Active illumination for high speed image acquisition and recovery of shape and albedo

Matis Hudon

ANNÉE 2016

UNIVERSITÉ DE RENNES 1

UNIVERSITE BRETAGNE LOIRE

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Bretagne Loire

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : informatique

Ecole doctorale MATISSE

présentée par

Matis Hudon

préparée à l'unité de recherche UMR 6074 IRISA
Institut de Recherche en Informatique et Systèmes Aléatoire
ISTIC

Active Illumination for High Speed Image Acquisition and Recovery of Shape and Albedo

Thèse soutenue à Rennes
le 13 octobre 2016

devant le jury composé de :

Jean-Michel DISCHLER
Professeur, Univ. de Strasbourg / rapporteur

Daniel MENEVEAUX
Professeur, Univ. de Poitiers / rapporteur

Julie DIGNE
Chercheur CNRS, Univ. de Lyon / examinateur

Philippe ROBERT
Ingénieur de Recherche, Technicolor / examinateur

Rémi COZOT
Maître de Conférences, Univ. de Rennes 1 / co-directeur de thèse

Kadi BOUATOUCH
Professeur, Univ. de Rennes 1 / directeur de thèse

Abstract

The objective of this thesis is to take advantage of controlled illumination to enrich a video acquisition with shape and reflectance reconstructions. Today, a lot of works have tried to meet this objective. Some of them take advantage of sequential controlled illumintation to recover high quality shape and reflectance, however they either require a costly and very cumbersome fixed setup, and/or do not run in real-time. Our aim is a low cost, fast, mobile and simple acquisition setup which has to be the less intrusive possible so as to provide a greater ease of use.

The first contribution of this thesis focuses on the applications of the well known photometric stereo method to a video acquisition. Moreover, as a high frame rate is required by such an application, a method using sequential illumination with high frame rate cameras (electronic rolling shutter cameras) is also considered. Despite the interesting results provided by photometric stereo, we found that this latter did not provide enough qualitative results. Moreover, by its nature, photometric stereo is not really suitable for the range of applications targeted.

We propose, as a second contribution, a method for recovering the shape (geometry) and the diffuse reflectance from an image (or video) using a hybrid setup consisting of a depth sensor (Kinect), a consumer camera and a partially controlled illumination (using a flash). The objective is to show how combining RGB-D acquisition with a sequential illumination is useful for shape and reflectance recovery. A pair of two images are captured: one non flashed (image under ambient illumination) and a flashed one. A pure flash image is computed by subtracting the non flashed image from the flashed image. We propose a novel and near real-time algorithm, based on a local illumination model of our flash and the pure flash image, to enhance geometry (from the noisy depth map) and recover reflectance information.

Finally, our last contribution concerns an automatic method for light compositing, using rendered images. Lighting is a key element in photography. Professional photographers often work with complex lighting setups to directly capture an image close to the targeted one. Some photographers reversed this traditional workflow. Indeed, they capture the scene under several lighting conditions, then combine the captured images to get the expected one. Acquiring such a set of images is a tedious task and combining them requires some skill in photography. We propose a fully automatic method, that renders, based on a 3D reconstructed model (shape and albedo), a set of images corresponding to several lighting conditions. The resulting images are combined using a genetic optimization algorithm to match the desired lighting provided by the user as an image.

Remerciements

Commençons par le commencement, je souhaiterais remercier l'équipe WP1 de Technicolor sans qui cette thèse n'aurai jamais vu le jour, et avec qui j'ai énormément appris. Un remerciement tout particulier à Arno Schubert qui m'a toujours encouragé, même dans les moments difficiles.

Viens ensuite le tour de l'équipe FRVSENSE, merci à vous, et notamment mes deux directeurs de thèse Rémi Cozot et Kadi Bouatouch. Kadi m'a énormément appris et aidé à développer mes compétences dans de nombreux domaines. Aussi ses encouragements et sa confiance en mes capacités m'ont aidé à appréhender cette thèse avec plus de sérénité. Rémi a eu les bons mots aux bons moments, de plus ses idées ont toujours été fructueuses.

Je voudrais aussi remercier les membres du jury qui ont accepté de juger mon travail et mon manuscrit. Je remercie donc Jean-Michel Dischler et Daniel Meneveaux d'avoir accepté d'être rapporteur de mon manuscrit, et Julie Digne d'avoir accepté d'être examinateur. Merci pour vos commentaires constructifs sur mon travail et pour vos conseils.

Je me dois de citer ici une autre personne : Adrien Gruson. Adrien a été pendant 3 ans un collègue/ami/encadrant/exemple. Autre chose, je ne t'en ai jamais voulu de ne pas être au niveau à Sc2. Je me rappellerai toujours de notre travail de cohésion d'équipe avec Billal (Merci à toi Billal, pour tous ces bons moments).

Je voudrais aussi remercier mes collègues et différentes équipes de Technicolor R& I pour ces trois années (Et pour le Ski..). Tout particulièrement, merci à Philippe et Paul pour leurs encadrements et leurs conseils.

Merci à tous les copains de Rennes, je ne pourrais pas citer tout le monde, mais merci d'abord à Ronan de m'avoir amené à l'Amaryllis puis à Adrien de m'y avoir gardé. Merci à Fabien, Fabien, Fanny, Martin, Merwan, Darya, Mehmet pour tout ce que l'on a partagé ensemble, notamment les marchés fous. Merci comme jamais à Papy, Chloé, Lambert et Jidé. Merci tout spécial à Flash pour toute la brasse et le surf. Merci à tous mes amis de Bordeaux et de Paris.

Je vais terminer en remerciant ma famille pour leur soutien depuis le début de mes études.

Enfin merci à toi, Marine.

Contents

List of figures 7

List of tables 11

1 Introduction 13
- 1 Context . . . 13
  - 1.1 Industrial and commercial context . . . 13
  - 1.2 Technological context . . . 14
- 2 Motivation . . . 15
- 3 Structure of the thesis . . . 15
- 4 Publications . . . 16
- 5 Patents . . . 16

2 Related Works 19
- 1 About reflectance measurement . . . 19
  - 1.1 Direct surface reflectance measurement . . . 19
- 2 Scene reconstruction (through active lighting) . . . 21
  - 2.1 Photometric stereo on still images . . . 21
  - 2.2 Photometric stereo for video sequences . . . 23
  - 2.3 Shape from Shading (SfS) . . . 25
  - 2.4 Depth Enhancement . . . 26
  - 2.5 Discussion . . . 28

3 Rolling Flash and Photometric Stereo 29
- 1 Introduction and Objectives . . . 29
- 2 Electronic rolling shutter cameras . . . 30
- 3 Related works . . . 33
  - 3.1 Computer vision and Electronic rolling shutter cameras. . . . 34
  - 3.2 Stroboscopic Illumination in computer vision. . . . 35
  - 3.3 Electronic shutters and stroboscopic illumination. . . . 35
  - 3.4 Discussion . . . 36
- 4 The Rolling Flash . . . 36
  - 4.1 Avoiding intra-rows artifacts . . . 36
  - 4.2 Reconstructing temporal-coherent rows . . . 38
  - 4.3 Flash illuminating 3 frames . . . 39
- 5 Framework and algorithm . . . 40

5.1 Framework . . . 40
5.2 Reconstruction Algorithm . . . 41
5.3 Sequential recording . . . 42
6 Experiment results . . . 44
6.1 Straightforward application to Photometric Stereo . . . 44
7 Conclusion . . . 46

4 Shape and Reflectance from RGB-D images using time sequential illumination 47
1 Introduction . . . 47
2 General Idea . . . 49
2.1 Light Source Modeling. . . . 50
2.2 Scene Illumination. . . . 50
2.3 Pure flash image from image pairs. . . . 51
3 Our Approach . . . 52
3.1 Chromaticity Assumption . . . 53
3.2 Computing Normal Map from Quantified Depth Data . . . 53
3.3 Normal Map Filtering . . . 54
3.4 Diffuse Reflectance Coefficients Estimation and Filtering . . . 54
3.5 Normal Map Refinement . . . 55
3.6 Global Convergence . . . 56
4 Results . . . 57
4.1 Performance improvement . . . 59
4.2 Application to direct relighting . . . 60
5 Conclusion . . . 61

5 Automatic Light Compositing using Rendered Images 63
1 Introduction . . . 63
2 Overview . . . 64
2.1 RGB-D Acquisition and refinement . . . 65
2.2 Rendering . . . 66
2.3 Genetic Algorithm . . . 66
3 Results . . . 67
4 Conclusion . . . 69

6 Conclusion 71
1 Conclusion . . . 71
2 Future Work . . . 72
2.1 Rolling flash and HDR acquisition . . . 72
2.2 Shape and reflectance from RGB-D images using time sequential illumination . . . 72
2.3 Automatic light compositing using rendered images . . . 73

Résumé en Français 75
3 Introduction . . . 75
3.1 Contexte . . . 75
3.2 Motivation . . . 76

4 Sommaire des Contributions 76
4.1 "Rolling flash" et photométrie stéréo 76
4.2 Reconstruction de la forme et de l'apparence à partir d'images RGB-D, et d'une illumination séquentielle. 77
4.3 Composition automatique d'éclairages à partir d'images de synthèse 78

Bibliography **84**

List of Figures

1.1 Different examples of AR/MR application, demonstrating the huge range of field concerned. Where AR and MR stand for Augmented reality and mixed reality respectively. 14

2.1 Murray-Coleman and smith gonioreflectometer, figure from [War92] . . . 20
2.2 Gonioreflectometer by [War92] 20
2.3 Illustration of the three fundamental properties used by [HMI10], figure from [HMI10] 21
2.4 Schematic illustration of the appearance model from [GCHS10]. A normal map (a) is used to remap the fundamental materials (b), and results are modulated by material map (c) and summed together (d). 22
2.5 Pictures of the acquisition system presented in [TFG+13], pictures taken from [TFG+13] 23
2.6 Results taken from [JK07]: One of eight views of a figurine rotating it in front of a camera and light (top left). The intermediate dense depth map recovered by the method – red is closer and blue further from the camera (top right) and renderings of the final surface (bottom row). . . 24
2.7 Example of input image (left) and corresponding 3D reconstruction (right) from [HVB+07], images taken from the project webpage. 24
2.8 Results from [KWBE10], from the left to the right: input frame acquired with the acquisition setup, result geometry, rendered geometry from another view, rendered with estimated albedo from camera view. 25
2.9 Results from [WGT+05a], from the left to the right: input frame acquired with the acquisition setup, result Normal Map, albedo, ambient occlusions. 25
2.10 Results from [NRDR05], from the left to the right: Rendering of 3D scanned range image, same scanned geometry, augmented with a measured normal-map (from photometric stereo), hybrid surface reconstruction, which combines both position and normal constraints, photograph. . 26
2.11 Results from [RSD+12], from the left to the right: Input depth map, spatial filtering and spatiotemporal filtering. 26
2.12 Results from [WZN+14], from the left to the right: RGB aligned image, input raw depth map, refined depth map. 27
2.13 Results from [OERW+15], from the left to the right: RGB aligned image, input raw depth map, bilateral filtered depth, refined depth map. 27

3.1 Our goal application, simple PS applied to video sequences. 30
3.2 Mechanical shutter functioning: a red square represents the reset of a row while a green square represents a reading time. 31

3.3 Partial lighting example: typically happens while filming lightning. Lightning occurs while recording a video with an electronic rolling shutter. As the rows of the sensor integrate light sequentially, the illumination from lightning is only sensed by one part of the rows (here bottom rows). As a result, the bottom rows of the image appear brighter than the upper ones. 32
3.4 Description of a CMOS electronic rolling shutter operation. Rows are read sequentially through vertical signal lines, then the output signal is extracted through horizontal signal lines. 33
3.5 Electronic shutter functioning: a red square represents the reset of a row while a green squares represents a reading time. (a) For a 360° shutter value and (b) for a 180° shutter value. 34
3.6 **(a)** Electronic shutter with periodic flashes, the two types of artifacts are shown in this schema. **(b)** Typical example of the two artifacts put forward when filming a rolling fan. 34
3.7 Illustration of the rolling flash method: our method allows to use flashes with electronic rolling shutter camera, fixing the stroboscopic period of flash to the sum of the exposure time Δ_e and the flash duration Δ_f. . . 37
3.8 Typical raw sequence taken under a rolling flash illumination. We can see that the darker rows rolling over the images. 38
3.9 Detail of rolling flash method: the same flash instant illuminates two camera frames, the rows in grey contain all the information needed to reconstruct a coherent image. 38
3.10 Example of reconstructed image: Image n is reconstructed from frame n and frame $n+1$. 39
3.11 Flash illuminating three frames. 40
3.12 Overall architecture of the control device. 41
3.13 Computing the index k of the minimal luminance energy row for the current and subsequent images. k is determined within the range $[k_{est} - \epsilon, k_{est} + \epsilon]$. For our experiments we chose $\epsilon = \frac{h}{100}$, h being the height of a frame. 43
3.14 An illustration of sequential recording: the original sequence is shot with a periodically varying illumination. Each of the blue, red and green frames represent one illumination in the original sequence. After extraction we obtain 3 sequences of the same scene, each sequence with a proper illumination. 43
3.15 Rolling flash and sequential recording. 44
3.16 **(a):** Two consecutive input frames from our experiments, **(b):** Linear version of the two input frames. **(c):** Our reconstruction result with gamma correction 45
3.17 Application of Photometric stereo to one frame. From right to left: Reconstructed images from our rolling flash method corresponding to the three illumination conditions, the reconstructed Albedo and the reconstructed Normal Map. 45
3.18 Example of 180° relighting of one frame. 46

4.1 Blue: Our method takes as input a flash image registered with a raw depth map (rendered with Normals and diffuse shading). The flash image is computed using flashed and non flashed image pairs which represent two successive video frames. With these inputs our algorithm use an optimization process to produce refined normal map (rendered with diffuse shading) and reflectance map. 48
4.2 Illustration of flash and non flashed sequential recording and the extraction of a pure flashed image. 49
4.3 Blue: Spectrum of a white Lambertian point under the general uncontrolled lighting. Red: Spectrum of a white Lambertian point under unknown general uncontrolled lighting and flash illumination (total spectrum). Green: Spectrum of a white Lambertian point under pure flash illumination. Purple: difference between the total spectrum and the general uncontrolled lighting spectrum, this difference spectrum completely matches the pure flash spectrum. 51
4.4 Our framework picture. Green boxes represent the different processings of our algorithm. Orange boxes represent the values given by the sensors at different times. Red box is the result of our algorithm. 52
4.5 The chromaticity image is used to cluster surfaces with similar diffuse reflectances. We can observe that the t-shirt, the background and the skin are classified into different clusters. 53
4.6 Plots representing the convergence of Normals and diffuse reflectance coefficients for each iteration 56
4.7 Our experimental setup, from top to bottom: white flash LED light, Ueye Industrial Camera and Kinect depth sensor. 57
4.8 The original burger scene used in [OERW+15]. Top row shows the dot product image (dot product between the normal and the view direction) using the normals of: the reference solution, the noisy map used as input, [OERW+15] and our method. Bottom row shows the RGB image and the false color error on the normal dot product. For better visualisation the error was multiplied by 3. 58
4.9 T-Shirt scene captured by our setup. From left to right : the normal maps corresponding respectively to the raw input depth map, the [OERW+15] processing and our refinement method. The fourth image is the diffuse reflectance coefficients map obtained with our method. 58
4.10 On the left: the refined diffuse reflectance coefficients map after convergence of our algorithm. On the right: false color image to show how the chromaticity-based clustering performs. We can observe that the tomatoes and the bread lie in the same cluster. However, thanks to the local nature of our filtering, these two materials have not been merged. . . . 59
4.11 The normal and diffuse reflectance coefficients maps refined by our algorithm can be used for relighting a scene. Images Relighting 1 & 2 are obtained with different artificial light source positions. Moreover, sequential lighting makes our technique capable of capturing video sequences. However, fast and large movements in the video could create artifacts due to motion blur. 60

5.1 Main Framework 65
5.2 Pipeline of the genetic algorithm. 66
5.3 Typical convergence curve for the original set $\mathbb{S}$ Fig. 5.4 67
5.4 Original real set $\mathbb{S}$ of 12 images 68
5.5 Green: Luminance Histogram of the target image, Blue: Luminance Histogram of the best candidate after 100 generations. 68
5.6 Left: Target Image, Right: Result image after 100 iterations 69
5.7 **Top**, *from left to right*: target image created from a real set of images, result after 20 iterations using a real set of images, result after 20 iterations using a rendered set of images; **Bottom**, *from left to right*: a given target image (independent of the scene), result after 20 iterations using a real set of images, result after 20 iterations using a rendered set of images 69
5.8 Left: Target Image, Right: Result image after 20 iterations 70

6.1 Differents exemples d'applications AR/MR. 76
6.2 Application de la stéréo photométrie pour la vidéo. 77
6.3 Résultats de notre méthode hybride de reconstruction. 77
6.4 Résultats de notre méthode de composition automatique d'éclairage. . . 78

List of Tables

4.1 Computation times of each step of our algorithm. Bold timings are obtained using the performance improvement described in subsection 4.1. To compute the total time, we need to multiply the refinement iteration time by the number of iterations needed to converge to the desired results. In practice for a 1280×960 image, only 5 iterations are required, which corresponds to 12.88 fps or 0.91 fps on a GTX 980 respectively using or not using the performance improvement. For a 640×480 image (kinect size), we reach 73.8 fps and 3.58 respectively using or not using the performance improvement. 60

Introduction 1

Contents

1 Context **13**
1.1 Industrial and commercial context 13
1.2 Technological context 14
2 Motivation **15**
3 Structure of the thesis **15**
4 Publications **16**
5 Patents **16**

Since the 90's, the word "virtual" has been associated with all computer or internet contents: a user experience related to a digital world as opposed to the real world. The digital world is fully controlled and represents a great tool and interface for mankind to create, project and publicize virtual contents. In computer graphics, scientists try to mimic the real world with physically-based models, in order to create the closest possible illusion to the real world. When illusion meets up with reality, we use the word realistic.

1 Context

The main concern of this thesis is the digitization of scenes containing various types of objects with different reflectances and shapes. We aim at enriching the acquisition of a video sequence by providing additional information about the scene, such as reflectance and shape reconstructed in real-time.

1.1 Industrial and commercial context

Definition

Mixed reality (MR): sometimes referred to as hybrid reality [eS09], is the merging of real and virtual worlds to produce new environments and visualizations where physical and digital objects co-exist and interact one to another in real-time. Mixed reality takes place not only in the physical world or the virtual world, but is a mix of reality and virtual reality, encompassing both augmented reality and augmented virtuality [MK94].

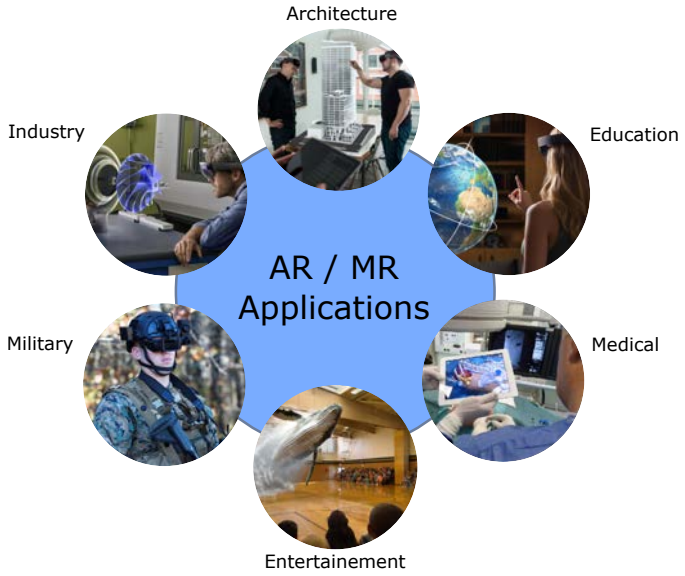

Figure 1.1 – Different examples of AR/MR application, demonstrating the huge range of field concerned. Where AR and MR stand for Augmented reality and mixed reality respectively.

There is no denying that mixed and virtual reality are the next big industrial challenges. Recent reports predict that within 10 years virtual and mixed reality hardware will be an $80 billion industry. There is a very long list of possible application domains for mixed reality that surely motivate both researchers and investors (see Fig.1.1), such as: medecine, education, entertainment, gaming, military, industry and architecture, to name a few. All those applications aim to add/project interactive virtual contents in a real and unknown environment, which suppose to be able to reconstruct a 3D representation of the scene in interactive or real-time. User experience can be improved even more, if the integration of a virtual content in a real environment looks realistic, which implies a realistic play of light between real objects and virtual contents. This is why there is also a need for reconstruction of the shape and the reflectance of real objects from video sequences.

1.2 Technological context

This subsection describe briefly the contextual technological progress.

Cameras. Digital cameras have improved a lot in the last two decades, and are now capable of producing highly detailed quality images of scenes . Furthermore, camera frame-rates are rising year after year, thanks to electronic rolling shutter sensors and CMOS technology.

Illumination. The recent development of powerful white Light-Emitting Diodes allowed researchers to design new acquisition setups based on active controlled illuminations. As an example the work presented in [Deb12a], shows how active illumination can be used to recover shape and reflectance per image, that are used for various types of artistic effects.

Consumer depth devices and scanners. An important contextual techno-

logical progress is also the emergence of consumer depth sensor and scanners capable of supplying in real-time rough depth information about the environment. Given their low price and small sizes, they open a wide area of research in computer vision. However those sensors still have a low resolution and are too noisy to directly digitize and reconstruct high quality 3D models of a scene.

A direct consequence of technological progress is the newly emerging market of more and more powerful wearable devices that are opening a wide field of research concerning virtual, augmented and mixed reality. In order to mix the real and virtual worlds in real-time, geometry (shape) and reflectance real-time reconstruction of a real scene is needed. Once this reconstruction is performed, it is possible to create interactions between real contents and virtual ones, thereby creating a mixed reality environment.

2 Motivation

The objective of this thesis is to take advantage of controlled illumination to enrich a video acquisition with shape and reflectance reconstructions. Today, a lot of works have tried to meet this objective. Some of them take advantage of sequential controlled illumination to recover high quality shape and reflectance, however they either require a costly and very cumbersome fixed setup, such as the Light Stage X [Deb12b], and/or do not run in real-time. Our aim is a low cost, fast, mobile and simple acquisition setup which has to be the less intrusive possible so as to provide a greater ease of use.

