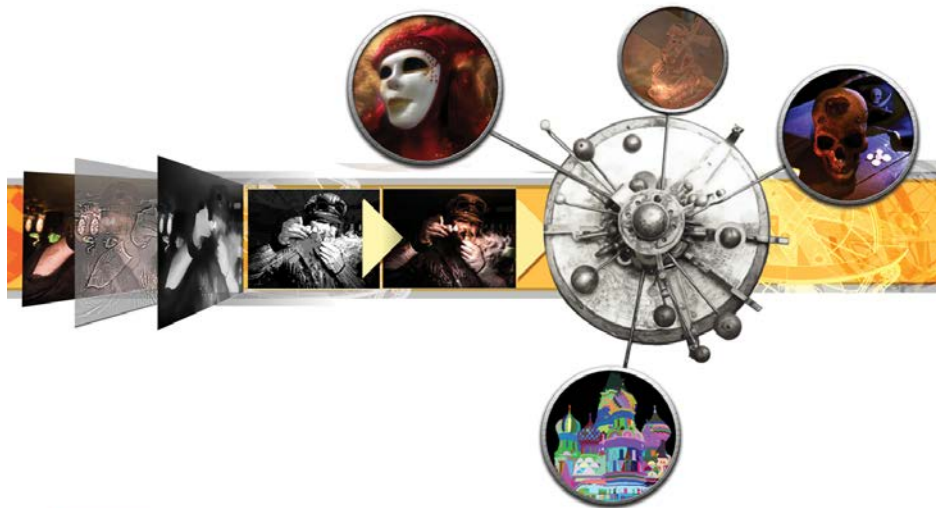


Perception Based Image Editing



CENTRO POLITÉCNICO
SUPERIOR

DEPARTAMENTO DE
INFORMÁTICA E INGENIERÍA
DE SISTEMAS

GRUPO DE INFORMÁTICA GRÁFICA
AVANZADA (ADVANCED COMPUTER
GRAPHICS GROUP)

Jorge Lopez-Moreno

Supervised by:

Dr. Diego Gutiérrez Perez

Dr. Erik Reinhard

Departamento de Informática e Ingeniería de Sistemas

Universidad de Zaragoza

To María Jesús, for so many reasons that I would need to write another book just to thank her.
To my parents for giving me only thing than no one can ever take from you, education.
To my friends and family because, in spite of my increasing computer graphics geekness over the last years, they still come to see movies with me, daring even to ask what was my last paper about.
To Diego, for telling me that what I really wanted was to become a PhD.
And to the one who just arrived, Juan. Lucky you. You were spared from hours of thesis-related ramblings.

Acknowledgments

All the research included in this book would have never ever been possible without the advice and cooperation of the following people:

- My supervisors: Diego Gutiérrez and Erik Reinhard. Special thanks for being near the fire at all the paper deadlines.
- The co-authors of past published papers: Adolfo Muñoz, Jorge Jimenez, Ken Anjyo, Adrián Gargallo, Jorge Fandos, Angel Cabanes, Francisco Sangorrín, Veronica Sundstedt, Francisco J. Serón, Sunil Hadap, Erik Reinhard and Diego Gutiérrez.
- The co-authors of (possibly) future published papers: Elena Garces and Adrian Jarabo.
- My colleagues of the Advanced Computer Graphics Group (GIGA) at the University of Zaragoza. Thanks for all those coffee time improvised lessons on computer graphics.
- My students and PFCs. I hope you learned as much as I learnt from you.
- My workmates and managers at Adobe Systems (Visual Computing Lab), who made feel at home in San Jose and sponsored this research. Thank you for making it possible.
- The disinterested thorough reviews of our papers done by dozens of anonymous reviewers.

This research was partly sponsored by:

- Adobe Systems Inc.
- The Spanish Ministry of Education and Research through the project TIN2010-21543.
- The Spanish Ministry of Science and Technology through the project TIN2007-63025.

PhD Summary and Contributions

This thesis is focused in extending the set of tools available to artists to effect high level edits in single images by relying on two facts: First, the human visual system has many limitations which, properly leveraged, allow for . And second, if we can extract some of the multiple variables which originated a two-dimensional image (like illumination, material, 3D shape,...), we will be able to perform advanced edits which, otherwise, would be almost impossible for an unskilled user.

In the side of publications related to this thesis, I have authored four journal papers indexed in the JCR list (two of them as first author), three international papers as first author and three papers on national conferences as first author. Additional awards, related research projects and stays are detailed in the introductory chapter of this document.

We cannot summarize this PhD without referring to our ongoing collaboration with Adobe Systems, which started as a result of this thesis, giving raise to: two internships (seven months in total) at the Visual Computing Lab (San Jose, CA. USA), two consecutive gifts of 20000\$ and 40000\$ supporting this PhD and three patents (co-authored with Sunil Hadap). Our main contributions to the field are:

- An approximated threshold for the accuracy of human vision when detecting lighting inconsistencies in images, used in the design of our light source estimation algorithms.
- New depth estimation techniques based either in the perception of depth or in the previous knowledge of the light sources.
- We have introduced and validated two novel light source estimation methods which are, to our knowledge, the first solutions in the literature to multiple light detection from arbitrary shapes in a single image (no depth information required).
- Regarding intrinsic image decomposition, we have explored the limits of bilateral filtering and proposed a novel algorithm based in albedo segmentation and optimization, which equals or even surpasses the results of previous approaches in the field.
- We have presented novel algorithms to simulate the complex process of light transport in participating media: fog and caustics. Our results match perceptually those achievable by ground truth simulation (photon mapping) if 3D information were available.

-
- We have applied our processing pipeline to the design of: novel relighting and compositing methods, non-photorealistic stylization techniques, and to the capture of complex materials with subsurface scattering properties from a single image.

Contents

1	Introduction	1
1.1	Perception	1
1.2	Recovering Dimensions from a Single Image	2
1.3	Goals	6
1.4	Contributions and Measurable Results	6
1.4.1	Publications	6
1.4.2	Patents	7
1.4.3	Awards	7
1.4.4	PFCs Supervised	8
1.4.5	Research Stays	8
1.4.6	Research Projects	8
1.4.6.1	Unrelated research projects	9
1.5	Dissertation Overview	9
	References	14
2	The perception of light inconsistencies	15
2.1	Introduction	15
2.2	Related Work	16
2.3	Experiment One: Overall Inaccuracy	17
2.3.1	Results	19
2.4	Experiment Two: Influence of Texture	21
2.4.1	Results	22
2.5	Experiment Three: Real World Images	22
2.6	Conclusions and Future Work	25
	References	28

CONTENTS

3	Light Detection in Single Images	29
3.1	Introduction	29
3.2	Previous Work	30
3.3	Perceptual Framework	31
3.4	Estimating Light Sources	31
3.5	Pre-processing	32
3.6	Estimating Azimuth Angles	33
3.6.1	K-means approach	33
3.6.2	Light Source Fitting Approach	35
3.6.2.1	Finding Light Source Candidates	36
3.6.2.2	Splitting a light source	38
3.6.2.3	Detecting point light sources	39
3.7	Estimating Zenith Angles and Intensities	41
3.7.1	Simple Normal Approximation	41
3.7.2	Normal Approximation by Osculating Arc	42
3.7.3	Zenith estimation	43
3.7.4	Grouping lights and ambient illumination	44
3.8	Results	44
3.8.1	Error Analysis	44
3.8.2	Visual Validation	51
3.8.3	Image Compositing	52
3.9	Discussion and Future Work	56
	References	61
4	3D Shape Reconstruction	63
4.1	Introduction	63
4.2	Selecting a Shape From Shading Method	64
4.2.1	Perception-based SFS	64
4.2.2	Parametric SFS based on light detection	65
4.3	Conclusions and Future Work	67
4.4	Annex A: Derivatives of the Error Function	68
	References	73

5	Intrinsic Images Decomposition	75
5.1	Introduction	75
5.1.1	Image Generation	76
5.2	Previous Work	77
5.2.1	State of the Art	77
5.3	Reflectance and Illumination Decomposition	79
5.4	Step 1: Image Segmentation	80
5.4.1	Graph-based Segmentation	81
5.4.2	The influence of color space: RGB and Lab	82
5.4.3	Filtering and Segmentation Refinement	83
5.4.4	Segmentation Results	84
5.5	Step 2: Normalization	84
5.5.1	Linearizing the Problem	86
5.5.2	Looking for the Luminance Steady State	87
5.5.3	Solving the System	90
5.6	Results	91
5.7	Conclusions	101
5.8	Limitations and Future Work	101
	References	105
6	Application 1: Light Transport in Participating Media. An Image Editing Approach	107
6.1	Introduction	107
6.2	Previous Work	109
6.3	Light in Participating Media	110
6.3.1	Assumptions	110
6.3.2	Simplifying the Physical Model	110
6.3.3	Perception of the Natural Process	111
6.3.4	Image Processing	112
6.3.4.1	Depth estimation	113
6.3.4.2	Image Processing Pipeline	114
6.4	Validation	118
6.4.1	Adding participating media	118
6.4.2	Psychophysical test	120

