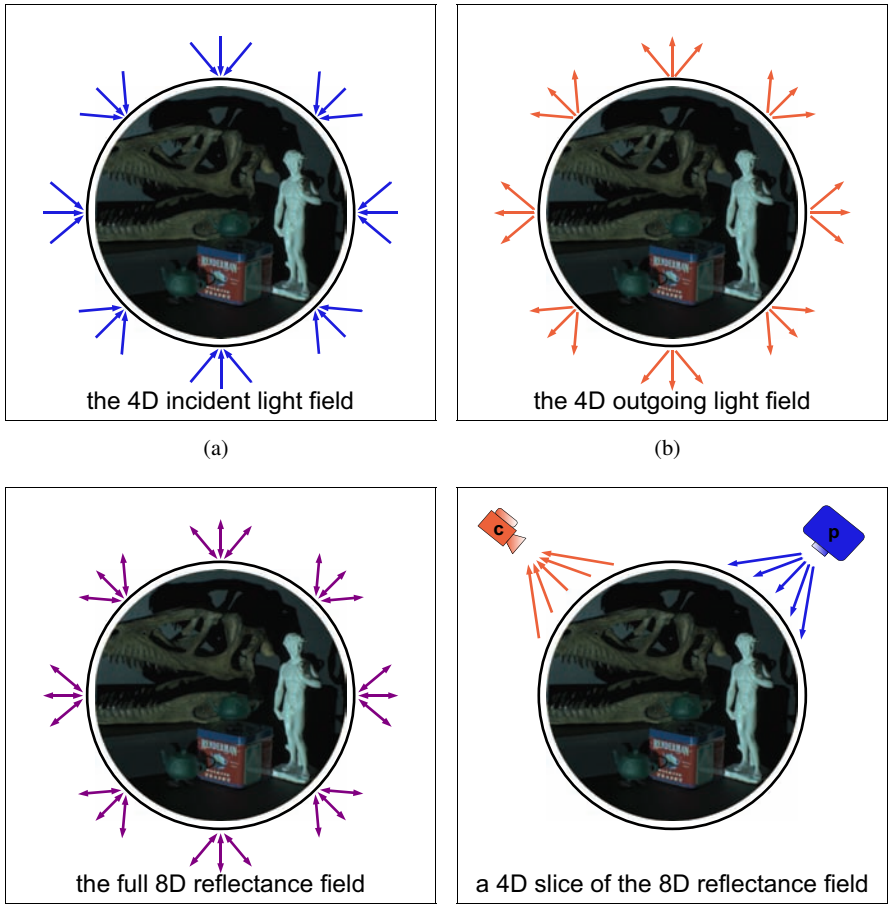

Figure 3.5: The 8D reflectance field of a scene (c) is defined by the incident light field (a) and the outgoing light field (b). A projector-camera system samples the reflectance field at discrete locations (d).

An alternative approach is given by an image-based model of an environment. Light rays are used as a basis of the scene description. A common representation is the 8D reflectance field [14]. Incoming light is defined for every point on a bounding volume and each incident direction. Usually this is referred to as the incident light field. Similarly, the outgoing light field described exitant radiance for each of the surface points in every direction. Figure 3.5 depicts a scene, its bounding sphere as well as the incident

light field (fig. 3.5 (a)), the outgoing light field (fig. 3.5 (b)) and the full 8D reflectance field (fig. 3.5 (c)). A projector-camera system samples the reflectance field at discrete locations (d).

Various simplifications of the 8D reflectance functions are possible for a camera-projector setup, depending on the number of cameras and projectors. The light transport of a single projector to one camera is defined by a 4D slice of the full 8D reflectance field (see figure 3.5 (d)). Capturing the illumination of multiple projectors with a single camera or using multiple cameras to record radiance emitted by a single projectors allows acquiring a 6D slice of the reflectance field. Determining the full function requires measuring the transport for multiple projector positions with more than one camera.

The radiance transfer within a scene can be described by the surface balance equation [26]:

$$L_o(x,\vec{w}) = L_e(x,\vec{w}) + \int_\Omega f_r\left(x,\vec{w}',\vec{w}\right) L_i\left(x,\vec{w}'\right)\left(\vec{w}'\cdot\vec{n}\right) d\vec{w}' + \int_\Omega f_t\left(x,\vec{w}',\vec{w}\right) L_i\left(x,\vec{w}'\right)\left(\vec{w}'\cdot\vec{n}\right) d\vec{w}', \quad (3.8)$$

where L_o is the outgoing light at a point x in a specific direction $\vec{w}$. L_e is the emitted light in the outgoing direction, L_i describes the incoming light at the point from a direction $\vec{w}'$. The surface reflectance of a point depending on an incident and an exitant ray is defined by f_r, similarly, f_t describes the transmitted light. $\vec{n}$ is the surface normal. Generally speaking, the outgoing light of a point in a certain direction within the scene or a point on the bounding volume is given by the sum of its self emission in that direction, the entire light reflected (described by the first integral) and transmitted light towards that direction (second integral).

The surface balance equation separates reflected and transmitted light. Simulating light transfer in computer graphics is often performed by discarding the transmitted light, hence, all transparent and refractive objects. A somewhat more general reformulation of equation 3.8 is given by

$$L_o(x,\vec{w}) = L_e(x,\vec{w}) + \int_\Omega \widetilde{T}\left(x,\vec{w}',\vec{w}\right) L_i\left(x,\vec{w}'\right) d\vec{w}', \quad (3.9)$$

where $\widetilde{T}$ describes any kind of light transfer, including all global illumination effects, within the scene and the geometric relations (form factors) between surfaces. The discrete version of equation 3.9 is given by:

$$L_o(x_i) = L_e(x_i) + \sum_j T(x_i, w_j) L_i(w_j), \quad (3.10)$$

where x_i samples the outgoing and w_j the incoming light field. $L_e(x_i)$ is the self emission of any surface visible at x_i. The continuous light transfer function $\widetilde{T}$ is replaced by the transport matrix T.

In case of a single-projector, single-camera system the transport matrix has the size $mn\ x\ pq$ with a camera of resolution mn and a projector illuminating pq different pixels. Each of T's columns completely

describes the contribution of a single projector pixel to the entire camera image, thus adding all of the columns results in a composition that simulates a photograph of the completely illuminated scene.

In general, the forward light transport from a projector through a scene into a camera can be expressed by the matrix multiplication[5]:

$$\begin{array}{ccccc} \vec{c}_\lambda & = & T_\lambda & \vec{p}_\lambda & + \quad \vec{e}_\lambda \\ \begin{bmatrix} c_{\lambda 0} \\ \vdots \\ c_{\lambda(mn-1)} \end{bmatrix} & = & \begin{bmatrix} t^0_{\lambda 0} & t^1_{\lambda 0} & \cdots & t^{(pq-1)}_{\lambda 0} \\ \vdots & & \ddots & \vdots \\ t^0_{\lambda(mn-1)} & t^1_{\lambda(mn-1)} & \cdots & t^{(pq-1)}_{\lambda(mn-1)} \end{bmatrix} & \begin{bmatrix} p_{\lambda 0} \\ p_{\lambda 1} \\ \vdots \\ p_{\lambda(pq-1)} \end{bmatrix} & + \begin{bmatrix} e_{\lambda 0} \\ \vdots \\ e_{\lambda(mn-1)} \end{bmatrix} \end{array} \tag{3.11}$$

where $\vec{c}_\lambda$ is a column vector representing a single color channel λ of the camera image (size mn) and $\vec{p}_\lambda$ is the projected pattern (size pq). T_λ is the matrix describing the transport from every projector pixel to each camera pixel (size $mn\ x\ pq$) including all possible illumination effects such as reflections, refractions, scattering etc. This representation can easily be extended to include multiple projectors and cameras. The size of the matrix would increase to $rmn\ \ x\ \ kpq$ for r cameras and k projectors (assuming a similar resolution for all cameras and all projectors). Emitted radiance and environment light including the projector's black level is denoted by $\vec{e}_\lambda$.

Once the light transport matrix is acquired, this representation allows to simulate a camera image under arbitrary projector illumination with a simple matrix multiplication. Equation 3.11 describes the most fundamental relation in a projector-camera system.

3.4.2 Scene Adaptive Hierarchical Light Transport Acquisition

Simple structured light schemes can determine a direct mapping from projector to camera space and vice versa. However, these techniques make certain preconditions, the most important one is that projector pixels only affect a very small, localized region in the camera image. While this assumption is true for convex diffuse objects, it does not account for more complex scenes containing global illumination effects, such as refractions, reflections, light scattering, diffraction, dispersion etc., where a direct mapping is not given. Thus, advanced patterns that allow capturing all of these effects, have to be employed.

A simple approach to capturing the full light transport would be to illuminate every projector pixel sequentially and filling T's columns successively with the pixel values of the appropriate camera images.

[5]The discrete outgoing light field L_o will furthermore be denoted by $\vec{c}$ to refer to a camera image, the incident light field as $\vec{p}$ for the projector illumination and the emission term L_e as $\vec{e}$. This is essential, since the light fields do not model incident irradiance at the sensor or exitant radiance at the projector. This is assumed to be static, hence, it is included in the acquired light transport matrix.

This is illustrated in figure 3.6 (a-c). In each of these images a single projector pixel is projected onto a scene and captured by a camera. The photograph is inserted in the appropriate column of the matrix until this is completely filled. Both, the projector pattern and the photograph can be represented as a single column vector by storing the individual pixel rows successively.

Projecting individual pixels on a scene may result in very low intensities in some parts and much higher values in others. Hence, the ratio between brighter and dimmer matrix entries can be, depending on the scene, significant. In order to measure all global illumination effects accurately, high dynamic range imaging techniques as described in section 3.3 are applied to acquire the camera images.

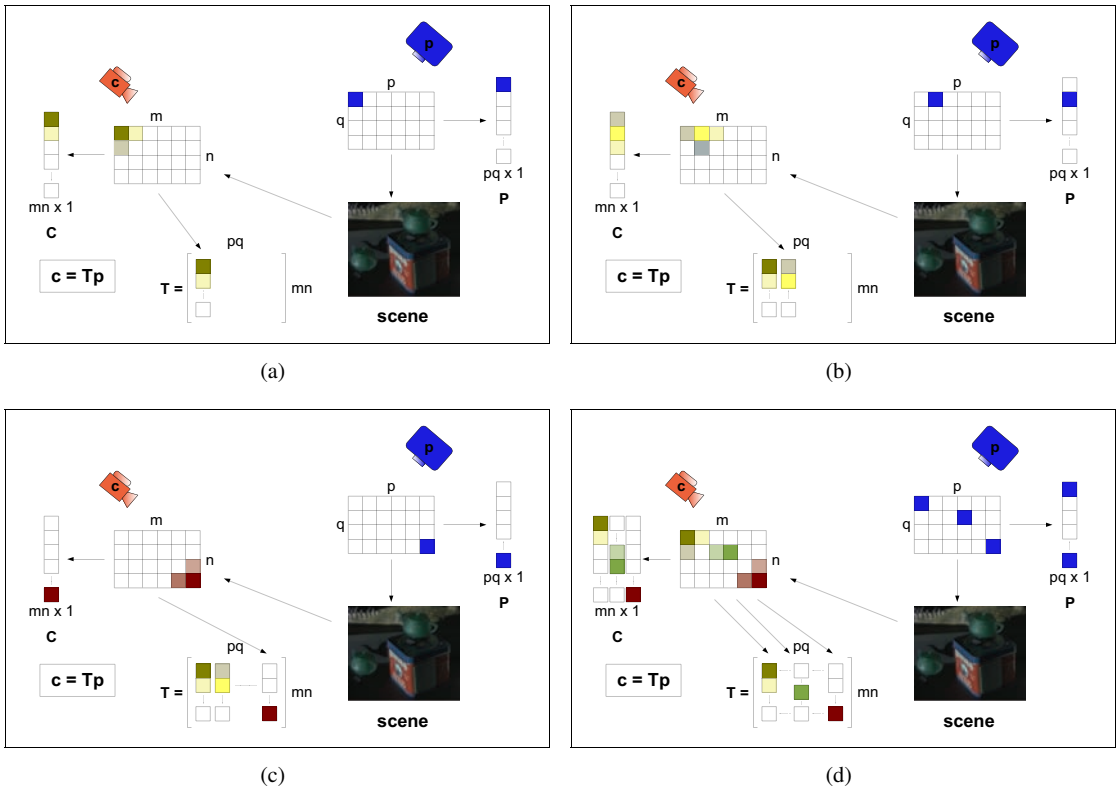

Figure 3.6: The light transport matrix can be acquired by illuminating each projector pixel individually capturing each illumination with a camera and inserting the photograph at the appropriate place in T (a)-(c). Conflict-free projector pixels can be acquired simultaneously when a unique reconstruction of individual contributions is possible (d).

However, illuminating pixel by pixel is very slow and requires to capture *pq* different high dynamic range images. The assumption that there are regions in the camera image that are affected by different areas of the projection can accelerate the acquisition process significantly. These regions have to be given along with a mapping that allows identifying a region in both, projector and camera space. This allows to simultaneously project and capture pixels of non-influencing areas and recover unambiguous contributions of the individual pixels from the photographs as indicated in figure 3.6(d). More parallely projected pixels

result in less images that have to be captured. The main question is how to receive the information about influencing areas. Several approaches to this problem exist.

Masselus et al. [43] proposed a method that subdivides the projector space into multiple uniform blocks and measures the affected areas in camera space for each block individually. Projector pixels of one block are assumed to possibly affect the same camera pixels, while projector pixels of different blocks are not. Thus, each pixel of one block must be illuminated sequentially. Due to the assumption that blocks do not interfere in camera space, the corresponding pixels in every block can be projected simultaneously. This approach accelerates the light transport acquisition at the expense of completely discarding global illumination effects.

Sen et al. [63] adapted Masselus' method to also take block interferences into account. This approach is based on the following idea:

> "...our adaptive algorithm tries to acquire the transport matrix with as few patterns as possible while ensuring that projector pixels affecting the same camera pixel are never illuminated simultaneously."

A hierarchical, scene adaptive scheme is employed, that starts by capturing the fully illuminated scene (floodlight image). Possible contributions from environment light are neutralized by subtracting a photograph of the scene containing the projector's blacklevel and incident environmental light. In this first hierarchy level, the image is processed by determining for every camera camera pixel k if it is affected by the projection or not. The decision is based on a threshold τ. In the second level only camera pixels that were affected in the first level have to be further processed. The projector image is subdivided into four equally sized sub-blocks. Each of the blocks receives a unique ID and is illuminated and captured one after the other. Just as in the first level, every camera pixel is processed in each of the captured images and a possible influence of that specific projector block is determined using τ. Now each camera pixel k has a list $B_k = \{B_0, ..., B_n\}$ with projector block IDs that affected it in level 2. Multiple blocks in one block list are conflicting, due to their contribution to one camera pixel.

Combining all possible conflicts of all camera pixels in one level of the hierarchy allows to compute lists of conflict free patterns for the next level. If two projector blocks are not conflicting in any of the B_ks, these are conflict-free and could be projected parallely. Again, the next hierarchy level requires to further subdivide each of the blocks into four sub-blocks. These subdivisions are assumed to be conflicting for one block and have to be projected sequentially. However, sub-blocks of conflict-free projector blocks can be projected at the same time and allow a unique reconstruction from the camera images. This procedure is repeated recursively until the projector blocks have the size of a single pixel.

Each of the hierarchy levels can be represented by a light transport matrix. The number of blocks in the

particular level determines the number of matrix columns. Level 0 has only one block (floodlight), thus the matrix is a column vector of size $mn\ x\ 1$, level two has 4 blocks resulting in a matrix of size $mn\ x\ 4$. This is continued to level n, where T is of size $mn\ x\ 4^n$. The energy for each element is only stored at the last level, in which the captured radiance was still measurable. This ensures that energy is not stored multiple times. It also means that T_0 actually does not contain the floodlit camera image, but only the energy of the floodlight image that would not be measurable in the next iteration.

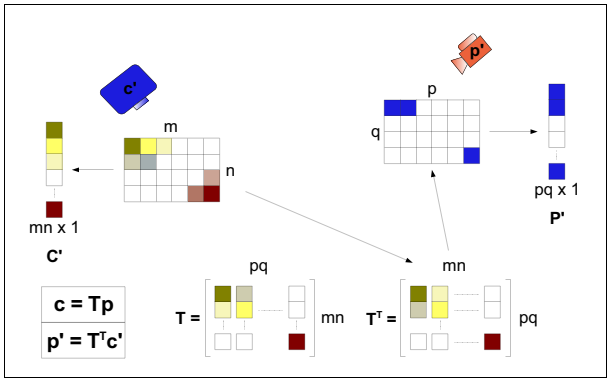

Figure 3.7: Transposing the light transport matrix allows to interchange camera and projector.

3.4.3 Application to Dual Photography

Dual photography is a photographic technique proposed by Sen et al. [63] that uses Helmholtz reciprocity to interchange a camera and a projector. The idea is, that the flow of light on a surface can be reversed without changing the transport properties. For a given BRDF f_r, the transfer between incoming ω_i and outgoing radiance ω_o can be interchanged: $f_r(\omega_i \rightarrow \omega_o) = f_r(\omega_o \rightarrow \omega_i)$. Mathematically, this can be formulated by transposing the light transport matrix T and multiplying with a desired illumination pattern

$$\begin{array}{ccc} \vec{p}'_\lambda & = & T_\lambda^T \quad \vec{c}'_\lambda \end{array}$$
$$\begin{bmatrix} p'_{\lambda 0} \\ p'_{\lambda 1} \\ \vdots \\ p'_{\lambda (pq-1)} \end{bmatrix} = \begin{bmatrix} t^0_{\lambda 0} & \cdots & t^0_{\lambda (mn-1)} \\ t^1_{\lambda 0} & & t^1_{\lambda (mn-1)} \\ \vdots & \ddots & \vdots \\ t^{(pq-1)}_{\lambda 0} & \cdots & t^{(pq-1)}_{\lambda (mn-1)} \end{bmatrix} \begin{bmatrix} c'_{\lambda 0} \\ c'_{\lambda 1} \\ \vdots \\ c'_{\lambda (mn-1)} \end{bmatrix} \tag{3.12}$$

where $\vec{p}'$ represents an image in dual space (indicated by the superscript $'$) from the projector's point of view with a resolution of pq and $\vec{c}'$ the illumination pattern of the camera with a resolution of mn in dual space. Any illumination pattern can be applied, a white image results in a picture, fully illuminated from the camera. Computing a dual image requires a captured light transport matrix. Note that T^T is not the mathematical inverse of T, it is only transposed. The environment light is discarded. The technique is illustrated in figure 3.7.