3 Structure of the thesis

The work presented in this thesis is mainly directed toward how a sequential illumination can be used to acquire shape and reflectance corresponding to a video sequence.

Chapter 1 - Related Works: As the first chapter concerns related scientific effort to the field, we survey relevant existing methods aiming at recovering shape and reflectance from a scene.

Chapter 2 - Rolling Flash and Photometric Stereo: The second chapter of this thesis focuses on the applications of the well known photometric stereo method [Woo80] to a video acquisition. Moreover, as a high frame rate is required by such an application, a method to use sequential illumination with high frame rate cameras (electronic rolling shutter cameras) is also considered. Despite the interesting results provided by photometric stereo, we found that this latter did not provide enough qualitative results. Moreover, by its nature, photometric stereo is not really suitable for the range of applications targeted. In particular, a three light strobe illumination setup is too cumbersome and intrusive to be considered as a real solution to augmented and mixed reality applications.

Chapter 3 - Shape and Reflectance from RGB-D images using time sequential illumination: In this chapter we propose a method for recovering the shape (geometry) and the diffuse reflectance from an image (or video) using a hybrid

setup consisting of a depth sensor (Kinect), a consumer camera and a partially controlled illumination (using a flash). The objective is to show how combining RGB-D acquisition with a sequential illumination is useful for shape and reflectance recovery. A pair of two images are captured: one non flashed (image under ambient illumination) and a flashed one. A pure flash image is computed by subtracting the non flashed image from the flashed image. We propose a novel and near real-time algorithm, based on a local illumination model of our flash and the pure flash image, to enhance geometry (from the noisy depth map) and recover reflectance information.

Chapter 4 - Automatic Light compositing Using rendered images: Lighting is a key element in photography. Professional photographers often work with complex lighting setups to directly capture an image close to the targeted one. Some photographers reversed this traditional workflow. Indeed, they capture the scene under several lighting conditions, then combine the captured images to get the expected one. Acquiring such a set of images is a tedious task and combining them requires some skill in photography. We propose a fully automatic method, that renders, based on a 3D reconstructed model (shape and albedo), a set of images corresponding to several lighting conditions. The resulting images are combined using a genetic optimization algorithm to match the desired lighting provided by the user as an image.

4 Publications

Most of the work presented in this thesis is published:

- **M. Hudon**, P. Kerbiriou, A. Schubert, K. Bouatouch, "High speed sequential illumination with electronic rolling shutter cameras", CVPRW, CCD 2015.
- **M. Hudon**, A. Gruson, P. Kerbiriou, R. Cozot, K. Bouatouch, "Shape and Reflectance from RGB-D Images using Time Sequential Illumination", VISAPP 2016.
- **M. Hudon**, R. Cozot, K. Bouatouch, "Automatic Light Compositing using Rendered Images", DIMAF 2016.

5 Patents

Also, several patents related to the work presented in this thesis were published or filled during my thesis.
Published:

- O. Bureller, **M. Hudon**, P. Kerbiriou, A. Schubert, "Method and apparatus for acquiring a set of images illuminated by a flash", US14578242.
- O. Bureller, **M. Hudon**, P. Kerbiriou, A. Schubert, "Method and device for capturing frames of a scene under different illumination configurations", EP20140305599 .

Filled:

- **M. Hudon**, P. Kerbiriou, P. Robert, "Method for normals and reflectance acquisition using sequential illumination and depth sensor".
- P. Robert, S. Jiddi, **M. Hudon**, "Reflectance parameter estimation in real scenes using an RGBD sequence".
- P. Robert, S. Jiddi, **M. Hudon**, "Estimation of specular light source and surface reflectance in a scene from a RGBD sequence".

Related Works 2

Contents

1 About reflectance measurement 19
1.1 Direct surface reflectance measurement 19
2 Scene reconstruction (through active lighting) 21
2.1 Photometric stereo on still images 21
2.2 Photometric stereo for video sequences 23
2.3 Shape from Shading (SfS) 25
2.4 Depth Enhancement 26
2.5 Discussion 28

The aim of this chapter is to survey the more relevant existing methods aiming at recovering the shape and the reflectance of real object from images. The measurement of the reflectance of real materials is also addressed in this chapter.

1 About reflectance measurement

Reflective properties of materials has been studied for hundreds of years. The progress in optical science, such as Snell-Descartes and Maxwell equations, has expanded the range of research about material reflectance properties with a better characterization of the interactions between objects and illumination. The function that describes this reflectance characteristic is the bidirectional reflectance distribution function (BRDF). Computer graphics has increased the research interest in materials properties, ranging from the creation of virtual objects with photo-realistic appearance to the characterization of real objects. However, the computation cost constrained researchers to simplify calculations and models so as to render objects in an acceptable amount of time.

1.1 Direct surface reflectance measurement

Devices such as classical gonioreflectometer have been invented to measure the BRDF of materials [MCS90]. They consist of a light source and a detector that can be controlled to include the necessary degrees of freedom. Those devices measure the radiant energy reflected by a surface material for numerous configurations

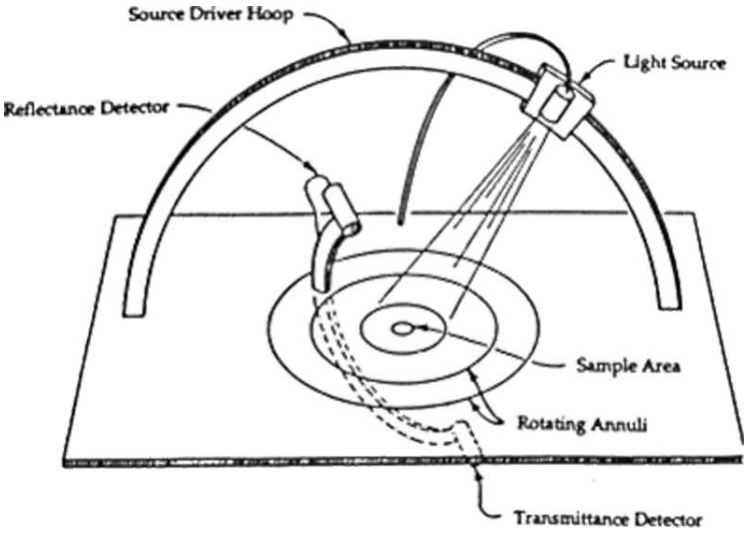

Figure 2.1 – Murray-Coleman and smith gonioreflectometer, figure from [War92]

of illumination and observation. The idea is to capture a comprehensive sampling of the BRDF. Fig. 2.1 shows a traditional gonioreflectometer.

[War92] has proposed to replace the reflectance sensor by a simple CCD camera equipped with a fish-eye lens. The surface is lit by a light source, but rather than observing directly the surface with the CCD camera, the surface is reflected by a half silvered hemisphere which is observed by the fish-eye camera. With the appropriate camera calibration, the entire hemisphere of reflected light is captured at once (see Fig.2.2).

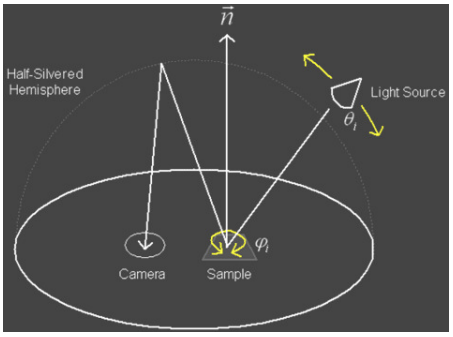

Figure 2.2 – Gonioreflectometer by [War92]

Ward's Gonioreflectometer is cheap and allows to rapidly sample the reflectance of a surface. However, a goniorefectometer is capapble of sampling, at once, the BRDF of a single point of an object of simple geometry, but cannot be applied to real world scenes containing objects of more complex geometry.

2 Scene reconstruction (through active lighting)

In this section we present a non-exhaustive state of the art of scene reconstruction (shape and/or albedo) methods using active illumination.

2.1 Photometric stereo on still images

Unlike gonioreflectometer-based acquisitions, which aim at capturing a comprehensive set of BRDF values of a surface, other works have tackled the problem of fitting simple BRDF models to BRDF measurements of real world scenes. Work on imaging processes has led to a better understanding of image formation. Indeed, the relation between the radiance values, recorded in an image of a scene, and the shape of the objects of this scene can be determined by modeling the way the objects' surface reflects light.

The first enlightening work [Woo80], on shape reconstruction from still images, relies on **photometric stereo**. The main idea of **photometric stereo** is to vary the direction of incident illumination between successive images while keeping the viewing direction constant, which allows the resulting images to provide enough information to determine the surface orientation at each of its points. The technique uses a simple reflectance model in which the scene's objects are assumed to be perfectly diffuse (Lambertian surfaces), which allows to express the radiance at each point (u, v) as a function of the surface orientation N and its reflectance k_d:

$$R(u, v) = k_d(u, v) \cdot |N(u, v).\omega_i|, \tag{2.1}$$

where ω_i is the direction of illumination. Three non co-planar directions of illumination are sufficient to determine the geometry for each pixel. However the author admits that the technique is best suited to smooth surfaces with no discontinuities.

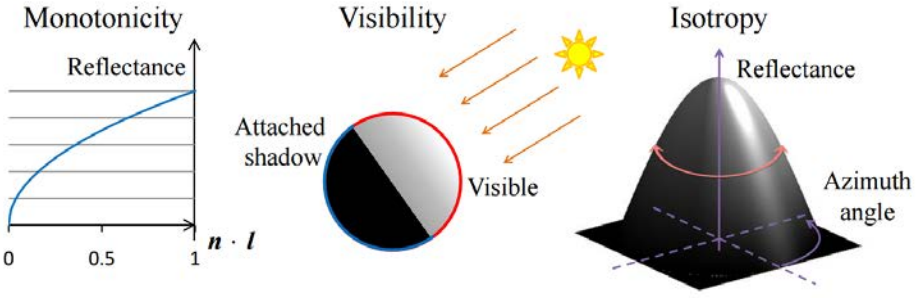

Figure 2.3 – Illustration of the three fundamental properties used by [HMI10], figure from [HMI10]

As a Lambertian model is very simplistic, to obtain more photo-realistic and accurate reconstruction of real world scenes, in [HMI10] the authors have expanded the photometric stereo concept to non-lambertian surfaces. Indeed, rather than fitting a reflectance model, to recover the geometry they use a consensus approach based on three reflectance properties: monotonicity, visibility and isotropy, observed

on many non-Lambertian surfaces, including specular ones Fig.2.3. Using a set of fifty input images with different incident illuminations, they compute three solution spaces corresponding to each reflectance property constraint. Surface orientation is then obtained by intersecting the three solution spaces. The method naturally avoids the need for radiometry calibration and is robust to general uncontrolled lighting, which finally allows to use photometric stereo in less constrained conditions (Dark rooms). It outperforms classical photometric stereo-based methods but is computationally heavy, more than $592s$ are required to process a dataset of 47 small images (480×490). Moreover as the isotropy and monotonicity constraints are surface dependent, the method can only handles surfaces that show either diffuse or specular reflection.

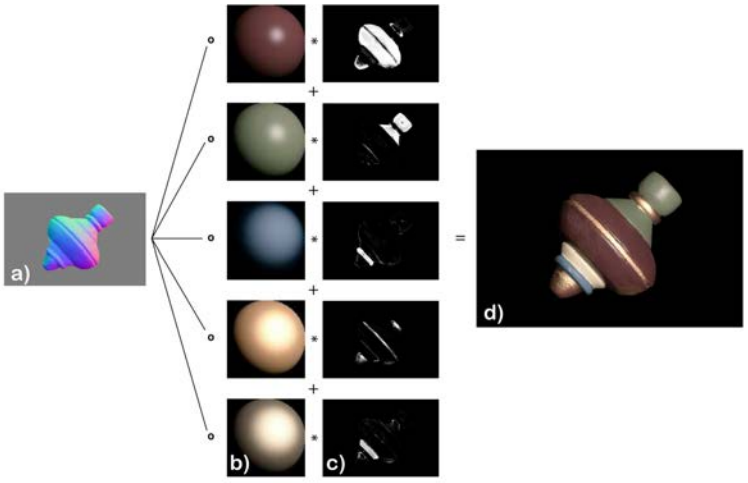

Figure 2.4 – Schematic illustration of the appearance model from [GCHS10]. A normal map (a) is used to remap the fundamental materials (b), and results are modulated by material map (c) and summed together (d).

In [GCHS10] the shape and Spatially-Varying BRDFs are recovered from multiple photographs with varying illumination (approximately 8 to 10 different illuminations). The authors describe materials as a convex combination of a small number of fundamental materials. Those latter are constructed with the isotropic Ward reflectance model [War92]. In other words, this contribution tries to directly project complex and spatially varying BRDFs into a basis function space, the basis functions being simple reflectance models fig.2.4.

In [RH01] it is shown that one only needs 9 spherical harmonics coefficients, corresponding to the lowest frequency modes of the illumination, to compute a diffuse shading with an error of 1%. A method is proposed in [TFG+13] for acquiring geometry and spatially varying reflectance properties using spherical harmonic illuminations. They have developed a system comprising a rotating arm capable of reproducing a spherical harmonic illumination in a real acquisitions fig.2.5. Then, from an object's response to several harmonic illumination conditions, specular and diffuse reflections are separated and world-space normals as well as anisotropic roughness parameters are obtained from a multi-view reconstruction. As a result, a high quality 3D model and merged reflectance maps are obtained on a variety of objects difficult

Figure 2.5 – Pictures of the acquisition system presented in [TFG+13], pictures taken from [TFG+13]

to acquire with other techniques.

Note that the computation constraints have drastically decreased since the work of [Woo80]. In more than 30 years an enormous progress in computer technology has been achieved and has gradually extended the boundaries of calculation time constraints. Finally, the capturing process in [TFG+13] and [HMI10] is really close to what gonioreflectometers tried to achieve with surface samples.

2.2 Photometric stereo for video sequences

Photometric stereo techniques are well suited to still images. However, real-time augmented and/or virtual reality applications require reconstruction methods for video sequences as they provide a better immersive user experience. Unfortunately, as photometric stereo techniques require several images of the same scene under different lighting conditions, they are by nature inconsistent with video sequences or moving objects. Nevertheless, several attempts have been made to adapt such techniques to video sequences with moving objects.

For moving rigid objects some methods have been proposed. They combine shading information with motion or multiview stereo while assuming a fixed illumination [BF08, MS06]. In [JK07] shape is recovered from several images of a rigid moving object and possibly under a varying illumination. Using camera projection matrices estimated from point correspondence across views, the authors' method computes a dense correspondence map by minimizing a multi-ocular photometric constraint. Once correspondence across views is established, photometric stereo is applied to estimate a surface normal field and 3-D surface fig.2.6. However, this method is difficult to apply to every video sequence scenario as it requires a rigid object. It cannot be applied, for example, to non-rigid deformable objects such as clothes.

Some approaches have tried to tackle the issue of using photometric stereo for video sequences by reducing the number of frames needed to apply a photometric stereo method. While in [Pet87] the authors use photometric stereo with multispectral illumination, in [HVB+07] a method makes use of photometric stereo with different colored lights, and shows how multispectral lighting allows to essentially capture three images (each with a different light direction) in a single snapshot, thereby making per-frame photometric reconstruction possible. The obtained results

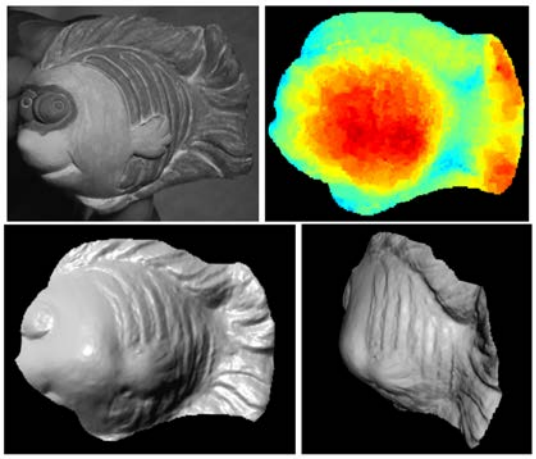

Figure 2.6 – Results taken from [JK07]: One of eight views of a figurine rotating it in front of a camera and light (top left). The intermediate dense depth map recovered by the method – red is closer and blue further from the camera (top right) and renderings of the final surface (bottom row).

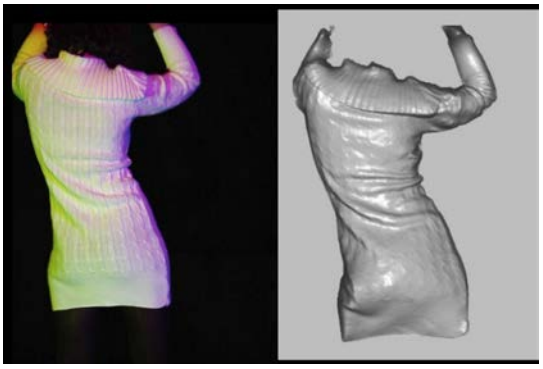

Figure 2.7 – Example of input image (left) and corresponding 3D reconstruction (right) from [HVB+07], images taken from the project webpage.

are similar to those provided by classical photometric stereo applied to a single uniform diffuse lambertian surfaces fig.2.7. Unfortunately, real world scenes are composed of a variety of non-uniform surfaces. A method [KWBE10] combines time-multiplexing and colored lights to compute surface normals of non-rigid objects. It computes directly instant normals from multi-spectral illumination as well as imaging coefficients using other images (with a carefully chosen lighting) obtained through time multiplexed illumination and optical flow alignment fig.2.8. Note that optical flow alignment does not need to be perfect as long as the material of aligned pixels remain the same, thereby the method does not rely on a perfect optical flow alignment. It is then robust to small optical flow misalignment. Time multiplexed illumination (TMI) can be used to reconstruct live action face performance through photometric stereo [WGT+05a]. The method uses Vision Research Phantom v7.1 high-speed digital camera to capture a scene at 2160fps and a resolution of 640×480 while synchronizing 156 light sources oriented toward the actor's face. Really impressive reconstruction results are obained, consisting of spatially varying BRDF

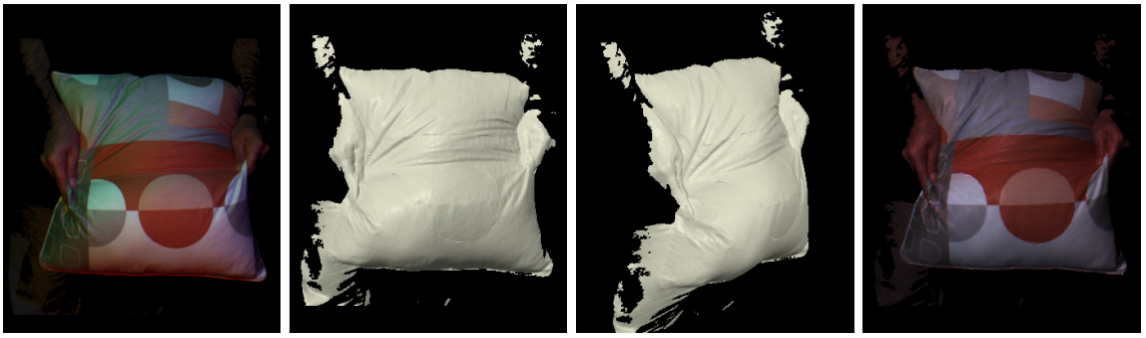

Figure 2.8 – Results from [KWBE10], from the left to the right: input frame acquired with the acquisition setup, result geometry, rendered geometry from another view, rendered with estimated albedo from camera view.

Figure 2.9 – Results from [WGT+05a], from the left to the right: input frame acquired with the acquisition setup, result Normal Map, albedo, ambient occlusions.

on the actor's face together with high quality geometry fig.2.9. However the system is really expensive and cumbersome. Furthermore, only 4.3 seconds of video can be recorded at this large frame rate. Once again, this last work requires a huge amount of data to reconstruct shape and reflectance, but the technological progress opens the door to this kind of work.

2.3 Shape from Shading (SfS)

Shading plays an important role in human perception of surface shape. Artist have long exploited lighting and shading to convey illusions of depth in paintings. The study of how images are formed leads to the SfS techniques. In [Hor70] a SfS technique is presented and described as a method for obtaining the shape of a smooth opaque object from one view. The method makes use of a monocular image and assumes some knowledge on the surface reflectance and the position of the light sources. It provides a first-order non-linear equation with two unknowns relating the intensity at the image points to the shape of the object. Since this first paper, a huge amount of contributions have been made to the SfS technique. In [ZTCS99] the authors give an exhaustive survey of SfS methods while classifying them in four categories: minimization, propagation, local and linear approaches. They also compare algorithms on virtual and real data and conclude that generally, all SfS algorithms produce poor results on synthetic data and even worse results on real images. Later, a method [Bru88] builds fine height maps from scenes illuminated from above, using a SfS method based on a recursive way of determining equal-height

contours. More recently, another approach [PF05, FKI+14] uses controlled light sources near the camera optical center and takes into account the inverse squared distance attenuation term of the illumination in a SfS approach.

2.4 Depth Enhancement

Recently, there has been a little revolution in computer vision with the emergence of consumer active depth sensor such as the Kinect. Those sensors use active infrared structured (Kinect 1) or unstructured (Kinect 2.0) illumination to extract in real-time depth information about the scene. Those sensors have opened the way to many new real-time applications in computer vision, computer graphics and human computer interaction. However those sensors suffer from significant drawbacks: they provide only coarse geometry because of their low resolution and noise. The ability to capture per-frame higher quality geometry could facilitate the arrival of new applications such as high detail feature tracking, real-time relighting or pre-visualization of computer graphics imaging effects. In [DT05] the authors used Markov Random Fields to fuse data from a low resolution depth scanner and a high resolution color camera. An efficient algorithm has been devised in [NRDR05] for combining depths

Figure 2.10 – Results from [NRDR05], from the left to the right: Rendering of 3D scanned range image, same scanned geometry, augmented with a measured normal-map (from photometric stereo), hybrid surface reconstruction, which combines both position and normal constraints, photograph.

and surface orientations (normals) while taking advantage of each to create the best geometry possible for computer graphics purposes fig.2.11.