CONTENTS

6.5	Conclusions and Future Work	121
	References	126
7	Application 2: Procedural caustics	127
7.1	Introduction	127
7.2	Motivation	129
7.3	Simulating Caustics	132
7.3.1	Depth Recovery	132
7.3.2	Phase Symmetry	132
7.3.3	Luminance Adjustment	133
7.4	Results	134
7.5	Psychophysics	137
7.5.1	Experiment 1: Validation against 3D Rendering	137
7.5.2	Experiment 2: Validation against Direct Painting	141
7.6	Conclusions	143
7.7	Annex A. Phase symmetry	144
	References	149
8	Application 3: Image Stylization and Non Photorealist Rendering	151
8.1	Introduction	151
8.2	Previous Work	152
8.3	Perceptual Background	154
8.4	Algorithm	155
8.4.1	Depth Recovery	156
8.4.2	Computing Visibility for New Light Sources	157
8.5	Stylization examples	158
8.6	Image retouching interface	163
8.7	Evaluation	167
8.8	Discussion	169
8.9	Conclusions	172
	References	178

9	Application 4: BSSRDF Estimation from Single Images	179
9.1	Introduction	179
9.2	Previous Work	181
9.3	BSSRDF Estimation	182
9.3.1	Algorithm	182
9.4	Estimation from Uncontrolled Single Images	188
9.5	Results and Discussion	190
9.6	Conclusions	193
	References	204
10	Conclusions and Future Work	205
10.1	Future Work	206

CONTENTS

Chapter 1

Introduction

Image editing and post-processing techniques have matured over the years, making it difficult (verging on impossible) to assess whether an image has been digitally enhanced or modified somehow. However, complex manipulations are still a time consuming process which relies on skilled user input, often requiring painstakingly painting over pixels.

In this thesis we present our work on advanced image editing techniques, extending current tools by leveraging the limitations of the human visual system in order to extract additional dimensions (like depth or texture) from a single two-dimensional image. Working in perceptual space, the validity of our results is assessed by psychophysical methodologies.

1.1 Perception

In the early years of science, Sir Isaac Newton studied the nature of light and optics, stating that our perception of colors is due to the pressure produced by the light (composed by *particles*) over the surface of our eyes. To prove it, he slid a darning needle around the side of his eye until he could poke at its rear side, dispassionately noting "white, darke & colored circles" so long as he kept stirring with "ye bodkin."

Nowadays we don't need to go as far as Newton to know that the perception of color (or light for what it matters) is not as simple as connecting a linear light meter to our brain. How we interpret images (light) depends on multiple factors, some well-known, some still a mystery. Take for instance the image in Figure 1: we all see two spirals (one green, one blue) on a pink background. If we look closer, we will notice that there are also some orange strips. There does not seem to be a lot more in this image. Well, actually, we have seen more than there actually is: in reality, the green and blue colors are exactly the same! A quick PhotoshopTM test will confirm this. So what is going on?

As stated by Diego Gutierrez (Gut09), it turns out that our visual system is designed to interpret visual information relying heavily on contrast and other contextual information. In other words, we cannot tell the exact physical magnitude of, say, luminance (an objective magnitude). Instead, we can only judge brightness (a subjective measure), that is, we can only tell whether something is lighter or darker than its surroundings. The same concept applies to color: the green spiral in Figure 1.1 is



Figure 1.1: The perceived green and blue spirals are just a visual effect. In reality, both colors are exactly the same. Image from <http://blogs.discovermagazine.com/badastronomy/>

crossed by orange stripes, whereas for the blue they turn magenta. So our brain computes color based on local information and comes out wrongly with two very different colors when there is only one.

Any image-editing algorithm that works in pixel-value space will miss out on the clear fact that the two spirals are perceived very differently, since the pixel values for both are exactly the same ((0, 255, 150) in RGB space, to be precise). This thesis explores algorithms that work in perceptual space instead, where there exists a clear distinction between the two spirals. Given that our perception, as we have seen, is not perfect, it makes sense to think that working in perceptual space we can sometimes get away with imperfect simulations (see Figure 1.2).

The key is to understand which imperfections will not be noticed by a human observer, and which will be easily spotted and thus must be avoided.

1.2 Recovering Dimensions from a Single Image

The image synthesis is a complex process produced by the transport of the light and its interactions with both media and objects. The final result for each pixel is the result of the collapse of several dimensions of information (3D geometry, material properties, illumination characteristics, variations in time,...) into a just few dimensions (usually five in RGB images): the X-Y coordinates of each pixel in the image and its corresponding color value.

Some extreme edits in a single image depend on the alteration of one of the "lost" input dimensions. For instance, if we want to add fog to a photograph, we would need to know the depth value of each pixel, and the behavior of the fog in function of this depth.

1.2 Recovering Dimensions from a Single Image

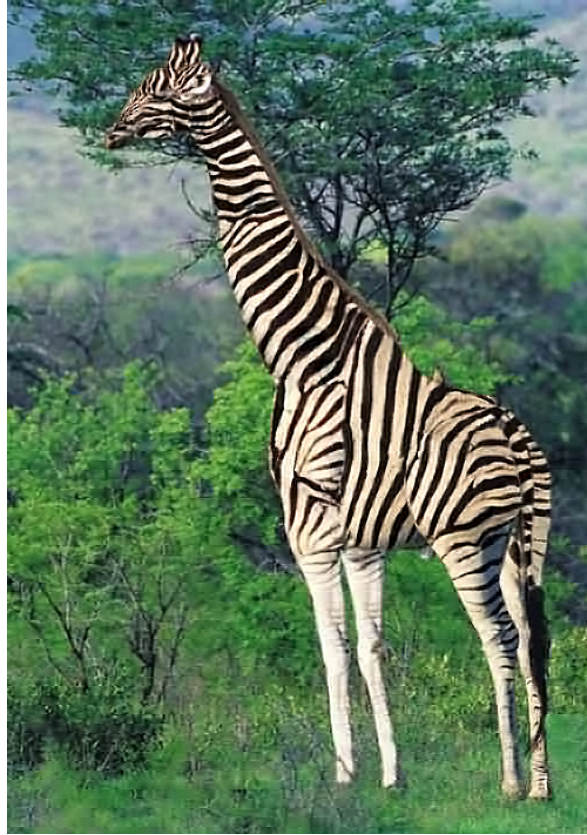


Figure 1.2: Example of digital manipulation. When asked to spot a deliberate mistake in the image, some people see it immediately, while others stare at it for a long time, before noticing. Some people do not see it at all. Image from <http://www.moillusions.com/2009/01/find-mistake.html>

As such, the inverse problem, recovering the original information is an ill posed problem with infinite possible solutions for a given image. In order to obtain an optimal solution we will rely on two bases, the limitations of the human visual system (HVS) and a progressive refinement of our results through iteration and isolation of these dimensions into material, geometry and illumination. Intuitively, this means that for instance, the better we know the amount of contribution of one component, the better we can extract the contribution of a complementary dimension. For instance, if we know the shading of an object is very straightforward to approximate its texture or albedo. The opposite is equally true.

Our thesis is that we can work with perceptual approximations of these modular components and use them in order to produce final results or even compute other components.

Naturally, the more accurate and physically correct our results are, the better would work any edits afterwards. However, we find that, as long as we work within certain perceptual thresholds, the results will be plausible and errors will tend to go unnoticed, even for the most trained eyes.