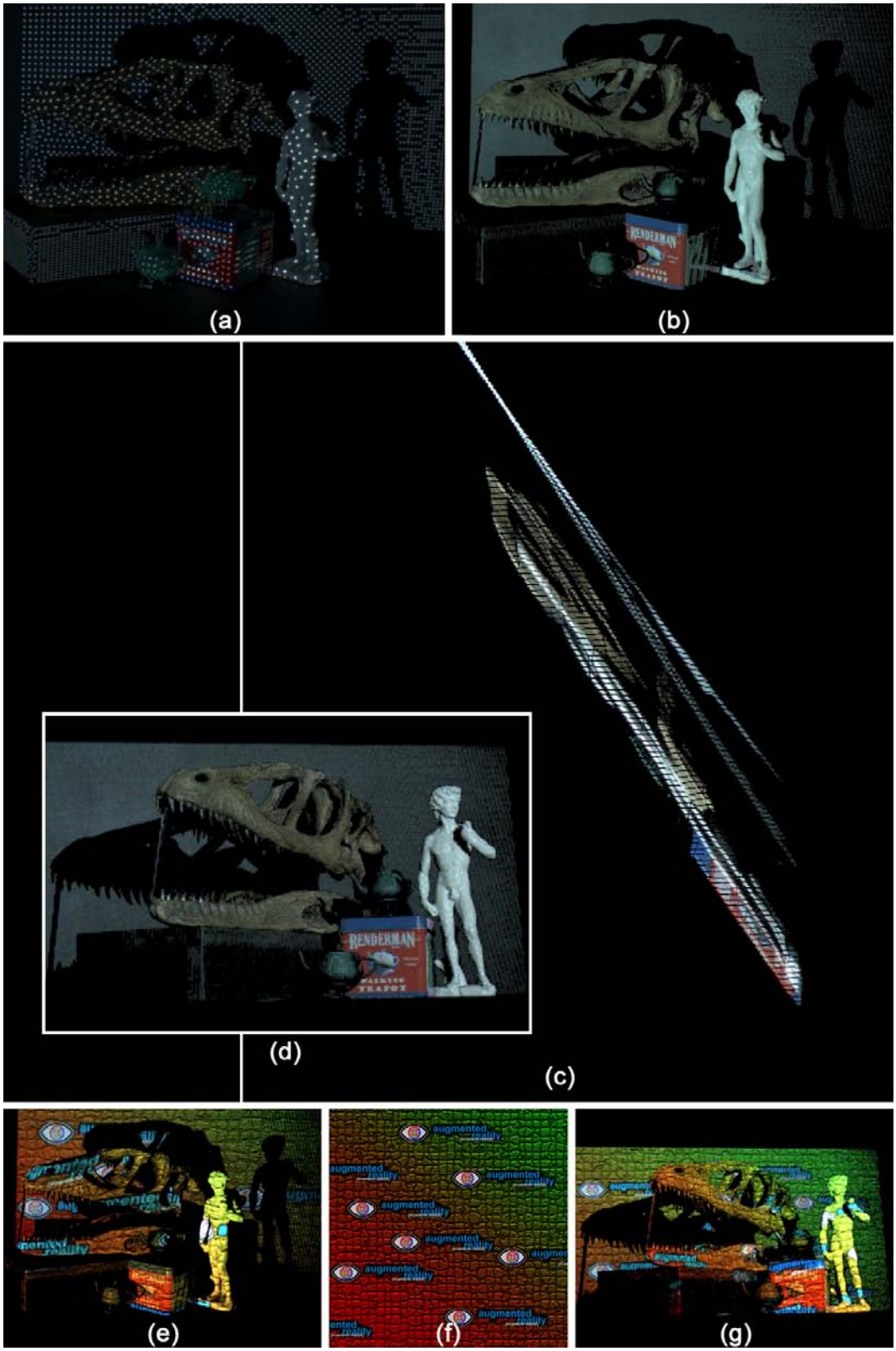

Figure 3.8: An sample light transport matrix (c) with a simulation of camera (b) and projector (d) space as well as novel illuminations (e+g). A sample illumination pattern, captured by the camera is shown in (a).

An acquisition of the light transport between a projector and a camera is demonstrated in figure 3.8 showing: one of the projected patterns as seen by the camera (a), a synthesized camera composition image using T (b), the light transport matrix itself (c) along with the dual image (d). Composition images under arbitrary projector illumination (f) can be simulated (e) by applying equation 3.11. The same can be done for dual images under a virtual camera illumination (f) using equation 3.12.

4 A Generalized Approach to Radiometric Compensation

The last chapter contained a detailed discussion on light transport acquisition, including one of the most fundamental equations: $\vec{c}_\lambda = T_\lambda \vec{p}_\lambda + \vec{e}_\lambda$ (see eq. 3.11). This also represents the foundation of the proposed radiometric correction approach which is modeled as a linear system. The camera image $\vec{c}$ is replaced by a desired image and T is assumed to be well-known. A compensation image can be computed by solving the equation system with respect to $\vec{p}$. Projecting this onto the scene results in a corrected view from the camera's perspective.

However, equation 3.11 is only a simplification of physical light transport. It does not take the spectral difference between projector colors and camera channels into account.

A mathematical formulation for generalized radiometric compensation with an arbitrary number of cameras and projectors is introduced in the next section. For practical application and further processing several simplification and matrix decomposition methods are proposed in the following sections.

4.1 Radiometric Compensation as a Linear System

The generalized equation for radiometric compensation is successively introduced by starting with a simple setup containing only a single camera and projector. This is further extended to allow an arbitrary amount of displays and imaging devices.

A very important assumption is that all cameras and projectors capture and synthesize light with three color channels. While it is possible to reduce this model to greyscale devices, it does not consider more channels. LCD projectors separate different colors using a spinning wheel of transparent colored films within the light path. In order to increase the maximal brightness of the display some projectors have four color channels: red, green, blue and an additional white channel.

4.1.1 Single Camera-Projector Systems

As mentioned before, equation 3.11 is based on the assumption that differences in the spectral sensivities of camera and projector are negligible. In general this is not the case, because when displaying red light with the projector, the camera's green and blue channel are affected as well. This has to be taken into account when deriving a generalized mathematical foundation. Therefore, one light transport matrix is required for every color channel of the projector (T^R, T^G and T^B). The general equation for radiometric compensation can be derived from the following set of equations:

$$\begin{aligned}
\vec{c_R} &= T_R^R \vec{p_R} + T_R^G \vec{p_G} + T_R^B \vec{p_B} + \vec{e_R}, \\
\vec{c_G} &= T_G^R \vec{p_R} + T_G^G \vec{p_G} + T_G^B \vec{p_B} + \vec{e_G}, \\
\vec{c_B} &= T_B^R \vec{p_R} + T_B^G \vec{p_G} + T_B^B \vec{p_B} + \vec{e_B}.
\end{aligned} \tag{4.1}$$

All $\vec{c_\lambda}$s are vectors with a size of $mn\,x\,1$ forming a single color channel λ of the entire camera image. Each of the light transport matrices T^λ (size $mn\,x\,pq$) also has three channels representing the contribution of a single projector color to all camera channels. Projector channels are denoted as $\vec{p_\lambda}$, each of size $pq\,x\,1$. The matrix of a specific projector color is indicated by the superscripts, T^R for example represents the light transport matrix that was acquired by only projecting red light. The subscripts represent the color channel of an element, $\vec{c_G}$ is the green color channel of the camera image, $\vec{e_B}$ the blue channel of the environmental light contribution, T_R^G is the red color channel of the light transport matrix acquired for the green projector channel.

Even though the coefficients in equation 4.1 are matrices, these form a set of linear equations:

$$\begin{bmatrix} \vec{c_R} - \vec{e_R} \\ \vec{c_G} - \vec{e_G} \\ \vec{c_B} - \vec{e_B} \end{bmatrix} = \begin{bmatrix} T_R^R & T_R^G & T_R^B \\ T_G^R & T_G^G & T_G^B \\ T_B^R & T_B^G & T_B^B \end{bmatrix} \begin{bmatrix} \vec{p_R} \\ \vec{p_G} \\ \vec{p_B} \end{bmatrix}, \tag{4.2}$$

which represents the general radiometric compensation equation for a single-camera, single-projector system. It is described by an equation system with a size of $3mn\,x\,3pq$.

When describing a direct relation between a single projector and camera pixel, each T_i^j would be a scalar. The coefficient matrix has in this case a size of $3x3$. This is exactly the compensation model used by Nayar et al. [46], where the coefficient matrix is referred to as the color mixing matrix V. However, equation 4.2 is its generalized form.

4.1.2 An Extension for Multiple Displays

So far, the radiometric compensation model has been restricted to a single camera and one projector per setup. However, it can easily be extended to support multiple projectors and/or multiple cameras. Creating correct images for multiple perspectives does actually not make much sense when projecting onto ordinary material. Lambertian surfaces for instance reflect the same radiance in every direction, thus, it is not possible to display one color for one perspective and a second color for another one. Applications for special projection materials are considered in the future work discussion (section 6.4).

Using multiple projectors in one setup does make much sense. Overall intensity is increased and shadows cast by one projector can be compensated with the other one. The simplified forward light transport in a multi-projector, single camera system is defined by

$$\vec{c}_\lambda \;=\; {}^0T^0_\lambda \vec{p}_\lambda \;+\; {}^1T^1_\lambda \vec{p}_\lambda \;+\; ... \;+\; {}^{(k-1)}T^{(k-1)}_\lambda \vec{p}_\lambda \;+\; \vec{e}_\lambda \;=\; \sum_{i=0}^{k-1} \left({}^iT^i_\lambda \vec{p}_\lambda\right) \;+\; \vec{e}_\lambda, \tag{4.3}$$

where $\vec{c}_\lambda$ is the camera image (each color channel λ is processed separately), ${}^i\vec{p}_\lambda$ the projector contribution of projector i, ${}^iT_\lambda$ the light transport matrix between projector i and the camera. $\vec{e}_\lambda$ is the environment light as well as all projector's black level contribution.

Such a setup describes a 6D slice of the full 8D reflectance field. Using multiple projectors allows to capture the 4D incident light field, instead of a 2-dimensional one for a single projector. The camera, however, constraints the outgoing light field to 2 dimensions.

In order to perform a radiometric compensation, all light transport matrices are concatenated to a single matrix with a size of $mn\ x\ kpq$, assuming a similar resolution of pq for each projector. If projector resolutions are varying the size of the matrix changes to $mn\ x\ \sum_{i=0}^{k-1} p_i q_i$. In a similar way, all projector contributions ${}^i\vec{p}_\lambda$ are inserted in a single column vector. Using equation 4.3 the compensation can be reformulated as follows:

$$\vec{c}_\lambda - \vec{e}_\lambda \quad = \quad \begin{bmatrix} {}^0T_\lambda & {}^1T_\lambda & \dots & {}^{(k-1)}T_\lambda \end{bmatrix} \quad \begin{bmatrix} {}^0\vec{p}_\lambda \\ {}^1\vec{p}_\lambda \\ \vdots \\ {}^{(k-1)}\vec{p}_\lambda \end{bmatrix},$$

$$\begin{bmatrix} c_{\lambda 0} - e_{\lambda 0} \\ c_{\lambda 1} - e_{\lambda 1} \\ \vdots \\ c_{\lambda (mn-1)} - e_{\lambda (mn-1)} \end{bmatrix} = \begin{bmatrix} {}^0t^0_{\lambda 0} & \dots & {}^0t^{(pq-1)}_{\lambda 0} & {}^1t^0_{\lambda 0} & \dots & {}^{(k-1)}t^{(pq-1)}_{\lambda 0} \\ {}^0t^0_{\lambda 1} & & {}^0t^{(pq-1)}_{\lambda 1} & {}^1t^0_{\lambda 1} & & {}^{(k-1)}t^{(pq-1)}_{\lambda 1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ {}^0t^0_{\lambda (mn-1)} & \dots & {}^0t^{(pq-1)}_{\lambda (mn-1)} & {}^1t^0_{\lambda (mn-1)} & \dots & {}^{(k-1)}t^{(pq-1)}_{\lambda (mn-1)} \end{bmatrix} \begin{bmatrix} {}^0p_{\lambda 0} \\ \vdots \\ {}^0p_{\lambda (pq-1)} \\ {}^1p_{\lambda 0} \\ \vdots \\ {}^{(k-1)}p_{\lambda (pq-1)} \end{bmatrix}. \tag{4.4}$$

The superscripts on the left of the ${}^it_\lambda$s and ${}^ip_\lambda$s indicate a specific projector, while the right superscripts and subscripts of the matrix elements are the indices within the matrix. All projector images are packed into the one-column right hand vector and the light transport matrices ${}^iT_\lambda$ are aligned next to each other in one huge matrix.

Again, equations 4.3 and 4.4 assume that color channels are independent. A generalization requires three light transport matrices per projector. Each of these matrices has three color channels representing the contribution of a single projector color to all camera channels:

$$\begin{aligned}
\vec{c}_R &= \textstyle\sum_{i=0}^{k-1} \left({}^{i}T_R^R \, {}^{i}\vec{p}_R + {}^{i}T_R^G \, {}^{i}\vec{p}_G + {}^{i}T_R^B \, {}^{i}\vec{p}_B \right) + \vec{e}_R, \\
\vec{c}_G &= \textstyle\sum_{i=0}^{k-1} \left({}^{i}T_G^R \, {}^{i}\vec{p}_R + {}^{i}T_G^G \, {}^{i}\vec{p}_G + {}^{i}T_G^B \, {}^{i}\vec{p}_B \right) + \vec{e}_G, \\
\vec{c}_B &= \textstyle\sum_{i=0}^{k-1} \left({}^{i}T_B^R \, {}^{i}\vec{p}_R + {}^{i}T_B^G \, {}^{i}\vec{p}_G + {}^{i}T_B^B \, {}^{i}\vec{p}_B \right) + \vec{e}_B,
\end{aligned} \tag{4.5}$$

yielding

$$\begin{bmatrix} \vec{c}_R - \vec{e}_R \\ \vec{c}_G - \vec{e}_G \\ \vec{c}_B - \vec{e}_B \end{bmatrix} = \begin{bmatrix} {}^{0}T_R^R & {}^{0}T_R^G & {}^{0}T_R^B & {}^{1}T_R^R & \cdots & {}^{(k-1)}T_R^B \\ {}^{0}T_G^R & {}^{0}T_G^G & {}^{0}T_G^B & {}^{1}T_G^R & \ddots & {}^{(k-1)}T_G^B \\ {}^{0}T_B^R & {}^{0}T_B^G & {}^{0}T_B^B & {}^{1}T_B^R & \cdots & {}^{(k-1)}T_B^B \end{bmatrix} \begin{bmatrix} {}^{0}\vec{p}_R \\ {}^{0}\vec{p}_G \\ {}^{0}\vec{p}_B \\ {}^{1}\vec{p}_R \\ \vdots \\ {}^{(k-1)}\vec{p}_B \end{bmatrix}. \tag{4.6}$$

The left hand vector, representing the camera image, has a size of $3mn \; x \; 1$, the coefficient matrix is of size $3mn \; x \; 3kpq$ and the right hand vector has a size of $3kpq \; x \; 1$.

4.1.3 Compensation with General Setups

A somewhat more theoretical construct is created if equation 4.4 is generalized to the full 8D reflectance field. First of all, the simplified case is discussed, which is then extended to account for spectral differences. Based on equation 4.3, a contribution of every projector to each camera is defined by summing the appropriate matrices. The amount of cameras r and projectors k is arbitrary, requiring rk light transport matrices. Indexing the light transport matrices is done by denoting two left-hand values, the upper one representing the projector, the lower one the camera. Thus, matrix ${}^{1}_{2}T_\lambda$ describes the light transport from projector 1 to camera 2. The simplified radiometric compensation equation for k projectors and r cameras is given by

$$\begin{bmatrix} {}_{0}\vec{c}_\lambda - {}_{0}\vec{e}_\lambda \\ {}_{1}\vec{c}_\lambda - {}_{1}\vec{e}_\lambda \\ \vdots \\ {}_{(r-1)}\vec{c}_\lambda - {}_{(r-1)}\vec{e}_\lambda \end{bmatrix} = \begin{bmatrix} {}^{0}_{0}T_\lambda & {}^{1}_{0}T_\lambda & \cdots & {}^{(k-1)}_{0}T_\lambda \\ {}^{0}_{1}T_\lambda & {}^{1}_{1}T_\lambda & \cdots & {}^{(k-1)}_{1}T_\lambda \\ \vdots & \vdots & \ddots & \vdots \\ {}^{0}_{(r-1)}T_\lambda & {}^{1}_{(r-1)}T_\lambda & \cdots & {}^{(k-1)}_{(r-1)}T_\lambda \end{bmatrix} \begin{bmatrix} {}^{0}\vec{p}_\lambda \\ {}^{1}\vec{p}_\lambda \\ \vdots \\ {}^{(k-1)}\vec{p}_\lambda \end{bmatrix}. \tag{4.7}$$

Its generalization forms an equation system of size $3rmn \; x \; 3kpq$:

$$\begin{bmatrix} {}_{0}\vec{c}_R - {}_{0}\vec{e}_R \\ {}_{0}\vec{c}_G - {}_{0}\vec{e}_G \\ {}_{0}\vec{c}_B - {}_{0}\vec{e}_B \\ {}_{1}\vec{c}_R - {}_{1}\vec{e}_R \\ \vdots \\ {}_{(r-1)}\vec{c}_B - {}_{(r-1)}\vec{e}_B \end{bmatrix} = \begin{bmatrix} {}^{0}_{0}T_R^R & {}^{0}_{0}T_R^G & {}^{0}_{0}T_R^B & {}^{1}_{0}T_R^R & \cdots & {}^{(k-1)}_{0}T_R^B \\ {}^{0}_{0}T_G^R & {}^{0}_{0}T_G^G & {}^{0}_{0}T_G^B & {}^{1}_{0}T_G^R & \cdots & {}^{(k-1)}_{0}T_G^B \\ {}^{0}_{0}T_B^R & {}^{0}_{0}T_B^G & {}^{0}_{0}T_B^B & {}^{1}_{0}T_B^R & \cdots & {}^{(k-1)}_{0}T_B^B \\ {}^{0}_{1}T_R^R & {}^{0}_{1}T_R^G & {}^{0}_{1}T_R^B & {}^{1}_{1}T_R^R & \cdots & {}^{(k-1)}_{1}T_R^B \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ {}^{0}_{(r-1)}T_B^R & {}^{0}_{(r-1)}T_B^G & {}^{0}_{(r-1)}T_B^B & {}^{1}_{(r-1)}T_B^R & \cdots & {}^{(k-1)}_{(r-1)}T_B^B \end{bmatrix} \begin{bmatrix} {}^{0}\vec{p}_R \\ {}^{0}\vec{p}_G \\ {}^{0}\vec{p}_B \\ {}^{1}\vec{p}_R \\ \vdots \\ {}^{(k-1)}p_B \end{bmatrix}. \tag{4.8}$$

Applying equation 4.8 enables to compensate for a desired image with a projector-camera systems. An arbitrary amount of cameras and projectors can be used to perform the compensation. Employing multiple projectors generally increases the projection's quality. All possible global illumination effects are considered with this compensation approach.

4.2 Light Transport Decomposition and Clustering

Solving an equation system for an acquired light transport matrix with equation 4.2 can be quite impractical. Assuming a camera resolution of $640\,x\,480$ and a projector resolution of $512\,x\,512$ would result in an equation system of size $921,600\,x\,768,432$. Allocating enough memory for the coefficient matrix containing single precision floating point values would require about 2638 GB.

In order to reduce required resources, equation 3.11 ($\vec{c}_\lambda = T_\lambda \vec{p}_\lambda + \vec{e}_\lambda$) is employed, where $\vec{c}_\lambda$ is replaced by a desired image that the user wishes to perceive (from the camera's point of view), $\vec{p}_\lambda$ is the pattern that has to be projected onto the scene and $\vec{e}_\lambda$ the environmental light contribution including the projector's blacklevel.

While this method is straightforward, solving huge linear equation systems requires an enormous amount of processing time and memory, hence it may still be impossible to be performed with a personal computer. An approach towards reduced redundancy and increased performance is to decompose the light transport matrix into clusters of mutually influencing camera and projector pixels. Each of these clusters represent a single linear equation system that can be processed separately.

The size of a cluster allows reasoning about the amount of implicitly given global illumination effects (reflection, refraction, scattering, diffraction etc.). Besides, effects induced by the camera and projector optics such as defocus as well as light sensor specific effects as blooming also affect connections within the matrix. A flat diffuse surface with an overall focused projector and camera is likely to produce very small and localized systems while capturing a scene with high depth variance and lots of reflecting concave objects will behave oppositional.