Figure 2.11 – Results from [RSD+12], from the left to the right: Input depth map, spatial filtering and spatiotemporal filtering.

A new possibility [RSD+12] is explored to augment a video camera with a consumer time-of-flight depth camera. It proposes an efficient algorithm to remove typical artifacts of such depth data and then apply efficient spatio-temporal denoising and up-samlping to obtain plausible RGB-D sequences. It also shows a direct application to special video effects such as the relighting of the resulting sequence. The proposed heuristic approach looks plausible but takes no advantage of shading. The methods described in [HLK13, YYTL13] also combine RGB images and depth data and try to solve the inverse rendering problem by estimating reflectance and illumination as well as geometry. A real-time method [WZN+14] tries to solve the in-

Figure 2.12 – Results from [WZN+14], from the left to the right: RGB aligned image, input raw depth map, refined depth map.

verse rendering problem using an effective parametrization of the shading equation. It estimates time varying incident lighting per-frame, which is then used for geometry refinement. The method relies on a highly parallel algorithm that reformulates the inverse rendering optimization problem, allowing to estimate per-frame lighting and shape for Lambertian scenes fig.2.12. Recently, a novel method [OERW+15]

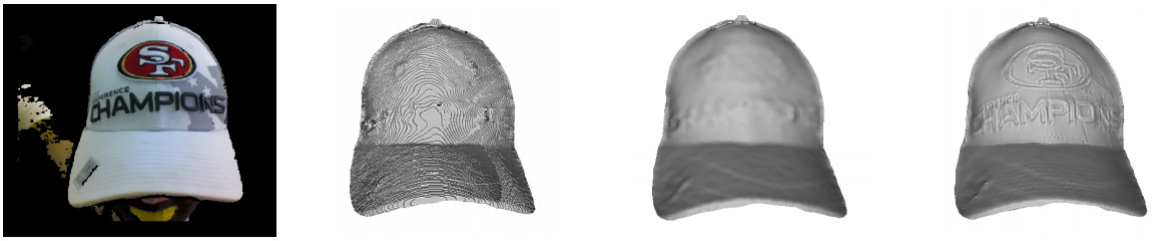

Figure 2.13 – Results from [OERW+15], from the left to the right: RGB aligned image, input raw depth map, bilateral filtered depth, refined depth map.

has been proposed to enhance the depth captured with low-cost RGB-D scanners without the need to explicitly determine and use surface normals. The method gives accurate results and runs in real-time. It achieves 10 fps for 640×480 depth profiles fig.2.13.

2.5 Discussion

We have described several methods aiming at reconstructing shape and reflectance of a real scene. In this thesis, our aim is to design a low-cost, fast (real-time), mobile and simple acquisition setup which as to be the less intrusive possible so as to provide a greater ease of use.

It appears that SfS methods are under-constrained, as prior knowledge about lighting and/or surface reflectance is required. Also, SfS methods do not provide accurate results, moreover, due to their high computational cost, they do not match with our mobile and real-time objectives.

On the other hand, Photometric Stereo methods provide very impressive results at a low computational cost. However Photometric Stereo often relies on too complex acquisition setups. In addition, it is not appropriate to most practical case scenarios as Photometric stereo methods require an environment where light has to be fully controlled.

As for depth enhancement methods, they help reconstruct shape from noisy and quantified depth maps provided by consumer depth sensors. Those sensors are not expensive and are easy to embed into a mobile setup. Nevertheless, those methods still suffer from reconstruction artifacts and do not run in real-time.

Given the above remarks, we propose several contributions in this thesis. First, we propose a simple photometric stereo for image sequences: at least 3 frames are needed to perform simple photometric stereo and we synchronize stroboscopic illumination with fast CMOS camera sensors equipped with an electronic rolling shutter (most frequently used sensors).

Second, we devised a new method to reconstruct shape and diffuse reflectance of real scenes with general uncontrolled lighting. This method relies on a hybrid acquisition setup consisting of a depth sensor, a consumer camera and a partially controlled illumination (Flash). The objective is to show that combining RGB-D acquisition (as depth enhancement methods) with a sequential illumination is beneficial for shape and reflectance recovery. Finally, we exploit our hybrid acquisition setup to perform an automatic light compositing.

Rolling Flash and Photometric Stereo

3

Contents

1 Introduction and Objectives **29**
2 Electronic rolling shutter cameras **30**
3 Related works **33**
3.1 Computer vision and Electronic rolling shutter cameras. 34
3.2 Stroboscopic Illumination in computer vision. 35
3.3 Electronic shutters and stroboscopic illumination. 35
3.4 Discussion 36
4 The Rolling Flash **36**
4.1 Avoiding intra-rows artifacts 36
4.2 Reconstructing temporal-coherent rows 38
4.3 Flash illuminating 3 frames 39
5 Framework and algorithm **40**
5.1 Framework 40
5.2 Reconstruction Algorithm 41
5.3 Sequential recording 42
6 Experiment results **44**
6.1 Straightforward application to Photometric Stereo 44
7 Conclusion **46**

1 Introduction and Objectives

Light is a key element in photography or cinema as it is the information vector. Flashes bring many possibilities to photographers, one direct benefit is the motion freeze. LED technology allows flashes to be more sustainable, cheaper and highly controllable. All those considerations make more tempting the idea of using flashes in cinema. Sequential recording with a stroboscopic flash is an interesting idea that is already used in many research projects such as light stage X [Deb12b]. This chapter presents our work on using stroboscopic illumination for video sequences. Our goal

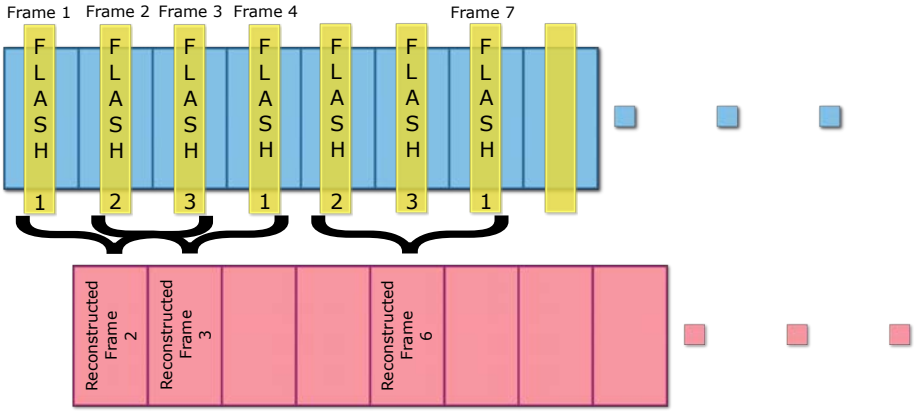

Figure 3.1 – Our goal application, simple PS applied to video sequences.

is to be able to directly apply simple Photometric Stereo (PS) to a video sequence to extract, per-frame, a normal map as well an albedo map. In order to apply PS on a frame, one needs 3 images, from a fixed point of view, each taken under a different direction of illumination (the 3 directions of illumination must not be co-planar [Woo80]). Fig.Fig.3.1 describes our objective, each direction of illumination is represented by a flash ($Flash1$, $Flash2$ and $Flash3$). At each exposure we trigger a different flash. As a result three consecutive frames are lighted by 3 different flashes. Thus if we consider that our frame rate is high enough to neglect motion in the video, we can reconstruct each frame with PS.

The strongest assumption in this technique is that the motion between consecutive frames is neglectable. Thus, we need our camera to have the higher frame rate possible. In the consumer and professional market, there are different types of camera and sensor. The most widespread type of camera with high frame rate capabilities are cameras equipped with a CMOS sensor and an electronic rolling shutter. But using flashes with video recording is not as easy as in photography. Actually, flashes cause many temporal artifacts in video recordings, especially with high speed CMOS cameras equipped with electronic rolling shutters. This chapter proposes a video recording method that allows the use of periodic strobbed illumination together with any electronic rolling shutter camera, even without any synchronization device between the camera and the controlled lights. The objective is to avoid recording artifacts by controlling the timings and periods of the flash lights, and then reconstruct images using rows that correspond to the same flash instant. In this chapter, first, the basic operation of a CMOS sensor is explained and a model of the electronic rolling shutter is described. Then, after presenting and modeling typical artifacts caused by a strobe illumination with those sensors, we present our solution to overcome these artifacts.

2 Electronic rolling shutter cameras

Traditional cameras use a mechanical shutter to block entering illumination while sliding the chemical film to a new frame. Nowadays, some digital cameras still

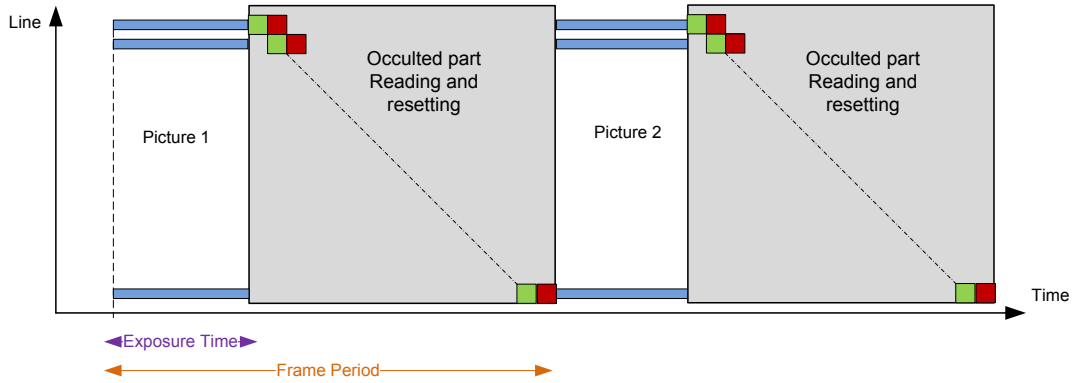

Figure 3.2 – Mechanical shutter functioning: a red square represents the reset of a row while a green square represents a reading time.

uses electronic rolling shutter. A model is shown in Fig.3.2. As rows can only be read one by one, there is time between the read of the first and the last row of the sensor. Just as traditional cameras mechanical shutter are used to avoid recording artifacts. When closed, the mechanical shutter prevents any light from reaching the sensor by blocking the focal image point. At the end of an exposure, the shutter is closed and every rows are read and reset sequentially. Once the last row has been read and reset, the shutter is opened for the next exposure. When using a mechanical shutter, the camera avoids any recording artifact, but the system is cumbersome, noisy and expensive. Moreover time spent for a read/reset cycle (shutter closed) is time lost for exposure. This limited time available for the exposure is a strong physical restriction to high speed acquisitions. For all those reasons, cameras equipped with a mechanical shutter are not well suited to our needs.

However, most digital cameras use an electronic rolling shutter as a substitute for the mechanical shutter. Electronic rolling shutters have been a revolution as they allowed to rise the frame rate while decreasing the price of both professional and consumer cameras. Unfortunately, electronic rolling shutters also cause new types of temporal artifacts, such as skew effect (spatio-temporal shear in images) or partial exposure lighting (an example of partial lighting during lightning is shown in figure Fig.3.3). Some digital cameras are now equipped with a global electronic shutter. This technology completely resolves temporal artifacts issues caused by electronic rolling shutters. However some designs of global shutter makes photosites less efficient, which means that, at equal exposure, those global shutters will be noisier than electronic rolling shutter. Furthermore, a large amount of available cameras (consumer and professional) are continued to be equipped with an electronic rolling shutter as the rolling shutter is the standard for almost a century.

In a typical camera with a CMOS digital sensor and an electronic rolling shutter, the rows are read only one by one, as described in Fig.3.4. The photodiode within the pixel receives light which is then converted to electrical charges and accumulated. Accumulated charges are converted to voltage for each pixel. The converted voltage is transferred to the vertical signal line. Finally the image signal voltage is output through the horizontal signal line and then converted from analog to digital values.

The electronic rolling shutter framework entails a significant time between the

Figure 3.3 – Partial lighting example: typically happens while filming lightning. Lightning occurs while recording a video with an electronic rolling shutter. As the rows of the sensor integrate light sequentially, the illumination from lightning is only sensed by one part of the rows (here bottom rows). As a result, the bottom rows of the image appear brighter than the upper ones.

reading of the first and the last row of an image, resulting in artifacts such as partial exposure.

Figure Fig.3.5 shows a model of an electronic rolling shutter of a camera. This model was learned after experiments we conducted on raw cameras. It describes how an electronic shutter operates. Each parallelogram depicts the integration (accumulation of light energy reaching the sensor) of the rows of an image, a blue row represents the integration of light by one row of the sensor. A red square represents a row reset, while a green square corresponds to a row read. As there is no mechanical shutter, the sensor is always integrating light. At the beginning the first row is reset, which means that a new integration cycle starts for this row. Then, the second row is reset, etc. When the exposure time is up for the first row, the camera reads and saves the information. Then a reset is performed and a new cycle begins. This process is performed for all rows, therefore all rows have the same exposure time. The shutter value is the ratio of the exposure time to the frame period multiplied by 360° (expressed in degrees). If this ratio is less than 360°, then the camera waits for a time equal to $(frame_{period} - exposure_{time})$ between read and reset.

An explanation of partial lighting artifact, according to the electronic rolling shutter model, is described in figure Fig.3.6. Parallelograms still represent different images taken by the camera (shutter value is set to 360°). Yellow rectangles have been added to represent stroboscopic flashes. As the rows are not read and reset at the same time, a flash of a certain duration Δ lights several images. However, depending on the flash duration Δ only a subset of rows of an image are lit by one flash. For the example shown in figure Fig.3.6 the flash 2 lights the top rows of *image* 3, while the bottom rows of *image* 3 are lit by the flash 3. As a result, some rows in *image* 3 integrate the light of the flash 2 and other rows integrate the light of the flash 3. This type of artifact will henceforth be called inter-rows temporal artifact. Figure Fig.3.6 shows another temporal artifact. Indeed, the rows of *image* 3 between the two black dotted rows are composed of both lights integrated during

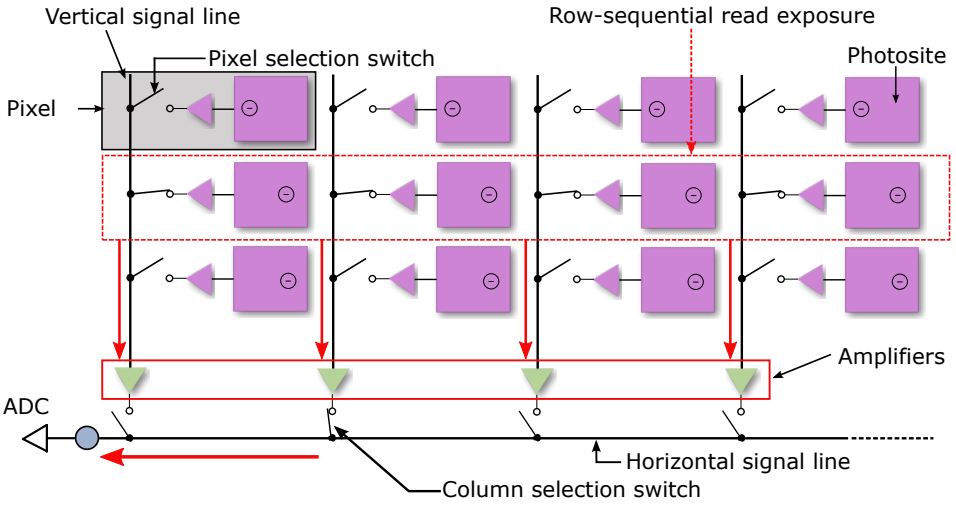

Figure 3.4 – Description of a CMOS electronic rolling shutter operation. Rows are read sequentially through vertical signal lines, then the output signal is extracted through horizontal signal lines.

flash 2 and 3. From now on, this type of artifact will be called intra-rows temporal artifact.

In this chapter, we propose a video recordering method that allows the use of stroboscopic flashes with high speed electronic shutter cameras while maximizing the output frame rate. Our method can be used with a shutter value of 360° in a controlled indoor environment such as a movie studio. Our main contributions are:

- an acquisition framework, relying on triggering stroboscopic flashes, which avoids only intra-rows temporal artifacts, this method allows the maximum possible frame rate using periodic flashes, it does not need any synchronization device between the camera and the flash;
- a method and a framework, robust to albedo variations, to reconstruct a coherent sequence from a sequence containing only inter-rows temporal artifacts;
- an adaptation the of above method to perform sequential recording with flashes of different durations;
- a straightforward application of our method to photometric stereo.

In the following, we first review the related work on electronic rolling shutter cameras and light synchronization, followed by a description of the theoretical aspects regarding our method. Then, we provide a framework and algorithms to use our method. Finally we present some results.

3 Related works

Temporal artifacts are inherent to electronic rolling shutter cameras and led in the past to the depreciation of this type of camera for computer vision applications.

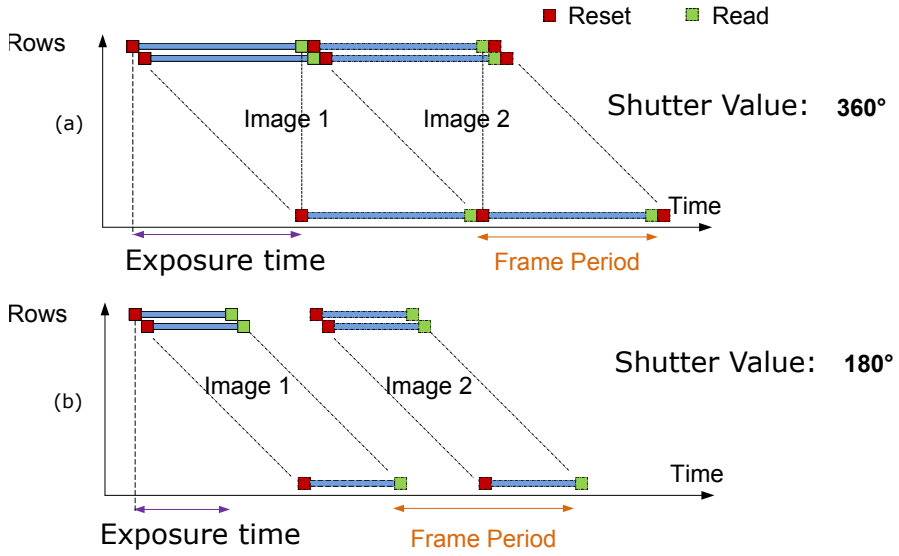

Figure 3.5 – Electronic shutter functioning: a red square represents the reset of a row while a green squares represents a reading time. (a) For a 360° shutter value and (b) for a 180° shutter value.

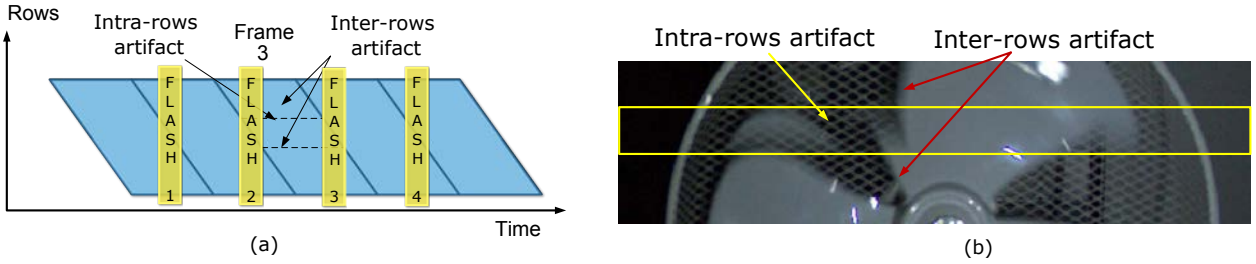

Figure 3.6 – **(a)** Electronic shutter with periodic flashes, the two types of artifacts are shown in this schema. **(b)** Typical example of the two artifacts put forward when filming a rolling fan.

Nonetheless, lower prices and high frame rates have pushed scientists and researchers to work with those cameras. More and more papers present methods either benefiting from those artifacts or creating a model in order to remove these artifacts.

3.1 Computer vision and Electronic rolling shutter cameras.

Wilburn et al. [WJV+05] used an array of electronic rolling shutter cameras with precisely timed exposure windows to perform very high speed recordings. They warped images from calibrated cameras, by merging scanlines from the different views into a virtual one to reconstruct very high speed sequences with no distortion artifacts. Grundmann et al. [GKCE12] presented a calibration free rolling shutter removal technique based on a novel mixture model of homographies which faithfully models rolling shutter distortions. This technique adapts to the camera without any calibration and is robust to a wide range of scenarios while having an efficient rate of 5 - 10 frames per second. Magerand et al. [MBAP12] proposed a method to

estimate a uniform motion, using constrained global optimization of a polynomial objective function, to automatically build robust 2D-3D correspondences. Magerand and Bartoli [MBAAP12] proposed a rolling shutter model capable of handling both global and uniform rolling shutters.

3.2 Stroboscopic Illumination in computer vision.

Theobalt et al. [TAH+04] used consumer cameras and stroboscopic illumination to capture high speed motions. In their experiments, they captured the motion of a hand and a ball in a baseball pitch. They used several still cameras with a long exposure time of 1 second and stroboscopic flashes at $75Hz$, which means that a high number of ball ghosts were visible on the output images. Linz et al. [LSM08] used stroboscopic flashes to capture multi-exposure images that allowed to generate intermediate exposure views and synthetic motion blurs. Methods based on photometric stereo [Woo80], such as those described in [KWB10] and [HVB+07], use colored lights to obtain several images (with different lighting directions) in a single snapshot. Decker et al. [dDKMB09] use both colored lights and time multiplexed images to perform photometric stereo (with more than 3 light directions) on video sequences. This method resolves low frame rate issues inherent in photometric stereo applied to video, but suffers from some issues. For example, the method fails when the spectra of one of the light sources and the object albedo do not overlap. Wenger et al. [WGT+05b] proposed a method to acquire live-action performance of an actor, allowing lighting and reflectance to be designed and modified in post-production. They perform a robust motion compensation allowing to increase the exposure times of their cameras. They obtain excellent results, but need a highly controlled environment and a synchronization device between lights and cameras.