In Figure 1.3 we can see the diagram of our image processing pipeline approach. We extend a two-dimensional image to three or even more dimensions in order to perform advanced edits in higher

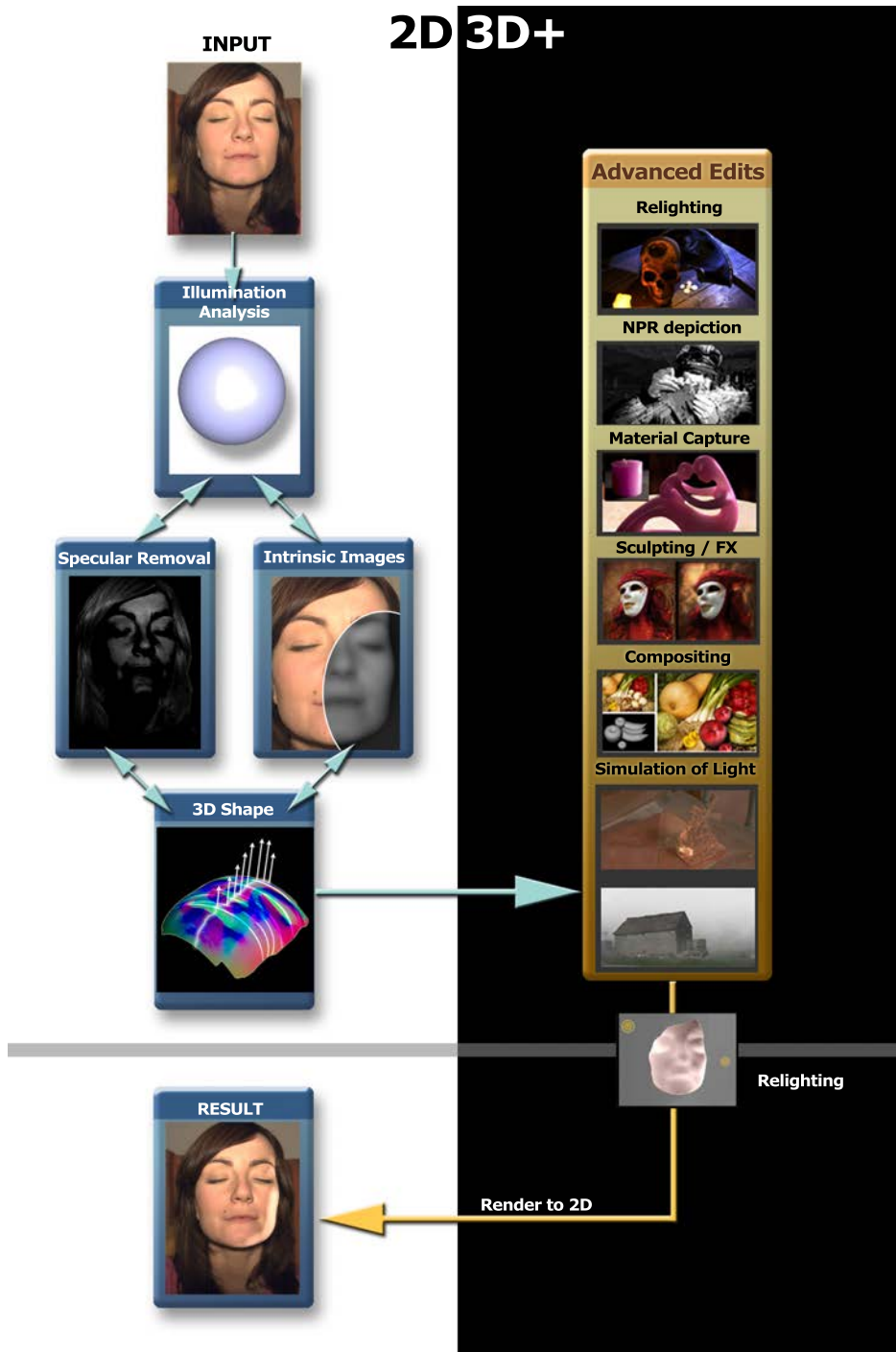


Figure 1.3: Diagram showing our image processing pipeline. The left part shows two-dimensional image processes. On the right, our algorithms use three or more dimensions in order to perform advanced edits on the image before rendering it back to its original two dimensions. The result shows an example of a relighting technique.

1.2 Recovering Dimensions from a Single Image

dimension levels and render the result back into a two-dimensional image. In the following paragraphs, we describe the main components of the pipeline.

Illumination Analysis: This module is focused on inferring the number of light sources, their spatial positions and their relative intensities of the input image. In order to approximate these, we rely on limited and unskilled user input (select a convex object in the image and contour it). Our algorithms (LMSSG10) are able to detect up to four light sources, with error within perceptual thresholds. This module uses approximated geometry and intrinsic images decomposition, therefore the bidirectional arrows in the figure.

Intrinsic Images Decomposition: The goal of this module is to separate albedo(texture) from illumination (shading). In our research this is achieved through albedo segmentation (Chapter 5) or frequency decomposition by bilateral filtering. As a general rule, the materials are assumed to be Lambertian and the specular component (if existing) is extracted in advance. We find that multilevel decomposition approaches (SSD09) might improve this module, however its study is beyond the scope of this thesis.

Specular Removal: In order to extract specularity (highlights) we rely on two techniques (see Figure 1.4). First, we perform a change of color space in order to detect the amount of specular component per pixel (MZBK06). The color of the light is required and provided by the user, detected through histogram thresholding in HDR images or assumed to be white. Second, we follow the approach by Qingxiong-Yang et al. (YWA10), and propagate color values from neighboring pixels with less specular component through anisotropic gaussian filtering. The anisotropy is guided by the specular values computed in the previous step. In general, this kind of image processing yields better results in HDR images.

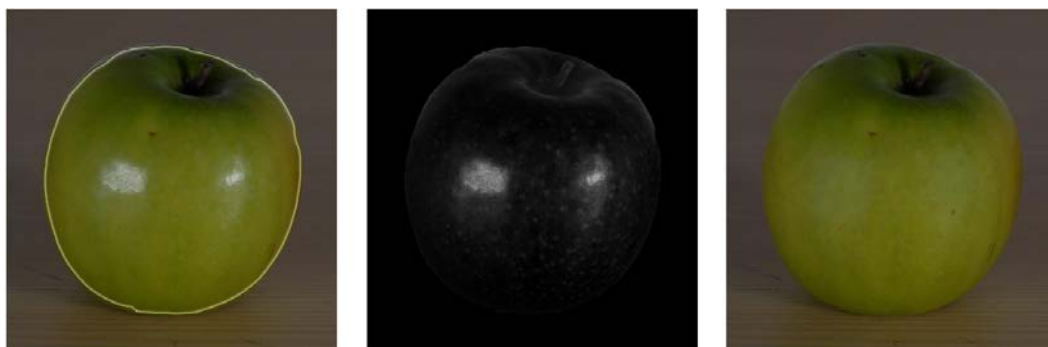


Figure 1.4: Left: Input image with contour selected by user. Middle: Specular component obtained by color space rotation. Right: Result of specular anisotropic erosion.

Depth Reconstruction: An ill posed problem such as 3D shape recovery from a single image has been tackled without achieving a general-purpose solution until the date. Some of the best results were obtained using shading and shadows information over the surface of the object of study (*shape from shading* (SFS), see (ZTCS99)).

In our applications we rely on an automatic approach based on *shape from shading* (SFS) which takes advantage of the dark-is-deep paradigm and the bas-relief ambiguity (BKY99) to extract depth from a single image. It can then be used to perform extreme material editing in objects from images without the observer noticing the obvious inconsistencies arisen by the simplicity of the SFS algorithm (KRFB06). This approach and more sophisticated methods are discussed in detail in Chapter 4.

Advanced Edits: In this thesis we will show how a wide range of advanced image edits become feasible to an unskilled user: relighting, NPR stylization, simulation of light transport (caustics), tonemapping, automatic composition, capture of complex material properties, etc.

1. INTRODUCTION

1.3 Goals

Our overall goal is to extend the set of tools available to artists to effect high level edits in single images, without the need to painstakingly paint over all pixels.

To perform the edits, we intend to extend a single image to the multidimensional space which originated it. We assume that such an ill-posed decomposition is feasible by relying on the limitations of the HVS.

When working in perceptual space, psychophysics and user tests become a crucial way to validate the results. Therefore, as a general rule, the applications shown in this thesis follow this scheme: First, we set a perceptual basis (assumptions) for the algorithm. Then we implement the algorithm and show our results. Finally, we validate our findings by means of psychophysics and user tests.

1.4 Contributions and Measurable Results

1.4.1 Publications

Part of the present PhD has already been published:

- Our K-means-based light detection method (Chapter 3) has been published in the Computers & Graphics Journal (LMHRG10). This journal has an impact factor of 0.787 and a 5-year impact factor of 0.978, ranking 67th out of 93 (Q3) in the JCR list. Previous results were published in the Spanish national conference on computer graphics, CEIG 2009, and selected as 2-top paper (LMHRG09).
- Our study on the limits on human perception of light inconsistencies was published in the Applied Perception on Graphics and Visualization (APGV 2010) (LMSSG10). This work was selected as cover of the conference proceedings. The initial results of our tests were published in CEIG 2009 (LMSLG09).
- Our image-based approach to procedural simulation of caustics (Chapter 7) was published in Siggraph 2008 (ACM Transactions on Graphics journal (GLMF⁺08)). Its impact factor in 2008 was of 3.383 (being the 3rd out of 86) of the JCR list, with a five-year average impact factor of 4.997 (the 1st out of 86).
- The first version of our single-image relighting and compositing tool (used to generate several examples of this PhD) was published in the IX International Conference on Human-Machine Interaction, INTERACCION 2008 (LMCG08).
- Our research on image-based simulation of participating media (Chapter 6) was published in CEIG 2008 (LMCG08)
- The results of our research on non photorealistic rendering of single images, shown in Chapter 8, have been published in the Computers & Graphics Journal (JCR listed) (LMJH⁺11) . Our previous work on the same topic received the best paper award at the 2010 NPAR conference, and was selected as cover of the proceedings (LMJH⁺10).