The acquired light transport matrix of a sample scene is visualized in figure 4.1. A contrast enhanced magnified part of this matrix indicates global illumination effects. These occur, whenever a single projector pixel affects multiple camera pixels or an individual camera pixel is affected by various projector pixels.

As discussed in section 3.4.2, a threshold τ is used to decide whether a particular camera pixel is affected by a projected block or not. The threshold mainly determines the amount of global illumination

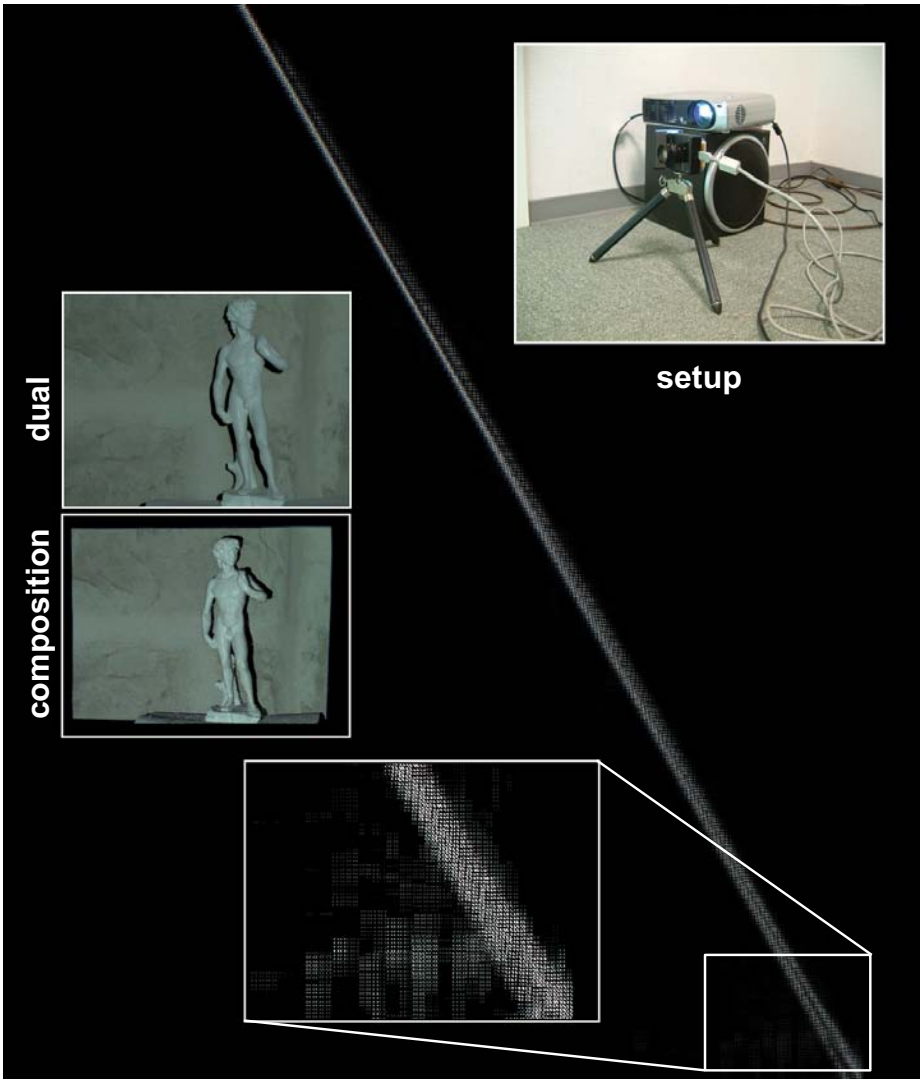

Figure 4.1: Global illumination effects within a light transport matrix.

effects captured by the light transport matrix. A lower threshold allows to capture detailed information about low-intensity illumination, but also increases the amount of high dynamic range images that have to be captured. Thus, a tradeoff has to be made between quality of the scan and capture/processing time. Knowledge about the projector blocks that affect each camera pixels can be used to determine connected cluster within the matrix already during its acquisition.

4.2.1 Deriving Clusters during Light Transport Acquisition

Since the light transport matrix acquisition is performed in a hierarchical way, implicit connections are computed for one level of the hierarchy and refined with each following. In level 0 only a single projector block is projected on the scene. Once its effect on every camera pixel is determined using τ, a cluster

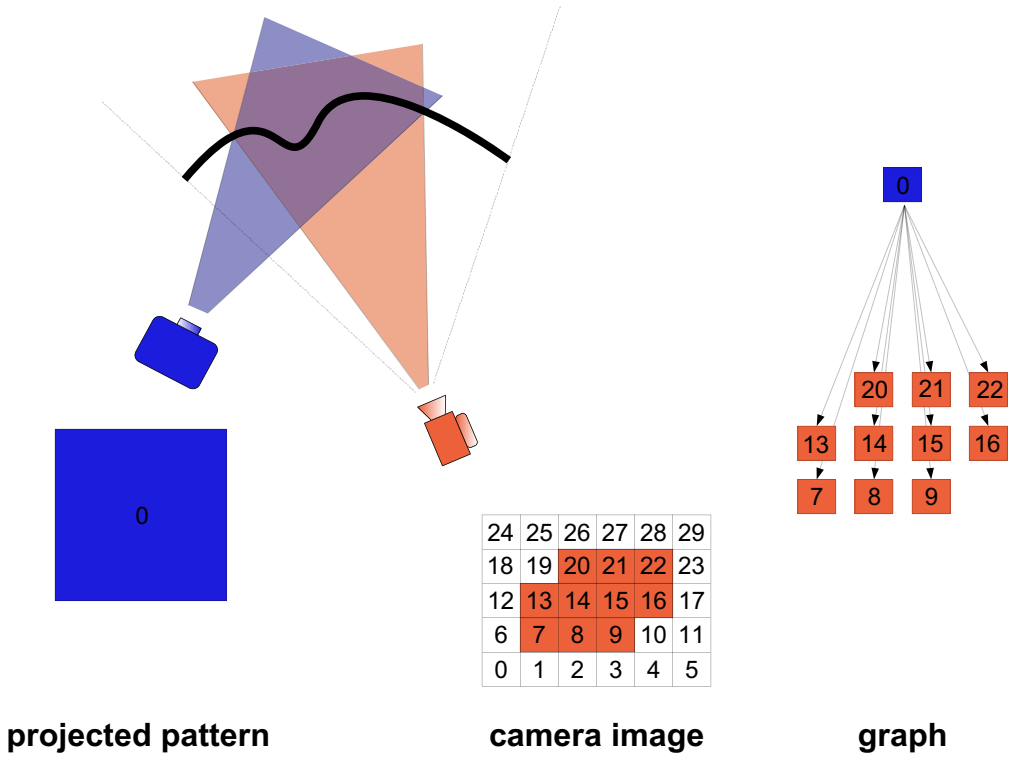

Figure 4.2: Projection of block 0 in the first hierarchy of the light transport acquisition. Depending on a threshold each camera pixel is either affected by the projection or not. The interconnections between projector blocks and camera pixels form a graph (right hand side).

can be computed. Note that there is at most one cluster in level 0, because the camera pixels are either affected by block 0 or not. Level 1 has four sub-blocks of the previous level and, after capturing the appropriate patterns, their effect on the camera pixels can be derived. Again, the maximum number of clusters is given by the number of projector blocks in this hierarchy. A synthetic example follows that is used to illustrate the cluster acquisition for successive hierarchy levels.

Figure 4.2 shows block 0 (floodlight image) projected on a scene in the first hierarchy. The captured camera image is processed and block dependency is assigned to each pixel that has a higher luminance that the threshold (in this case blocks 7,8,9,13,14,15,16,20,21,22). All pixels that are not affected at all can be discarded from the equation system for hierarchy 0. Note that it would not make much sense to perform a radiometric compensation with the equation system resulting from level 0, since this is heavily underdetermined. In this example, the number of projector blocks is far less than the amount of affected camera pixels.

However, the size of the equation system can effectively be reduced from $30x1$ to $10x1$:

$$\begin{array}{ccccc} \vec{c}_\lambda - \vec{e}_\lambda & = & T_\lambda & \vec{p}_\lambda & \\ \begin{bmatrix} c_{\lambda 0} - e_{\lambda 0} \\ c_{\lambda 1} - e_{\lambda 1} \\ \vdots \\ c_{\lambda 29} - e_{\lambda 29} \end{bmatrix} & = & \begin{bmatrix} t^0_{\lambda 0} \\ t^0_{\lambda 1} \\ \vdots \\ t^0_{\lambda 29} \end{bmatrix} & [p_{\lambda 0}] & \rightarrow \begin{bmatrix} c_{\lambda 7} - e_{\lambda 7} \\ \vdots \\ c_{\lambda 22} - e_{\lambda 22} \end{bmatrix} = \begin{bmatrix} t^0_{\lambda 7} \\ \vdots \\ t^0_{\lambda 22} \end{bmatrix} [p_{\lambda 0}], \end{array} \tag{4.9}$$

where the superscripts of T's entries indicate the column and the subscripts the row within the light transport matrix.

The same scheme can be utilized in the next hierarchy to compute block dependencies of the camera pixels as shown in figure 4.3. Hierarchy 2 contains four projector blocks (index 1-4) that affect different regions in the camera image. Representing the clusters as graphs (right part of figure 4.3 allows to distinguish between individually connected parts. The left hand side illustrates the projection of one of the blocks on the scene and the affected portions in the camera image. These portions form a subset of the entire region affected by the previous hierarchy. A pixel-precise mapping from blocks to camera pixels rarely occurs in real-world setups, since the boundaries of projected blocks are likely to overlap in the camera space. This is indicated for blocks 7,8 and 13,14 in figure 4.3.

The connections of the graph in hierarchy 2 can be easily derived. Thus, the equation system (size $30x4$) can efficiently be decomposed into three smaller sets of equations of sizes $5x2$, $3x1$ and $2x1$:

$$\begin{array}{ccccc} \vec{c}_\lambda - \vec{e}_\lambda & = & T_\lambda & \vec{p}_\lambda & \\ \begin{bmatrix} c_{\lambda 0} - e_{\lambda 0} \\ c_{\lambda 1} - e_{\lambda 1} \\ \vdots \\ c_{\lambda 29} - e_{\lambda 29} \end{bmatrix} & = & \begin{bmatrix} t^1_{\lambda 0} \cdots t^4_{\lambda 0} \\ t^1_{\lambda 1} \quad t^4_{\lambda 1} \\ \vdots \ \ddots \ \vdots \\ t^1_{\lambda 29} \cdots t^4_{\lambda 29} \end{bmatrix} & \begin{bmatrix} p_{\lambda 1} \\ \vdots \\ p_{\lambda 4} \end{bmatrix} & \rightarrow \begin{array}{l} \begin{bmatrix} c_{\lambda 9} - e_{\lambda 9} \\ c_{\lambda 15} - e_{\lambda 15} \\ c_{\lambda 16} - e_{\lambda 16} \end{bmatrix} = \begin{bmatrix} t^3_{\lambda 9} \\ t^3_{\lambda 15} \\ t^3_{\lambda 16} \end{bmatrix} [p_{\lambda 3}], \begin{bmatrix} c_{\lambda 21} - e_{\lambda 21} \\ c_{\lambda 22} - e_{\lambda 22} \end{bmatrix} = \begin{bmatrix} t^2_{\lambda 21} \\ t^2_{\lambda 22} \end{bmatrix} [p_{\lambda 2}], \\ \begin{bmatrix} c_{\lambda 7} - e_{\lambda 7} \\ c_{\lambda 8} - e_{\lambda 8} \\ c_{\lambda 13} - e_{\lambda 13} \\ c_{\lambda 14} - e_{\lambda 14} \\ c_{\lambda 20} - e_{\lambda 20} \end{bmatrix} = \begin{bmatrix} t^1_{\lambda 7} \\ t^1_{\lambda 8} \\ t^1_{\lambda 13} \\ t^1_{\lambda 14} \\ t^1_{\lambda 20} \end{bmatrix} \begin{bmatrix} p_{\lambda 1} \\ p_{\lambda 4} \end{bmatrix}. \end{array} \end{array} \tag{4.10}$$

This scheme can recursively be continued down to the last level in the hierarchy. Usually, only the pixel-block lists resulting from the last acquisition hierarchy are used to determine the clusters for the radiometric compensation. This restricts the compensation to account for global illumination effects included in the clusters of the last hierarchy only. The clusters of planar diffuse surfaces are assumed to be rather small and localized while reflections within the scene may lead to clusters whose camera pixels

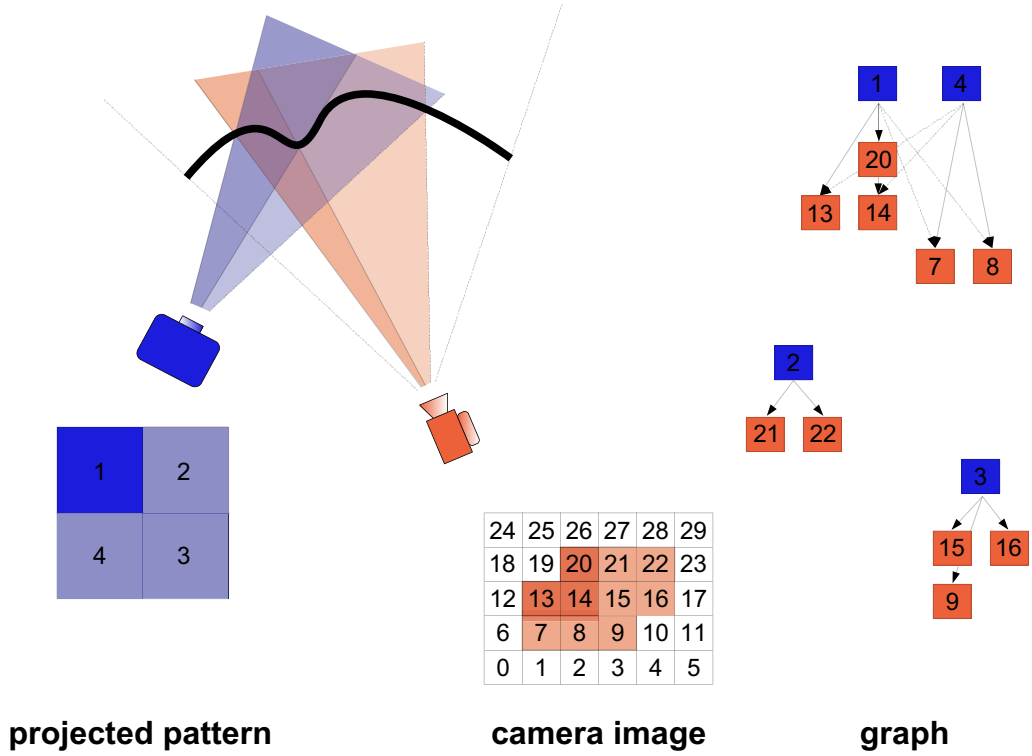

Figure 4.3: Projection of block 1 in the second hierarchy of the light transport acquisition. A list with associated blocks is assigned to each camera pixel after all blocks have been illuminated and clusters can be derived (right).

are distributed all over the camera image. Often, single projector pixels affect a region within the camera image and neighboring projector pixels are likely to overlap also on diffuse surfaces. This leads to many inter-connections and relatively huge clusters. Since available memory and other resources are limited, the size of huge clusters has to be reduced in order to solve the resulting equation system. Approaches for cluster decomposition are discussed in the next subsection.

4.2.2 Cluster Processing and Representation

The acquired pixel-block lists can easily be converted to weighted, bipartite graphs. The graphs have two different types of vertices (camera pixels and projector blocks) with the pixel's irradiance contributed from a specific block being the weight of that edge. Each vertex stores only direct connections to the vertices of the opposite type making the graph bipartite. While different representations for graphs exist (e.g. adjacency matrix), adjacency lists are preferable due to less required memory for sparse graphs.

Figure 4.4 depicts the internal weighted graph representation used for storing the individual clusters from figure 4.3. A cluster's adjacency lists are separated into two subsets including either all of the graph's

pixel-vertices or the block-vertices respectively. This form of storing the structures also allows to perform algorithms on either the camera or projector pixels very efficiently. Iterating over the elements of one of the sets allows for example to draw a graphical representation of a cluster in camera or projector (dual) space.

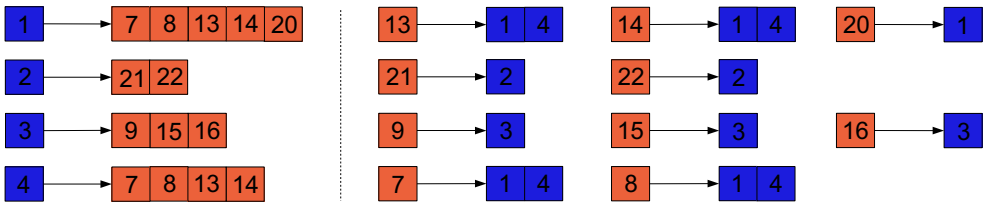

Figure 4.4: Clusters of mutually influencing camera and projector pixels are internally stored as weighted bipartite graphs using the adjacency list representation.

As mentioned before, neighboring projector pixels are likely to overlap in the camera image. This effect may for instance be a result of projector or camera defocus, camera lens imperfections or blooming induced by the camera sensor. While it is possible to adjust the threshold τ during light transport acquisition so that the overlap in one region of the image is neutralized, it is hardly possible to find such a threshold for a complex scene with varying radiances. On the other hand, global illumination effects that fall below the threshold would be filtered out already during the scan. Hence, τ has to be chosen so that all desired indirect illumination contributions are captured.

Conventional graph partitioning and clustering approaches often use a spectral analysis of the graph's Laplacian matrix. The spectrum of a matrix is represented by its eigenvectors, ordered by the magnitude of their corresponding eigenvalues. The Laplacian of the light transport matrix's graph has a size of $(mn+pq)\ x\ (mn+pq)$, thus, it is larger than T itself. This makes it difficult to allocate enough memory for the matrix structure. A common x86 32 bit processor can only address up to 4 GB of memory, storing a single color channel of the full light transport matrix using 32 bit floating point values with a camera resolution of 640x480 and a projector resolution of 512x512 requires 300 GB. These dimensions require alternative approaches to decompose the light transport.

What we usually want, is to split local pixel connections while preserving global illumination effects. Figure 4.5(a) illustrates the local and global effects using a scene that includes diffuse scattering.