3.3 Electronic shutters and stroboscopic illumination.

Furlan et al. [FB08] patented a method to use flash lights with an electronic rolling shutter (with no temporal artifacts). It makes use of a rolling shutter timing mechanism based on an alterable translucent material in the optical path of light going to the sensor. Bradley et al. [BAIH09] used a controlled illumination to synchronize an array of rolling shutter cameras. The method consists in triggering flashes, with a high enough period to avoid any temporal artifact, and reconstructing images that are integrated in consecutive frames. As the duration of the flashes is short, the temporal shear of rolling shutter cameras is also avoided. With their approach the maximum frame rate is divided by two. It is mentioned that the frame rate can be rose with more computational effort to explicitly search for unexposed scanlines that separate the frames. Unfortunately the authors do not provide any theoretical nor experimental details.

3.4 Discussion

There is no denying that imaging objects under variable lighting is of foremost importance in computer vision and image-based rendering. Time-multiplexed illumination (TMI) is often considered for different applications, as mentioned above. However, TMI does not tackle the problem of getting the highest frame rate possible. Several TMI-based methods make use of stroboscopic illumination in computer vision, even with electronic rolling shutter cameras. However, no method focuses on increasing the frame rate. In this part, we present a method that allows TMI (and sequential recording) with high speed electronic rolling shutter cameras, while maximizing the output frame rate. Also, a direct application to Photometric Stereo is presented.

4 The Rolling Flash

First of all, our method consists in removing intra-rows artifacts, which cannot be removed in a post-processing step because pixels of the same row integrate light multiple times which is non-reversible Fig.3.6. We then reconstruct a coherent sequence combining rows corresponding to the same flash instant. From now on, the time, during which a flash is on, will be called flash instant. For our experiments we used a LED illumination system. Average LED response and falloff time is of the order of 10 nanoseconds. A Full HD electronic rolling shutter camera approximately reads one of its rows in 10 microseconds. Average LED response time and falloff time are negligible with respect to the camera time constants, therefore the illumination can be considered as perfect square.

4.1 Avoiding intra-rows artifacts

Our objective is to avoid intra-rows artifacts while maximizing the frame rate. Our strategy is to increase the period of stroboscopic flashes. To this end, we set the period T_f of the stroboscopic flashes as follows:

$$T_f = \Delta_f + \Delta_e, \tag{3.1}$$

where Δ_f is the flash duration and Δ_e the exposure time. With this formula we prevent a row from integrating light twice (at two different flash instants), which allows a maximum frame rate.

An illustration of this method can be seen in figure Fig.3.7, each parallelogram represents an image, and each colored pattern represents rows that have integrated light from the same flash instant. A flash stops emitting light during the integration of the n^{th} row of frame k, the next flash starts $T_f = \Delta_f + \Delta_e$ seconds later, which exactly corresponds to the begining of the n^{th} row of frame $k+1$. Thus, each row only integrates light from a single flash, at a single flash instant (which duration is Δ_f). However not all the rows integrate light for the whole duration Δ_f. In a classical camera model the digital value N_d of a pixel is in direct relation with the

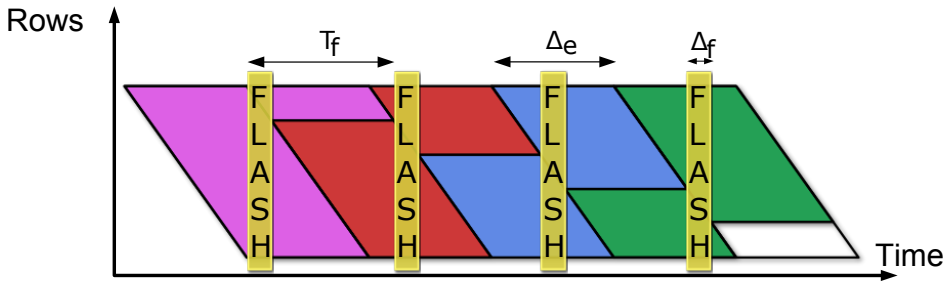

Figure 3.7 – Illustration of the rolling flash method: our method allows to use flashes with electronic rolling shutter camera, fixing the stroboscopic period of flash to the sum of the exposure time Δ_e and the flash duration Δ_f.

exposure time Δ_e, the aperture f_s, the ISO sensitivity S, the calibration parameter K_c of the camera and the luminance L of the scene [PDH11]:

$$N_d = \int_{\Delta_e} K_c \times \frac{S}{f_s^2} \times L \, \mathrm{d}t \tag{3.2}$$

L can be approximated to zero when there is no flash, thus the interval of integration is reduced to the duration of the flash Δ_f:

$$N_d = \int_{\Delta_f} K_c \times \frac{S}{f_s^2} \times L \, \mathrm{d}t \tag{3.3}$$

In our case of controlled illumination we can suppose L to be constant for a period Δ_f, and so:

$$N_d = \Delta_f \times K_c \times \frac{S}{f_s^2} \times L \tag{3.4}$$

Consequently, if a row integrates light for a duration Δ shorter than Δ_f, its digital value will be lower. Therefore those rows will appear darker on the resulting image. As the stroboscopic period of flashes is higher than the exposure time, the index of darker rows is never the same. On the resulting sequence darker rows roll over the image along consecutive frames Fig.3.8. That is why we chose to call that method rolling flash. T_f is the minimum stroboscopic period that makes possible to completely avoid intra-rows artifacts, which allows to maximize the output sequence's frame rate. Actually, the output frame rate is:

$$F = \frac{1}{\Delta_f + \Delta_e} = \frac{\frac{1}{\Delta_e}}{1 + \frac{\Delta_f}{\Delta_e}} \tag{3.5}$$

As explained before, our method works with a camera shutter value set to 360°, which means that the inverse of camera exposure time $\frac{1}{\Delta_e}$ is equal to the camera frame rate F_c:

$$F = F_c \times \frac{1}{1 + \frac{\Delta_f}{\Delta_e}} \tag{3.6}$$

Typically the flash duration is much lower than the exposure time. As an example, if we use a $60H_z$ camera and a flash duration of $200\mu s$ we obtain an output frame rate of $59.29H_z$, which represents a loss of $0.71H_z$.

Figure 3.8 – Typical raw sequence taken under a rolling flash illumination. We can see that the darker rows rolling over the images.

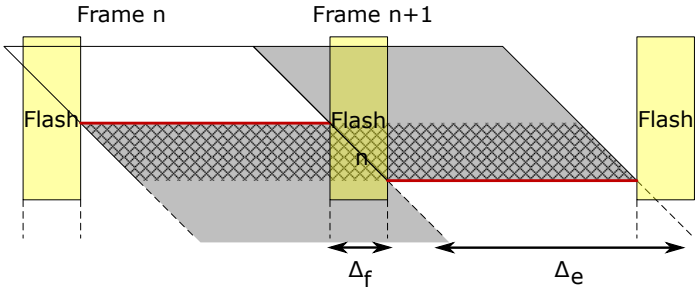

Figure 3.9 – Detail of rolling flash method: the same flash instant illuminates two camera frames, the rows in grey contain all the information needed to reconstruct a coherent image.

4.2 Reconstructing temporal-coherent rows

Now, we have an acquisition method that allows to avoid intra-rows temporal artifact. Unfortunately our sequence still contains inter-rows temporal artifacts. In this subsection we present a method to obtain a fully coherent sequence reconstructed from the rows (belonging to two subsequent captured images) integrating light from a same flash instant.

In figure Fig.3.9, each row, colored in grey, has integrated light from the same flash instant. On frame n, the red row delimits rows that have integrated light from flash $n-1$ and rows that have integrated light from flash n. On frame $n+1$, the red row delimits rows that have integrated light from flash n and rows that have integrated light from flash $n+1$. Here we are interested in reconstructing a coherent image from flash n. On frame n and $n+1$ rows in grey integrated light from flash n. Rows that integrated light for the full flash duration Δ_f can be directly used to reconstruct our coherent image (bottom rows of frame n and top rows of frame $n+1$). The hatched rows did not integrate for the full duration Δ_f. Let us reconsider the integration equation presented in equation Fig.5.1 for a pixel N of the n^{th} frame and the i^{th} row that did not integrate for the full flash duration:

$$N_n^i = (\alpha^i \times \Delta_f) \times K_c \times \frac{S}{f_s^2} \times L(n),\ 0 < \alpha < 1 \tag{3.7}$$

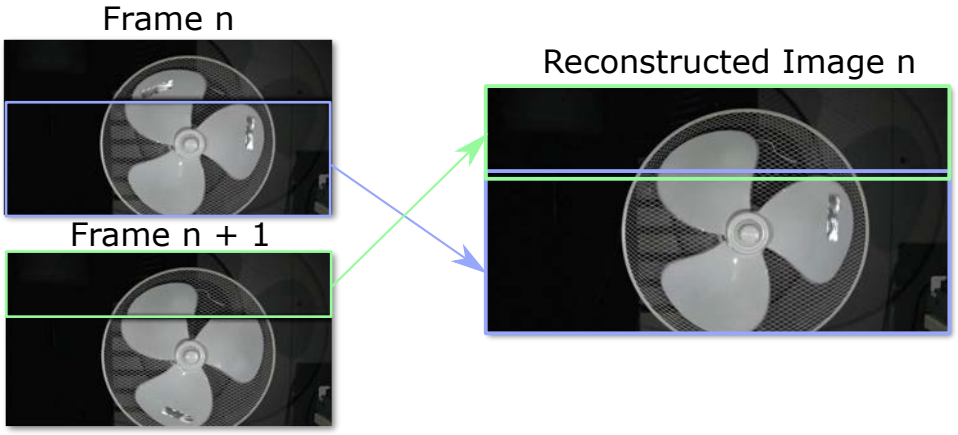

Figure 3.10 – Example of reconstructed image: Image n is reconstructed from frame n and frame $n+1$.

$L(n)$ is the luminance acquired of the scene illuminated by flash n. Let δ_n^i be the time between the beginning of flash n and the read of a row i that did not integrate during the full duration of flash n ($\delta_n^i < \Delta_f$), then α^i is given by:

$$\alpha^i = \frac{\delta_n^i}{\Delta_f} \tag{3.8}$$

If we consider that the time of one row read and reset is negligible compared to flash duration, then the digital value of the same pixel but in frame $n+1$ (which is illuminated by flash n) is expressed as:

$$N_{n+1}^i = ((1-\alpha^i) \times \Delta_f) \times K_c \times \frac{S}{f_s^2} \times L(n) \tag{3.9}$$

Consequently the sum of the two digital values is:

$$N_n^i + N_{n+1}^i = \Delta_f \times K_c \times \frac{S}{f_s^2} \times L(n), \tag{3.10}$$

which is the digital value corresponding to the flash instant n as if it had been fully integrated in one frame. Thus, to reconstruct those rows, we just need to sum the rows of same indices in frame n and frame $n+1$.

An example of reconstruction is shown in figure Fig.3.10. The rows on the top of frame n integrated light during flash $n-1$, while the rows of the bottom of frame n integrated light during flash n. As the top rows of frame $n+1$ integrated light during flash n, a coherent image can be reconstructed from the top rows of frame $n+1$ and the bottom rows of frame n. The yellow hatched area on the reconstructed image represents the rows that have to be reconstructed as a combination of the same rows from frame n and $n+1$.

4.3 Flash illuminating 3 frames

As a flash has a non zero duration, and the shutter value of the camera is 360°, there is a strong probability for a flash to illuminate rows of 3 consecutive images.

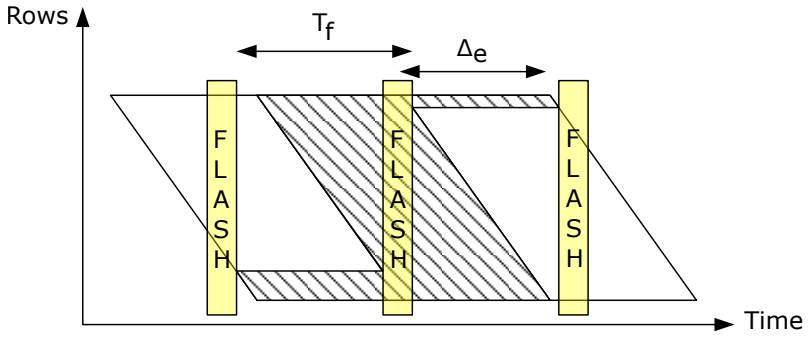

Figure 3.11 – Flash illuminating three frames.

Then the reconstruction works exactly the same way as explained in the previous subsection. Indeed, the bottom rows have to be reconstructed from frames $n-1$ and n, while the top rows have to be reconstructed from frames n and $n+1$. For simplicity reasons, this issue can be avoided by choosing a flash duration as follows:

$$\Delta_f = \frac{\Delta_e}{n}, n \in \mathbb{N}\backslash\{0,1\} \tag{3.11}$$

With a flash duration δ_f, being a divisor of the exposure time Δ_e, and a first flash triggered at the beginning of an exposure, the n^{th} flash ends exactly at the end of the exposure of frame n, which avoids the issue (of a flash illuminating rows of three frames) from happening. Unfortunately, in this case a synchronization device between the camera and the illumination device is needed to trigger the first flash at the right time.

5 Framework and algorithm

In this section we present the framework and the algorithm we used for our experiments.

5.1 Framework

We designed an electronic device that allows to drive our LEDs (flashes). A micro controller is used to send a pulse width modulation to the power stage driving the LEDs. LED response to current is fast enough to be neglected compared to the camera hardware. We have created a small interface to rapidly control the micro controller. In this device the following parameters can be tuned:

- the stroboscopic frequency $f = \frac{1}{T_f}$,
- the duty cycle of a burst allowing to control the power of the output light,
- the frequency of a burst (linked to duty cycle),
- the possibility to control several flashes with different parameters.

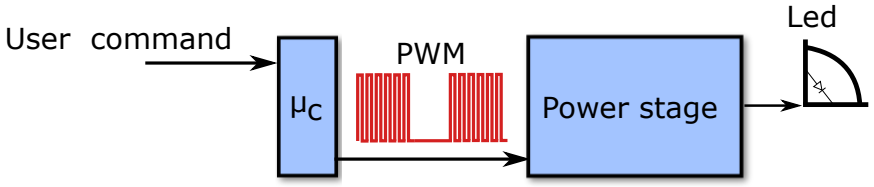

Figure 3.12 – Overall architecture of the control device.

An overall architecture of the control device is shown in Fig.3.12. We control the parameters presented above through a C++ interface linked to the micro controller. The micro controller sends timed pulses width modulation to the power stage which generates current to the LEDs.

5.2 Reconstruction Algorithm

In this subsection, we propose an algorithm to reconstruct temporally coherent rows. The main idea of this algorithm is to search for the row with the lowest luminance energy. A color pixel consists of 3 components (red, blue and green). For each pixel i the luminance Y_i is calculated. Let N be the number of columns in the image, then the luminance energy of a row n is:

$$L_e^n = \sum_{i=0}^{N-1} Y_i \tag{3.12}$$

Unfortunately, finding the row with the minimum energy can be problematic when dealing with noisy images and/or scenes containing albedo variations. To improve robustness to albedo variations and noise, we have added a term that defines a zone in which the row of minimum luminance energy is excepted. Furthermore, this term drastically reduces the search window. The reconstruction algorithm is described in algorithm Fig.1. This algorithm uses raw images. The first input image (captured image) is transformed to a linear gamma image (line 1). Then, the row of that linear image with the minimum luminance energy is determined (line 2). This row corresponds to the end of a flash instant and the beginning of the next one. We calculate the expected index offset between the rows of minimal energy of two consecutive images (line 3). Given that our shutter speed is set to 360°, we can calculate the duration of a row read as the ratio of the exposure time Δ_e to the number of rows h of the sensor, therefore the index offset o is simply:

$$o = \frac{\Delta_f}{\Delta_e} \times h, \tag{3.13}$$

where Δ_f the flash duration. Then for each image in the sequence, we proceed as follows. In the current image $CurrentImage_l$, all the rows with an index lower than the one of the row with minimal luminance energy are set to zero (line 9). The index k_{est} of the minimal energy row for the next frame is estimated by adding (modulo the image height h) the offset o to the index of the minimum energy row calculated for the current image (line 10). After that, we compute the index k of the local

minimum luminance energy row (in $NextImage$) around k_{est} ($k \in [k_{est} - \epsilon, k_{est} + \epsilon]$) and update an adjustment variable o_{adj} which avoids artifacts caused by minor drifts that could have occurred between the camera capture and the strobe illumination (line 13). An illustration of the latter process is given in figure Fig.3.13. Rows of $NextImage_l$, with an index higher than the one of the row with minimal luminance energy, are set to zero (line 14). Then the two resulting images are summed (line 15). Finally we transform the summed image back into the original gamma space (line 16). All those steps are performed for all the images of the sequence.

Algorithm 1 ReconstructImage(InputVideoSequence M, OutputVideoSequence M_o) :

1: $FirstImage_l$ = LinearGamma($FirstImage$);
2: i = FindRowMinimalEnergy($FirstImage_l$);
3: o = CalculateOffset(Δ_e, Δ_f, h);
3:
4: $CurrentImage_l = FirstImage_l$; // Current Image
5: $k = i$;
6: $k_{est} = i$;
7: $o_{adj} = 0$;
8: **for each** $Image \in M$ **do**
9: $CurrentImage_l$.SetToZeroRows(index $\leq k$);
9:
10: $k_{est} = k_{est} + o + o_{adj} \pmod h$;
11: $NextImage_l$ = LinearGamma($NextImage$);
12: k = RowMinEnergyAround($NextImage_l$, k_{est});
13: $o_{adj} = k - k_{est}$;
13:
14: $NextImage_l$.SetToZeroRows(index $\geq k$);
14:
15: $FinalImage_l = CurrentImage_l + NextImage_l$;
16: $FinalImage$ =OriginalGamma($FnialImage_l$);
17: Output($FinalImage$);
18: **end for**

5.3 Sequential recording

The rolling flash method is an efficient way to use periodic flashes with an electronic rolling shutter camera. The use of high speed cameras originally aimed at performing sequential recordings. The goal of sequential recordings is to obtain several sequences of the same scene in a single shot. As an example, to perform stereo photometry [Woo80] on a video the same scene is lit with at least 3 different illuminations. Another example is HDR (High Dynamic Range) recording which consists in capturing several sequences of the same scene with different exposure times. Usually a stereoscopic rig with two cameras (with aligned optical axes) is used to capture

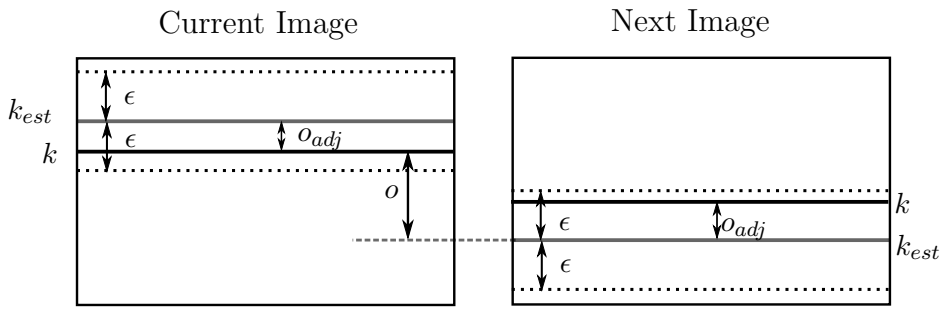

Figure 3.13 – Computing the index k of the minimal luminance energy row for the current and subsequent images. k is determined within the range $[k_{est} - \epsilon, k_{est} + \epsilon]$. For our experiments we chose $\epsilon = \frac{h}{100}$, h being the height of a frame.

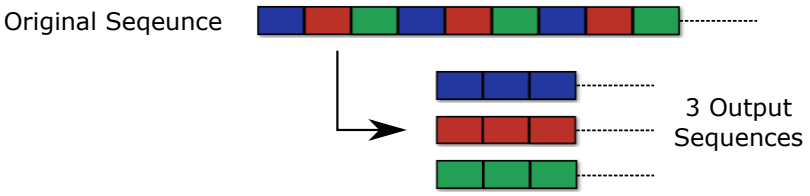

Figure 3.14 – An illustration of sequential recording: the original sequence is shot with a periodically varying illumination. Each of the blue, red and green frames represent one illumination in the original sequence. After extraction we obtain 3 sequences of the same scene, each sequence with a proper illumination.

two sequences of the same scene with two different exposure times. In our method the real exposure time (when the sensor is currently integrating light) is completely controlled by the duration of flashes. Consequently, if consecutive flashes have different durations and if the frame rate is high enough to neglect motion in consecutive frames (or if there is a motion estimation process as in [WGT+05b]), our method can be used for HDR recordings with a single camera. An issue, well addressed in HDR processing, is the difference of motion blur in the different sequence acquired (due to different exposure times). So, changing the power (but not the duration) of the flash between consecutive frames could provide HDR sequences with the same motion blur. An illustration of sequential recordings is shown in figure Fig.3.14, the original sequence is shot with varying illuminations. In this example, three different illuminations are used. Each colored frame represents an illumination. Three different sequences are obtained by extracting frames with the same illumination. The output sequence frame rate is then divided by three, consequently it is very important to maximize the original sequence frame rate.

So, recordings with sequential illuminations are feasible by simply changing the duty cycle of the pulse width modulation at each illumination. But in order to obtain more different illuminations, we also need to be able to change the duration of the flash for each illumination. In figure Fig.3.15 we show how to use the rolling flash method to perform sequential recordings with two different illuminations (by varying Δ_f). Note that the method works with any number of different illuminations. The following equations can easily be adapted to three or more illuminations. The idea is exactly the same, except that the period changes each time a flash is triggered.

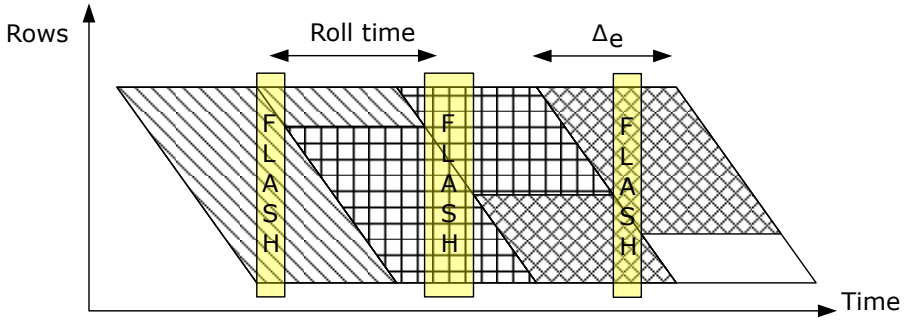

Figure 3.15 – Rolling flash and sequential recording.