1.4 Contributions and Measurable Results

- Finally, our work on single-image capture of material properties was published in Eurographics (MELM⁺11). This conference's proceedings are included in the journal Computer Graphics Forum, which in 2009 had an impact index of 1.681 (2:059 is the average of the last five years), which is the 22nd out of 93 of the subject category Computer Science, Software Engineering of the JCR list.

Our planned research include:

- Our work on automatic intrinsic images decomposition (Chapter 5) is to be submitted next March 2011 to the International Conference on Computer Vision, ICCV (ICCV has a CiteSeer impact factor ranking in the top 5% of all Computer Science journals and conferences).
- Our light detection method based in optimization and osculating arc (Chapter 3 has shown better accuracy than our previously published method, and we expect to submit it this year (the venue is still to be decided).
- In the long term, our current line of work will focus on the interaction of our RBF-based shape from shading implementation with our light detection method and intrinsic images decomposition, in order to develop more accurate solutions for single-image 3D edition.
- Ongoing collaboration with Adobe Systems in single image editing techniques.

1.4.2 Patents

- US Patent App 20090110322, *Methods and Systems for Estimating Illumination Source Characteristics from a Single Image*. Inventors Sunil Hadap and Jorge Lopez (alphabetically listed).
- US Patent pending (61/093,174), *Determining Characteristics of Multiple Light Sources in a Digital Image*. Inventors Sunil Hadap and Jorge Lopez (alphabetically listed).
- US Patent pending (61/093,168), *Determining Three-Dimensional Shape Characteristics in a Two-Dimensional Image*. Inventors Sunil Hadap and Jorge Lopez (alphabetically listed).

1.4.3 Awards

- Best paper award at 2010 NPAR conference, Annecy (France).
- 2007 Most Innovative Intern Project for *Multiple Light Source Detection in Single Images*. Adobe Systems Inc.

1. INTRODUCTION

1.4.4 PFCs Supervised

In Spain, in order to obtain the degree in engineering, all the students have to successfully finish a *Proyecto Fin de Carrera* (PFC), literally: *End of Degree Project*, which could be considered equivalent to a master thesis in most countries.

- *Descomposición de imágenes en sus componentes intrínsecas* (Image Decomposition into intrinsic components). 2010, by Elena Garcés García.
- *TANGIBLE: Sistema de bajo coste para localización y detección de gestos 3D para entornos inmersivos* (TANGIBLE: Low cost system for location and gesture tracking in 3D immersive environments). 2009, by Alvaro Fernandez Tuesta. Co-supervised with Francisco Serón.
- *Fotografía Computacional: Estudio de límites de captura y percepción visual para el diseño de algoritmos* (Computational Photography: A study on visual and capture limitations for algorithm design). 2009, by Francisco Sangorrín Perdices.
- *Diseño e implementación de un entorno de desarrollo con interfaz gráfico multiplataforma para fotografía computacional* (Design and implementation of a multiplatform environment with GUI for computational photography research). 2008, by Adrián Gargallo Pérez.

1.4.5 Research Stays

- Jul-Oct, 2007 (four months). First internship at Advanced technology Labs, Adobe Systems Inc. San Jose, CA (USA). Research in multiple light detection in single images.
- Jun-Aug, 2008 (three months). Second internship at Advanced technology Labs, Adobe Systems Inc. San Jose, CA (USA). Research in multiple light detection and 3D shape reconstruction from single images.
- Nov-Dec 2009 (two months). Stay at MOVING Group, Universitat Politècnica de Catalunya (UPC). Barcelona (Spain). Research in RBF-based shape from shading techniques.

1.4.6 Research Projects

- *MIMESIS: Low-Cost Techniques for Appearance Model Acquisition of Materials*. (TIN2010-21543). From 2010 to the present day. Funded by the Spanish Ministry of Science and Technology. Main researcher: Dr. Diego Gutierrez.
- *TANGIBLE: Humanos Virtuales Realistas e Interacción Natural y Tangible*. (TIN2007-63025) from October 2007 until the present day. Funded by the Spanish Ministry of Science and Technology. Main researcher: Dr. Francisco J. Seron.
- *Fotografía Computacional* (UZ2007-TEC-06) from January to December 2008. Project about Computational Photography. Funded by the Universidad de Zaragoza. Main researcher: Dr. Diego Gutierrez.

1.4.6.1 Unrelated research projects

During this PhD, I participated in a series of research projects, which, although not directly related with this thesis, entailed a good research experience.

- *SELEAG: Serious Learning Games*(UZ2007-TEC-06) from March 2010 to Sept 2011. Funded by the European Commission (Lifelong Learning Programme). Main researcher: Dr. Carlos Vaz de Carvalho (University of Oporto, Portugal).
- *Development of multidisciplinary management strategies for conservation and use of heritage sites in Asia and Europe*. Asia link Program, REF ASI/B7-301/98/679-051 (072471). Year 2006. Lead researcher: Dr. Diego Gutierrez.
- *INSide, 3D reenactment of neurosurgery interventions*. Instituto de Neurociencia de Aragon. Oct-Dec, 2006. Lead Researcher: Dr. MD. Vicente Calatayud and Dr. Francisco Seron.
- *Virtual reconstruction of the lost gothic Cathedral of El Pilar*. LSLUZ. OTRI project. Aug-Oct 2006. Lead researcher: Emilio Sobreviela.
- *Domus Novo*: DVD for e-learning of domotics. European Leonardo project. 2005-2006 (6 months). Lead Researcher: Dr. Francisco Seron.
- *Proyecto ejecutivo parque lineal en la plataforma logstica de Zaragoza*. Government of Aragon. OTRI project. Feb-Apr, 2005. Lead Researcher: Dr. Francisco Seron.
- *Technical consulting and Multimedia DVD for SIMA*. GRUPO PLANNER SL. OTRI project. 2003 (4 months). Lead Researcher: Dr. Francisco Seron.
- *Virtual reenactment of Sinhaya, 10th century Muslim Neighborhood of Zaragoza*. Zaragoza city council, LSLUZ. OTRI project. 2003 (4 months). Lead Researcher: Dr. Francisco Seron.

1.5 Dissertation Overview



Figure 1.5: In this image, new objects were automatically relit and inserted, mimicking the light detected on neighboring objects. Could you spot them? The solution is shown in Chapter 3.

This document starts with the analysis of illumination in Chapter 3. We rely on psychophysics to try to quantify a well known aspect of human perception: its inability to detect light directions accurately in an image. Since it is actually an ill posed problem for which no precise solution can

1. INTRODUCTION

be inferred, the goal is to understand the limits of our human visual system in order to design light detection algorithms within perceptual limits: as long as the error of the algorithm is less than the accuracy of our perception, the results, although physically inaccurate, will be perceived as correct. We propose and validate two light detection methods based on this premise, which are subsequently applied to image editing techniques such as: image composition (see Figure 1.5), 3D reconstruction (Chapter 4, Figure 1.6) or acquisition of translucent materials from photographs (Chapter 9).

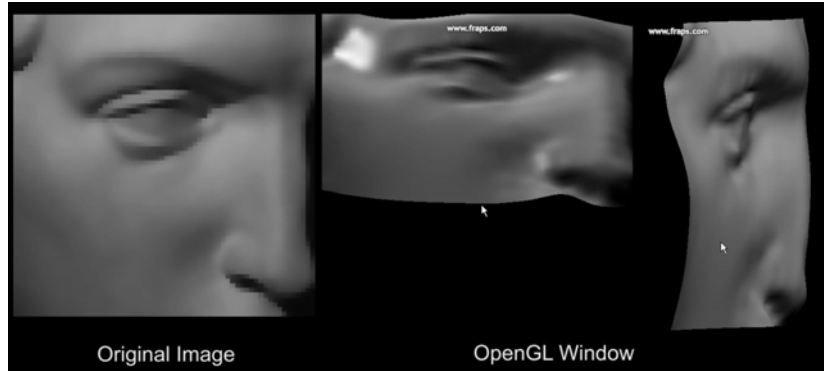


Figure 1.6: Left: input image. Middle, right: automatic 3D reconstruction based on light detection.

In our pipeline, the reflectance (albedo) and illumination (shading) decomposition plays an important role. Most of the image editing techniques proposed in this thesis rely on decomposing images in their high and low frequency components, associated to texture and illumination respectively. Thanks to limitations in the HVS, we are able to produce plausible results in most cases. However, we found that certain applications like relighting or 3D reconstruction would benefit of a better texture extraction approach: In Chapter 5 we propose a novel technique to decompose an image into illumination and reflectance (albedo, texture). Figure 1.8 shows the decomposition in *intrinsic images* using our technique and the corresponding high and low frequency components.