In order to separate local from global light effects, the spatial distribution of projector pixels affecting a camera pixel can be taken into account. This is done by grouping all projector pixels that contribute to a single camera pixel into blocks of neighboring pixels. The area of such a block is determined by its radius. A radius of one would only take direct neighbor pixels into account. A subsampling operation can be performed by restricting the amount of projector pixels per block that can contribute to a specific

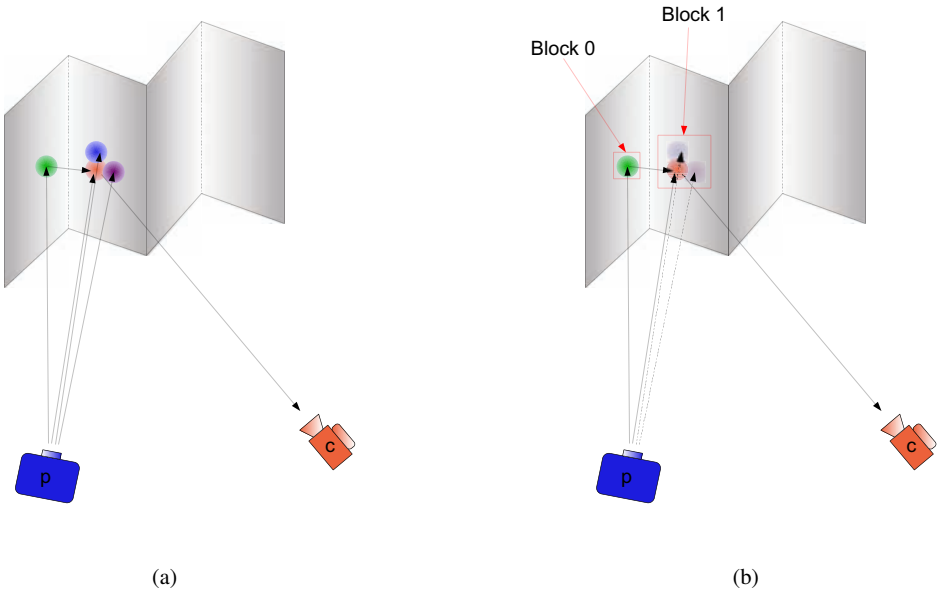

Figure 4.5: A camera pixel may be affected by several projector pixels either due to direct or by indirect illumination (a). These projector pixels can be grouped into blocks of spatially neighboring pixels (b). In order to separate global from local illumination the contribution for a camera pixel can be artificially restricted to a certain amount of projector pixels per block.

camera pixel as shown in figure 4.5(b). It is straightforward to introduce a threshold that filters luminance values in each block with respect to the highest projector pixel contribution.

Decomposing the clusters in this way may also result in discarded projector pixels. This leads to undesired visual artifacts (holes) in the compensated projector image. Thus, it must be ensured that each projector pixel that had connections before the decomposition has at least one connection afterwords. Preferably, these are the connections with the highest luminances. Again, reinserting such connections can be performed using a relative threshold.

An example of such a decomposition is shown in figure 4.6, where the scene's unmodified clusters are visualized on top of the composed floodlight image (c) and on the dual image (d) with the same colors for the same graph in either space. Using the above described decomposition method, the clusters can efficiently be cut so that diffuse surface segments consist only of small, localized clusters while global illumination effects are preserved (see figure 4.6(e) and (f)).

The radius of the neighboring block regions has to be determined empirically based on the acquisition setup. A small radius, e.g. 1, can be sufficient for scenes where individual projector pixels are well visible in the camera image (camera subsamples the projection). However, a higher radius might have to be chosen for setups where the camera supersamples the projected image (multiple projector pixels are

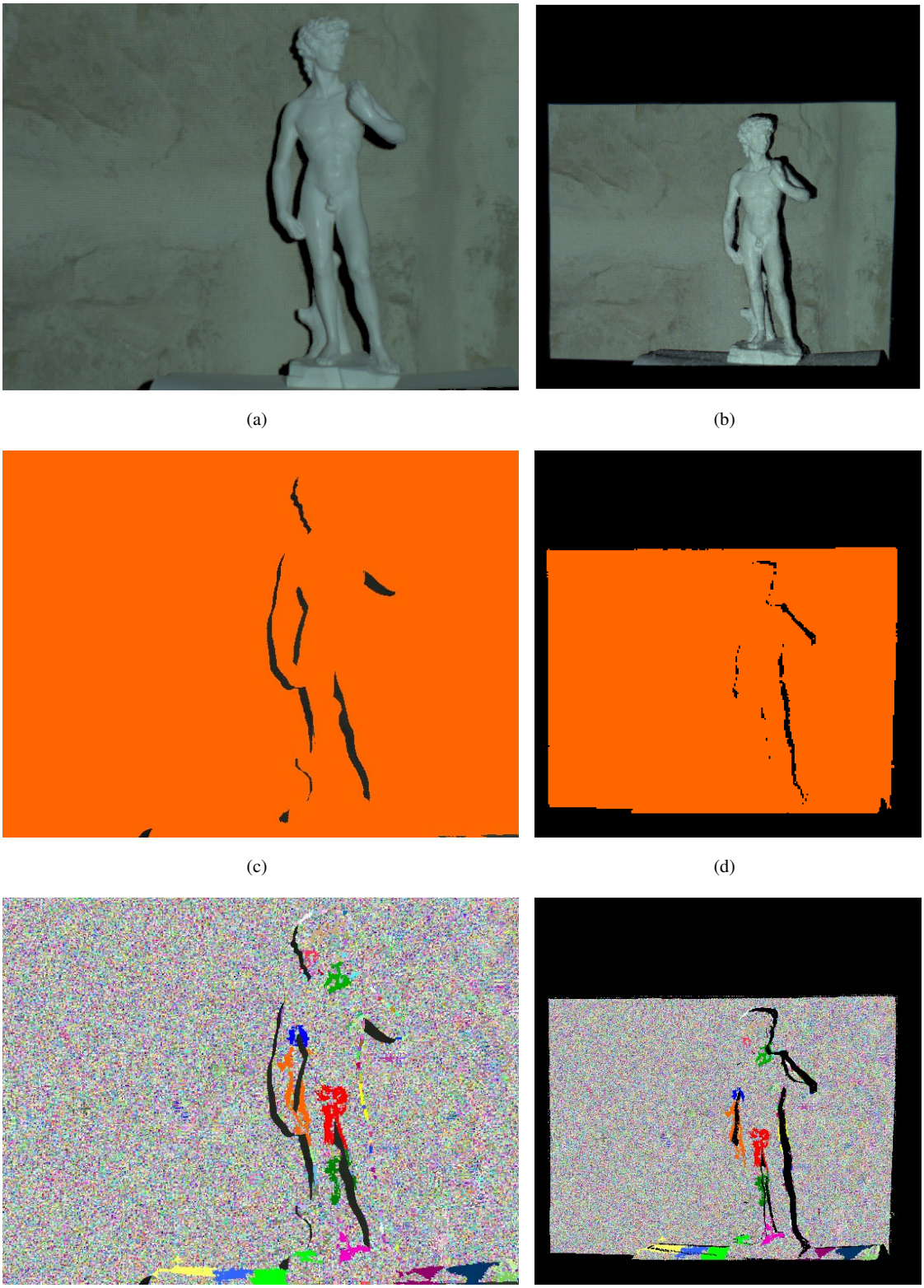

Figure 4.6: Acquiring the light transport allows to synthesize a composition image (a) and its dual (b) as well as clusters of mutually influencing camera and projector pixels (c), (d). Further decomposition is necessary to split local from global illumination effects (e), (f).

captured by a single camera pixel). On the other hand, the user may want to limit the size of the clusters in order to keep the resulting equation systems of a manageable size (see section 4.2.4). Using a higher neighborhood radius is more likely to produce smaller clusters. Both, the radius of the neighborhood and the thresholds determine the size of the decomposed clusters. Appendix A includes pseudo-code for the cluster decomposition algorithm.

4.2.3 Impact on the Light Transport

Care must be taken when removing connections from the graph. Energy cannot simply be discarded, since this would unbalance the light distribution and affect the radiometric compensation by inducing visual artifacts and uncorrect results. Thus, all radiant energy has to be forced to add up to T's original composition. Figure 4.7 illustrates the effect of the cluster decomposition on T. Regular patterns of varying intensities in the composition and dual image are visible. These patterns in the composition image show the desired effect: individual projector pixels in the camera image are implicitly separated from the regions in between, whose intensity is decreased much more.

The decomposed light transport composition represents only a portion of the illumination that was created for separating local and global illumination effects[6]. In order to preserve at least an approximation of the energy that can actually be achieved with the projector the connections within the graph are scaled to fit the original matrix composition as seen in figure 4.7 (e+f). The patterns in the dual image are still visible, indicating the modifications within the light transport. These may influence the stability of the resulting equation systems. However, a direct comparison to a compensation with the unmodified matrix is due to its enormous size hardly possible.

Decomposing the clusters in the described way may be problematic for specific setups. Figure 4.8 visualized clusters of a scene that contains corners with lots of scattering and inter-reflections. A larger neighborhood eventually results in a cut of clusters by discarding spatially close projector pixels. This also leads to a loss of illumination information, especially near the corners. The effect of applying various neighborhood radii is shown in figure 4.8 (b) to (e).

In this way, clusters resulting directly from the light transport acquisition can be decomposed to a size that can be used for further processing. The decomposition is designed to separate global illumination effects from local effects such as overlapping neighboring projector pixels due to defocus and blooming. Energy in the whole light transport system is modulated so that it adds up to the unmodified composition.

[6]Note that this also weakens the projector and camera defocus included in the light transport matrix.

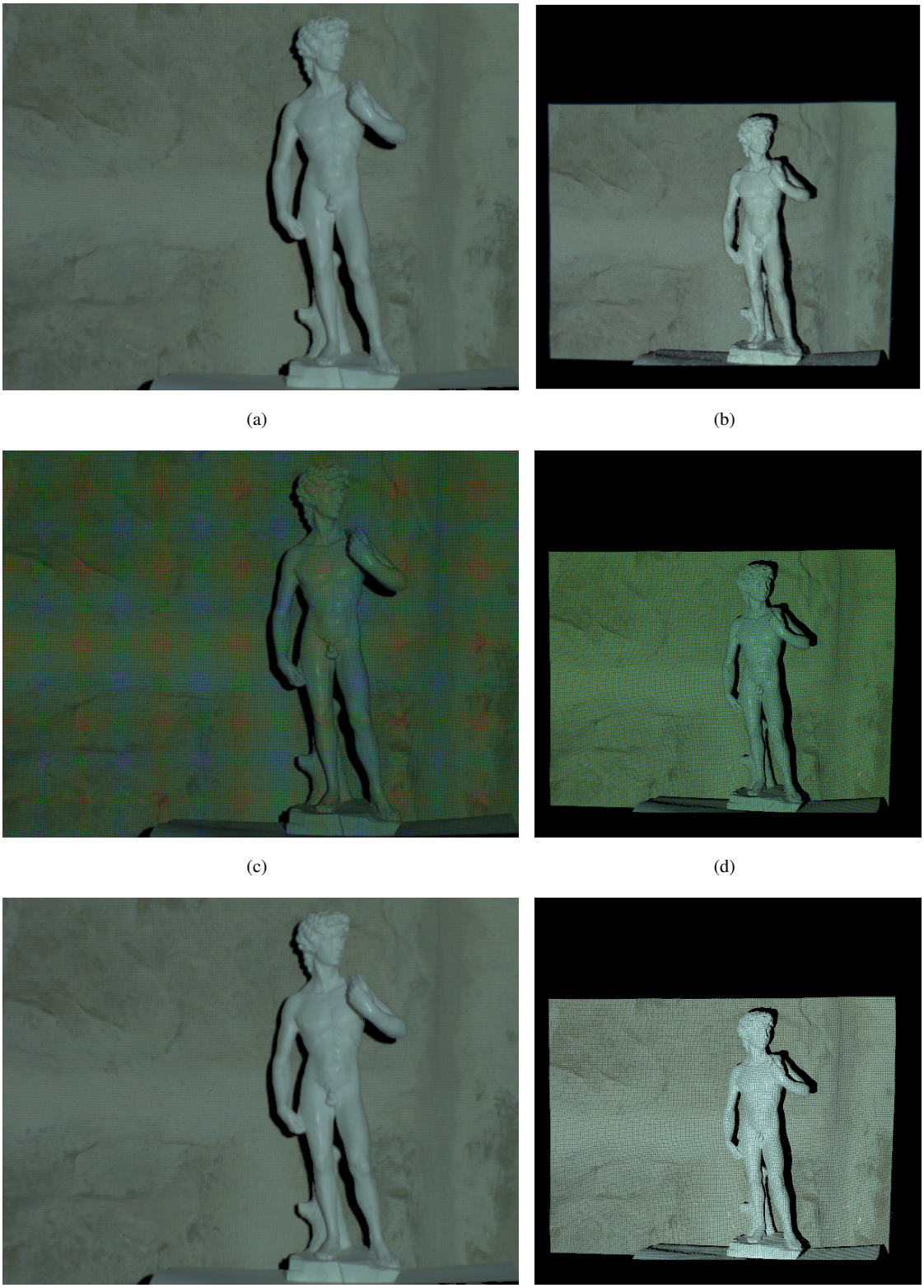

Figure 4.7: The synthesized composition and dual images after scanning (a+b), after decomposing the clusters (c+d) and after fitting the decomposition to the original composition (e+f).

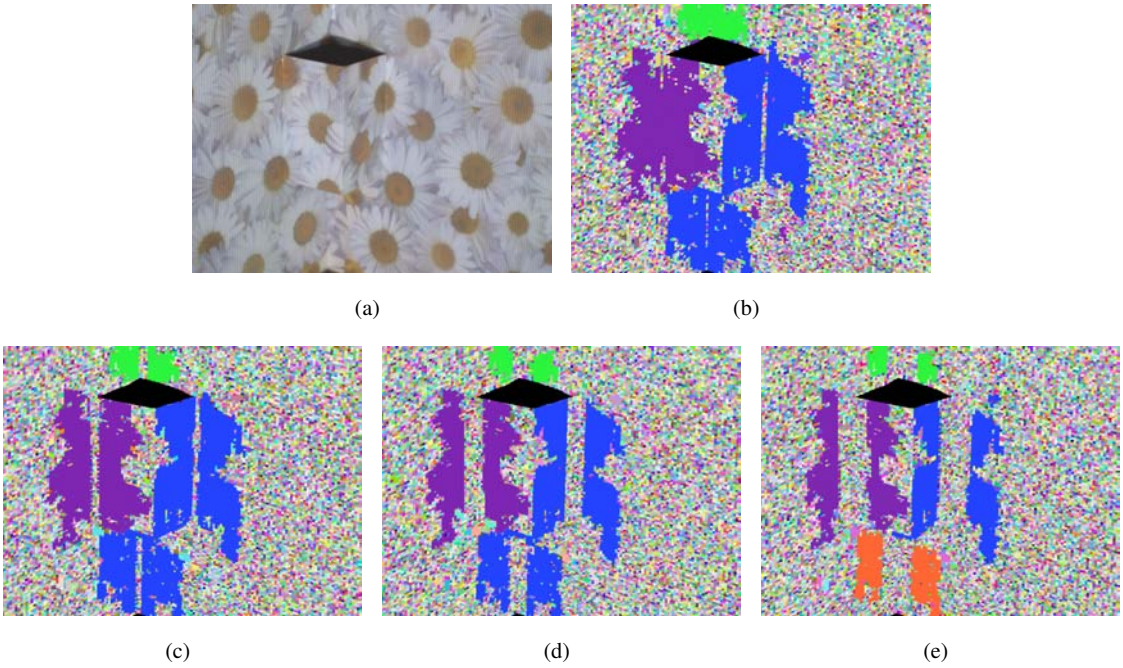

Figure 4.8: Decomposing the light transport of a scene containing lots of inter-reflections (a). The projector pixel neighborhood determines resulting clusters. The decomposed clusters are depicted for a neighborhood of 3 (b), 5 (c), 10 (d) and 15 (e) pixels.

4.2.4 Cluster-Based Radiometric Compensation

Once the clusters are cut to a predefined size[7] the radiometric compensation can be performed separately for each cluster. The compensation is based on equation 3.11 ($\vec{c}_\lambda = T_\lambda \vec{p}_\lambda + \vec{e}_\lambda$) that describes the forward light transport, where $\vec{p}_\lambda$ is the column vector formed by the projector pixels of a specific cluster, $\vec{c}_\lambda$ are the cluster's camera pixels and T_λ is in this case a subset of the acquired light transport matrix that contains the transport only from $\vec{p}_\lambda$ to $\vec{c}_\lambda$.

In order to perform the compensation for arbitrary input images, these have to be scaled to camera resolution. Each of the cluster's camera pixels is replaced by the appropriate sample of the input image. Thus, $\vec{c}_\lambda$ and T_λ are given, $\vec{p}_\lambda$ has to be computed and projected onto the scene. This equation forms a simple linear equation system, that can be solved easily using high-performance libraries such as LAPACK [2]. Note that the scene as well as projector and camera have to be arranged in the same way as during light transport acquisition.

The equation system formed by a cluster can be underdetermined (the cluster has more projector than camera pixels), overdetermined (more camera than projector pixels) or square. In most cases T_λ will

[7] The maximum size of the clusters has to be chosen so that the resulting equation systems are processable using available computational resources.

not be square and regular, thus, there is no exact solution (overdetermined case) or an infinite amount of solutions (underdetermined case) can be computed. A common approach to approximate a solution in the overdetermined case is to compute $\vec{p}_\lambda$ in a least squared error sense. This methods uses the sum of the quadratic differences between computed and desired result as a measurement for quality. In other words $\sum_{i=0}^{(M-1)} (c_{\lambda i} - e_{\lambda i} - (T_\lambda \vec{p}_\lambda)_i)$ has to be minimized, with M being the number of camera pixels and $(T_\lambda \vec{p}_\lambda)$ the computed solution as a matrix-vector multiplication of T_λ and a possible $\vec{p}_\lambda$. Underdetermined sets of linear equations have an unlimited amount of possible solutions. Often the minimum two-norm solution is used for these cases. This ensures that the solution vector $\vec{p}_\lambda$ satisfies the set of equations with the lowest possible values. The two-norm of a vector is defined as the square root of the sum of the squared vector elements, i.e. $\|\vec{p}_\lambda\|_2 = \sqrt{\sum_{i=0}^{(N-1)} (p_{\lambda i})^2}$.

In general, solving linear equation systems can be carried out in the described way. The LAPACK functions *sgels* and *dgels* implement these approaches for single and double precision floating point values respectively. The overdetermined case is solved using a QR factorization of the matrix and the underdetermined case using a LQ factorization. Although these methods are straightforward, a matrix of full rank is required. This cannot be guaranteed for the decomposed light transport cluster matrices or at least it has not been proved to be this way. A numerically more stable method to solve such equation systems with a matrix that may be rank-deficient is provided by the singular value decomposition (SVD). LAPACK provides the functions *sgelss* and *dgelss* to solve a set of linear equations in a minimum-norm sense.

Although the SVD optimizes the standard approach by approximating a solution for rank-deficient matrices, both approaches allow negative values in the solution vector. While this is a suitable numerical construct, no real-world projector is capable of producing negative light. Hence, all negative values will be clamped to zero, increasing the error between desired and achieved result. Actually, the radiometric compensation application requires a solution that does not consider negative values and still provides a solution with a minimal error and evenly distributed values (as low values as possible). Such approaches have been explored and are referred to as non-negative least square (NNLS) error solutions to linear equation systems [31]. The NNLS solution is an iterative method that is suitable for solving the cluster-based equation systems so that no negative and only minimal positive clipping occurs, while preserving a high-quality solution. Figure 4.9 shows results for a radiometric compensation of a sample scene using a NNLS solution to the equation systems. The computation of a single compensation image required approximately 3 minutes on a Pentium 4 3GHz with 2GB RAM.

Figure 4.9: Radiometric compensation of a screenshot (a) from the movie "9" (courtesy: Focus Features and 9, LLC). The compensation image (b) is projected onto the scene (c+d). An artificial simulation of the outcome (e) by multiplying the compensation image with the light transport matrix and the actual projection captured by the camera (f).

4.3 Proof of Concept

The compensation depicted in figure 4.9 contains some inter-reflections on the statuette, but mostly diffuse surfaces. In order to validate the proposed techniques, multiple sample scenes including different global illumination effects are compensated and presented in the following subsections.