Actually, when the i^{th} flash is triggered with a duration Δ_f^i, the time T_f^{i+1} to wait before the next illumination, to avoid intra-rows temporal artifact, is:

$$T_f^{i+1} = \Delta_e + \Delta_f^i, \tag{3.14}$$

where Δ_e is the exposure time. Unfortunately the output frame rate is necessarily lower:

$$F \approx F_c \times \frac{1}{1 + \frac{\Delta_f^1 + \Delta_f^2}{2 \times \Delta_e}}, \tag{3.15}$$

where Δ_f^1 and Δ_f^2 represent the duration of the two different flashes.

6 Experiment results

For our experiments we used a Sony F65 cinema camera at $60Hz$, using its electronic rolling shutter at 360°. We shot a rolling fan because its frequency puts forward temporal artifacts due to periodic illumination. Some images can be found in figure Fig.3.16, on the first column show the raw images from the F65 camera, on the second column we can see those images converted to a linear gamma images. The last image is a reconstructed image from the two previous images. There are no more temporal artifacts in the output image.

6.1 Straightforward application to Photometric Stereo

We conducted another experiment, setting the Sony F65 camera at 120Hz, and using a sequential rolling flash illumination, we were able to apply a straightforward Photometric Stereo as in illustrated in Fig.3.1 to 4K frames. In this section we show qualitative results of our application of photometric stereo to video sequences. On Fig.3.17 we show three consecutive reconstructed frames, from our acquisition with the rolling flash method, corresponding to three different flash illuminations needed to apply PS from [Woo80]. As the acquisition is made at high speed motion between frames is neglectable, however the reconstructed sequence still have some motion artifact as the three illuminations are consecutive and not perfectly aligned. Moreover, the most annoying artifact is caused by flash projected shadows, indeed

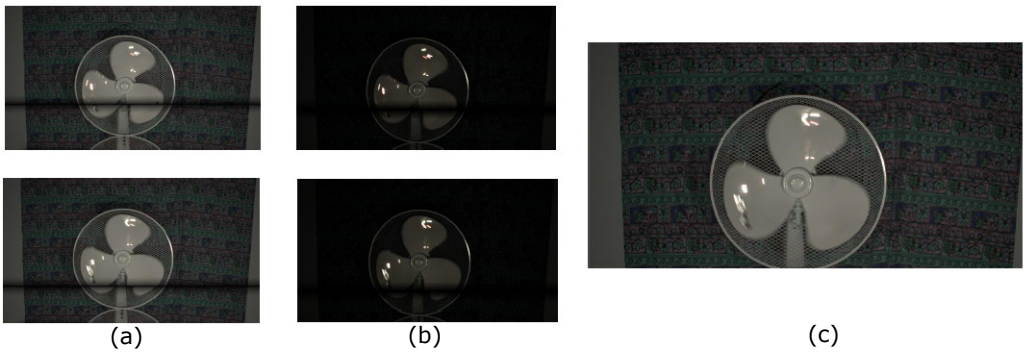

Figure 3.16 – **(a):** Two consecutive input frames from our experiments, **(b):** Linear version of the two input frames. **(c):** Our reconstruction result with gamma correction

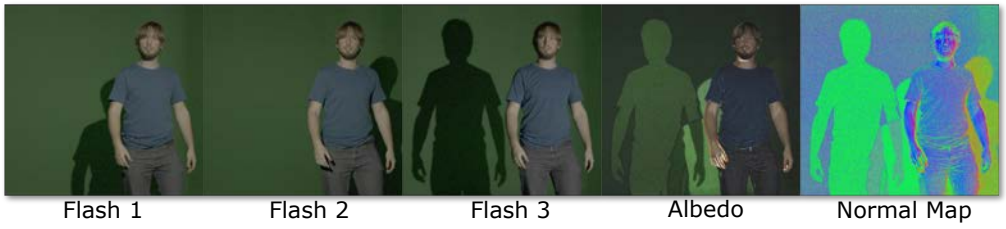

Figure 3.17 – Application of Photometric stereo to one frame. From right to left: Reconstructed images from our rolling flash method corresponding to the three illumination conditions, the reconstructed Albedo and the reconstructed Normal Map.

as the three flashes have three different directions of illumination, the projected shadows in the scene are also different. As a consequence, each part of the scene that is shadowed, in at least one of the three frames used for PS reconstruction, has an erroneous reconstruction as it can be seen in the reconstructed albedo and the reconstructed normal map. Furthermore, the reconstruction of the normal map is highly sensitive to the noise present in the 3 input frames.

Despite those artifacts we were able to render (direct illumination) a quite qualitative relighting of the scene Fig.3.18. In order to render the scene we used a direct illumination (no ray-casting) and the Lambertian shading equation at each pixel.

$$R(u,v) = k_d(u,v) \cdot |N(u,v).\omega_i|, \tag{3.16}$$

Where $k_d(u,v)$ is the albedo at pixel (u,v), $N(u,v)$ is the normal at pixel (u,v) and ω_i is the direction of illumination. In Fig.3.18 we rendered ten images varying the ω_i direction around the scene. Another drawback of this application is the flash strobe annoyance. Indeed, actors can be annoyed by this stroboscopic illumination even at $120Hz$, each flash as a frequency of $40Hz$ which is noticeable by human eye. At least a $50Hz$ per flash frequency is needed to avoid any annoyance.

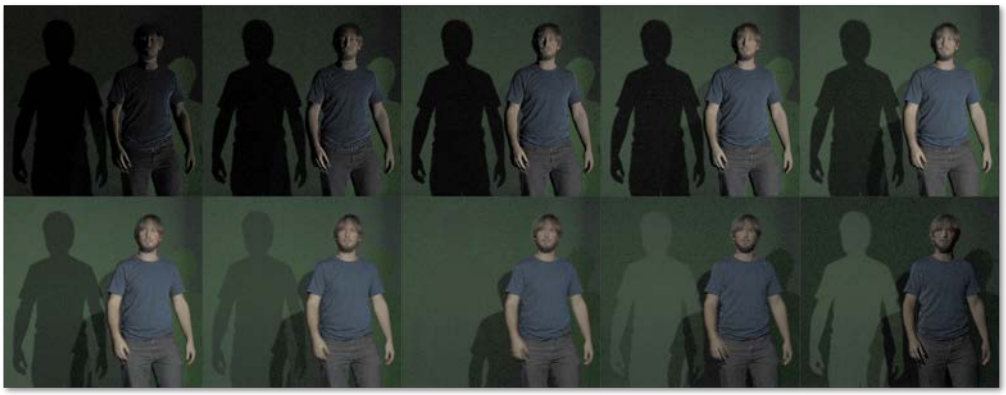

Figure 3.18 – Example of 180° relighting of one frame.

7 Conclusion

We described a model of electronic rolling shutter camera and explained why those types of camera may cause two types of temporal artifact when using stroboscopic flashes. We presented a method that allows to use periodic flashes with these high speed cameras and to remove the two types of artifact presented while maximizing the output sequence frame rate. The method requires a shutter value of 360° and an active illumination setup in an indoor environment. On the other hand, the method needs a very light setup and does not require any synchronization device between the camera and the illumination setup. We described an acquisition framework for timing the stroboscopic illumination and provided a simple algorithm to reconstruct the output video frames. We also described an adapted method to perform sequential recordings with periodic flashes aiming at applications such as photometric stereo. Experiments have been conducted to demonstrate the efficiency of our method to avoid artifacts. Our method relies on flash duration to control the exposure time of the camera. As the flash duration is lower than the exposure time of the camera, motion blur can be avoided. Consequently, when the flash is turned off, the camera still integrates (shutter value of 360°), which can results in a higher noise in the output sequence. We showed qualitative 1920×1080 (Full HD) results concerning the direct application of photometric stereo. We also showed an example of direct relighting of a frame easily applicable to video sequences. However this PS with rolling flash can only be applied indoor in a controlled environment such as a laboratory. Thus this method is non applicable to most common scenes, and cannot be incorporated in a traditional movie framework. Furthermore, when taken with a frame rate under $150Hz$, flash strobes can be very disturbing for actors. Finally, studying the electronic rolling shutter allowed to develop a technique to apply directly photometric stereo to video sequences, however the application framework is very restrictive and cannot be applied to most common scenes.

Shape and Reflectance from RGB-D images using time sequential illumination 4

Contents

1 Introduction **47**
2 General Idea **49**
2.1 Light Source Modeling. 50
2.2 Scene Illumination. 50
2.3 Pure flash image from image pairs. 51
3 Our Approach **52**
3.1 Chromaticity Assumption 53
3.2 Computing Normal Map from Quantified Depth Data 53
3.3 Normal Map Filtering 54
3.4 Diffuse Reflectance Coefficients Estimation and Filtering 54
3.5 Normal Map Refinement 55
3.6 Global Convergence 56
4 Results **57**
4.1 Performance improvement 59
4.2 Application to direct relighting 60
5 Conclusion **61**

1 Introduction

Low-cost RGB-Depth scanners have recently led to a little revolution in computer graphics and computer vision areas with many direct applications in robotics, motion capture and scene analysis. The main concern of such depth sensors is their low accuracy due to noise and their inherent quantization (see raw depth image, rendered with normals and diffuse shading, in fig.4.1). The idea of improving depth using the information contained in the associated RGB image has been widely explored [DT05, RSD+12, NRDR05, WZN+14]. It relies on building a complete model

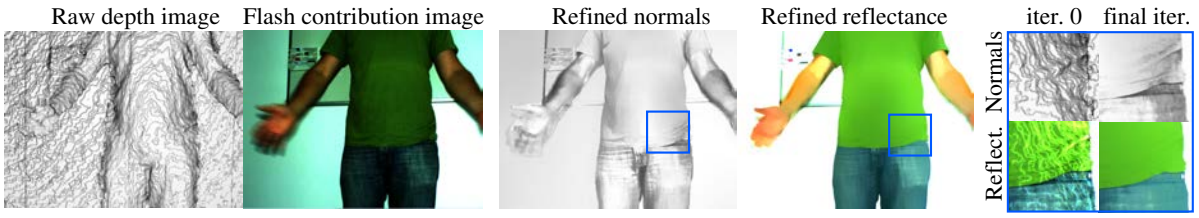

Figure 4.1 – Blue: Our method takes as input a flash image registered with a raw depth map (rendered with Normals and diffuse shading). The flash image is computed using flashed and non flashed image pairs which represent two successive video frames. With these inputs our algorithm use an optimization process to produce refined normal map (rendered with diffuse shading) and reflectance map.

of the scene by estimating and ideally extracting separately materials, 3D shape and illumination. The depth sensor usually provides a rough estimate of the scene geometry which is then refined using lighting and materials (extracted from RGB images) as well as shape from shading-based algorithms. On the other hand, stereo photometry methods have been used for years to extract finer geometry and materials from images of a scene [Woo80, KWB10, Deb12a]. Unfortunately, the use of stereo photometry is inappropriate in most of the shooting scenarios as it requires finer calibration and a complex lighting setup. Therefore stereo photometry methods cannot be easily incorporated into a traditional movie framework. In our new technique, we explore the possibilities given by a hybrid setup consisting of a depth sensor, a RGB sensor and a partially controlled illumination. We target a low-cost and the least intrusive possible setup. Our idea is to use RGB flashed and non-flashed image pairs. After the report of [XDCW01], that showed that, using active light sources, it is possible to measure object information independently of the passive illuminant; [DXW01] had the idea of combining two images, one with general uncontrolled lighting and the other with an additional controlled and known illumination, to obtain an image without general uncontrolled lighting and then estimate the spectral power distribution of the general uncontrolled lighting. More recently [PSA+04] used pairs of flashed and non flashed images for various applications in digital photography, including denoising, detail transfer, white balancing and red-eye removal. To obtain a flash no-flash video sequence, we perform a time sequential illumination by triggering flash illuminations on half the frames of the RGB camera and then extracting two sequences of the same scene: one corresponding to the scene with its natural illumination and no alterations, preserving then the shooting framework, and another containing flashed images fig.4.2. With a proper combination of the two images of an image pair, we create a pure flash image, as if the general uncontrolled lighting had been switched off, which amounts to take a picture of the scene under the flash illumination only [DXW01]. This provides us with a sequence of images with a controlled and simple illumination, which simplifies the ill-posed problem of retrieving separately shapes and diffuse reflectance coefficients from a single image.
Compared to the simple photometric stereo approach from last chapter, only one flash is needed. In this method only one flash has to be triggered, which implies an important gain of strobe frequency and less annoyance to potential actors. Moreover

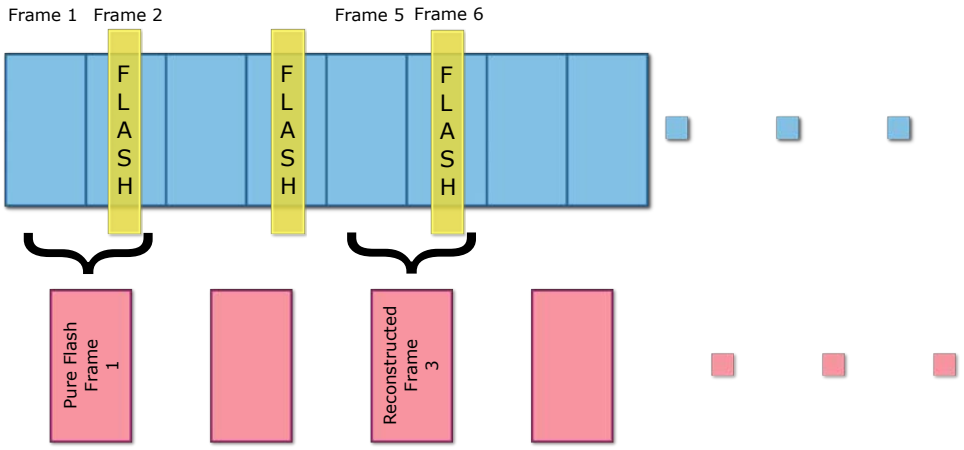

Figure 4.2 – Illustration of flash and non flashed sequential recording and the extraction of a pure flashed image.

in this chapter the flash is not supposed to be a distant light source, so it can be placed anywhere, which is more convenient and results in a very less complex setup compared to photometric stereo.
We use raw depth input to estimate a rough normal map, and our knowledge of the flash illumination to reconstruct high quality normals and diffuse reflectance coefficients for each pixel using a simple iterative mean square optimization. The main contributions of this paper are:

- a new method, to efficiently recover geometry and diffuse reflectance coefficients from image sequences, using a hybrid setup combining sequential illumination, consumer depth sensor and a RGB sensor;
- robustness to multiple albedo scenes;
- near real-time performance.

2 General Idea

Under the assumption of Lambertian scene a photometric stereo-baed method uses three observations (three images) of the same scene, under different illuminations, to compute surface normals and diffuse reflectance coefficients. Now imagine a single albedo scene illuminated by a single known light source. It is possible to estimate a rough normal map with a low-cost RGB-D sensor, and so for each point in the scene it is easy to compute the diffuse reflectance coefficients from the shading equation and the rough normal map. As the measured normals are not perfectly estimated the diffuse reflectance coefficients computed for each point is different rather than similar (errors on normal estimation affect the diffuse reflectance coefficients estimation), but the scene is supposed to be a single albedo scene, which means that all the diffuse reflectance coefficients have to be equal. If we suppose that the errors on normal estimation are equally distributed over the range of possible normal directions, a

single albedo of the scene can be estimated by averaging all the obtained diffuse reflectance coefficients. We now have an estimate of the scene diffuse reflectance coefficients together with a knowledge of the only light source within the scene. The normal map estimate can then be improved using the shading equation. Those new computed normals can be used as input for a novel per-point estimation of diffuse reflectance coefficients. This process can be repeated until convergence.

For natural scenes, we relax the assumption of single albedo to consider multiple albedos by assuming that the reflectance of these scenes is sparse [SY11]. Using chromaticity, we can cluster points in the scene so that each cluster contains points of nearly the same diffuse reflectance. As for illumination, we use pairs of flashed and non flashed images to extract pure flash images [DXW01].

2.1 Light Source Modeling.

We assume that the flash LED light source is small, consequently the light source will be characterized by its intensity I_s:

$$I_s(\omega_i) = L_s \cdot \Delta S \cdot |N(S) \cdot \omega_i| \tag{4.1}$$

where ΔS is the surface area of the flash light source, L_s its emitted luminance, $N(S)$ its normal and ω_i its emission direction. With this assumption, we can easily compute the reflected luminance L of a point P as seen through a pixel in direction ω_c as:

$$L(P, \omega_c) = fr(P, \omega_i \rightarrow \omega_c) \cdot I_s(\omega_i) \cdot \frac{|N(P).\omega_i|}{\|P - S\|^2} \tag{4.2}$$

where fr is the bidirectional reflectance distribution function (BRDF) of the surface, $\omega_i = \overrightarrow{P - S}$ the emission direction of the light source, $N(P)$ the surface normal at P and $\|P - S\|$ the distance between the flash light source and P.

2.2 Scene Illumination.

Let p be a pixel of coordinates (u, v) on the camera sensor (centered coordinates). This pixel can be projected onto the scene as a 3D point $P(u, v)$ expressed in the camera coordinate system by using the camera parameters:

$$P(u, v) = \begin{pmatrix} u.\mathcal{D}(u, v)/f_x \\ v.\mathcal{D}(u, v)/f_y \\ -\mathcal{D}(u, v) \end{pmatrix} \tag{4.3}$$

where (f_x, f_y) are the camera focals and $\mathcal{D}(u, v)$ is the depth value given by the depth sensor expressed in the rgb camera coordinate system. For a Lambertian surface, the luminance of a point P can be expressed as:

$$L(P(u, v), \omega_c) = k_d(u, v) \cdot I_s(\omega_i) \cdot \frac{|N(u, v).\omega_i|}{d^2} \tag{4.4}$$

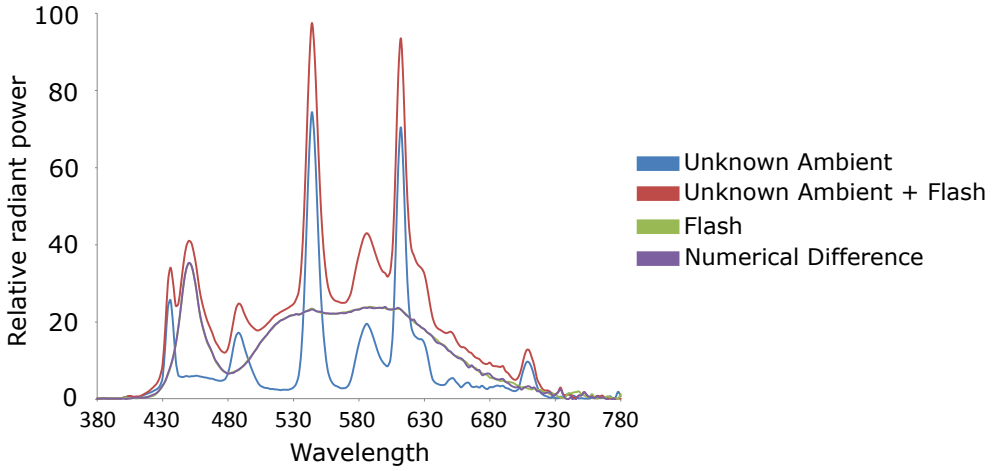

Figure 4.3 – Blue: Spectrum of a white Lambertian point under the general uncontrolled lighting. Red: Spectrum of a white Lambertian point under unknown general uncontrolled lighting and flash illumination (total spectrum). Green: Spectrum of a white Lambertian point under pure flash illumination. Purple: difference between the total spectrum and the general uncontrolled lighting spectrum, this difference spectrum completely matches the pure flash spectrum.

where $d = \|P - S\|$. The diffuse reflectance coefficients k_d (which are the coefficients corresponding to the diffuse part of the BRDF, for each color component RGB) and the surface normals N are stored into two different 2D buffers. From now on, we will use $p = (u, v)$ and $L(P(u, v), \omega_c) = L(p)$ for each RGB component c.

2.3 Pure flash image from image pairs.

Our goal is to create a pure flash image from a pair of two images: a flashed and non-flashed one (general uncontrolled lighting image, also referred to as ambient image). As we record our images with a time sequential illumination we use the same aperture and exposure time for the two images. The flashed image can be recovered by subtracting the ambient image from the flashed one, provided that the images are linear and do not contain any underexposed and saturated pixels. As shown in Figure 4.3, to validate this substraction (to compute the pure flash image) we have captured a white Lambertian point under several illuminations with a spectrometer and subtracted the spectrum of the general uncontrolled lighting from the total spectrum (obtained after trigging the flash). This results in a spectrum that matches the spectrum obtained with a pure flash illumination. To make sure that the combination of the two images of a pair provides a pure flash image, three caveats should be considered:

- the two images must be taken with the same camera parameters (exposure time, aperture, focal length),
- the images have to be linear,
- the pixels color should not be saturated nor underexposed in the two images.

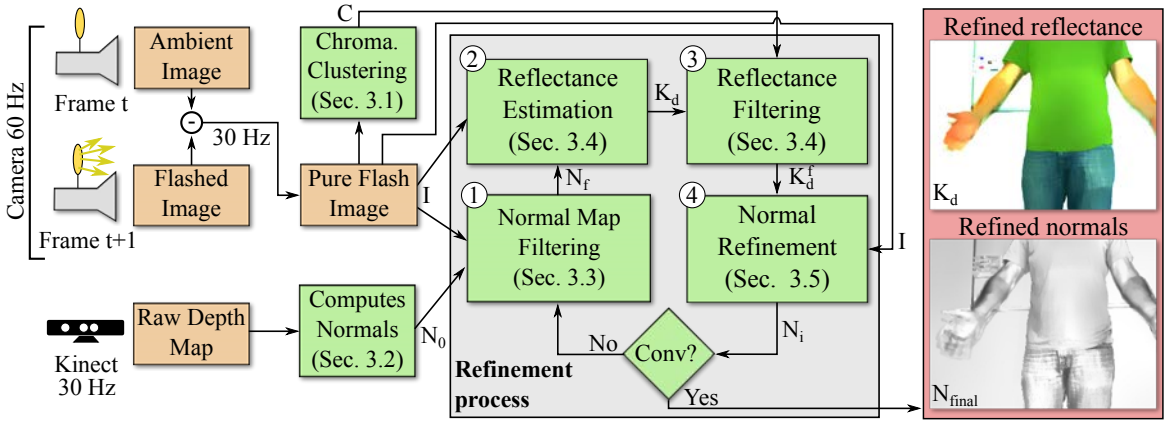

Figure 4.4 – Our framework picture. Green boxes represent the different processings of our algorithm. Orange boxes represent the values given by the sensors at different times. Red box is the result of our algorithm.