Figure 1.7: Some examples of global illumination effect. From left to right: input image, relighting with $\alpha = 1.0$ and $\beta = 1.0$ and light source at $(80,1000,500)$, relighting with $\alpha = 1.0$ and $\beta = 2.0$ and light source at $(570,500,597)$. In this case the offset is set to 0 to over illuminate the image, producing an interesting glow effect. Finally, relighting with two light sources at $(50,920,230)$ and $(315,400,438)$. α and β are set to $(1.0,0.8)$. Note the color bleeding (red) produced at the jaw.

In Chapter 8 we propose a new class of methods for stylized depiction of images based on approximating significant depth information at local and global levels. Our psychophysical study suggests that the human visual system is more forgiving in a non-photorealistic context, and thus larger errors go unnoticed. We show that a simple methodology suffices to stylize 3D features of an image, showing a variety of 3D lighting and shading possibilities beyond traditional 2D methods, without the need for explicit 3D information as input (See Figure 1.7). A real-time implementation of our image-processing pipeline is presented in this chapter.

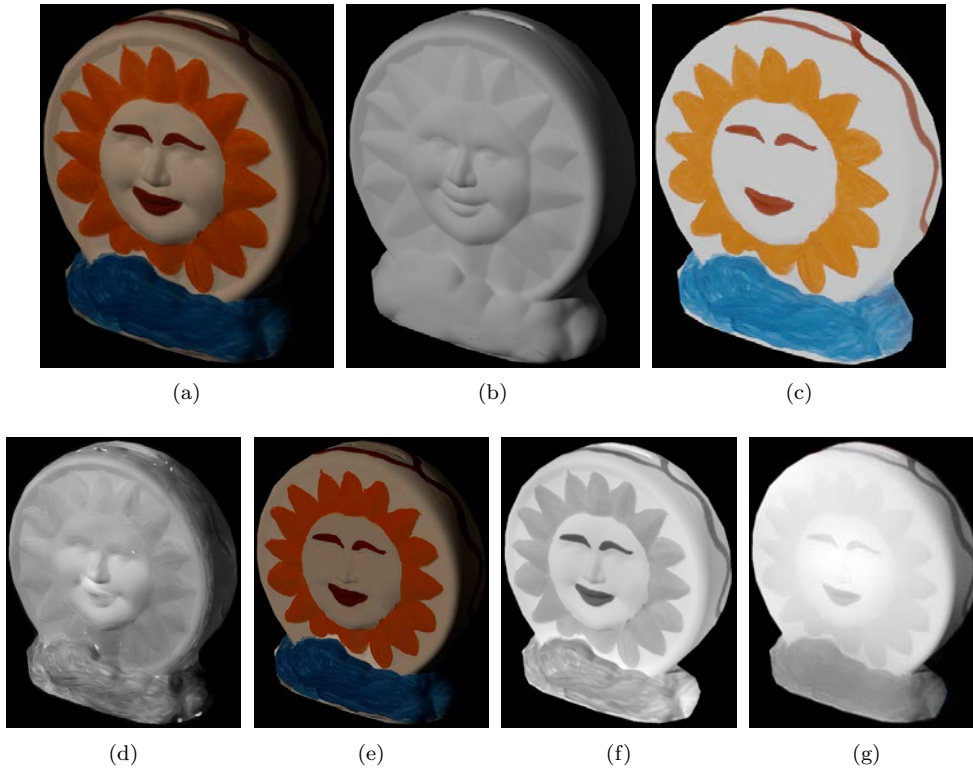


Figure 1.8: *Comparison with other decomposition methods.* (a) Input image. (b) Ground truth shading. (c) Ground truth reflectance. (d) and (e) shading and reflectance with our method. (f) and (g) high an low frequency components, obtained by bilateral filtering.

Figure 1.9 shows another example of a complex image edit, which would require painstakingly painting over pixels by a skilled user. The image on the left is the original picture; on the right, the effect of light transport in a participating media (thick fog) has been simulated. In Chapter 6, we present a novel algorithm which leverages the findings by Narasimhan and Nayar (NN03), who model the effects of different kinds of atmospheric haze and fog by measuring their characteristic point-spread function. In our work, the user simply draws a mask separating foreground and background objects and sets some intuitive fog parameters: its corresponding point-spread function, plus color desaturation, are automatically applied based on the relative distance of the objects in the image.

Chapter 7 introduces an extreme image editing: procedural caustics are simulated in an image based on statistical information of the input image (see the right images of Figure 1.10). The object's geometry is approximated and analyzed to establish likely caustic patterns that such an object may cast. This analysis takes the form of symmetry detection, for which we employ an algorithm that works

1. INTRODUCTION



Figure 1.9: Left: input image. Right: Result of approximating the light transport in fog with image processing filters.

in frequency space and makes minimal assumptions on its input. Finally, the luminance channel of the image is varied according to the projected caustic patterns. In this chapter, psychophysics were run to show how the results were perceptually on par with photon-mapped caustics, but without the need for any 3D geometry.

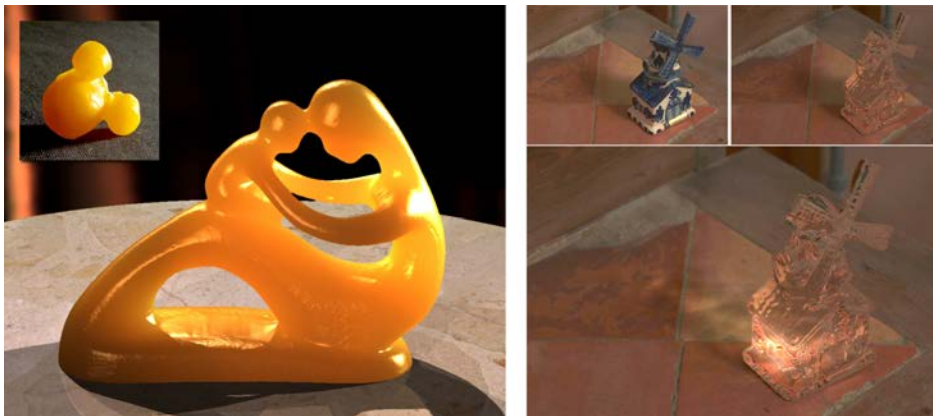


Figure 1.10: **Left:** Example of material transfer, captured from a single photograph of a yellow soap (shown in the inset) and used to render the figurine. **Right:** Top left: input image. Top Right: Object material edited to be transparent (KRFB06). Bottom: Image-based caustics, generated with our method.

Chapter 9 shows an application of our image processing pipeline to the capture of complex material properties like sub-surface scattering from a single photograph. By using light detection and depth approximation this kind of capture is possible even from objects with arbitrary 3D shapes (See the left image of Figure 1.10). Finally, Chapter 10 summarizes the conclusions of this dissertation.

References

- [BKY99] Peter N. Belhumeur, David J. Kriegman, and Alan L. Yuille, *The bas-relief ambiguity*, Int. J. Comput. Vision **35** (1999), no. 1, 33–44. 5
- [GLMF⁺08] Diego Gutierrez, Jorge Lopez-Moreno, Jorge Fandos, Francisco Seron, Maria Sanchez, and Erik Reinhard, *Depicting procedural caustics in single images*, ACM Transactions on Graphics (Proc. of SIGGRAPH Asia) **27** (2008), no. 5, 120:1–120:9. 6
- [Gut09] Diego Gutierrez, *Perception-based image editing*, IEEE ICAT'09 (invited keynote paper), 2009. 1
- [KRFB06] Erum Arif Khan, Erik Reinhard, Roland Fleming, and Heinrich Bülthoff, *Image-based material editing*, ACM Transactions on Graphics (SIGGRAPH 2006) **25** (2006), no. 3, 654–663. 5, 12
- [LMCG08] Jorge Lopez-Moreno, Angel Cabanes, and Diego Gutierrez, *Image-based participating media*, CEIG 2009, Sep 2008, pp. 179–188. 6
- [LMHRG09] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Light source detection in photographs*, CEIG 2009, Sep 2009, pp. 161–168. 6
- [LMHRG10] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Compositing images through light source detection*, Computers & Graphics **In press** (2010). 6
- [LMJH⁺10] Jorge Lopez-Moreno, Jorge Jimenez, Sunil Hadap, Erik Reinhard, Ken Anjyo, and Diego Gutierrez, *Stylized depiction of images based on depth perception*, NPAR '10: Proceedings of the 8th international symposium on Non-photorealistic animation and rendering, ACM, 2010. 6
- [LMJH⁺11] Jorge Lopez-Moreno, Jorge Jimenez, Sunil Hadap, Erik Reinhard, Ken Anjyo, and Diego Gutierrez, *Non-photorealistic, depth-based image editing*, Computers & Graphics **In press** (2011). 6
- [LMSLG09] Jorge Lopez-Moreno, Francisco Sangorrin, Pedro Latorre, and Diego Gutierrez, *Measuring the accuracy of human vision*, CEIG 2009, Sep 2009, pp. 145–152. 6
- [LMSSG10] Jorge Lopez-Moreno, Veronica Sundstedt, Francisco Sangorrin, and Diego Gutierrez, *Measuring the perception of light inconsistencies*, Symposium on Applied Perception in Graphics and Visualization (APGV), ACM Press, 2010. 5, 6
- [MELM⁺11] Adolfo Muñoz, Jose I. Echevarria, Jorge Lopez-Moreno, Francisco Serón, Mashhuda Glencross, and Diego Gutierrez, *Bssrdf estimation from single images*, Computer Graphics Forum (Proc. of EUROGRAPHICS) (2011). 7