4.3.1 Projecting on Refractive Material

Highly refractive materials such as glass are a challenge for structured light range scanning. It is often not possible to determine a precise mapping of an individual projector pixel to a single camera pixel. Figure 4.10 (a) shows a glass in front of wallpapered surface. The projection of text[8] visualizes the geometrical distortions due to refractions. Multiple characters are visible at different locations within the camera image. These effects are especially visible near the boarders of the glass.

Not only the complex geometrical distortion, but also the shadows and caustics cast from the illuminated glass, as well as the colored background, make a radiometric compensation difficult. The acquired light transport matrix (fig. 4.10 (b)) shows characteristic features that indicate global illumination effects. Missing portions (left magnified part of figure 4.10 (b)) within a region of the matrix represent the thicker glass on top of the glass. This and its shadow forms an ”8” in the camera image and is not captured in the matrix. The upper right magnified part depicts twisted narrow bands that illustrate refractions. A single projector pixel or neighboring ones affects spatially different regions in the camera image. Usually, the same effect can be observed at object boundaries (depth discontinuities within either the camera or the projector image). However, in this case there are no hard cuts, but a smooth movement of two bands (boundaries of the glass) away from the normal diagonal.

An interesting effect is highlighted in the lower right part of figure 4.10 (b). This is a group of projector pixels that affect a different part of the camera image as all others. It refracts the bottom part of the glass, where the glass is much thicker resulting in heavier refraction. The glass' foot is visible in the camera image only through refraction.

The photograph of a restaurant[9] (figure 4.10 (c) is roughly aligned to fit the corners of the camera image as shown in figure 4.10 (d). Geometrical distortions and color modulation due to the background are clearly visible. Using the proposed technique, a compensation image (fig. 4.10 (e)) is computed and projected onto the scene (fig. 4.10 (f)). Even regions in projector space that are not directly visible in the camera image (foot of the glass) are taken into account.

[8] The poem is ”Jabberwocky” by Lewis Carroll from the book ”Through the Looking-Glass” (1872)

[9] Jonny Rockets serves great (veggie) burgers and fantastic milk shakes.

Figure 4.10: A refractive glass in front of a colored wallpaper (a). The acquired light transport matrix (b) is decomposed and used to compute a compensation image (e) depending on a desired image (c). The result captured from the camera's point of view (f) and a roughly aligned uncompensated projection (d).

The compensation image contains several bright spots that are due to clipping. These often occur near object boundaries, especially if there is a shadowed region next to the particular pixel. However, if the projector and the camera remain in place during the whole procedure, the artifacts are not visible for the camera. Only slight misalignments (even a shift of a single pixel) would unveil them, leading to strongly visible artifacts. Discarding primary colors of the produced compensation image allows to remove the artifacts at the expense of all primary colors of the image itself.

Greenish caustics are cast from the glass onto the surface near the boundary of the shadow. These are hardly visible in the floodlight image, because when illuminating the scene with a uniform color these regions are darker. The same projector pixels are much brighter in the compensation image for creating the desired uniform brightness. Although, these try to compensate for the caustics as well (magenta portions near the glass boundaries in the compensation image) a complete correction is not achieved.

This example clearly demonstrates several physical limitations of radiometric compensation in general. Portions of the camera image that are not illuminated by the projector cannot be compensated. Using multiple projectors would account for shadows, but not for refractions of the environment that is not illuminated by any projector. Several parts of the background are visible at multiple different locations within the camera image. In this case it results from refractions. It is physically not possible to create different colors at those camera pixels that are actually the same surface segments. Such an effect can be seen in figure 4.10 (f). The 'B' of the 'hamburger' writing is visible directly on the surface and refracted in the glass.

4.3.2 Compensating Diffuse Scattering and Inter-Reflections

Among the most often occurring global illumination effects are diffuse scattering and inter-reflections. Scattering is a view-independent effect that results from light being reflected equally in every direction on a diffuse surface. How much of the scattered radiance reaches another surface depends on the geometric relation (form factor) between the two surface patches. The form factor is defined as $\frac{cos\alpha}{d^2}$, where α is the angle between the surface normals and d the distance of the patches. Hence, close patches facing each other receive more scattered radiance than patches that are further apart.

Inter-reflections occur whenever a surface directly reflects a portion of the incoming light. The angle between a reflected light ray and an incident light ray is equal. Organic materials such as transparent films and glossy paper often reflect portions of the incident light spectrum towards certain directions. Figure 4.11 depicts a scene with a wallpapered background and two folded cardboard pieces. The left one is coated with a self-adhesive transparent film. Global illumination effects on these are clearly visible in the floodlight image (fig. 4.11 (a)): inter-reflections on the left piece and diffuse scattering on the right one. There are also some specular reflections on the wallpaper and on the upper left part of the left cardboard piece.

Projecting a color pattern (fig. 4.11 (b)) onto both patches leads to an increased brightness within the corners and color modulations of the neighboring patches (fig. 4.11 (c+f)). These can be compensated with the described technique as seen in figure 4.11 (d+g). While a compensation of color and intensity modulation due to diffuse scattering is well-visible, it is very difficult or even impossible to compensate all effects that result from inter-reflections. Just as seen in the last subsection, if one pixel or region of the projection contributes to different portions of the camera image it is physically not possible to create different colors for both. Besides, the specular reflections on the upper left part of the left cardboard piece cannot be compensated completely, because of the projector's blacklevel. Figure 4.11 (e+h) show the difference between the compensated and uncompensated camera images. These were contrast enhance by 50%.

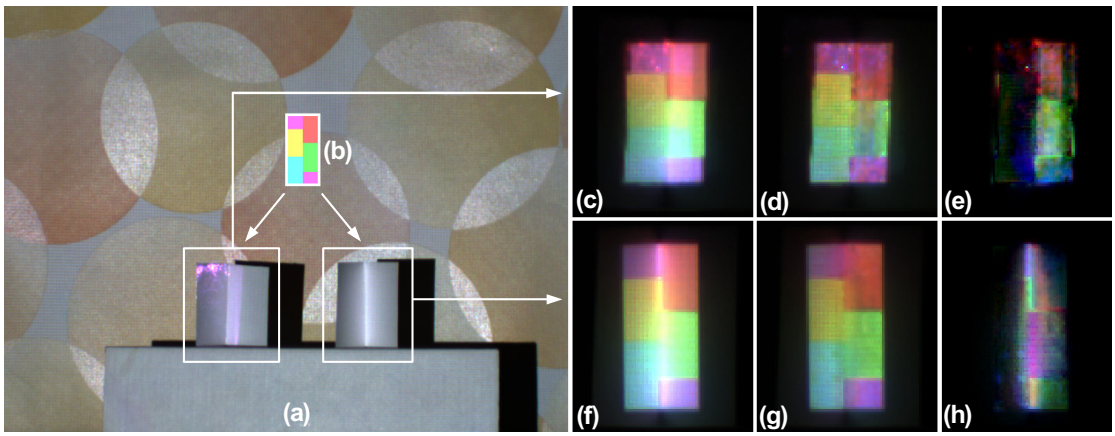

Figure 4.11: Compensation of inter-reflections and diffuse scattering. A colored patch (b) is projected onto a scene (a) and leads to color and intensity modulation (c+f). These can be compensated with the proposed technique (d+g), (e+f) depict the contrast enhanced difference.

The light transport matrix was not decomposed into clusters to compensate for the color patches in figure 4.11. Only the non-black camera pixels, all directly connected projector pixels and camera pixels that are connected to these (several black pixels around the image boundary) were used for the compensation. In order to correct a desired image that covers the whole camera, a matrix decomposition is necessary. This leads to small clusters on the background and larger ones on the cardboard pieces (see fig. 4.12 (c)). Note that, even though the clusters cover the entire cardboard areas, energy distribution within the light transport matrix is modulated due to the composition. An original image (fig. 4.12 (a)) is aligned with the camera image and projected onto the scene (fig. 4.12 (b)). This leads to geometric distortion and color modulations, which can be corrected by computing a compensation image (fig. 4.12 (c)) and displaying it (fig. 4.12 (d)).

This experiment shows, that it is possible to compensate for illumination effects such as scattering and inter-reflections. Just as seen in the previous section is is hardly possible to correct direct specular reflec-

tions. Employing multiple projectors as described by Park et al. [49] could solve this problem, but has not been implemented yet. It is difficult, but partly possible, to produce correct colors in different camera regions that are affected by the same projector pixels.

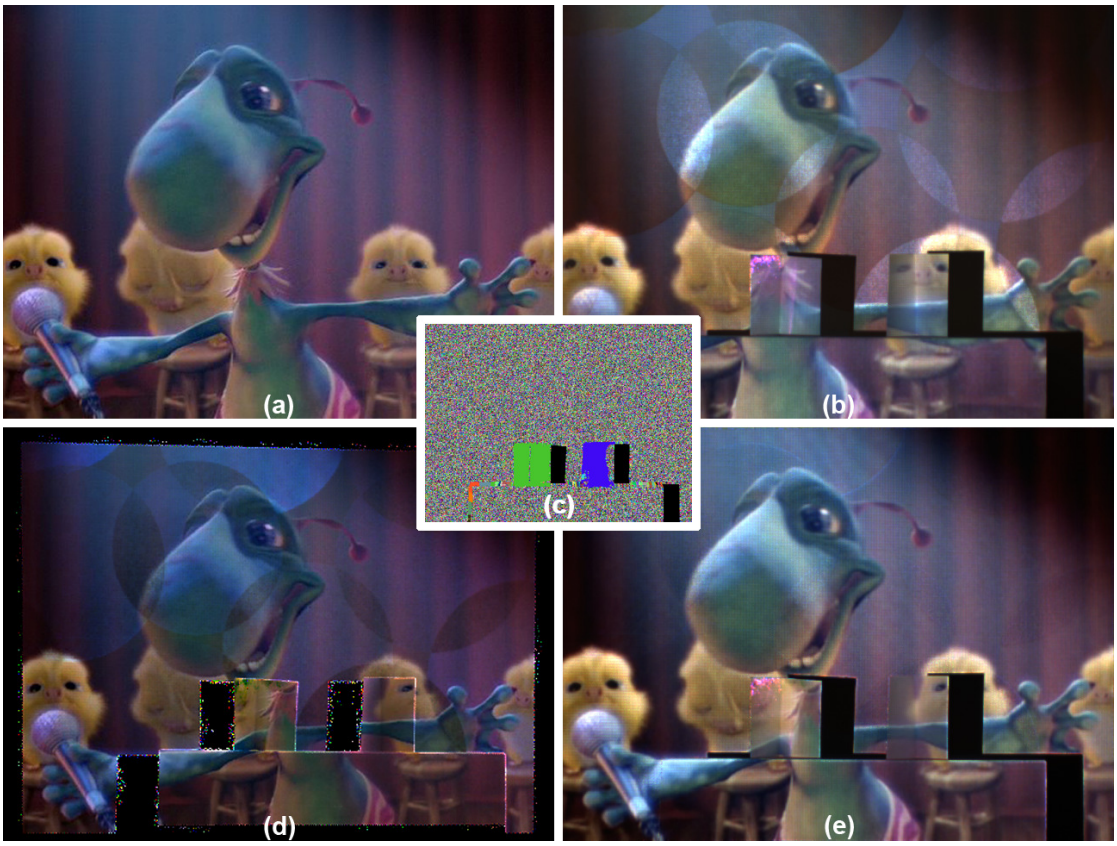

Figure 4.12: A screenshot of the shortfilm "The Chubb Chubbs" (courtesy: Pixar) (a) is aligned with the camera image and displayed (b). Projecting a compensation image (d) onto the scene leads to a correct view (e). The clusters in camera space are shown in (c).

4.3.3 Defocus Compensation

Projectors can display focused images on a single fronto-parallel plane only. They have large apertures, thus, narrow depth of fields. When projecting onto scenes with high depth variance this leads to partly defocused imagery. Bimber and Emmerling [5] proposed to use multiple projectors, each focused on a different plane, to create more brilliant and overall focused images. Another approach was presented by Zhang and Nayar [70], who estimated the 3D geometry of a scene with depth-from-defocus techniques using a coaxial projector-camera system. This information was used to sharpen the desired image, which after being projected on the scene, resulted in an increased image quality.

Since projector defocus is included in the light transport matrix, an alternative idea is to apply the ra-

diometric compensation to increase projection focus. However, the decomposition described in section 4.2.2 is likely to affect and modify the encoded defocus information, since local overlaps are discarded. Hence, the matrix should not be decomposed. Due to its enormous size, it is difficult and time consuming to solve an equation system for the full matrix. However, for original images that cover only a part of the camera image the equation system can be efficiently reduced to only the corresponding camera and affecting projector pixels, as described in the last subsection for the scene depicted in figure 4.11.

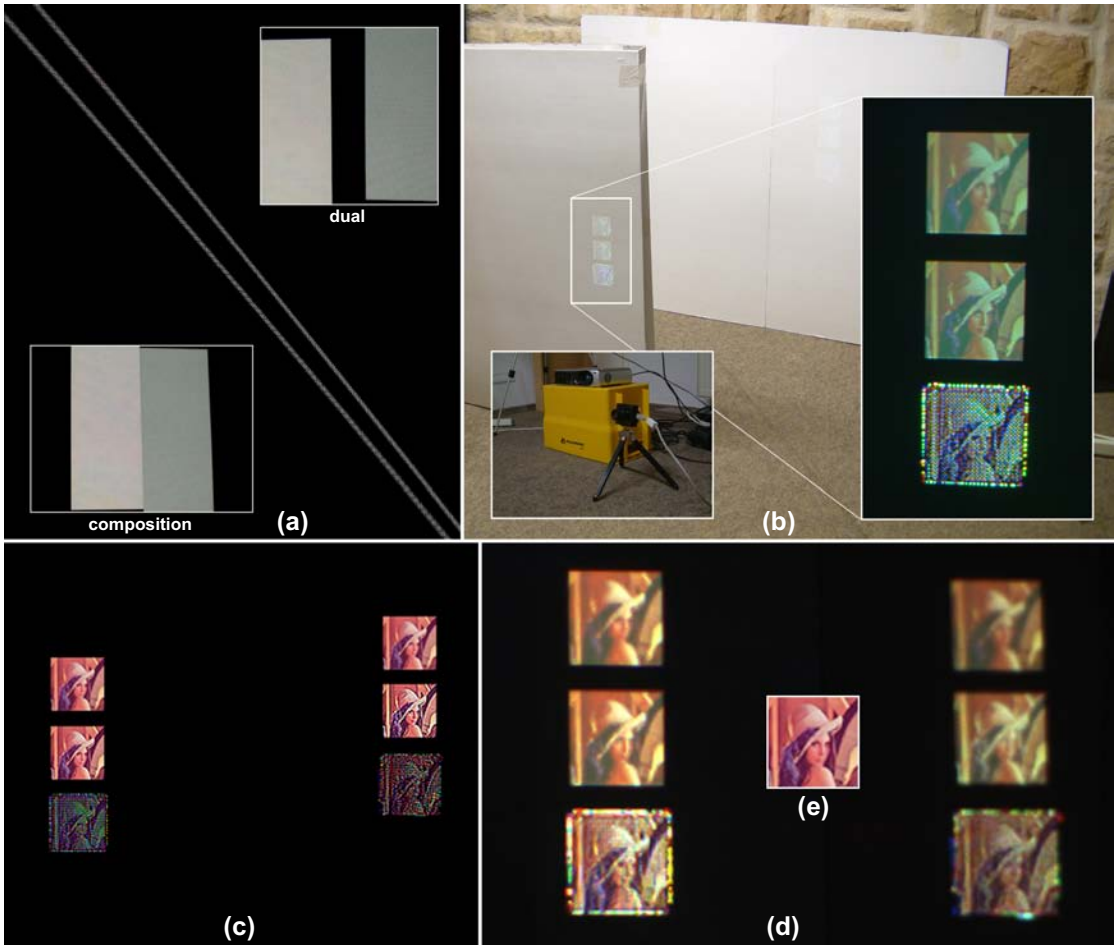

Figure 4.13: The acquired light transport matrix (a) shows two narrow bands indicating the two cardboard planes that were captured and used for projection (b). The compensation image (c) includes the original image (e), a sharpened version and the compensation, which leads to an increased sharpness from the camera's point of view (d).

The limitations of any technique that tries to enhance the sharpness of imagery displayed by a single projector are given by projector's defocus and the sharpness of the original image. It is not possible to increase the sharpness of a straight edge in any way. It makes only sense to modify soft contours, thus, these techniques are highly dependent on the projected content. All completely sharp imagery will have the defocus of the projector when being displayed. Hence, the achievable quality is directly proportional to the difference of projector defocus and image blur. As long as the projector's defocus is less than or

equal to the frequencies within the original images these can be compensated. The resulting quality is decreased if the defocus is higher and the contours in the image harder.

An experiment was performed using a single projector, single camera setup with two cardboard planes at different distances (see figure 4.13 (b)). The resulting light transport matrix along with the composition and dual images are presented in figure 4.13 (a). The left hand side in the camera image is only slightly defocused, while the defocus in the right hand side is quite significant. Compensating for an original image with a size of 48x48 pixels resulted in equation systems with an approximate[10] size of 2304x2700 matrix elements. The compensation was performed in about 25 minutes using a non-negative least squared error solution on a Pentium 4 3GHz with 4 GB RAM. The projected compensation image 4.13 (c) shows the original image (upper row, left side is the neared plane), a sharpened image[11] and the result of the radiometric compensation. The view from the camera's perspective shows a significant increase of perceived image sharpness.

Several issues of the compensation image and the captured result have to be discussed. First of all, the compensation image is not a simply sharpened version of the original picture. In this case several projector pixels are "turned off". This leads to the impression that the compensation is actually darker, but appears as bright in the camera image. In fact, the lit pixels are brighter than the average in the original image leading to good results for the calibrated camera, but suboptimal results for different viewpoints as indicated in the magnified part of figure 4.13 (b). One should keep in mind that the proposed compensation technique produces corrected images for the camera's point of view without making any assumptions on the scene's geometry, while Zhang and Nayar [70] perform a correction using acquired depth information. Besides, not only the projector defocus in included in the light transport matrix, but also the camera's as well as color interpolation artifacts due to the camera's color filters and other camera specific effects such as blooming. The numerical methods for solving the equation systems are likely to produce bright colored spots at the boundaries of the object.

The experiment proofed that it is possible to apply the presented radiometric compensation technique to increase the overall sharpness of projected imagery with a single projector. Although only small pictures were used for the compensation, it is possible to perform the operation on large images if the necessary hardware resources are available. Optimal results are produced for the calibrated camera.

[10] The amount of influencing projector pixel varied slightly for different locations within the camera image

[11] The sharpening was performed using standard PhotoShop filters. The image on the right hand side was sharpened more to simulate a compensation for more defocus.

5 TOWARDS INTERACTIVE COMPENSATION ON PROGRAMMABLE GRAPHICS HARDWARE

A straightforward approach to compensate radiometry and correct for geometrical distortions in a projector-camera system was introduced in the last chapter. However, computing a single compensation image takes, depending on the size of the clusters and available computing resources, several seconds to minutes. This does not allow to display interactive content such as movies. A minimum of 25-30 frames per second is necessary to watch a movie without motion artifacts. As mentioned in section 3.3.2, programmable graphics processing units (GPUs) offer a powerful platform for accelerating the performance of many (graphical) applications tremendously.

Modern GPUs provide inexpensive and highly parallel processing capabilities, however they also have their limitations - not each algorithm is suitable to be implemented on graphics cards. Only algorithms that follow the SIMD (single instruction multiple data) principle can efficiently be processed on programmable graphics hardware. Fragment shader provide a tool to control the output of every pixel on the computer screen or an off-line buffer. A single shader can be executed in parallel for multiple fragments depending on the amount of pixel-pipelines of the graphics hardware. This is only possible if the same instructions (as specified in the shader) are used for each fragment, input data may vary. The maximum amount of output parameters in a shader is limited by the capabilities of the framebuffer. This often stores up to four 8 bit values (the color channels for red, green, blue and alpha). Recent hardware supports 16 and 32 bit floating point buffers as well as multiple render targets (up to four output buffers per fragment).