Moreover, as a luminous power decreases with the inverse squared distance, objects too far away might not receive enough light energy from the flash illumination. This restraints the scenario setup to scenes not too far from the camera.

3 Our Approach

Our approach is summarized in Figure 4.4. First, our hybrid setup (a camera, a Kinect and a Flash) is completely calibrated to register the Kinect depth image to the RGB camera. The illumination of the scene is known (pure flash image) thanks to our time sequential illumination. Moreover, as we also know the extrinsic and intrinsics parameters of our setup, we can project the depth map (provided by the Kinect) onto a 3D point map in the camera coordinate system. Before refining the normal and the diffuse reflectance coefficients maps, we compute a rough normal map from the raw depth map provided by the Kinect (Section 3.2). Then we use the pure flash image to cluster (K-means clustering) the 3D points seen through the pixels camera, each cluster containing points with nearly the same diffuse reflectance coefficients (Section 3.1). Then we start our iterative refinement process consisting of 4 steps. In the first step (Section 3.3), we filter the normal map, the weights depending on luminance to preserve geometry details and coherence. The second step (Section 3.4) performs an estimation of a diffuse reflectance coefficients map from the pure flashed image and the filtered normal map. In third step (Section 3.4), a filter is applied to the diffuse reflectance coefficients map with weights depending on chromaticity. During the last step (Section 3.5), the normal map is refined thanks to a shading least square minimization that allows to fit our model to the pure flash image. Finally, we repeat those steps until convergence, say when both the normal and diffuse reflectance coefficients maps do not vary anymore. Now, we will detail the different steps in the following subsections. For convenience purpose, as the images and the depths are registered, from now on, a point is either a pixel p of an image captured by the camera or the 3D point P seen through p.

3.1 Chromaticity Assumption

As in [DXW01], we suppose that the chromaticity of the scene is sparse. This assumption allows us to perform a color segmentation of the scene based on quadratic chromaticity distance. The segmentation consists in applying a K-means clustering to the input image so that each pixel is assigned a cluster of a given chromaticity (fig.4.5). Each pixel is projected onto a 3D point with a diffuse reflectance. Each cluster is supposed to contain pixels with nearly the same diffuse reflectance coefficients. We initialize the K-means centers (10 in our current implementation) by spreading them in the chromaticity gamut. More robust clustering techniques are left for future work.

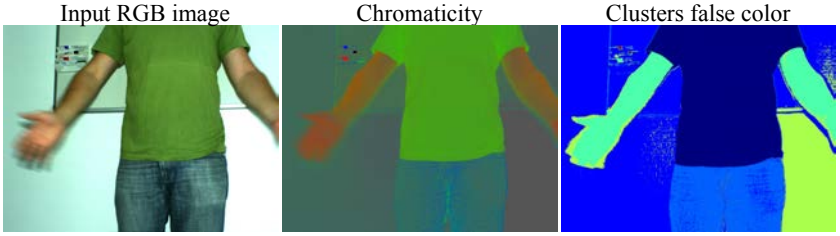

Figure 4.5 – The chromaticity image is used to cluster surfaces with similar diffuse reflectances. We can observe that the t-shirt, the background and the skin are classified into different clusters.

3.2 Computing Normal Map from Quantified Depth Data

Let us use the raw depth map (captured by the Kinect) to compute $N(p)$, the normal associated with a pixel p. To compute this normal, we need to express the depth changes $(\delta_x(u,v), \delta_y(u,v))$ as follows:

$$\delta_x(u,v) = \mathcal{D}(u+1,v) - \mathcal{D}(u-1,v) \tag{4.5}$$
$$\delta_y(u,v) = \mathcal{D}(u,v+1) - \mathcal{D}(u,v-1) \tag{4.6}$$

The normal, associated with the pixel p, can be estimated as:

$$T_x(p) = \begin{pmatrix} 2 \cdot \frac{\mathcal{D}(p)}{f_x} & 0 & -\delta_x(p) \end{pmatrix}^T \tag{4.7}$$
$$T_y(p) = \begin{pmatrix} 0 & 2 \cdot \frac{\mathcal{D}(p)}{f_y} & -\delta_y(p) \end{pmatrix}^T \tag{4.8}$$
$$N(p) = T_x(p) \times T_y(p) \tag{4.9}$$

where $T_x(p)$ and $T_y(p)$ are the tangents to the surface according to the X and Y axes respectively. Note that, as the Kinect depth map is quantized, it represents a piece-wise constant approximation of the real depth map (see fig.4.1). Due to this quantization artifact, most of the normals, computed from this depth map, will be oriented toward the camera, which is a poor initial guess for our normal map refinement algorithm. This quantization artifact makes the normal map noisy. Consequently, to overcome the quantization artifact and other possible artifacts, an important part of our algorithm consists in filtering the normal map. This filtering is detailed in the next subsection.

3.3 Normal Map Filtering

To remove noise in the normal map, we apply a joint-bilateral filter (as introduced by [PSA+04]) to the normals N (computed from the depth map) to get filtered normals N_f:

$$N_f(p) = \frac{1}{W_n(p)} \sum_{s \in \Omega(p)} \Psi_N(s,p)\, N(s) \tag{4.10}$$

where $W_n(p)$ is a normalization factor, $\Omega(p)$ a neighborhood of the pixel p and Ψ_N the weighting function. The expression of the weighting function depends on the luminance L of the pixels of the pure flash image and on the normals:

$$\Psi_N(s,p) = \exp\left(-\frac{(L(p)-L(s))^2}{2\sigma_l^2} - \frac{\|N(p)-N(s)\|^2}{2\sigma_n^2}\right) \tag{4.11}$$

where σ_l and σ_n are weighting parameters related to luminance and normal respectively. Numerical values used in our experiments are given in the results section. The reason of adding the luminance information in the weighting function is to preserve geometry details, as the luminance value changes locally with the normal orientation.

3.4 Diffuse Reflectance Coefficients Estimation and Filtering

Let us consider the case of Lambertian surfaces. The RGB pixel values of p captured by the camera can be estimated as follows:

$$I(p) = \frac{I_s(\omega_i)}{d^2} \cdot k_d(p) \cdot |N(p).\omega_i| \tag{4.12}$$

I_s, d, ω_i are known as the source illumination is controlled, $k_d(p)$ is the diffuse reflectance coefficient of a Lambertian surface. k_d is commonly expressed for each $\{r,g,b\}$ component. Refining the normal map requires an estimation of those diffuse reflectance coefficients.

Once the first rough estimation of the normals has been performed, Equation 4.2 is used to estimate the diffuse reflectance coefficients map:

$$k_d(p) = \frac{d^2}{I_s(\omega_i)} \cdot \frac{I(p)}{|N(p).\omega_i|}, \tag{4.13}$$

where $I(p)$ is the $\{r,g,b\}$ value of a pixel of the pure flash image and I_s the intensity of the flash light source. The diffuse reflectance coefficients map is rough as it is computed from a rough depth map. However, the pure flash image can help improve the diffuse reflectance coefficients map, provided that the two following assumptions are satisfied:

1. if two points have the same normal, the difference between their pixel values is only due to albedo change,

2. the distribution of reflectance over the image is sparse to ensure a reliable segmentation (Section 3.1).

According to the two above assumptions, any albedo change is related to chromaticity change [DXW01], in other words any chromaticity change entails an albedo change. Consequently, the impact of normal aberration, on the diffuse reflectance coefficients map, can be reduced by averaging the diffuse reflectance coefficients of points lying in a neighborhood. This averaging operation is performed using another joint-bilateral filter:

$$k_d^f(p) = \frac{1}{W_d(p)} \sum_{s \in \Omega(p)} \Psi_d(s,p)\, k_d(s) \tag{4.14}$$

where $W_d(p)$ is the normalization factor, Ψ_d being the weighting function which depends on chromaticity similarities. Indeed, if two points are not assigned the same chromaticity cluster (Section 3.1), their weights are set to zero. The condition that two pixels p and s belong to the same cluster is:

$$C(s,p) = (C(s) = C(p))\ and\ (|m(s) - m(p)| < t_m) \tag{4.15}$$

where $C(p)$ is the cluster id associated with pixel p, $m(p)$ is the maximum of the r, g, b values of the pixel p and t_m a threshold value that we set to 0.5. The second term is added to make the distinction between black and white points. Finally, the expression of the weight used in Equation 14 is given by:

$$\Psi_d(s,p) = \begin{cases} exp\left(\frac{-\|m(s)-m(p)\|^2}{2\sigma_m^2}\right) & \text{if } C(s,p) \\ 0 & \text{otherwise} \end{cases}$$

where σ_m is a weighting parameter. Numerical values used in our experiments are given in the results section. To avoid that black, white and grey pixels mingle during the filtering process, we use a weight which depends on the maximum m of the r, g, b values of each pixel.

3.5 Normal Map Refinement

Once the diffuse reflectance coefficients is filtered, the next step consists in refining the normal map. Our refinement relies on the lambertian model and a least-square error like algorithm for the three channels of each pixel. Equation 4.4 can be written as:

$$I(p) = \frac{I_s(\omega_i) \cdot k_d(p)}{d^2} \cdot (N_x\omega_x + N_y\omega_y + N_z\omega_z) \tag{4.16}$$

Let us assume that the right k_d diffuse reflectance coefficients are available, the goal is to find the three components of normal N that minimize ξ over the set of the three rgb components:

$$\xi(p) = \left(\sum_c (\mathcal{S}(p,c) - (N_x\omega_x + N_y\omega_y + N_z\omega_z))\right)^2$$

$$\mathcal{S}(p,c) = \frac{I(p,c) \cdot d^2}{k_d(p,c) \cdot I_s(\omega_i)}$$

where $k_d(p,c)$ and $I(p,c)$ are respectively the diffuse reflectance coefficient and the value of pixel p for the color channel c. We want to find the minimum error with respect to (N_x, N_y, N_z), which is reached when:

$$\begin{aligned}
\frac{\partial \xi}{\partial N_x}(p) = 0 \rightarrow N_x &= \frac{\sum_c \left(\mathcal{S}(p,c) - N_z\omega_z - N_y\omega_y\right)}{3 \cdot \omega_x^2} \\
\frac{\partial \xi}{\partial N_y}(p) = 0 \rightarrow N_y &= \frac{\sum_c \left(\mathcal{S}(p,c) - N_x\omega_x - N_z\omega_z\right)}{3 \cdot \omega_y^2} \\
\frac{\partial \xi}{\partial N_z}(p) = 0 \rightarrow N_z &= \frac{\sum_c \left(\mathcal{S}(p,c) - N_x\omega_x - N_y\omega_y\right)}{3 \cdot \omega_z^2}
\end{aligned}$$

In the minimization process, each normal $N = \begin{pmatrix} N_x & N_y & N_z \end{pmatrix}^T$ is initialized with the rough normal map computed from he Kinect depth data, then each component of the normals is computed through an iterative scheme until convergence of the normal map refinement.

3.6 Global Convergence

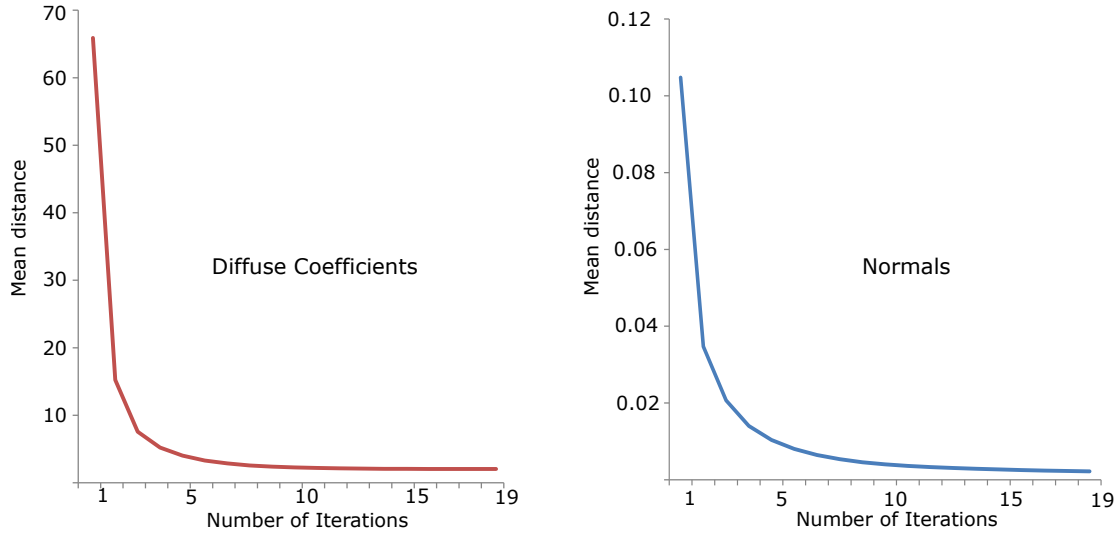

Figure 4.6 – Plots representing the convergence of Normals and diffuse reflectance coefficients for each iteration

The plots on fig.4.6 illustrate the convergence of the Normal Map and the diffuse reflectance coefficients map when applying our iterative algorithm. We computed the mean distance between the normal map (respectively the diffuse reflectance coefficients map) evaluated at the current iteration and the normal map (respectively the diffuse reflectance coefficients map) evaluated at the preceding iteration. fig.4.6 demonstrates that only a few iterations are necessary to reach a steady state. Indeed, only $4-5$ iterations are necessary to reach a high quality Normal and diffuse reflectance coefficients maps.

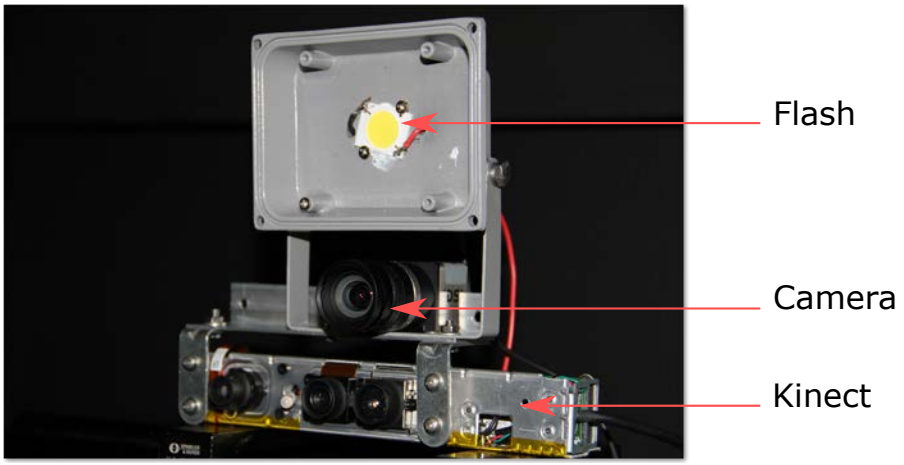

Figure 4.7 – Our experimental setup, from top to bottom: white flash LED light, Ueye Industrial Camera and Kinect depth sensor.

4 Results

Our experimental setup consists of: a Kinect depth sensor (depth res. 640×480, frame rate 30 fps), a Ueye Industrial Camera (res. 1280×960, adjustable frame rate, form 15 fps to 120 fps with a global shutter) and a flash LED light fig.4.7. An electronic board is used to synchronize the camera with the flash. This allows to sequentially acquire an image pair: flashed and non flashed. So, we are able to generate an image pair at a maximum of 60 fps. However, the higher the frame rate of the camera, the shorter the flash duration, thus the harder part is to obtain a suitable pure flash image (see subsection.2.3). The Kinect is not synchronized due to its own limitations. The depth image is upsampled to match the camera resolution.

We have implemented our technique in C++ and used CUDA 7 to speedup all the steps of our algorithm. The details of the algorithm timings are summarized in Table 4.1. All the timings in this table have been measured on a Xeon X5680 CPU 3.33GHz (12GB Ram) and an Nvidia Geforce GTX 980. The algorithm parameters were set to: $\sigma_m = 0.002$, $\sigma_l = 0.0004$ and $\sigma_n = 0.02$, these values were carried throughout all experiments. For the two filtering operations (normal and diffuse reflectance coefficients maps), we use a kernel of 20 pixels size except for the **Burger scene** (fig.4.8) for which we use a kernel of 10 pixels size. This is due to the fact that the image of this scene is twice smaller than that of the other test scenes (fig.4.1 and 4.9).

Our method is compared to the one proposed in [OERW+15]. To this end, we used the Matlab code provided by the authors. Their method aims at enhancing a depth map by fusing intensity and depth information to create detailed range profiles. For this purpose, they use a lighting model that can handle natural illumination. This model is integrated into a shape-from-shading technique to improve the reconstruction of objects. Note that, unlike our method, their approach refines the depths rather than the normals.

Fig.4.8 shows a comparison between our method and the one of [OERW+15] for

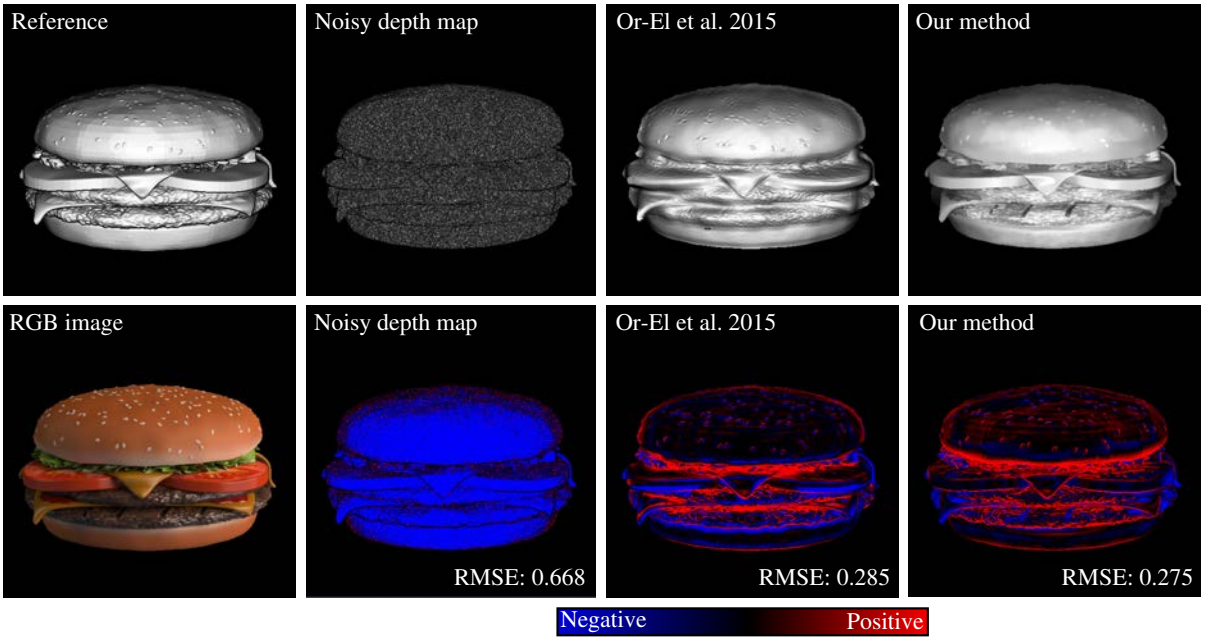

Figure 4.8 – The original burger scene used in [OERW+15]. Top row shows the dot product image (dot product between the normal and the view direction) using the normals of: the reference solution, the noisy map used as input, [OERW+15] and our method. Bottom row shows the RGB image and the false color error on the normal dot product. For better visualisation the error was multiplied by 3.

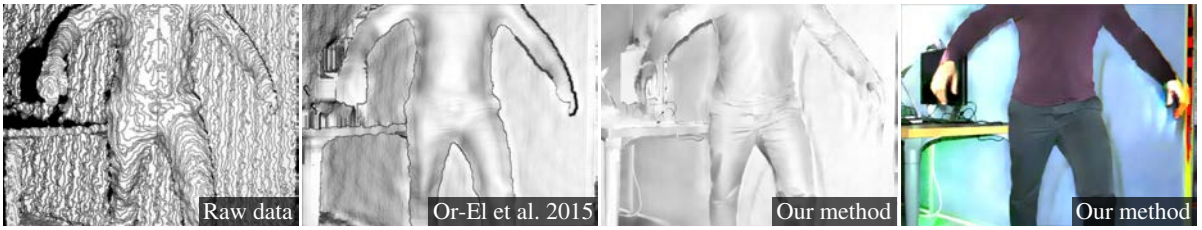

Figure 4.9 – T-Shirt scene captured by our setup. From left to right : the normal maps corresponding respectively to the raw input depth map, the [OERW+15] processing and our refinement method. The fourth image is the diffuse reflectance coefficients map obtained with our method.

a synthetic **Burger scene**. The input depth map is perturbed by adding a gaussian white noise. To use our algorithm, we rendered the scene using a small light source that simulates the flash. Our method produces results with an error smaller than the one obtained with the method of [OERW+15]. This is particularly visible on the tomatoes where fine details are well recovered. However, there is more noise on the bread due to clustering issues (fig.4.10).