REFERENCES

- [MZBK06] Satya Mallick, Todd Zickler, Peter N. Belhumeur, and David Kriegman, *Specularity removal in images and videos: A pde approach*, European Conference on Computer Vision (ECCV), May 2006, pp. 550–563. 5
- [NN03] Srinivasa G. Narasimhan and Shree K Nayar, *Shedding light on the weather*, Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, June 2003, pp. 665 – 672. 11
- [SSD09] Kartic Subr, Cyril Soler, and Frédo Durand, *Edge-preserving multiscale image decomposition based on local extrema*, , Annual Conference Series, ACM Press, dec 2009. 5
- [YWA10] Q. Yang, S. Wang, and N. Ahuja, *Real-time specular highlight removal using bilateral filtering*, ECCV, 2010. 5
- [ZTCS99] R Zhang, P Tsai, J Cryer, and M Shah, *Shape from shading: A survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence **28** (1999), no. 8, 690–706. 5

Chapter 2

The perception of light inconsistencies

In this chapter we present our study of the limits of the human visual system in the perception of light inconsistencies (e.g.: an object which is lit by a different light than its surrounded objects, like in a tampered image).

Part of this work has been presented in Los Angeles (USA) at the Applied Perception on Graphics and Visualization conference (APGV 2010) (LMSSG10), being selected as cover of the conference proceedings. The initial results of our tests were published in the Spanish national conference of computer graphics CEIG 2009 (LMSLG09). We are currently working in an extension of this work for additional spatial positions of the light sources, multiple visual rendering styles and degrees of visual complexity.

The thresholds suggested by this study have been taken into account in the design our light detection algorithms (LMHRG10), described in Chapter 3.

2.1 Introduction

The process of perception in the human visual system (HVS) is a complex phenomenon which starts with the formation of an image in the retina. This image is subsequently analyzed and processed by the HVS in order to extract significative data while disregarding unnecessary information.

Areas such as computer graphics deal with the creation of images by simulating the complex interactions of light and matter in its path towards the retina. However, if we disregard the remaining part of the perception process it is likely that most of these computations could have been avoided. For instance, JPEG format achieves great image compression ratios by removing frequencies which are not easily perceived by the HVS.

Multiple technologies like augmented reality (WS02; ZY01), image editing (YWAC06) or image forensics (JF05; JF07) strongly rely in the process of detecting the lighting environment and inserting

2. THE PERCEPTION OF LIGHT INCONSISTENCIES

new objects relate in the same fashion as their neighbors. For this, the ability to estimate the light direction in the original scene becomes a crucial step. This can be done in controlled environments, but when there is limited information (like in a single image), this task becomes difficult or simply impossible. The influence of shape, material or lighting becomes integrated into a single pixel value and disambiguating this information is not possible without any prior information. This may be further complicated due to uncontrolled factors in the input images such as lens distortion or glare. In these uncontrolled environments, light detection algorithms are expected to yield large errors in their estimations. However, these errors might go completely unnoticed by users in an image while they are easily spotted in another.

In this chapter we are interested in determining an error threshold below which variations in the direction vector of the lights will not be noticed by a human observer. This threshold is very valuable in order to design the light detection methods like the ones proposed in the next chapter, as the errors produced by the unavoidable approximations in such an ill posed problem might go unnoticed if they are below the HVS accuracy. To this end we performed a set of psychophysical experiments where we analyze several factors involved in the general light detection process, while measuring their degree of influence for its future use in computer applications.

There are several aspects involved in the process of light detection. For example the object material, texture frequency, the presence of visual cues such as shadows, light positions and the level of user training are all relevant. The most frequent scenarios to acquire a useful measure studied in present tests have focused on different aspects. Work by Ostrovsky *et al.* (OCS05) studied the influence of the light positions. They anticipated that a greater presence of shadows (produced when the light source is behind the object) increases the accuracy of the HVS.

Our overall goal is to obtain a valid range of values in which the HVS is not able to distinguish lighting errors in very general scenarios. Scenarios we would like to consider are scenes with multiple light sources and material properties and a complete range of light positions. It is important to notice that all our tests preclude the presence of strong visual shadow cues in horizontal surfaces by the objects of the scene. These scenes were excluded based on two main reasons: (1) the subject has been studied in great depth in previous work and its influence has been clearly stated and more importantly (2) it is a visual cue that might not be present in many scenarios in opposition to shading, materials, or self shadowing which are ever-present features.

2.2 Related Work

Todd and Mingolla (TM83) showed the low accuracy of the HVS in determining the light direction by observing a lightprobe. They stated that the presence of highlights did not help in the estimation of the illuminant's direction. However, their measures were limited to cylinders (a simple geometry which varies in only one axis) and the users were asked for the direction of light (the inverse of the present case). In the same line, the same authors disproved the general belief that the HVS assumes objects as diffuse by default (MT86).

Additionally Koenderik *et al.* (KvDP04) showed how human perception is much better at azimuth estimates than at zenith estimates. They also proved that when shadows are present, the shadow boundaries (a first order discontinuity in shading) increased the accuracy of HVS in detecting the light field direction.

Previous research has shown that the visual system assumes that light is coming from above and slightly to the left of a shaded object (SP98; MG01). A recent work by O'Shea *et al.* (OBA08) confirmed this light-from-above prior and provided the quantifiable evidence that for unknown geometries

2.3 Experiment One: Overall Inaccuracy

	a	b	c	d	e	f	g	h
Diffuse	Yes	Yes	Yes	No	No	Yes	No	No
Textured	No	P(h)	No	CHK	CHK	No	P(l)	No

Table 2.1: Description of materials per object (a-h) shown in the images of the test. The top row indicates if the material is only diffuse, otherwise it has a highly specular (Phong) reflectance. **P(h)** and **P(l)** describe a texture obtained through Perlin’s Noise at different spatial scales (high and low frequency respectively) and **CHK** corresponds to a black and white checkerboard texture.

the angle between the viewing direction and the light direction is assumed to be 20°-30° above the viewpoint. Ostrovsky *et al.* (OCS05) show that humans can easily spot an anomalously lit object in an array of *identical* objects with the *same* orientation and lit *exactly* the same, but performance drops when altering orientations of the equally-lit objects. In a similar manner, in this work we aim to extend previous results (OCS05) by providing a wider set of scenarios, adding eye tracking data and quantifying the results. We first present an extension of the experiments published in CEIG 2009 (LMSLG09). Second, we analyze the influence of light position adding new insights by analyzing eye tracking data. Finally, we present two additional experiments which analyze the influence of texture frequency and extrapolate our findings to real-world images, respectively.

2.3 Experiment One: Overall Inaccuracy

In the first experiment our goal is to check how capable the human visual system is of spotting illumination errors in three different lighting situations. Images with several objects are shown (see Figure 2.1), all of them lit from the same angle, except for one, which is lit with a varying degree of divergence with respect to the rest. We limit the study to the less restrictive case of the zenith angle, according to previous findings (KvDP04).

Four of the objects have no texture, two have high-frequency and two have low-frequency textures. Four of the objects are shiny, while four are diffuse. Table 2.1 summarizes their characteristics. The motivation of the scene and the diversity of materials is chosen to represent a wide enough range. In particular, the shape of the objects has been chosen to be abstract in order to avoid semantical significance and globally convex (according to global convexity default assumption of the HVS (LB01)). They have a relatively complex surface, but with limited variance (to avoid the influence of geometry (VLD07)) and are arranged to avoid direct side-by-side comparisons of exactly equal geometries.

We consider the Y axis as the vertical axis of the screen plane XY and Z as the positive XY-plane direction. In each of the 60 images, all the objects are illuminated with an ambient light made up by two directional sources. One is located at 45° between the axis +Z and the axis -X and the other situated on top of the axis Y. Their intensities are four times weaker in terms of luminance than the main light. This main light is also a directional light and is the same for seven of the eight objects, while the eighth is lit from a different direction. Thus, we will refer to these as the *two main lights* in the image: the “correct” one, illuminating seven objects and the “wrong” one, illuminating the eighth.