Solving linear equation systems of the form $A\vec{x} = \vec{b}$, where A is a given MxN matrix, $\vec{b}$ the desired output vector and $\vec{x}$ the unknown, is not a particular well suited problem to be solved on GPUs. It is difficult to parallelize such sets of linear equations, since these represent coupled systems that cannot be solved for individual elements of the solution.

In this chapter the approach to radiometric compensation, as introduced in section 4, is modified to be suitable for an implementation on programmable graphics hardware. This allows interactive framerates.

5.1 An Efficient Approach to Solving Linear Equation Systems on the GPU

The overall goal for the compensation is to produce an image that, projected on the scene from the projector, produces an optimal desired image from the camera's point of view. Thus pixel intensities for each projector pixel have to be computed. A very efficient way to produce such a corrected image on GPUs would be to solve the equation systems in a way that can be performed separately for each projector pixel.

Solving the set of equations $\vec{c}_\lambda = T_\lambda \vec{p}_\lambda + \vec{e}_\lambda$ can be reformulated by applying the inverse light transport matrix to both sides, yielding $T_\lambda^{-1}(\vec{c}_\lambda - \vec{e}_\lambda) = \vec{p}_\lambda$. A matrix inverse, however, exists only for regular (non-singular) square matrices. Most likely the projector and camera will not have the same resolution, thus the matrix is not square. A numerical approximation to a general matrix inverse exists, which is called pseudo-inverse or Moore-Penrose inverse [52, 45] and denoted by T^+. The pseudo-inverse of a matrix can be computed using the same methods as introduced in section 4.2.4 (LQ/QR factorization or SVD). If T is an invertible square matrix T^+ and T^{-1} are equal and computing the pseudo-inverse is an expensive way to generate the inverse. The pseudo-inverse allows to reformulate the compensation problem as follows:

$$\begin{array}{ccccc} \vec{c}_\lambda - \vec{e}_\lambda & = & T_\lambda & & \vec{p}_\lambda \\ \begin{bmatrix} c_{\lambda 0} - e_{\lambda 0} \\ c_{\lambda 1} - e_{\lambda 1} \\ \vdots \\ c_{\lambda(mn-1)} - e_{\lambda(mn-1)} \end{bmatrix} & = & \begin{bmatrix} t_{\lambda 0}^{0} & \cdots & t_{\lambda 0}^{(pq-1)} \\ t_{\lambda 1}^{0} & & t_{\lambda 1}^{(pq-1)} \\ \vdots & \ddots & \vdots \\ t_{\lambda(mn-1)}^{0} & \cdots & t_{\lambda(mn-1)}^{(pq-1)} \end{bmatrix} & & \begin{bmatrix} p_{\lambda 0} \\ \vdots \\ p_{\lambda(pq-1)} \end{bmatrix} \end{array}$$

$$\Downarrow \tag{5.1}$$

$$\begin{array}{ccccc} T_\lambda^+ & & (\vec{c}_\lambda - \vec{e}_\lambda) & = & \vec{p}_\lambda \\ \begin{bmatrix} t_{\lambda 0}^{0} & t_{\lambda 1}^{0} & \cdots & t_{\lambda(mn-1)}^{0} \\ \vdots & & \ddots & \vdots \\ t_{\lambda 0}^{(pq-1)} & t_{\lambda 1}^{(pq-1)} & \cdots & t_{\lambda(mn-1)}^{(pq-1)} \end{bmatrix}^+ & & \begin{bmatrix} c_{\lambda 0} - e_{\lambda 0} \\ c_{\lambda 1} - e_{\lambda 1} \\ \vdots \\ c_{\lambda(mn-1)} - e_{\lambda(mn-1)} \end{bmatrix} & = & \begin{bmatrix} p_{\lambda 0} \\ \vdots \\ p_{\lambda(pq-1)} \end{bmatrix}. \end{array}$$

Note that the elements $t_{\lambda i}^{j}$ of T^+ are not the same elements as in T. The ith column of the pseudo-inverse semantically represents a projector pattern, that has to be projected to illuminate a single camera pixel at a time. Summing all columns of T^+ generates the projector pattern that, theoretically, generates a uniform white illumination of the scene from the camera's point of view (see figure 5.3).

Computing the pseudo-inverse is numerically expensive and relatively unstable compared to solving a conventional linear equation system, thus it is more a theoretical construct than of practical use in applied mathematics. However, for the special case of radiometric compensation it provides a solution that includes a computational relatively expensive preprocessing step (computing the pseudo-inverse) and a simple vector dot product of each of T^+'s rows and the desired image in camera space. The vector multiplication can be performed in parallel for each projector pixel: $p_{\lambda i} = \sum_{j=0}^{(mn-1)} \left(\left[t_{\lambda j}^{i} \right]^+ (c_{\lambda j} - e_{\lambda j}) \right)$. This approach also separates the inverse light transport from the input data (images, video, 3D visualization etc.).

A robust, stable and open-source implementation of matrix pseudo-inversion is included in the linear al-

gebra package (LAPACK) [2]. Although LAPACK is implemented in Fortran, freely available wrappers for C/C++ (CLAPACK) exist.

Compared to solving the sets of linear equations explicitly, a non-negative least squared error solution for computing the pseudo-inverse does not necessarily provide the best solution. Since the inverse is independent of possible input images it does not make much sense to only consider positive matrix entries. Clipping artifacts may still occur, however, these are mostly only individual pixels near the borders of projector or camera space and next to shadowed regions in either space.

5.2 Optimizing Data Structures

Using programmable graphics hardware allows to compute the results for individual projector pixels separately and in parallel. However, every pixel has to be provided with the data that is essential for the computation. Projector pixel i requires the entries of the ith row of the pseudo-inverse for each color channel as well an index for x and y to access the original image at each position. This allows to perform a vector-vector multiplication of the original image and a specific row of the inverse light transport matrix.

Textures provide the only kind of manageable memory interface for larger amounts of data on GPUs. Hence, all necessary information for each projector pixel has to be encoded using 32 bit floating point textures. Although the 16 bit half format could be used as well this only provides up to 1024 different values, which may be to few to correctly access the entries of the pseudo-inverse. A texture of projector resolution (projector pixel lookup table) is used to provide every projector pixel with an index x and y to access a second texture that contains the actual entries of the appropriate row in the pseudo-inverse as well as the length of this vector. Actually, the length of all these vectors is the same, but in order to accelerate computation time and required storage space only non-zero matrix entries are taken into account. This results in a varying vector length.

The texture that is accessed via the projector pixel lookup table (correction map) contains the elements of the light transport matrix's pseudo-inverse. Since there are actually three matrices, one for each color channel, their entries can be encoded in the different color channels of the correction map. This is used to continuously store the non-zero matrix row elements in incremental x texels. An efficient implementation sorts these vectors in a preprocessing step so that no row breaks for the vectors occur and the texture is optimally filled according to the maximal available texture width.

Each of the correction map's entries also need a reference to the appropriate coordinate in camera space. This is used to access the desired image. Texture lookups on the GPU are performed using one index for x and one for y. Since there is only one color channel left in the correction map (alpha) the two indices can either be encoded in a single scalar using $index = x + y * textureWidth$ and later recovered

by $x = index \,/\, textureWidth, \quad y = index \,\%\, textureWidth$ or by using a second texture to only store the indices x and y.

Figure 5.1 illustrates the requires textures. On the left is the projector pixel lookup table that contains access to the vector of inverse light transport matrix elements stored in the correction map as well as the length of the appropriate vector. The correction map is used to perform the radiometric compensation by multiplying a row of the matrix with the appropriate elements of the original image (on the right) that is accessed by an index stored in the correction map.

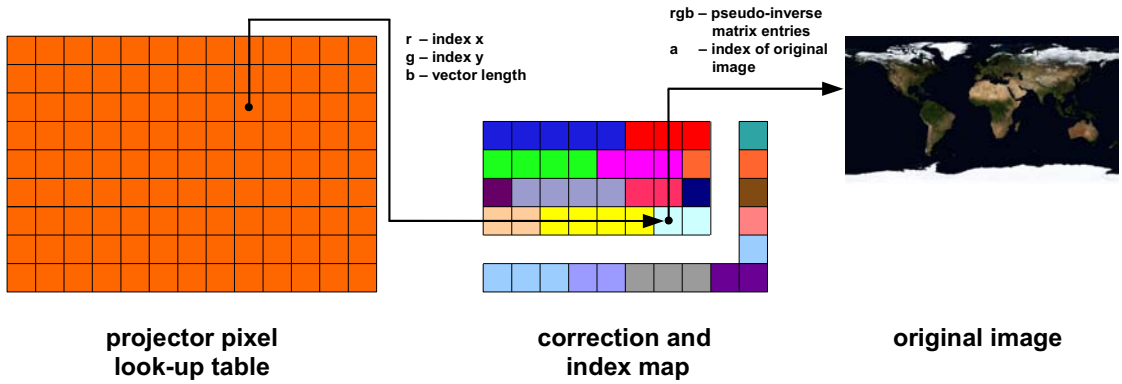

Figure 5.1: Optimal data structures for performing a radiometric compensation on the GPU. A per-projector pixel lookup table (left) allows access to the entries of the inverted light transport matrix (center) and the original image (right).

As mentioned before, only 32 bit floating point textures are used to allow precise access to the textures. If the correction map is neither wider nor higher that 1024 pixels the projector pixel lookup table can be efficiently stored using only 16 bit per entry. Converting the correction map to 16 bit half values would not only reduce the dynamic range of the inverse light transport matrix entries, but also limit the index stored in the alpha component to a non-acceptable size. Hence, using 32 bit textures in all cases is in general the only exact solution.

Note that the camera and projector resolution as well as the size of the clusters determine the memory required for storing the correction map. The maps of the example scene shown in figure 4.1 had a size of approximately 120 MB.

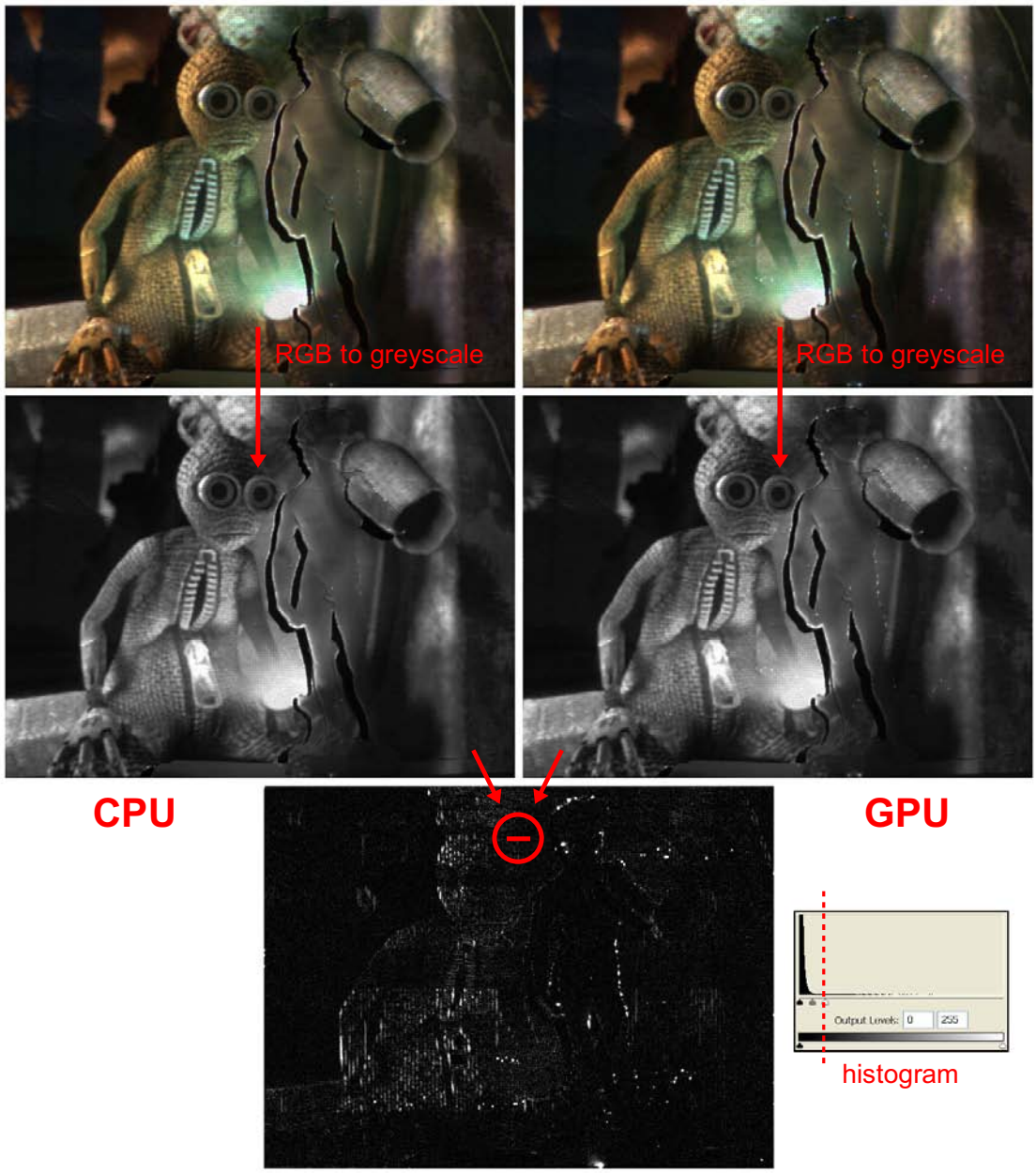

Figure 5.2: The upper images show a compensated scene using the CPU (left) and the GPU (right) approach. Both images are converted to greyscale (images in the center). The lower 25% intensities of the difference between both is shown in the lower part. This is contrast enhanced and shows slight variations of the compensated geometry and color.

5.3 Results and Evaluation

A comparison between the approach introduced in chapter 4 and the GPU approach described in this chapter is shown in figure 5.2. The same scene as in figure 4.9 is compensated using the light transport matrix's pseudo-inverse on programmable graphics hardware. The upper left camera image is the projected compensation image, which was computed by explicitly solving the equation systems on the CPU. The compensated image that was generated in real-time on the GPU using T's pseudo-inverse is on the upper right. Both pictures are converted to greyscale as illustrated in the second row of figure 5.2. The absolute difference between the greyscale images indicates variations. These, however, are very small, thus the lower 25% of image intensity are streched to fill the entire range from 0 to 255. This is indicated by the associated histogram on the lower right. No difference occurs in black regions, brighter parts indicate higher variations. These appear as slight shifts of the compensated image geometry on the puppet's body and near borders between shadowed and non-shadowed regions in either the composition or the dual image. However, these differences are hardly perceivable when seeing the wohle compensated image. Several clipping artifacts (bright colored spots) as seen in the GPU method are filtered out on the CPU. These are actually present in both, but for achieving maximal framerates the filtering was not implemeted on the GPU. Solving the equation systems explicitly on the CPU using LAPACK's NNLS solution took approximately 3 minutes on the CPU (Pentium 4, 3 GHz, 2 GB RAM). The GPU implementation achieved 7 frames per second on a GeForce 7900 GTX with 512 MB memory, with a preprocessing step (computing T^+) of app. 15 minutes.

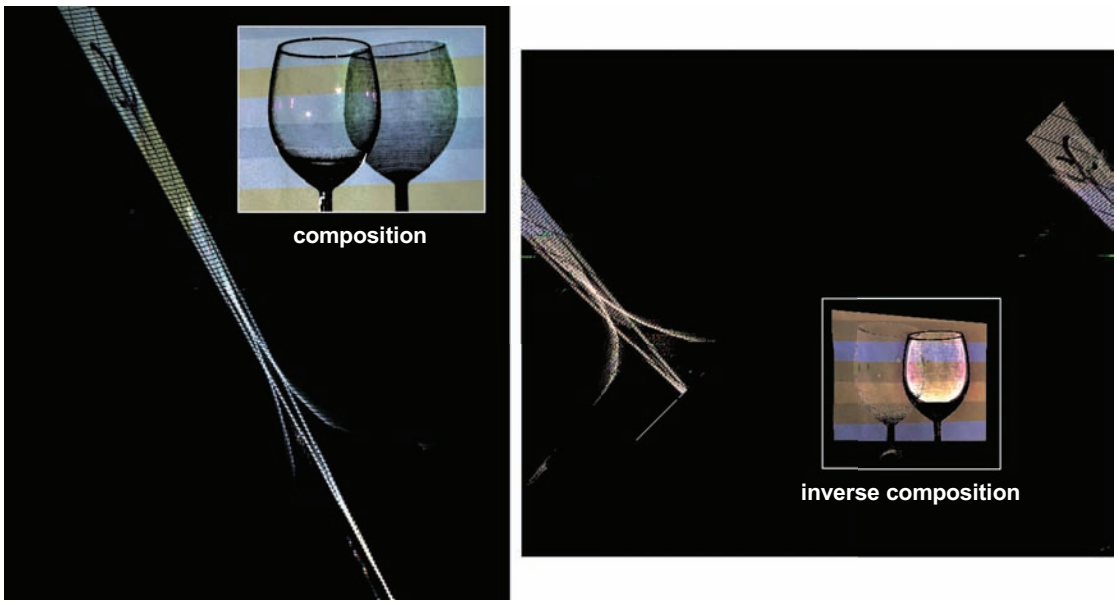

Figure 5.3: Visualization of the light transport matrix, its composition and dual (left). The matrix's pseudo-inverse, the inverse composition and the inverse dual (right).

The light transport matrix's pseudo inverse is presented in figure 5.3. The left image shows the acquired matrix (see section 4.3.1) superimposed with its composition image. The right hand side depicts the in-

verted light transport matrix with its composition. The inverse dual image represents the image that would have to be displayed from the projector to achieve a uniform white image captured by the calibrated camera.