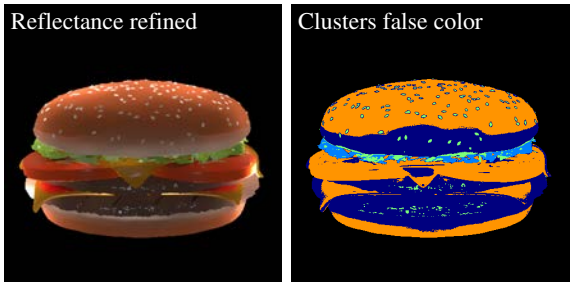

Figure 4.10 – On the left: the refined diffuse reflectance coefficients map after convergence of our algorithm. On the right: false color image to show how the chromaticity-based clustering performs. We can observe that the tomatoes and the bread lie in the same cluster. However, thanks to the local nature of our filtering, these two materials have not been merged.

The scene (fig.4.9), is a real scene with relatively slow arm movements. Unlike the method of [OERW+15] no mask is used to select the main object. The input depth map is noisy as it is directly provided by the Kinect sensor. This noise is different from the synthetic one used in fig.4.8. Compared to [OERW+15], our method provides results that have less artifacts and fine geometry details are correctly recovered. The main issue of [OERW+15] method is the use of a bilateral filter which is applied to filter a lot of data due to the high level of noise in the raw depth map. Our two filtering operations are performed using bilateral filters. But, as we use them at each iteration, a smaller kernel size is needed, which reduces the artifacts inherent in large kernel bilateral filters. Furthermore, at each iteration we compute new normals (Normal Refinement operation) and new diffuse reflectance coefficients (Reflectance Estimation operation). Consequently, artifacts due to successive bilateral filtering operations are avoided.

4.1 Performance improvement

One of the bottleneck of our algorithm is the two filtering operations (Normal Filtering and Reflectance Filtering) which take 92.205 ms and 120.405 ms respectively. These two operations use non separable filters, which are time consuming. However, to reduce the computing time, those filters can be approximated by separable filters with weights carefully computed. The error due to those approximations are hardly visible. The computing times needed by the separable version of these two filters are 5.409 ms (22.26× faster) and 4.702 ms (19.60× faster). All the timings are summarized in Table 4.1. Except for the **Burger scene**, due to the reduced image size, this optimization has been used for all the scenes (fig.4.1 and fig.4.9). Better and more robust optimization is left for future work.

Table 4.1 – Computation times of each step of our algorithm. Bold timings are obtained using the performance improvement described in subsection 4.1. To compute the total time, we need to multiply the refinement iteration time by the number of iterations needed to converge to the desired results. In practice for a 1280×960 image, only 5 iterations are required, which corresponds to 12.88 fps or 0.91 fps on a GTX 980 respectively using or not using the performance improvement. For a 640×480 image (kinect size), we reach 73.8 fps and 3.58 respectively using or not using the performance improvement.

Operations		Time (ms)			
		640×480		1280×960	
Init.	Image Alignment	1.143		2.973	
	Chroma Clustering	1.339		5.052	
	3D Points Estim.	0.044		0.156	
Refine iter.	Normal Filtering	22.604	**(1.133)**	92.205	**(4.702)**
	Dot Calculation	0.086		0.335	
	Reflectance Estim.	0.053		0.992	
	Reflectance Filtering	32.001	**(1.069)**	120.405	**(5.409)**
	Normal Refinement	0.552		2.452	
Total Refinement Iteration		55.295	**(2.202)**	216.387	**(13.89)**
Total Time (5 iterations)		279.001	**(13.536)**	1090.116	**(77.631)**

4.2 Application to direct relighting

One direct application of our algorithm is the relighting of a captured scene. Indeed, our algorithm provides the diffuse reflectance coefficients and the normal maps of a scene that we relight with artificial light sources (fig.4.11). In this scene, the movement of the arms is fast. This explains the artifacts on the arms due to motion blur.

Figure 4.11 – The normal and diffuse reflectance coefficients maps refined by our algorithm can be used for relighting a scene. Images Relighting 1 & 2 are obtained with different artificial light source positions. Moreover, sequential lighting makes our technique capable of capturing video sequences. However, fast and large movements in the video could create artifacts due to motion blur.

5 Conclusion

We showed that even with a consumer camera and a Kinect (capturing noisy depth data) it is possible to recover finer geometry and precise diffuse reflectance coefficients from an image or a video thanks to the use of sequential illumination provided by a flash. A pair of two images are captured: one non flashed image (image under general uncontrolled lighting) and a flashed one. A pure flash image is computed by substracting the non flashed image from the flashed image. We proposed an efficient iterative algorithm to recover shapes and diffuse reflectance coefficients from the pure flash image. The fact of knowing the illumination (flash light source) makes the extraction of normals and diffuse reflectance coefficients easier and more efficient. Indeed, as the position and the photometry of the flash light source is known, we used a local illumination model to express the normal and the diffuse reflectance coefficients for each pixel. From the computed normals we used the illumination equations to determine the reflectances. In turn, these diffuse reflectance coefficients are fed to a process that determines new normals. This process is repeated until convergence. We showed that only a few iterations are needed to converge to the desired results.

Automatic Light Compositing using Rendered Images 5

Contents

1 Introduction **63**
2 Overview **64**
2.1 RGB-D Acquisition and refinement 65
2.2 Rendering 66
2.3 Genetic Algorithm 66
3 Results **67**
4 Conclusion **69**

1 Introduction

Lighting is a key element in photography. Professional photographers often work with complex lighting setups to directly capture an image close to the targeted one. Some photographers reversed this traditional workflow. Indeed, they capture the scene under several lighting conditions, then combine the captured images to get the expected one. Acquiring such a set of images is a tedious task and combining them requires some skill in photography. We propose a fully automatic method, that renders, based on a 3D reconstructed model (shape and albedo), a set of images corresponding to several lighting conditions. The resulting images are combined using a genetic optimization algorithm to match the desired lighting provided by the user as an image.

Lighting is of foremost importance in photography. It can not only make a difference between a poor and a great photography, but also conveys an artistic and aesthetic point of view of the photographer. One of the main skill of a photographer is his ability to tune the lighting to produce an image that best matches his intent. Professional photographers usually rely on complex lighting setups: a set of flashes with light modifiers such as softboxes, reflectors, etc. However, the task of setting up a "good" lighting for a scene is not only artistically challenging but can also be tedious due to the large number of parameters associated with each light source (position, power, color, size, diffuser, etc.) That is why an extensive literature has been devoted to transfer the lighting style from a target to an input image

[HLMCB15] [RAGS01]. The main goal in those works is to automatically transfer the style of a target image to an input image. In addition, photo editing softwares have been developed to improve input images, their main problem is the difficulty of modifying the lighting once a photograph is taken.

That is why some photographers reversed the traditional workflow: rather than setting up a complex lighting for a single photo, they take several photographs of a scene by moving around a single light source. Then, they fuse the captured images to get the expected final image that could be hardly obtained when taking a single photograph with a complex lighting setup. This approach has first been proposed by [Hae92], then taken over by [BPB13] who introduced a set of optimizations to help even novice photographers to easily create compelling images from an original set of images with varying illumination. However, the process is still user-driven.

This is why we propose, in this chapter, a fully automated framework which also relies on the use of a set of images to compute an image with a certain lighting style. Our approach allows the user to choose a target image corresponding to a desired lighting style. Then it reconstructs the geometry and the albedos [WZN+14] [OERW+15] [HGK+16] of the scene's objects, then uses the reconstructed 3D model to render a set of input images with varying illuminations, which avoids the tedious acquisition of several photographs and makes possible to handle moving objects. Afterwards, it makes use of a global optimization algorithm to find a weighted combination of our set of images that matches the desired lighting style. Our main contributions are:

- an automatic method based on a global optimization algorithm to fuse a set of images (resulting from the rendering of the 3D recovered model) to obtain complex lighting;

- the description of the desired style using a target image as in color transfer.

2 Overview

The main objective of our method is to make easy the production of an image with a given lighting style, based on images fusion.

First, while other methods require a set of images shot from a single point of view but with different lighting setups, we only use a single flashed RGB-D acquisition, which allows to handle dynamic scenes.

The output of this acquisition is a 3D model: shape and albedos.

Secondly, we render a set rendered images of the scene lit with a single point light. In order to achieve a photo-realistic quality we use a ray-tracing algorithm. The rendering engine is configured so as the rendered images are well-exposed. The set of rendered images is the first input of the fusing engine.

Thirdly, we automatically fuse the rendered images to obtain the final image with a given lighting style. The target lighting style is described by a target image. The final image I_f is expressed as a linear combination of the images in the input

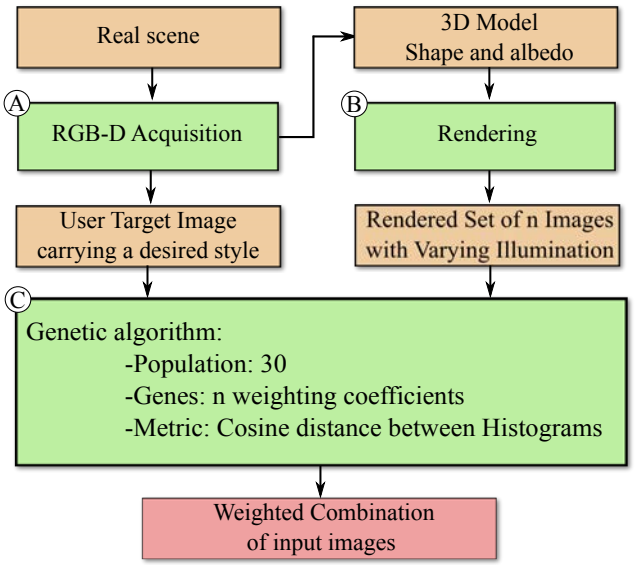

Figure 5.1 – Main Framework

set $\mathbb{S}$:

$$I_f(c) = \sum_{i=1}^{|\mathbb{S}|} c_i I_i, \tag{5.1}$$

where c_i is the weighting coefficient associated with the i^{th} image and $|\mathbb{S}|$ is the cardinal of $\mathbb{S}$. As explained in [MGPFS08], luminance histograms can be used to express an image aesthetics. In our method a lighting style is represented by an image luminance histogram (ILH). The difference between two lighting styles is expressed as the distance between the two corresponding ILH. The optimization of this distance results in an optimal set of weighting coefficients. We use a histogram cosine distance as proposed in [Cha07]. Finally, the final resulting image weights c can be found by minimizing:

$$\arg\min_{c} ||H_L(I_t), H_L(I_f(c))||_{Cosine}, \tag{5.2}$$

where $H_L(I)$ is the luminance histogram of the image I and c is the set of weighting coefficients to be optimized. As the number of coefficients in c to be optimized can be high, a gradient-based descent minimization is inappropriate, that is why we chose a genetic algorithm.

2.1 RGB-D Acquisition and refinement

We use the refinement process described in our previous work [HGK+16] to recover albedos and point-based 3D model of the scene. The approach makes use of a hybrid setup (a camera, a Kinect and a Flash) completely calibrated to register the Kinect depth image corresponding to the RGB camera. A pair of two images are captured: one non flashed (image under general uncontrolled lighting) and a flashed one. A

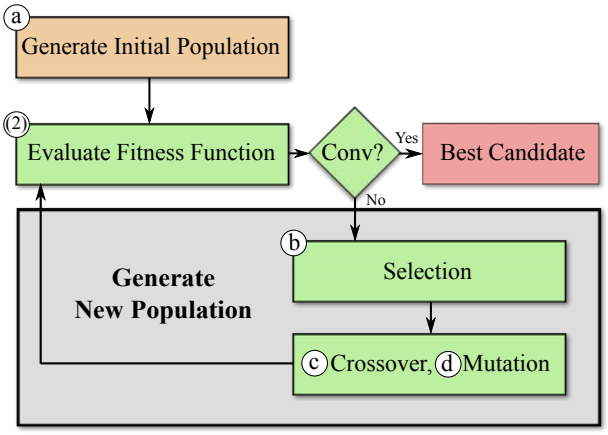

Figure 5.2 – Pipeline of the genetic algorithm.

pure flash image is computed by subtracting the non flashed image from the flashed image. The method then uses this known illumination to compute and refine the normal and the reflectance maps, based on a local illumination model of the flash and the pure flashed image. This method is all the more efficient for still scenes, which is the case in our method, as the pure flash image does not suffer from artifacts due to motion in the scene (ie motion between general uncontrolled lighting and flashed image). Furthermore using flash and no flash image pairs is very convenient when it comes to recover shape and albedo for scenes with unknown and general uncontrolled lighting.

2.2 Rendering

Our previous work [HGK+16] provides us with a 3D point cloud of the scene with refined normals. To render the scene we assign a splat with each point of the point cloud [RL00], each splat is oriented accordingly to the refined normal of the 3D point. Then we ray trace the splats to get images as described in [WWB+14]. More realistic soft shadows are obtained through bilateral filtering.

2.3 Genetic Algorithm

In this section we describe minimization process. The genetic algorithm is a search heuristic that mimics the process of natural selection. It is very useful for the optimization of under-determined problems. The pipeline of our genetic algorithm is described in Fig. 5.2.

A gene is a weighting coefficient c_i to be optimized. A candidate is an individual consisting of $|\mathbb{S}|$ genes, $|\mathbb{S}|$ is the number of input images. The used fitness function is the cosine histogram distance Eq. 2. The population is a set of k candidates (In our experiments, we use $k = 30$).

a) Initialization

To unsure a good distribution the of weighting coefficients over the initial population, the values of the weighting coefficients assigned to each candidate, are initialized

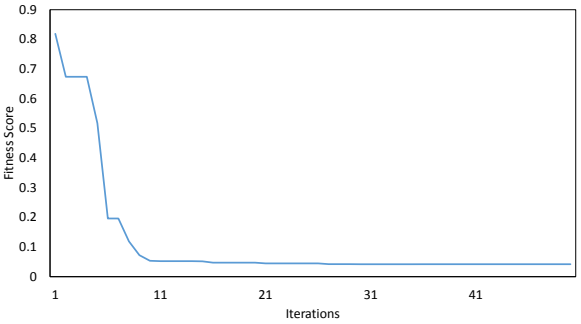

Figure 5.3 – Typical convergence curve for the original set $\mathbb{S}$ Fig. 5.4

with random values. Furthermore, these coefficients are normalized and scaled by a random factor. This prevents the creation of inconsistent candidates:

- Over-exposed images: corresponding to high weighting coefficients;
- Under-exposed images: corresponding to low coefficients.

b) Selection
During each successive generation, a fitness-based selection of candidates breed a new generation. The fitness function, based on the cosine distance between ILH, assigns each candidate a score, then individuals are selected by tournament (non-stochastic tournament so as same candidates can be selected multiple times). Finally the selected individuals are used to breed a new generation, by including mutation and crossover.

c) Crossover
The genes of two individuals are randomly mingled to breed a new generation individual. In our implementation, each of the selected candidates undergoes a crossover with a probability of 0.25.

d) Mutation
When mutating one individual can either be completely regenerated with a probability of 0.25 or its genes are altered randomly. Mutation is used to maintain genetic diversity in the population, which amounts to modifying or creating new individuals to avoid local minimum in the optimization process. Each of the selected candidates undergoes a mutation with a probability of 0.25.

3 Results

We have performed several tests to evaluate the quality and the accuracy of our method. We show numerical results that validate the convergence of the genetic algorithm as well as qualitative results to assess the quality of output images.

In a first experiment, to test the convergence of our genetic algorithm, we have acquired a set of 12 images (Fig. 5.4) of a static scene from a single point of view, with light source moving around as similarly to [BPB13]. From this set $\mathbb{S}$ of images

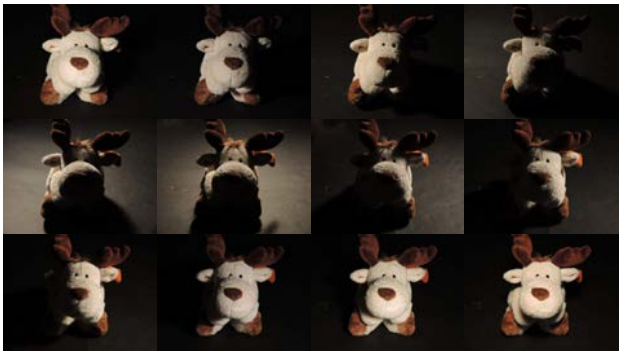

Figure 5.4 – Original real set $\mathbb{S}$ of 12 images

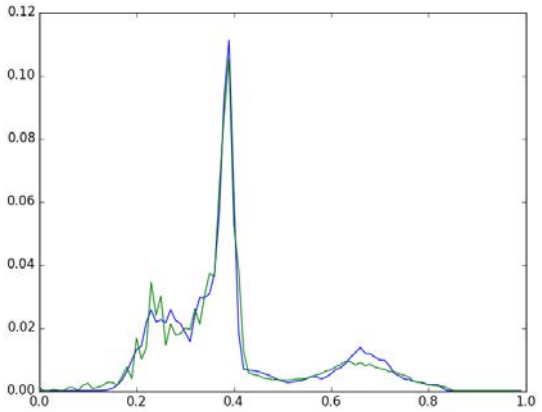

Figure 5.5 – Green: Luminance Histogram of the target image, Blue: Luminance Histogram of the best candidate after 100 generations.

a target image I_t is computed using a given vector c^t of weighting coefficients:

$$I_t = \sum_{i=1}^{|\mathbb{S}|} c_i^t I_i, \tag{5.3}$$

where c_i^t is the coefficient corresponding to the i^{th} image I_i. Using this target and a set $\mathbb{S}$ as inputs, we run our algorithm to find to optimal coefficients c^b. Then the validation consists in comparing the two sets c^t and c^b using the euclidean norm $||c^t - c^b||_{L^2}$, which is on average equal to 0.09 after 100 iteration. Fig. 5.5 shows a comparison between the luminance histogram of the target image and the one of the best candidate image after 100 iterations of the genetic algorithm. The fitness score of the best candidate is 0.0146. Fig. 5.6 shows the best candidate image after 100 iterations as well as the target image. This two images are visually close to each other, which validates our method. On Fig. 5.3 we plotted a typical curve of the best candidate's fitness score for each iteration. The curve shows a fast convergence in the first few iterations, 95% of the final fitness score is obtained in less than 15 iterations.

We have conducted a second experiment as follows. Two target images are used in this experiment. The first one is computed as in the first experiment but with another scene (using real images of the scene lit with real light sources). The second

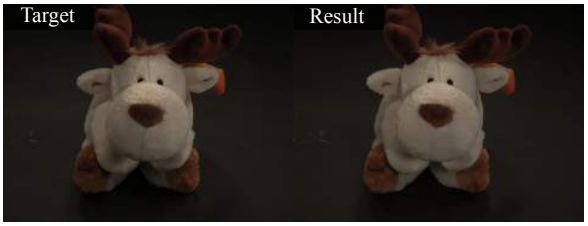

Figure 5.6 – Left: Target Image, Right: Result image after 100 iterations

Figure 5.7 – **Top**, *from left to right*: target image created from a real set of images, result after 20 iterations using a real set of images, result after 20 iterations using a rendered set of images; **Bottom**, *from left to right*: a given target image (independent of the scene), result after 20 iterations using a real set of images, result after 20 iterations using a rendered set of images

one is a given image independent of the scene. For this experiment we have also computed a set of rendered images from a reconstructed 3D model of the scene, using virtual light sources placed at the same positions as those of the real light sources. Fig. 5.7 shows results obtained with both real and rendered set of input images for the two different targets images. The result confirms the efficiency of our approach regarding lighting transfer.

We have also tested our algorithm for a set of virtual images, and we used an aesthetic image as target (Fig. 5.8). This experiment qualitatively demonstrates the efficiency of the the genetic algorithm.

4 Conclusion

We have presented an approach based on image fusion that simplifies the process of producing images with a complex lighting. The main features of our approach are: (A) 3D model acquisition of the scene, (B) rendering of a set of images corresponding to various key lighting conditions and (C) automated fusion (using a genetic optimization algorithm), of rendered images to obtain a lighting style close to the one provided by a target image. The main benefits of our approach are: (1) it is totally automated while related approaches are user-driven (in our case the user only provide an image that describes the intended lighting style), (2) it can obtain a wider range of lighting styles compared to color transfer approaches. In summary,

Figure 5.8 – Left: Target Image, Right: Result image after 20 iterations

our approach combines the best of the two alternative approaches: inverse lighting and color transfer between images.

Conclusion 6

Contents

1 Conclusion **71**

2 Future Work **72**

2.1 Rolling flash and HDR acquisition 72

2.2 Shape and reflectance from RGB-D images using time sequential illumination 72

2.3 Automatic light compositing using rendered images 73

1 Conclusion

In this thesis we focused on how a sequential illumination can be used to reconstruct shape and reflectance of a scene.

First, we explored how to efficiently use photometric stereo for video sequences. To this end, we designed a method that makes use of LED-based illumination with a high speed electronic rolling shutter video camera (maximizing the output frame rate). Then, we successfully recovered the shape and reflectance from video sequences using photometric stereo. However photometric stereo is only applicable in a controlled indoor environment and is too restrictive for the kind of application we are targeting. In particular, a three light sequential illumination setup is too cumbersome and intrusive to be considered as a real solution to augmented and mixed reality applications. To fulfill requirements of those applications while using an active illumination, we had to reduce to a minimum the number of light sources needed to reconstruct shape and reflectance.

Second, we designed a new acquisition framework, in-between photometric stereo, depth refinement and shape from shading techniques. This hybrid setup is less intrusive as it only uses a flash illumination one in two frames. Moreover, using flash and no flash image pairs allows to reconstruct shape and reflectance from scenes with general uncontrolled lighting. The robustness and simpleness of our method allowed us to outperform recent state of the art methods in terms of quality and speed.

Third, we proposed an approach, based on image fusion, that simplifies the process of producing images with a complex lighting. Indeed, we used our hybrid acquisition setup to acquire shape and reflectance of the input scene. From the reconstructed data, we rendered a set of key images, corresponding to various key

lighting conditions, and fuse them automatically to obtain an image with a lighting style close to the one provided by a target image.

2 Future Work

2.1 Rolling flash and HDR acquisition

One of the possible applications of the rolling flash technique is High Dynamic Range acquisition. The idea could be to setup a sequential illumination with two flashes of different power and/or duration. As a result we could reconstruct an HDR image using consecutive frames. It is an interesting possibility, as opposed to current passive HDR acquisition techniques that use several different exposures of camera to acquire input images, the rolling flash provides the possibility to try active lighting for HDR reconstruction. Unlike the state of the art methods that combine images of different exposures to get an HDR image, the proposed method acts directly on the power and/or the duration of light source. Acting on the power of the light source, rather than on its duration, could facilitate the removal of ghosting artifacts due two moving objects.