The two main lights vary their angle ϕ along the XZ plane between different images (top row in Figure 2.2). The absolute difference in ϕ between the two directional lights increases from 0° to a maximum difference of 90° in 10°-increments (5° in each angular direction). We thus obtain ten test images. To further analyze the influence of light direction, we repeat this procedure with three different situations: First with both sources illuminating the frontal hemisphere of the object, secondly

2. THE PERCEPTION OF LIGHT INCONSISTENCIES

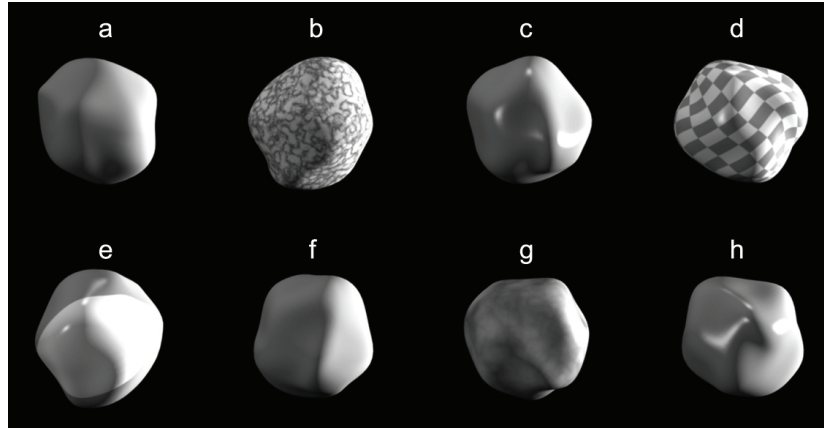


Figure 2.1: Example image for our first experiment: eight abstract objects with a main light coming from the right.

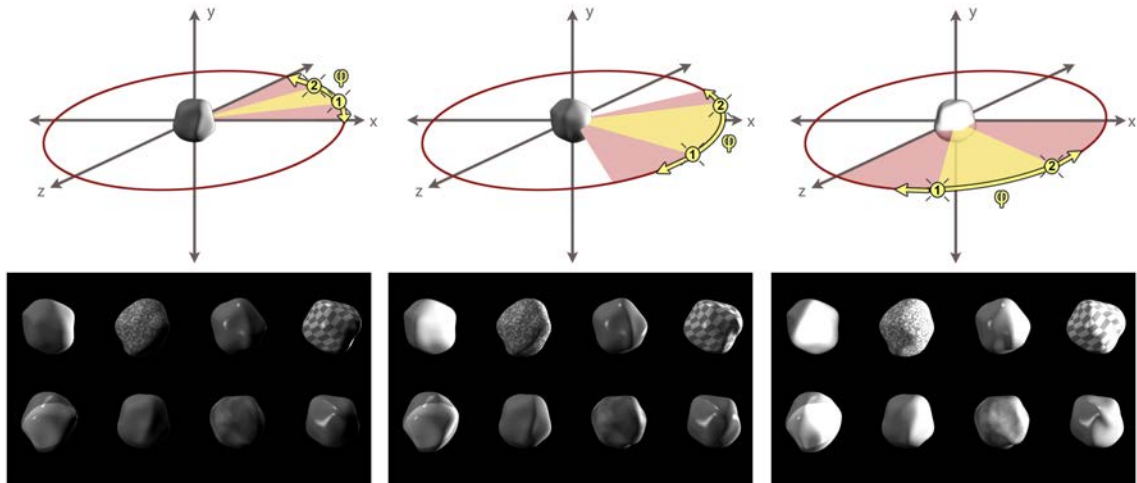


Figure 2.2: **Top Row:** 3D representation of the scenes rendered in our images. Light 2 is the global light of the scene and light number 1 is the "wrong" light affecting a single object. The angular divergence of the direction of the two light sources is shown in yellow for the case of 60° of divergence, while the maximum 90° of divergence is displayed in red for each case. **Bottom Row:** the correspondingly lit objects.

with both sources illuminating from behind the object and finally with one light coming from the back and the other from the front (Figure 2.2).

Half the times a shiny object is incorrectly lit and the other half a diffuse object is incorrectly lit. There are thus 60 images in total (10 increasing degrees of divergence, times three light configurations, times two types of inconsistently lit objects), each showing eight asymmetrical objects with different textures and degrees of shininess. Each image has a resolution of 1024 pixels wide by 600 pixels high. The order in that images were displayed was randomized, as well as the object that was inconsistently lit in each image. The test was performed through a web application, where users were asked, after an

2.3 Experiment One: Overall Inaccuracy

introductory explanation, to simply select the inconsistently lit object in each image. Although the time it takes each participant to complete the test is measured, there is no limitation in that regard. 55 participants took the test (ages 16-58; 33 male, 22 female), 18 of which had an artistic background.

2.3.1 Results

We analyze the number of correct answers (which we term *hits*) depending on the difference between the two lights for the two material cases: diffuse and shiny, according to the different configuration of lights (Figure 2.3). We can observe that up to 20° of divergence the probability of detection is around chance (12.5%). In the case that both lights are in the front this probability keeps on being below chance up to 30° . On the contrary when the lights are at the back the probability of detection is higher at 20° of divergence. This seems to agree with previous studies (KvDP04), suggesting that shaded areas and self-shadows increase our accuracy inferring light directions from images.

Furthermore, we can observe that for any position of the light source, the performance of HVS is slightly lower when highlights are present. Although further analysis should be carried out to find out why highlights have an apparently negative effect, this seems to agree with Todd and Mingolla's (TM83) previous work, which diverges from some computer vision approaches which do use highlights as visual cues (LF06).

We found no statistical difference across genders for this particular task, as opposed to other tasks like *mental rotation*, which has shown different reasoning strategies per gender (HTE06). Our results also showed that participants with an artistic background had significantly better results at judging light directions, achieving about 15% more correct answers on average.

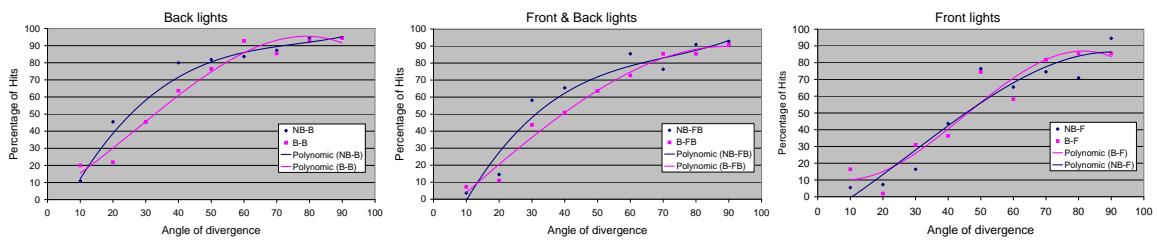


Figure 2.3: Hit probability by quadrant for both shiny (B, pink) and diffuse (NB, blue) materials. **Left:** with frontal position. **Middle:** with back position. **Right:** with front-back position.

Regarding the time spent per image, the average was 15.13 seconds. For the diffuse material, as expected, times were shorter as the error increased, meaning it was easier to spot (see Figure 2.4). However, the trend is less obvious in the presence of highlights: again, highlights seem to play a negative role for this particular task that is worth studying further.

2. THE PERCEPTION OF LIGHT INCONSISTENCIES

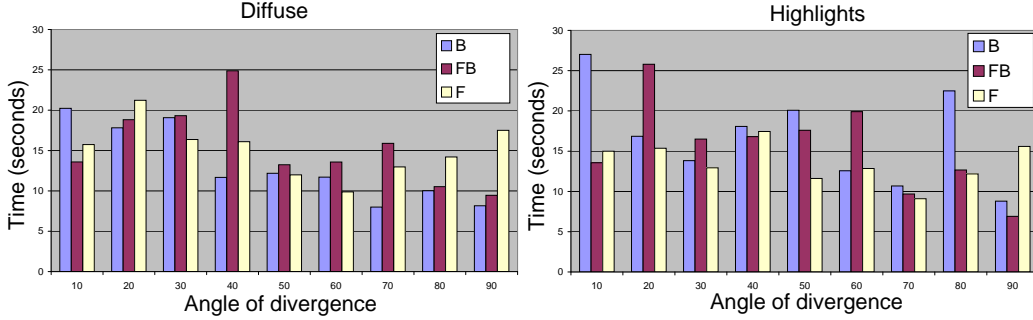


Figure 2.4: Time used to make decisions in our test, shown by increasing divergence and grouped by quadrant: Front (F), back (B) and front-back (FB). Please note that the questions were randomized and this is not a trend produced by fatigue or training.