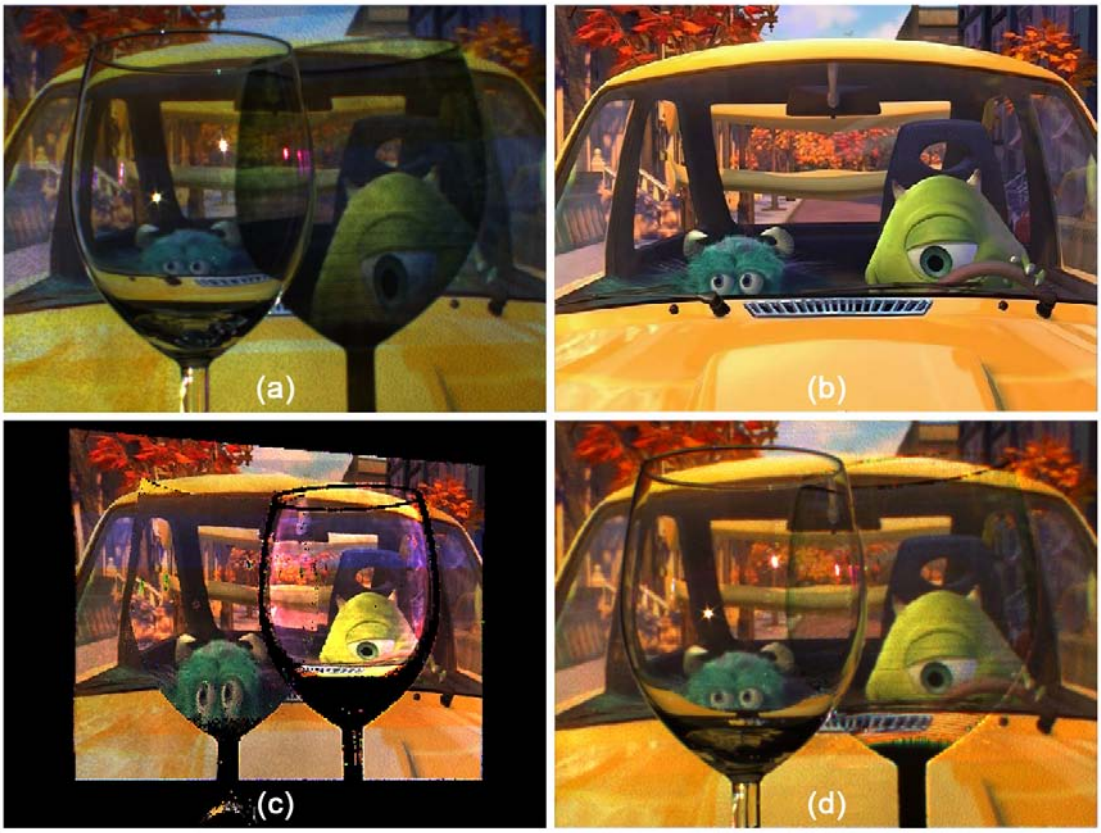

Figure 5.4: A scene from the shortfilm "Mike's new Car" (b) (courtesy: Pixar) and an uncorrected projection (a) onto the glass scene presented in figure 4.10. The compensation image (c) and captured from the camera's viewpoint (d).

Figure 5.4 illustrates another setup that shows the quality of the GPU accelerated compensation. The scene is the same as presented in section 4.3.1. Although color modulation of the uncompensated projection (fig. 5.4 (a)) and the wallpapered background are, due to the content, not well visible, geometrical distortions are obvious. The eyes of both characters are clearly deformed and compensated in figure 5.4 (d). Not all refraction can be compensated as seen in the bottom of the glass. This shows the physical limitations of any projector-camera system used for radiometric compensation. The data structures necessary for the GPU computation were generated in app. 13 minutes (P4, 3GHz, 2GB RAM), the on-line compensation was performed at interactive framerates with about 25 fps on a GeForce 7900 GTX.

The results presented in this section proof that a real-time compensation is possible using programmable graphics hardware. For achieving this performance, the light transport matrix's pseudo-inverse is required, which is calculated in a computationally relative expensive preprocessing step. Computing the

pseudo-inverse using singular value decomposition takes app. 3-8 times longer than solving the set of equations with the iterative NNLS approach. T^+ can be used for compensating multimedia content in real-time, while the explicit method compensates only a single image at a time. The quality of the GPU compensation and the explicit NNLS solution to the resulting equation system is almost the same - visible differences are marginal.

6 DISCUSSION

6.1 Comparision to SmartProjector

The radiometric compensation approach proposed by Bimber et al. [6] assumes a direct mapping between projector and camera pixels. This is theoretically given when projecting onto diffuse surfaces. In such a case, the light transport matrix would reduce to a single (or very few non-overlapping) entries in each column, if this particular projector pixel is visible from the camera's perspective. Decomposing this matrix would result in clusters of one or few camera pixels and a few projector pixel. Hence, the cluster-based light transport matrices of the global illumination approach would reduce to a single scalar or few entries. A similar radiometric compensation equation results for both approaches if the light transport matrix has only one element.

In practice, the direct approach estimates a single projector pixel for each camera pixel. An inverse mapping is achieved through triangulation and interpolation, resulting in a look-up table that stores one camera pixel for each projector pixel. Subpixel accuracy is achieved, but local light interaction on non-planar surfaces and the effect of mutually influencing, neighboring projector pixels is discarded.

In order to compare both techniques, a setup that includes only lambertian surfaces must be employed. The goal of such an experiment is to proof that both methods produce similar results under similar settings. It is not to show that the proposed method compensates for global illumination effects and the previous does not. The results presented in section 4.3 already proofed that it is possible to compensate such effects. A precise geometric mapping between projector and camera, as required by all previous approaches, is impossible to acquire in some of these setups.

An experiments has been carried out using an artificial brickstone wall as shown in figure 6.1. A screenshot of the shortfilm "For the Birds" (courtesy: Pixar, fig. 6.1 (a)) is compensated using a projector-camera system (fig. 6.1 (b)). A pixel-precise mapping between projector and camera is acquired using projected structured light patterns. This is employed to predistort the desired image (fig. 6.1 (d)), which appears geometrical correct when projected onto the scene and captured by the camera (fig. 6.1 (c)). Applying the radiometric compensation described in [6], the colors can be corrected as well (fig. 6.1 (e+f)). The same image is compensated using the global illumination approach presented in chapter 4.

The results of the compensations are comparable, however, they are not the same under similar settings. There are various reasons that may explain the differences. First of all, the images acquired for the global illumination approach are all high dynamic range images. While the same linearized camera is used for the direct approach, this only captures 8 bit images, resulting in a lower contrast. The HDR images are composed out of 12 exposures, each averaged out of three LDR images. Hence, each HDR image is composed out of 36 captured LDR photographs. This results in more reliable and precise irradiance values.

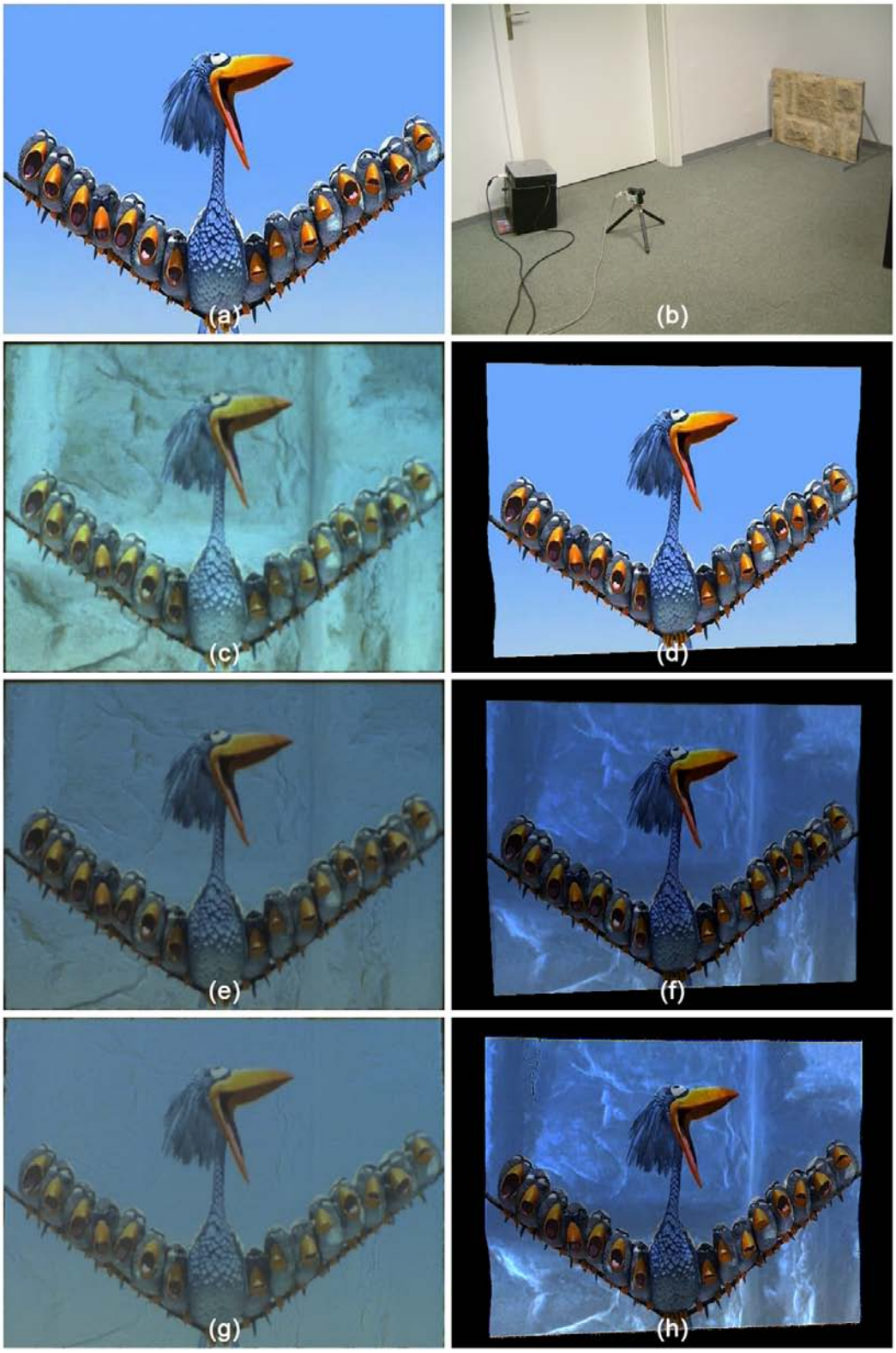

Figure 6.1: Compensation of a screenshot from the shortfilm "For the Birds" (a) (courtesy: Pixar) using the direct (c+d+e+f) and the global illumination (e+f) approach. The setup is shown in (b).

Although an LCD projector is employed, flickering is captured with the camera, especially for shorter exposures.

On the other hand, the clusters resulting from the light transport decomposition are small regions, not individual pixels. This seems to compensate better for local scattering on the rough surface. Although, misregistrations of the direct mapping could lead to less overall quality these are usually visible in the compensated image and not present in this example.

This experiment proofs that it is possible to achieve at least comparable results when applying the described approaches using the same setup. Both implementations discard spectral differences between projector and camera. However, the image acquisition used for the global illumination approach are more sophisticated and deliver more precise radiometric values. The direct compensation achieved over 100 fps while the novel method's performance was about 30 frames per second. Acquiring the light transport with an unoptimized implementation took app. 4.5 hours, while the direct approach required about 30 seconds for capturing all necessary data.

6.2 Summary and Conclusion

Acquiring the light transport between a projector and a camera allows to capture all global illumination effects. The chosen threshold represents a tradeoff between accuracy of the captured effects and acquisition time as well as required storage space. The light transport is a transformation from projector space to camera space ($\Re^{pq} \rightarrow \Re^{mn}$) and can be represented by a matrix T of size $mn\ x\ pq$, with mn being camera resolution and pq projector resolution. Since the matrix is usually sparse it is computationally more efficient to internally store and process it as a weighted bipartite graph. The graph has two different kinds of nodes: camera and projector pixels. No connections exist between nodes of the same type, the weighted connections between nodes represent radiance contribution from specific projector to camera pixels. The three color channels are handled separately.

A radiometric compensation can be performed using the simplified fundamental relation of forward light transport $\vec{c}_\lambda = T_\lambda \vec{p}_\lambda + \vec{e}_\lambda$, with $\vec{c}_\lambda$ being color channel λ of the camera image, T_λ the light transport matrix, $\vec{p}_\lambda$ the projector contribution and $\vec{e}_\lambda$ the incident environment light including the projector's black level. After T_λ and $\vec{e}_\lambda$ are captured from the camera, $\vec{c}_\lambda$ can be replaced by a desired image of the same resolution. This forms a linear equation system with mn equations and pq unknowns that can be solved with respect to $\vec{p}_\lambda$, which is the compensation image.

A comprehensive mathematical foundation for radiometric compensation in multi-projector, multi-camera setups has been derived. This is expressed as a set of linear equations and takes the spectral differences between projectors and cameras into account. The simplified form is based on the assumption that individ-

ual projector color channels only affect the corresponding camera channels. While this is in general not the case, all presented experiments described have been carried out using the simplified form. Neglecting the spectral differences generally leads to satisfying results and allows a speedup of light transport acquisition time, a faster radiometric compensation and less required storage space.

In a typical projector-camera setup the light transport matrix is huge. However, it can be decomposed into clusters of mutually influencing camera and projector pixels. An individual equation system can be solved for each cluster. In a real-world setup neighboring projector pixels are likely to overlap in the camera image. This is due to projector and camera defocus as well as blooming and often leads to interconnections between many camera and projector pixels resulting in a single or few very large clusters of an acquired scene. Therefore, a decomposition of the matrix can be employed that splits up local connectivity and preserves global illumination effects.

Decomposing the light transport matrix in this way discards local overlaps, thus projector and camera defocus. Most of the presented experiments demonstrate how effects such as refractions, reflections, scattering and caustics can be compensated. However, if a compensation for projector or camera defocus is required the matrix cannot be decomposed. It has been shown, that it is possible to optimize overall sharpness of projected imagery of a single, defocused projection system using the described techniques.

Conventional (spectral) graph partitioning and cutting approaches are difficult to utilize due to the enormous size of the graph, its adjacency matrix and T. Hence, a customized technique can be applied that groups all connected projector pixels of each camera pixel into blocks of neighboring pixels. The neighborhood is characterized by its radius and center, which is the projector pixel with the highest radiance contribution within this neighborhood. Only a limited amount of projector pixels per neighborhood is allowed for every camera pixel, lower weighted connections are discarded. Individual projector pixels are preserved from being cut-off the graph by reinserting connections according to a second threshold. The decomposition modifies the energy withing the light transport, thus this has to be scaled to fit the original composition of the matrix.

Due to the decomposition small clusters occur on diffuse surfaces with only local illumination, while areas with lots of inter-reflections are likely to produce larger clusters. An optimal solution to the resulting equation systems is provided by an iterative non-negative least square (NNLS) solution. This is necessary because projectors are not capable of displaying negative light. Depending on the cluster sizes a compensation image can be calculated in several minutes. A simulation of the captured camera image can be performed by inserting the compensation image $\vec{p}_\lambda$ into the forward light transport equation $\vec{c}_\lambda = T_\lambda \vec{p}_\lambda + \vec{e}_\lambda$.

Solving the equation systems explicitly does not allow to perform a compensation in real-time. The problem of finding $\vec{p}_\lambda$ for a desired $\vec{c}_\lambda$ can be reformulated using T_λ's pseudo-inverse, yielding $T_\lambda^+(\vec{c}_\lambda - \vec{e}_\lambda) = \vec{p}_\lambda$. This is numerically less stable but suitable for an efficient implementation on programmable graphics

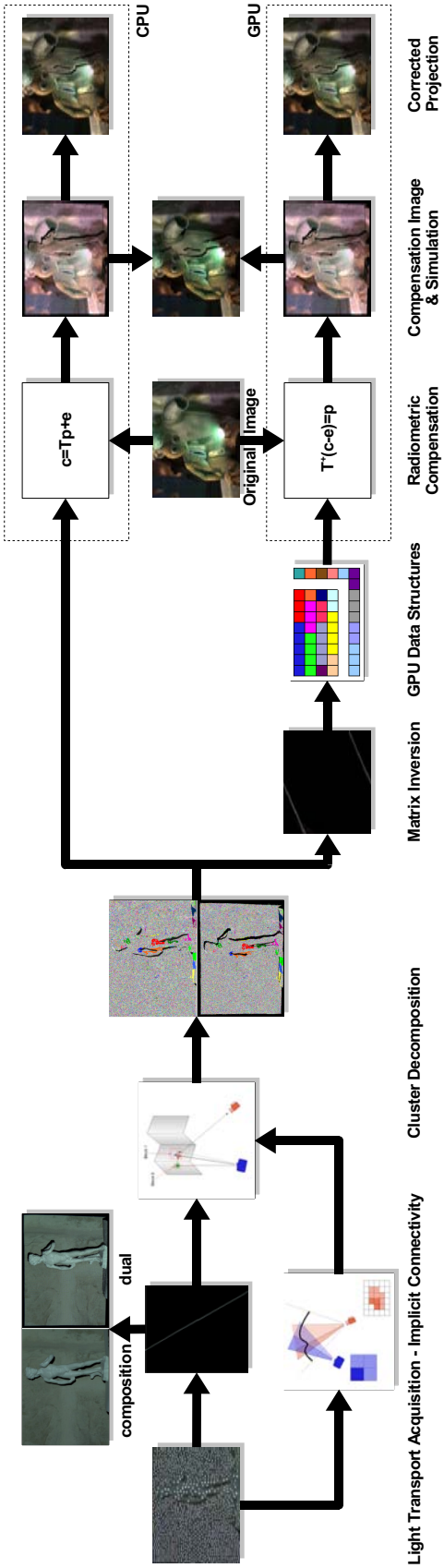

Figure 6.2: Workflow chart of the radiometric compensation approach.

hardware. T^+ is usually calculated using SVD, because it allows to solve equation systems with a rank-deficient coefficient matrix (it has not been proven that the decomposed subsets of T are of full rank). In this way the compensation is split into a computationally relative expensive preprocessing step (computing T^+) and an on-line matrix-vector multiplication. Latter is implemented using a fragment shader as a multiplication of T^+'s appropriate non-zero row elements and the desired image for each projector pixel. Textures are filled with optimized data structures to provide each fragment with the necessary data. It has been shown that real-time framerates can be achieved depending on the complexity of the scene. Even though, applying a matrix's pseudo-inverse to solve a set of linear equations is numerically less stable and rarely used in applied mathematics, it has been shown to produce similar results compared to explicitly solving the equation systems.

The entire workflow for the radiometric compensation approach is illustrated in figure 6.2. The procedure starts with the light transport and cluster acquisition. If necessary the clusters can be decomposed and individual equation systems can be solved for each cluster depending on a desired original image. This is usually carried out on the CPU using an NNLS implementation provided by LAPACK. To allow for interactive compensation, the matrices of the cluster's equation systems can be inverted in a preprocessing step and used as basis of an optimized GPU implementation.

scene	acquired HDR images	approximate acquisition time	compensation time [NNLS]	computing T^+	framerate [fps]
dinosaur skull [sec. 3.4.2]	205	6h	n.a.	n.a.	n.a.
simple wall [sec. 6.1]	281	4.5h	1m	8m	30
David statuette and wall [chapter 4]	813	4h	3m	15m	7
glass & wallpaper [sec. 4.3.1]	1129	5h	3m	13m	25
diffuse scattering & inter-reflections [sec. 4.3.2]	2645	9h	1h	n.a.	n.a.
focus compensation [sec. 4.3.2]	257	2h	25m	n.a.	n.a.

Table 6.1: Summary of required acquisition frames and time of the testscenes.

A summary of all acquired scenes along with the number of captured HDR images, approximate acquisition and compensation times as well as framerates is illustrated in table 6.1. Note that the implementation of the light transport acquisition changed significantly in between the scans - resulting in varying acquisition times. Computational resources varied as well. The scene showing the dinosaur skull has not been used for a radiometric compensation and the focus scene from section 4.3.3 has not been used in combination with the explicit solution to the equation systems.

6.3 Limitations

The proposed compensation technique has clear limitations. These are first of all given by the physical setup of the scene. It is impossible to compensate for several effects. Imaging that multiple regions in a camera image show the same projector pixels, for example due to refractions, it would not be possible to display different colors with the same projector pixels for different locations in the camera image. Also, portions of the camera image that are not illuminated by the projector cannot be compensated. Using

multiple projectors would account for shadows, but not for refractions of the environment that is not illuminated by any projector. The limited brightness of projectors does not allow to compensate all surfaces, it may for instance not be possible to let a red object appear blue. Conventional LCD or DLP projectors have a limited contrast and a relatively high blacklevel contribution. Hence, it is not possible to compensate all kinds of material pigments with conventional projection displays.