2.2 Shape and reflectance from RGB-D images using time sequential illumination

In our work we focused on estimating shape as normal maps, however AR/MR applications also require a refined depth. Our algorithm could be improved using joint depth and normal estimation as proposed in [RSD+12]. The quality and precision of the pure flash image is of foremost importance as it is the basis of our reconstruction and refinement algorithm. However it can be difficult to obtain usable pure flash images, as a flash and non flash image pair can contain underexposed and/or overexposed pixels. To get a pure flash image, from a flash and non flash image pair, one needs to find the appropriate balance between camera frame rate, aperture, flash power and flash duration. Setting up those parameters is not hard, but gets tedious when it comes to do it for each captured scene. A future work could be to make this setting automatic.

A high flash frequency is very important for visual comfort. Actually, beyond $50 - 60Hz$ the flash strobe is perceived (human vision) as a constant light source. However, rising the flash frequency would decrease the flash duration, as a consequence the energy captured by the camera is reduced. If the captured energy is too low, then the pure flash image quality decreases, thereby our shape and reflectance reconstruction is directly altered. However, it is still possible to consider more powerful flash to prevent those issues.

Also, as our algorithm considers only lambertian surfaces, however we are currently working on a new method, to handle specular objects.

2.3 Automatic light compositing using rendered images

In this work we used a classic genetic algorithm to match the lighting style of the target image. This algorithm gave us qualitative results, however there are plenty of other genetic - like algorithms in the state of the art that could potentially outperform our approach.
Another alternative to our method could be to use a non-fixed set of input rendered images. Indeed, the coefficients, obtained with the genetic algorithm for a fixed set of input rendered images (each corresponding to one light source position), could be used to determine new input positions of virtual light sources. By doing so, we could refine the initial set of images and furthermore improve the final compositing result.

Résumé en Français

Contents

3 Introduction 75

3.1 Contexte 75

3.2 Motivation 76

4 Sommaire des Contributions 76

4.1 "Rolling flash" et photométrie stéréo 76

4.2 Reconstruction de la forme et de l'apparence à partir d'images RGB-D, et d'une illumination séquentielle. 77

4.3 Composition automatique d'éclairages à partir d'images de synthèse 78

3 Introduction

Cette thèse concerne la numérisation de scènes composées d'objets aux apparences et formes diverses. Notre objectif est d'enrichir la capture vidéo de contenus tels que la reconstruction en temps réel de la apparence (réflectance) et de la forme.

3.1 Contexte

Aujourd'hui on ne peut ignorer que la réalité virtuelle RV, augmentée RV et mixte RM sont des secteurs en plein essor. Plusieurs études récentes prévoient une forte progression dans la prochaine décennie de l'industrie liée à ces technologies, avec un budget prévisionnel de 80 milliards de dollars. Les applications possibles de ces domaines sont nombreuses et couvrent la majeure partie des grands secteurs économiques : la médecine, l'éducation, le divertissement, le secteur militaire, l'architecture et l'industrie pour n'en nommer que quelques-uns. Toutes ces applications tentent de projeter, aussi naturellement que possible, un contenu virtuel interactif dans le réel. L'idée sous-jacente est de donner l'illusion à l'utilisateur, que le contenu qui lui est projeté artificiellement, est réel ou s'inscrit de manière évidente dans le réel. Pour que le rendu soit photo-réaliste, il est nécessaire de disposer, non seulement d'un dispositif capable de transmettre l'illusion à l'œil humain (casques de réalité virtuel, lunettes connectées etc.), mais aussi de moyens d'appréhender, de capter et de modéliser le monde qui entoure l'utilisateur (géométrie, apparence) en temps réel ou temps interactif. L'évolution rapide des technologies du numérique (capteurs de profondeurs, caméras, LED, etc.) permettent d'entrevoir les dispositifs de capture

Figure 6.1 – Differents exemples d'applications AR/MR.

du futur qui ouvrent la voie à un vaste champ de recherche dans les domaines de la vision par ordinateur, la réalité virtuelle, augmentée et mixte.

3.2 Motivation

Cette thèse se veut une contribution à la capture et la modélisation du monde réel en temps réel ou interactif. L'objectif principal est de tirer parti d'un éclairage (illumination) partiellement contrôlé pour enrichir l'acquisition vidéo avec des reconstructions de l'apparence et de la forme des objets. Aujourd'hui, un nombre substantiel de travaux proposent de telles reconstructions. Certains tirent avantage d'une illumination contrôlée pour obtenir des reconstructions de grande qualité, en revanche ces méthodes requièrent un dispositif coûteux et encombrant, c'est le cas du "Light Stage X" [Deb12b], et/ou ne fonctionnent pas en temps réel. Dans notre cas, nous visons un dispositif peu onéreux, rapide et mobile, devant être le moins envahissant (emcombrant) possible.

4 Sommaire des Contributions

Cette section résume les contributions présentées dans cette thèse.

4.1 "Rolling flash" et photométrie stéréo

Dans le second chapitre, nous proposons une méthode d'utilisation de la stéréo photométrie [Woo80] pour la vidéo Fig.6.2. La stéréo photométrie, telle qu'elle est décrite dans [Woo80], reconstruit l'apparence et la forme d'une scène fixe, en la capturant sous au moins trois illuminations directionnelles différentes (les trois directions devant être non-coplanaires). Quant à son application à l'acquisition vidéo,

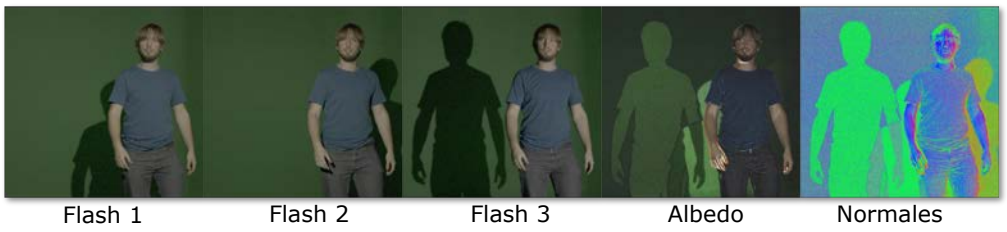

Figure 6.2 – Application de la stéréo photométrie pour la vidéo.

nous utilisons une illumination séquentielle associée à une fréquence de capture élevée. Cependant, utiliser une illumination séquentielle (flashs) pendant l'enregistrement de séquences d'images n'est pas si simple. En vidéo, les flashs provoquent des artefacts temporels, surtout lorsque l'on utilise des caméras à haute vitesse, de type "electronic rolling shutter". C'est pourquoi nous proposons également une méthode permettant d'utiliser une illumination séquentielle avec ce type de caméras. Malgré les résultats intéressants obtenus avec notre méthode, la qualité des reconstructions de l'apparence et de la forme n'étaient pas à la hauteur de nos espérances. De plus, la stéréo photométrie est une méthode qui, de nature, n'est pas très adaptée aux applications visées dans cette thèse. En particulier, la technique de capture utilisant trois illuminations est intrusive et encombrante, de plus cette méthode ne fonctionne que dans un environement contrôlé (pas de lumière extérieure au sytème de capture). Pour toutes ces raisons, nous ne pouvons réellement considérer la stéréo photométrie comme une solution d'acquisition pour les applications visées.

4.2 Reconstruction de la forme et de l'apparence à partir d'images RGB-D, et d'une illumination séquentielle.

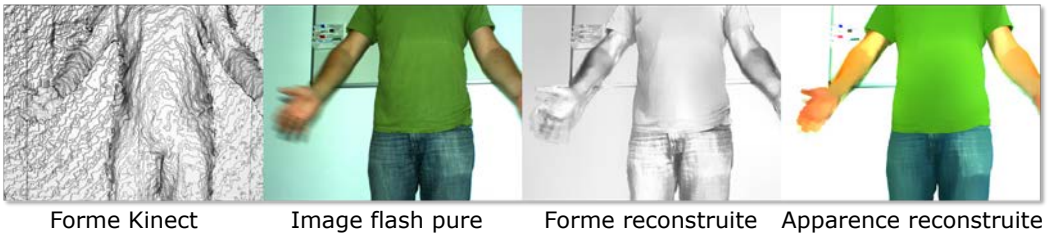

Figure 6.3 – Résultats de notre méthode hybride de reconstruction.

Dans ce chapitre nous proposons une méthode de reconstruction de la forme (géométrie) ainsi que de la réflectance diffuse à partir d'une image (d'une séquence) en utilisant un système de capture hybride composé d'un capteur de profondeur (Kinect), d'une caméra grand public et d'un flash Fig.6.3. L'objectif est de montrer qu'en combinant une acquisition RGB-D (image couleur + profondeur) avec illumination séquentielle, on peut obtenir une reconstruction qualitative de la forme et de la réflectance d'une scène dans le cas où l'éclairage n'est pas connu. Un couple d'images est capturé : une image non flashée (image sous une illumination ambiante) et une image flashée. Une image dont l'illumination ne provient que du flash (image

flash pure) peut être calculée en soustrayant l'image non flashée de l'image flashée. En quelque sorte, l'image flash pure représente la scène sous un éclairage connu, comme si l'on avait éteint l'éclairage ambiant pour ne laisser que le flash comme unique source de lumière. Nous proposons un nouvel algorithme temps réel, qui, basé sur un modèle local d'illumination de notre flash et de l'image flash pure, améliore l'information de forme fournie par le capteur de profondeur tout en retrouvant les informations de réflectance diffuse. Le système est, par nature, moins intrusif et moins encombrant que la stéréo photométrie. De plus l'utilisation d'une paire d'images (flashée et non flashée), rend possible l'utilisation de notre méthode pour des scènes comportant un éclairage ambiant inconnu. La robustesse et la simplicité de notre méthode nous a permis de surpasser des méthodes de l'état de l'art récentes, en termes de qualité et de vitesse d'exécution.

4.3 Composition automatique d'éclairages à partir d'images de synthèse

Figure 6.4 – Résultats de notre méthode de composition automatique d'éclairage.

L'éclairage est un élément clé de la photographie. Les professionnels travaillent régulièrement avec des systèmes d'éclairage complexes afin de capturer directement des images esthétiques. Récemment, certains photographes ont tenté une nouvelle approche : plutôt que photographier une scène directement sous un éclairage complexe, ils capturent la scène sous plusieurs éclairages simples, permettant ainsi un post-traitement permettant combiner les différentes illuminations de la scène. Cette approche apporte une nouvelle dimensionnalité intéressante au post-traitement. Cependant la combinaison des images requiert des compétences en matière de photographie, et l'acquisition sous différentes conditions d'éclairage n'en est pas moins fastidieuse. Nous proposons une méthode totalement automatisée, qui, à partir d'un modèle 3D (forme et albedo) reconstruit à partir de capture d'une scène réelle, produit virtuellement les images correspondant aux différentes conditions d'éclairages. Ensuite, ces images sont combinées automatiquement, à l'aide d'un algorithme génétique, pour correspondre à un style d'éclairage fourni par l'utilisateur sous forme d'une image cible de son choix 6.4.

Bibliography

[BAIH09] Derek Bradley, Bradley Atcheson, Ivo Ihrke, and Wolfgang Heidrich. Synchronization and rolling shutter compensation for consumer video camera arrays. *Computer Vision and Pattern Recognition Workshops, CVPR Workshops,IEEE Computer Society Conference on*, 2009. 35

[BF08] Ronen Basri and Darya Frolova. A two-frame theory of motion, lighting and shape. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7. IEEE, 2008. 23

[BPB13] Ivaylo Boyadzhiev, Sylvain Paris, and Kavita Bala. User-assisted image compositing for photographic lighting. *ACM Trans. Graph.*, 32(4):36–1, 2013. 64, 67

[Bru88] Alfred M Bruckstein. On shape from shading. *Computer Vision, Graphics, and Image Processing*, 44(2):139–154, 1988. 25

[Cha07] Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *City*, 1(2):1, 2007. 65

[dDKMB09] Bert de Decker, Jan Kautz, Tom Mertens, and Philippe Bekaert. Capturing multiple illumination conditions using time and color multiplexing. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 2536–2543, 2009. 35

[Deb12a] Paul Debevec. The light stages and their applications to photoreal digital actors. *SIGGRAPH Asia Technical Briefs*, 2012. 14, 48

[Deb12b] Paul Debevec. The light stages and their applications to photoreal digital actors. In *SIGGRAPH Asia*, Singapore, November 2012. 15, 29, 76

[DT05] James Diebel and Sebastian Thrun. An application of markov random fields to range sensing. In *Advances in neural information processing systems*, pages 291–298, 2005. 26, 47

[DXW01] Jeffrey M DiCarlo, Feng Xiao, and Brian A Wandell. Illuminating illumination. In *Color and Imaging Conference*, volume 2001, pages 27–34. Society for Imaging Science and Technology, 2001. 48, 50, 53, 55

[eS09] Adriana de Souza e Silva. *Digital cityscapes: Merging digital and urban playspaces*, volume 57. Peter Lang, 2009. 13

[FB08] J.L.W. Furlan and A. Braunstein. Digital photography device having a rolling shutter, November 18 2008. US Patent 7,453,514. 35

[FKI+14] Sean Ryan Fanello, Cem Keskin, Shahram Izadi, Pushmeet Kohli, David Kim, David Sweeney, Antonio Criminisi, Jamie Shotton, Sing Bing Kang, and Tim Paek. Learning to be a depth camera for close-range human capture and interaction. *ACM Transactions on Graphics (TOG)*, 33(4):86, 2014. 26

[GCHS10] Dan B Goldman, Brian Curless, Aaron Hertzmann, and Steven M Seitz. Shape and spatially-varying brdfs from photometric stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(6):1060–1071, 2010. 7, 22

[GKCE12] Matthias Grundmann, Vivek Kwatra, Daniel Castro, and Irfan Essa. Effective calibration free rolling shutter removal. *IEEE ICCP*, 2012. 34

[Hae92] Paul Haeberli. Synthetic lighting for photography. *Grafica Obscura*, 3, 1992. 64

[HGK+16] Matis Hudon, Adrien Gruson, Paul Kerbiriou, Remi Cozot, and Kadi Bouatouch. Shape and reflectance from rgb-d images using time sequential illumination. In *Intenational Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, 2016. 64, 65, 66

[HLK13] Yudeog Han, Joon-Young Lee, and In Kweon. High quality shape from a single rgb-d image under uncalibrated natural illumination. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1617–1624, 2013. 27

[HLMCB15] Hristina Hristova, Olivier Le Meur, Rémi Cozot, and Kadi Bouatouch. Style-aware robust color transfer. In *Proceedings of the workshop on Computational Aesthetics*, pages 67–77. Eurographics Association, 2015. 64

[HMI10] Tomoaki Higo, Yasuyuki Matsushita, and Katsushi Ikeuchi. Consensus photometric stereo. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1157–1164. IEEE, 2010. 7, 21, 23

[Hor70] Berthold KP Horn. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. 1970. 25

[HVB+07] C. Hernández, G. Vogiatzis, G. J. Brostow, B. Stenger, and R. Cipolla. Non-rigid photometric stereo with colored lights. In *ICCV*, October 2007. 7, 23, 24, 35

[JK07] Neel Joshi and David J Kriegman. Shape from varying illumination and viewpoint. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–7. IEEE, 2007. 7, 23, 24

[KWB10] Hyeongwoo Kim, Bennett Wilburn, and Moshe Ben-Ezra. Photometric stereo for dynamic surface orientations. In *Computer Vision - ECCV 2010, 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part I*, pages 59–72, 2010. 35, 48

[KWBE10] Hyeongwoo Kim, Bennett Wilburn, and Moshe Ben-Ezra. Photometric stereo for dynamic surface orientations. In *Computer Vision–ECCV 2010*, pages 59–72. Springer, 2010. 7, 24, 25

[LSM08] Christian Linz, Timo Stich, and Marcus Magnor. High-speed motion analysis with multi-exposure images. In *Proc. Vision, Modeling and Visualization (VMV) 2008*, October 2008. 35

[MBAAP12] Ludovic Magerand, Adrien Bartoli, Omar Ait-Aider, and Daniel Pizarro. Global optimization of object pose and motion from a single rolling shutter image with automatic 2d-3d matching. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part I*, ECCV'12, pages 456–469, Berlin, Heidelberg, 2012. Springer-Verlag. 35

[MBAP12] Ludovic Magerand, Adrien Bartoli, Omar Ait-Aider, and Daniel Pizarro. Global optimization of object pose and motion from a single rolling shutter image with automatic 2d-3d matching. In *Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part I*, pages 456–469, 2012. 34

[MCS90] JF Murray-Coleman and AM Smith. The automated measurement of brdfs and their application to luminaire modeling. *Journal of the Illuminating Engineering Society*, 19(1):87–99, 1990. 19

[MGPFS08] Miguel Martin, Diego Gutiérrez Pérez, Roland Fleming, and Olga Sorkine. Understanding exposure for reverse tone mapping. Technical report, 2008. 65

[MK94] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994. 13

[MS06] Yael Moses and Ilan Shimshoni. 3d shape recovery of smooth surfaces: Dropping the fixed viewpoint assumption. In *Computer Vision–ACCV 2006*, pages 429–438. Springer, 2006. 23

[NRDR05] Diego Nehab, Szymon Rusinkiewicz, James Davis, and Ravi Ramamoorthi. Efficiently combining positions and normals for precise 3d geometry. *ACM transactions on graphics (TOG)*, 24(3):536–543, 2005. 7, 26, 47

[OERW+15] Roy Or-El, Guy Rosman, Aaron Wetzler, Ron Kimmel, and Alfred M Bruckstein. Rgbd-fusion: Real-time high precision depth recovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5407–5416, 2015. 7, 9, 27, 57, 58, 59, 64

[PDH11] P.Eng Peter D. Hiscocks. Measuring luminance with a digital camera. 2011. 37

[Pet87] AP Petrov. Light, color, and shape. *Cognitive processes and their simulation*, pages 350–358, 1987. 23

[PF05] Emmanuel Prados and Olivier Faugeras. Shape from shading: a well-posed problem? In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 870–877. IEEE, 2005. 26

[PSA+04] Georg Petschnigg, Richard Szeliski, Maneesh Agrawala, Michael Cohen, Hugues Hoppe, and Kentaro Toyama. Digital photography with flash and no-flash image pairs. *ACM transactions on graphics (TOG)*, 23(3):664–672, 2004. 48, 54

[RAGS01] Erik Reinhard, Michael Ashikhmin, Bruce Gooch, and Peter Shirley. Color transfer between images. *IEEE Computer graphics and applications*, (5):34–41, 2001. 64

[RH01] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 497–500. ACM, 2001. 22

[RL00] Szymon Rusinkiewicz and Marc Levoy. Qsplat: A multiresolution point rendering system for large meshes. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 343–352. ACM Press/Addison-Wesley Publishing Co., 2000. 66

[RSD+12] Christian Richardt, Carsten Stoll, Neil A Dodgson, Hans-Peter Seidel, and Christian Theobalt. Coherent spatiotemporal filtering, upsampling and rendering of rgbz videos. In *Computer Graphics Forum*, volume 31, pages 247–256. Wiley Online Library, 2012. 7, 26, 27, 47, 72

[SY11] Li Shen and Chuohao Yeo. Intrinsic images decomposition using a local and global sparse representation of reflectance. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 697–704. IEEE, 2011. 50

[TAH+04] Ch. Theobalt, I. Albrecht, J. Haber, M. Magnor, and H.-P. Seidel. Pitching a Baseball – Tracking High-Speed Motion with Multi-Exposure Images. In *Proc. SIGGRAPH '04*, pages 540–547. ACM SIGGRAPH, 2004. 35

[TFG+13] Borom Tunwattanapong, Graham Fyffe, Paul Graham, Jay Busch, Xueming Yu, Abhijeet Ghosh, and Paul Debevec. Acquiring reflectance and shape from continuous spherical harmonic illumination. *ACM Transactions on graphics (TOG)*, 32(4):109, 2013. 7, 22, 23

[War92] Gregory J Ward. Measuring and modeling anisotropic reflection. *ACM SIGGRAPH Computer Graphics*, 26(2):265–272, 1992. 7, 20, 22

[WGT+05a] Andreas Wenger, Andrew Gardner, Chris Tchou, Jonas Unger, Tim Hawkins, and Paul Debevec. Performance relighting and reflectance transformation with time-multiplexed illumination. *ACM Transactions on Graphics (TOG)*, 24(3):756–764, 2005. 7, 24, 25

[WGT+05b] Andreas Wenger, Andrew Gardner, Chris Tchou, Jonas Unger, Tim Hawkins, and Paul Debevec. Performance relighting and reflectance transformation with time-multiplexed illumination. *j-TOG*, 24(3):756–764, July 2005. 35, 43

[WJV+05] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High performance imaging using large camera arrays. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, pages 765–776, New York, NY, USA, 2005. ACM. 34

[Woo80] Robert J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):191139–191139–, 1980. 15, 21, 23, 30, 35, 42, 44, 48, 76

[WWB+14] Ingo Wald, Sven Woop, Carsten Benthin, Gregory S Johnson, and Manfred Ernst. Embree: A kernel framework for efficient cpu ray tracing. *ACM Transactions on Graphics (TOG)*, 33(4):143, 2014. 66

[WZN+14] Chenglei Wu, Michael Zollhöfer, Matthias Nießner, Marc Stamminger, Shahram Izadi, and Christian Theobalt. Real-time shading-based refinement for consumer depth cameras. *Proc. SIGGRAPH Asia*, 2014. 7, 27, 47, 64

[XDCW01] Feng Xiao, Jeffrey M DiCarlo, Peter B Catrysse, and Brian A Wandell. Image analysis using modulated light sources. In *Photonics West 2001-Electronic Imaging*, pages 22–30. International Society for Optics and Photonics, 2001. 48

[YYTL13] Lap-Fai Yu, Sai-Kit Yeung, Yu-Wing Tai, and Stephen Lin. Shading-based shape refinement of rgb-d images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013. 27

[ZTCS99] Ruo Zhang, Ping-Sing Tsai, James Edwin Cryer, and Mubarak Shah. Shape-from-shading: a survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(8):690–706, 1999. 25