On the other hand, the proposed compensation is only capable of considering effects that are included in the light transport matrix. Acquiring an accurate light transport is a challenging task. Depending on the scene a huge amount of HDR images may be required. Selecting a threshold that accounts for all desired illumination effects and results in a manageable amount of images that have to be acquired is very difficult. Especially low intensity effects, such as diffuse scattering, are difficult to capture precisely.

6.4 Future Work

Finally, several possible future experiments and applications are suggested. First of all it would be desirable to implement and evaluate the general approach to radiometric compensation described in section 4.1. Therefore, one light transport matrix has to be acquired for each projector channel. Since the maximum of addressable main memory in 32 bit systems is 4 GB it is suggested to switch to a 64 bit architecture for allowing more flexibility in terms of computational resources. The compensation framework should be extended to support multiple projectors. Also, a more detailed analysis of the matrix decomposition's numerical stability should be performed.

Another possible extension is the support of view-dependency. As described by Bimber et al. [9], multiple source cameras could be used along with image-based rendering and interpolation techniques to support moving users. This leads, eventually, to a light field rendering approach. If the light transport is well-known for multiple cameras, rays in between can be synthesized and rendered in real-time (see Levoy and Harahan [34]). In order to be able to compensate view-dependent global illumination effects such as reflections and refractions lots of source cameras are necessary. However, the light transport from a single projector to multiple cameras can be acquired simultaneously. Hence, overall acquisition time would not be increased a lot.

The employment of special canvases such as lenticular sheets allows autostereoscopic projections. Using the described technique and the suggested extension of compensating for multiple cameras simultaneously (see section 4.1) would allow to automatically generate corrected imagery for multiple fixed viewpoints. This enables a fully automated registration of such surfaces with a projector.

One of the main challenges of this work was the accurate light transport acquisition. This is only determined by a single threshold, setting this too high discards low intensity effects, setting it too low results in

an enormous amount of images that have to be captured. Peers and Dutré [50] proposed a novel approach to projected structured light patterns for environment matting based on wavelets. Any image can be decomposed into wavelet basis functions. This is a standard procedure often applied to image compression. The projector space (in their case monitor space) is decomposed into these basis functions, which are projected onto the scene. The order of the projected basis functions is chosen, so that those contributing more to the final composition are projected first. Thus, whenever the procedure is stopped, the most important contributions are already captured. For the matting application, the monitor image is replaced by a photograph, simulating an illumination of the captured scene. The desired image is decomposed into its wavelet bases and the appropriate previously captured monitor wavelets are simply combined and displayed. However, the radiometric compensation application requires an inverse rendering since the desired image is given in camera, not in projector space. Hence, the proposed method is not applicable in this way, but the idea to encode some form of basis functions in the projected patterns is quite inspiring. The approach would have to be extended to four dimensions, so that a mapping between a basis function in camera space to a basis function in projector space is given. Such or similar techniques might increase light transport acquisition time.

Yet another idea is to reconstruct information about geometrical structures and BRDFs of the scene. This could, for instance, be done by analyzing the light transport matrix. One idea for segmenting objects within the matrix is to use common spectral matrix analysis and clustering methods. However, the gigantic size of the matrix should be considered. Estimating the geometry and BRDF of objects within the scene might allow a rerendering, thus radiometric compensation, from different perspectives. This is more or less a return to traditional projection technology, where projectors, cameras and a well-known scene are registered and corrected using projective texture mapping. In these applications global illumination effects are not considered yet.

APPENDIX A: PSEUDO CODE FOR CLUSTER DECOMPOSITION

```
for each camera pixel k do
    // group the connected projector pixels into groups of spatially neighboring blocks
    ComputeNeighborhoodBlocks(k);
    for each neighborhood block b do
        // remove all connections that fall below a threshold
        RemoveLowerConnections(b);
    end for
end for

for each projector pixel p do
    // do not allow single projector pixel to be cut-off the graph
    if p is cut-off then
        // choose the highest connections of p according to a threshold and reinsert them
        ReinsertHigherConnections();
    end if
end for

function ComputeNeighborhoodBlocks(camera pixel k)
    while k has connections do
        // search k's highest weight in the connections to projector pixels
        w = FindHighestWeight(k);
        // add w as center of a new neighborhood
        currentBlock = CreateNewBlockNeighborhood(w);
        // find all connected projector pixels that fall into w's neighborhood
        for each connected projector pixel p of k do
            if Distance(w,p) ≤ neighborhoodRadius then
                // add p to w's block
                AddToBlock(p, currentBlock);
            end if
        end for
        // remove all projector pixels of w's neighborhood from k's connections
        RemoveBlockEntryConnections(k,currentBlock);
    end while
end function

function RemoveLowerConnections(neighborhoodBlock b)
    // use highest weighted connection in b as reference
    highestLuminance = FindHighestConnectionInBlock(b);
```

```
    for each connection c in b do
        luminance = ComputeLuminance(c);
        if luminance ≤ τ1*highestLuminance then
            RemoveConnectionFromBlock(c, b);
        end if
    end for
end function

function ReinsertHigherConnections(projector pixel p)
    // use highest weighted connection of p as reference
    highestLuminance = FindHighestConnectionOfPixel(p);
    for each original connection c of p do
        luminance = ComputeLuminance(c);
        if luminance ≥ τ2*highestLuminance then
            ReinsertConnectionToGraph(c);
        end if
    end for
end function
```

REFERENCES

[1] Colour and Vision Research Laboratories, Institue of Ophthalmology. http://www.cvrl.org/, 2005. iii, 13

[2] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999. 43, 55

[3] Mark Ashdown, Takahiro Okabe, Imari Sato, and Yoichi Sato. Robust Content-Dependent Photometric Projector Compensation. In *IEEE International Workshop on Projector-Camera Systems (PROCAMS 2006)*, 2006. 7

[4] Oliver Bimber, Franz Coriand, Alexander Kleppe, Erich Bruns, Stefanie Zollmann, and Tobias Langlotz. Superimposing Pictorial Artwork with Projected Imagery. In *IEEE MultiMedia*, pages 16–26, January-March issue 2005. 1, 5, 6

[5] Oliver Bimber and Andreas Emmerling. Multi-Focal Projection: A Multi-Projector Technique for Increasing Focal Depth. In *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, volume 12, 2006. 7, 50

[6] Oliver Bimber, Andreas Emmerling, and Thomas Klemmer. Embedded Entertainment with Smart Projectors. In *IEEE Computer*, pages 56–63, 2005. 1, 5, 6, 61

[7] Oliver Bimber, Anselm Grundhöfer, Gordon Wetzstein, and Sebastian Knödel. Consistent illumination within optical see-through augmented environments. In *ISMAR '03: Proceedings of the The 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality*, page 198, Washington, DC, USA, 2003. IEEE Computer Society. 9

[8] Oliver Bimber, Anselm Grundhöfer, Thomas Zeidler, Daniel Danch, and Pedro Kapakos. Compensating Indirect Scattering for Immersive and Semi-Immersive Projection Displays. In *IEEE Virtual Reality (IEEE VR'06)*, 2006. 10

[9] Oliver Bimber, Gordon Wetzstein, Andreas Emmerling, and Christian Nitschke. Enabling View-Dependent Stereoscopic Projection in Real Environments. In *International Symposium on Mixed and Augmented Reality (ISMAR'05)*, pages 14–23, 2005. 1, 7, 67

[10] James F. Blinn and Martin E. Newell. Texture and reflection in computer generated images. *Commun. ACM*, 19(10):542–547, 1976. 8

[11] Jeff Bolz, Ian Farmer, Eitan Grinspun, and Peter Schröder. Sparse Matrix Solvers on the GPU: Conjugate Gradients and Multigrid. *ACM Trans. Graph.*, 22(3):917–924, 2003. 19

[12] Yuqun Chen, D. W. Clark, Adam Finkelstein, T. C. Housel, and Kai Li. Automatic Alignment of High-Resolution Multi-Projector Displays Using an Uncalibrated Camera. In *IEEE Visualization 2000*, pages 125–130, October 2000. 3

[13] Carolina Cruz-Neira, Daniel J. Sandin, and Thomas A. DeFanti. Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE. In *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 135–142, New York, NY, USA, 1993. ACM Press. 3

[14] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the Reflectance Field of a Human Face. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 145–156, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. 1, 8, 10, 21

[15] Paul Debevec, Andreas Wenger, Chris Tchou, Andrew Gardner, Jamie Waese, and Tim Hawkins. A Lighting Reproduction Approach to Live-Action Compositing. *ACM Transactions on Graphics*, 21(3):547–556, July 2002. 9

[16] Paul E. Debevec and Jitendra Malik. Recovering High Dynamic Range Radiance Maps from Photographs. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 369–378, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co. 16, 17, 20

[17] K. Fujii, M.D. Grossberg, and S.K. Nayar. A Projector-Camera System with Real-Time Photometric Adaptation for Dynamic Environments. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 814–821, Jun 2005. 1, 5, 6

[18] James Fung. Computer Vision on the GPU. In *GPU Gems 2, chapter 40*, pages 547–556. Addison Wesley, 2005. 19

[19] James Fung, Steve Mann, and Chris Aimone. OpenVIDIA: Parallel GPU Computer Vision. *Proceedings of the ACM Multimedia*, pages 849–852, 2005. 19

[20] Gaurav Garg, Eino-Ville Tavala, Marc Levoy, and Hendrik P. A. Lensch. Symmetric Photography: Exploiting Data-sparseness in Reflectance Fields. *Eurographics Symposium on Rendering (ESGR)*, 2006. 1, 9

[21] Abhijeet Ghosh, Matthew Trentacoste, Helge Seetzen, and Wolfgang Heidrich. Real Illumination from Virtual Environments. *Eurographics Symposium on Rendering*, pages 243–252, 2005. 9

[22] Michael Goesele, Hendrik P. A. Lensch, Jochen Lang, Christian Fuchs, and Hans-Peter Seidel. DISCO: Acquisition of Translucent Objects. *ACM Trans. Graph.*, 23(3):835–844, 2004. 8

[23] M.D. Grossberg, H. Peri, S.K. Nayar, and P.N. Belhumeur. Making One Object Look Like Another: Controlling Appearance using a Projector-Camera System. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume I, pages 452–459, Jun 2004. 1, 5, 6

[24] Michael D. Grossberg and Shree K. Nayar. High Dynamic Range from Multiple Images: Which Exposures to Combine? In *ICCV Workshop on Color and Photometric Methods in Computer Vision (CPMCV)*, 2003. 16

[25] Anselm Grundhöfer and Oliver Bimber. Real-Time Adaptive Radiometric Compensation. In *SIGGRAPH Poster*, 2006. 7

[26] Pat Hanrahan. The rendering equation. *Lecture Slides for CS348B: Image Synthesis Techniques*, 2000. 22

[27] S. Inokuchi, K. Sato, and F. Matsuda. Range Imaging System for 3D Object Recognition. pages 806–808, 1984. 6

[28] Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High Dynamic Range Video. *ACM Trans. Graph.*, 22(3):319–325, 2003. 16

[29] Craig Kolb, Don Mitchell, and Pat Hanrahan. A Realistic Camera Model for Computer Graphics. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 317–324, New York, NY, USA, 1995. ACM Press. 15

[30] Jens Krüger and Rüdiger Westermann. Linear Algebra Operators for GPU Implementation of Numerical Algorithms. *ACM Trans. Graph.*, 22(3):908–916, 2003. 19

[31] C L Lawson and R J Henson. *Solving Least Square Problems*. Prentice Hall, Englewood Cliffs, NJ, 1974. 44

[32] Hendrik Lensch, Michael Goesele, Philippe Bekaert, Jan Kautz, Marcus Magnor, Jochen Lang, and Hans-Peter Seidel. Interactive Rendering of Translucent Objects. *Computer Graphics Forum*, 22(2):195–205, 2003. 8

[33] Marc Levoy, Billy Chen, Vaibhav Vaish, Mark Horowitz, Ian McDowall, and Mark Bolas. Synthetic Aperture Confocal Imaging. *ACM Trans. Graph.*, 23(3):825–834, 2004. 7

[34] Marc Levoy and Pat Hanrahan. Light Field Rendering. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, New York, NY, USA, 1996. ACM Press. 10, 67

[35] Industrial Light and Magic. OpenEXR High Dynamic Range Image Format., 2005. 20

[36] A. Majumder and R. Stevens. LAM: Luminance Attenuation Map for Photometric Uniformity in Projection Based Displays, 2002. presented at Virtual Reality and Software Technology, Hong Kong, China. 3, 4

[37] Aditi Majumder. Properties of Color Variation Across Multi-Projector Displays. In *SID Eurodisplay*, 2002. 3

[38] Aditi Majumder and M. Gopi. Modeling Color Properties of Tiled Displays. In *Computer Graphics Forum*, 2005. 3

[39] Aditi Majumder, Zhu He, Herman Towles, and Greg Welch. Achieving Color Uniformity Across Multi-Projector Displays. In T. Ertl, B. Hamann, and A. Varshney, editors, *Proceedings Visualization 2000*, pages 117–124, 2000. 3

[40] Aditi Majumder and Rick Stevens. Color Non-Uniformity in Projection Based Displays: Analysis and Solutions. *IEEE Transactions on Visualization and Computer Graphics*, 10(2), 2004. 3

[41] Aditi Majumder and Rick Stevens. Perceptual Photometric Seamlessness in Tiled Projection-Based Displays. *ACM Transactions on Graphics*, 24(1), 2005. 3

[42] S. Mann and R. Picard. Being 'undigital' with Digital Cameras: Extending Dynamic Range by Combining Differently Exposed Pictures. *In IS&T's 48th Annual Conference*, pages 422–428, 1995. 16

[43] Vincent Masselus, Pieter Peers, Philip Dutré;, and Yves D. Willems. Relighting with 4D Incident Light Fields. *ACM Trans. Graph.*, 22(3):613–620, 2003. 1, 9, 25

[44] T. Mitsunaga and S. K. Nayar. Radiometric Self Calibration. *Proceedings of CVPR*, pages 374–380, 1999. 16, 17

[45] E H. Moore. On the Reciprocal of the General Algebraic Matrix. In *Bulletin of the American Mathematical Society*, volume 26, pages 394–395, 1920. 54

[46] Shree K. Nayar, Harish Peri, Michael D. Grossberg, and Peter N. Belhumeur. A Projection System with Radiometric Compensation for Screen Imperfections. In *International Workshop on Projector-Camera Systems (ICCV)*, 2003. 1, 5, 6, 30

[47] S.K. Nayar and T. Mitsunaga. High Dynamic Range Imaging: Spatially Varying Pixel Exposures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 472–479, Jun 2000. 16

[48] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis. Geometric Considerations and Nomenclature for Reflectance. National Bureau of Standards (US), 1977. 14

[49] Hanhoon Park, Moon-Hyun Lee, Sang-Jun Kim, and Jong-Il Park. Specularity-free projection on nonplanar surface. In *Proceedings of Pacific-Rim Conference on Multimedia (PCM)*, pages 606–616, 2005. 7, 50

[50] Pieter Peers and Philip Dutré. Wavelet Environment Matting. In *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, pages 157–166, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association. 1, 68

[51] Pieter Peers, Karl vom Berge, Wojciech Matusik, Ravi Ramamoorthi, Jason Lawrence, Szymon Rusinkiewicz, and Philip Dutré. A Compact Factored Representation of Heterogeneous Subsurface Scattering. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 25(3), July 2006. 8

[52] R Penrose. A Generalized Inverse for Matrices. In *Cambridge Philosophy Society*, volume 51, pages 406–413, 1955. 54

[53] J. L. Posdamer and M. D. Altschuler. Surface Measurement by Space-Encoded Projected Beam Systems. *Computer Graphics and Image Processing*, 18(1):1–17, 1982. 6

[54] Andrew Raij, Gennette Gill, Aditi Majumder, Herman Towles, and Henry Fuchs. PixelFlex2: A Comprehensive, Automatic, Casually-Aligned Multi-Projector Display. In *IEEE International Workshop on Projector-Camera Systems (PROCAMS-2003), Nice, France*, 2003. 3, 4

[55] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. iLamps: Geometrically Aware and Self-configuring Projectors. In *ACM SIGGRAPH*, 2003. 3

[56] Ramesh Raskar, Michael S. Brown, Ruigang Yang, Wei-Chao Chen, Greg Welch, Herman Towles, Brent Seales, and Henry Fuchs. Multi-Projector Displays Using Camera-Based Registration. In *VIS '99: Proceedings of the conference on Visualization '99*, pages 161–168, Los Alamitos, CA, USA, 1999. IEEE Computer Society Press. 3

[57] Ramesh Raskar, Greg Welch, Matt Cutts, Adam Lake, Lev Stesin, and Henry Fuchs. The Office of the Future: A Unified Approach to Image-Based Modeling and Spatially Immersive Displays. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 179–188, New York, NY, USA, 1998. ACM Press. 5

[58] Ramesh Raskar, Greg Welch, Kok-Lim Low, and Deepak Bandyopadhyay. Shader Lamps: Animating Real Objects With Image-Based Illumination. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, pages 89–102, London, UK, 2001. Springer-Verlag. 5

[59] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec. *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting*. Morgan Kaufmann Publishers, December 2005. v, 12, 14

[60] Mark A. Robertson, Sean Borman, and Robert L. Stephenson. Estimation-Theoretic Approach to Dynamic Range Enhancement Using Multiple Exposures. *Journal of Electronic Imaging*, 12(2):219–228, April 2003. 16, 17

[61] J. Salvi, J. Pages, and J. Batlle. Pattern Codification Strategies in Structured Light Systems, 2004. 6

[62] Steven M. Seitz, Yasuyuki Matsushita, and Kiriakos N. Kutulakos. A Theory of Inverse Light Transport. In *ICCV*, pages 1440–1447, 2005. 1, 10

[63] Pradeep Sen, Billy Chen, Gaurav Garg, Stephen R. Marschner, Mark Horowitz, Marc Levoy, and Hendrik P. A. Lensch. Dual Photography. *ACM Trans. Graph.*, 24(3):745–755, 2005. 1, 8, 9, 25, 26

[64] Edward H. Stupp and Matthew S. Brennesholtz. *Projection Displays*. John Wiley and Sons, 1999. 14

[65] Greg Ward. Real Pixels. In James Arvo, editor, *Graphics Gems II*, pages 80–83. Academic Press, 1991. 16

[66] Andreas Wenger, Andrew Gardner, Chris Tchou, Jonas Unger, Tim Hawkins, and Paul Debevec. Performance Relighting and Reflectance Transformation with Time-Multiplexed Illumination. *ACM Trans. Graph.*, 24(3):756–764, 2005. 9

[67] R. Yang, D. Gotz, J. Hensley, H. Towles, and M. Brown. PixelFlex: A Reconfigurable Multi-Projector Display System. In *IEEE Visualization 2001*, 2001. 3

[68] Ruigang Yang, Aditi Majumder, and Michael Brown. Camera Based Calibration Techniques for Seamless Multi-Projector Displays. *IEEE Transactions on Visualization and Computer Graphics*, 11(2), 2005. 3

[69] Yizhou Yu, Paul Debevec, Jitendra Malik, and Tim Hawkins. Inverse Global Illumination: Recovering Reflectance Models of Real Scenes from Photographs. In *SIGGRAPH 99*, pages 215–224, 1999. 10

[70] L. Zhang and S. K. Nayar. Projection Defocus Analysis for Scene Capture and Image Display. *ACM Trans. on Graphics (also Proc. of ACM SIGGRAPH)*, Jul 2006. 8, 50, 52

[71] Douglas E. Zongker, Dawn M. Werner, Brian Curless, and David H. Salesin. Environment Matting and Compositing. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 205–214, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. 1, 9