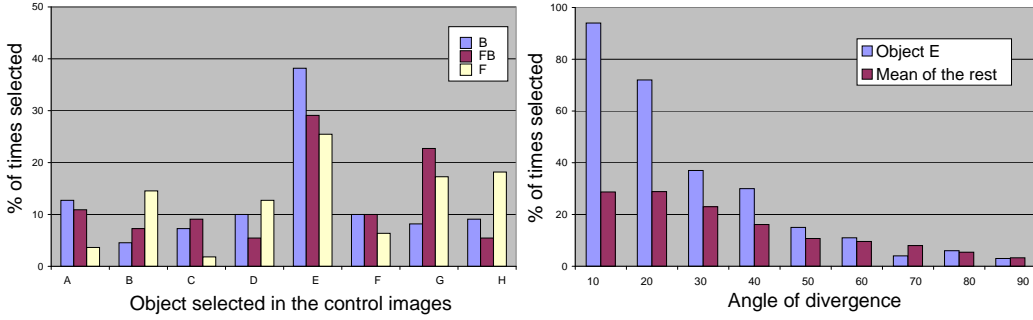


Figure 2.5: **Left:** Chosen object in the control images, grouped by quadrant: Front (F), back (B) and front-back (FB). The users have a preference for object E. **Right:** The relative saliency of the object E, computed as the number of times when it is chosen while missing the right choice. This is plotted in relation with the saliency of the remaining objects.

Object saliency: Amongst the 60 images there are six *control* images (0-degree divergence) in which all objects are illuminated correctly; this can help us detect potential salient objects. Figure 2.5 shows a bar chart with the different options that users have selected for these images. Each of the three bars corresponds to the three positions of the lights (both lights behind the object predominating the shadows versus the lights, one front and one back and two lights in the front, predominating the lights versus the shadows). It is interesting to notice that there is a clear outlier, object *E*, probably due to its particular geometry and white albedo patch. In the chart of Figure 2.5 we can observe how its saliency compared with the remaining objects is reduced in direct relation with the increase of divergence. In other words, for low or no divergence in light direction, object *E* was selected due to salient features outside the purpose of this test. But as the degrees of divergence increase, its saliency becomes less apparent due to the presence of a clearly incorrectly-lit object.

Additionally, five users were shown the same series of images as in our previous test, but in this case they were not given any specific task and were asked just to observe the images during a limited time, which was set to 15 seconds based on the average time per question of the previous test. We divided each image in eight regions of interest (ROI) corresponding to the eight synthetic objects and tracked their average eye fixation time in order to analyze the evolution of saliency per object.

2.4 Experiment Two: Influence of Texture

From the resulting heat maps (see Figure 2.6), we can analyze the gradient of the saliency for a incorrectly lit object. This can be done due to the design of this test: the inconsistently lit objects alternate between being incorrectly lit and being illuminated as the rest. For instance, at 10° of divergence F is inconsistently lit and A is correct while for 20° , A is correct and F is wrong, etc. Figure 2.7 shows the results, where an overall alternancy in saliency can be observed, as expected. However, more experiments need to be carried out to disambiguate other factors such as highlights, texture and geometry.

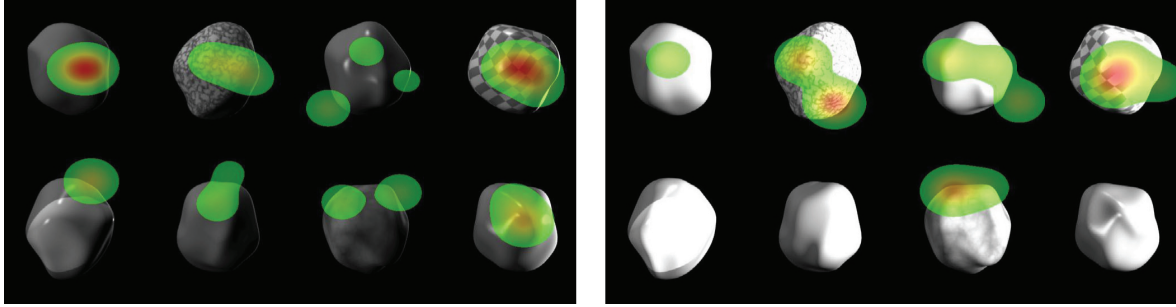


Figure 2.6: Example of heat maps representing average fixation time at two images for one user.

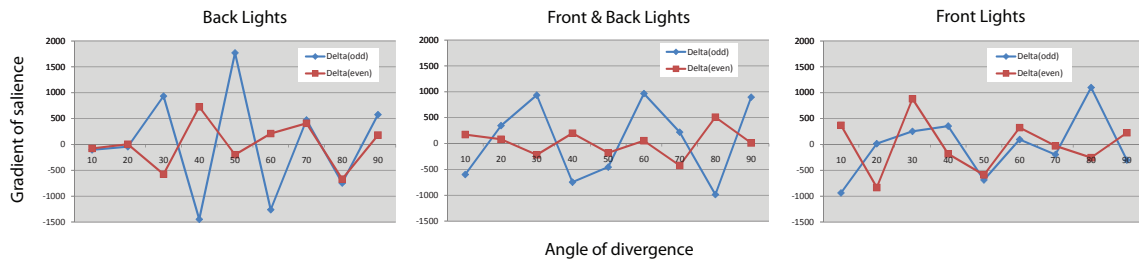


Figure 2.7: Gradient of the ratio between time spent watching the reilluminated object and the average time spent watching the rest of objects. At each graph, object A is represented in red (inconsistently lit at 20° , 40° , 60° and 80°) and object F is represented in blue (inconsistently lit at 10° , 30° , 40° , 50° , 70° and 90°).

2.4 Experiment Two: Influence of Texture

In this experiment we aim to analyze the influence in the perception process of the spatial frequency of the texture. The psychophysical test consists of a new series of images, which has been shown to 32 users (ages 22-57; 23 male and 9 female). The test was displayed using the same methodology as in Experiment One.

We analyze four different checkerboard textures of increasing spatial frequency (which we term *low*, *medium*, *medium-high* and *high*). Each one has a tile size two times smaller than the previous one. We do not aim to explore the luminance frequency, instead we fix the luminance ratio between the two albedos so that shading cue is always perceivable. With this configuration (AP96), the luminance of a clear tile in shadows is similar to the luminance of a dark tile in a lit area (See Figure 2.8). The shininess of the material is set to a 50% of the value used for shiny objects in the previous test. This

2. THE PERCEPTION OF LIGHT INCONSISTENCIES

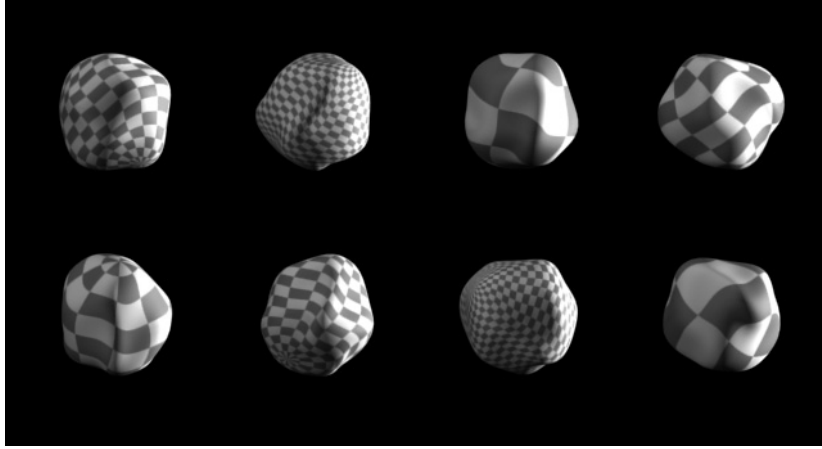


Figure 2.8: An example of an image used in our test. Four different texture patterns are assigned to eight random objects.

is done in order to analyze the results. The shape of the curve should fit between the curves for diffuse and shiny objects of the previous test.

Each user observes a series of 40 images (4 textures x 10 divergence values) with lights being modified in the same fashion as in our previous test. In order to reduce dimensionality, we limit the movement of the lights to the front-back quadrant. For each image, a random object is selected to be inconsistently lit (with a certain texture) and for the remaining objects both the textures and the geometries are randomly set.

2.4.1 Results

In Figure 2.9 we can observe a similar curve as in the first experiment, with some differences for the four textures. From the data collected, it seems that higher frequencies do mask lighting inaccuracies up to the detection threshold of 20° - 30° , making the detection task more difficult. For divergence angles above 40° we found no significant difference ($p > 0.05$) in the results. This shows that, at least for the pattern shown and the frequencies used, no amount of high frequency texture information can mask large inaccuracies in low frequency lighting information. This seems to coincide with the results of Khang *et al.* (KKK06) which suggest that the visual system may not take intensity variations due to the surface material or the light field into account when estimating the direction of illumination. We find an interesting line of future work in analyzing the transition area from masking to non-masking effects of the texture and the interplay between high and low frequency information in an image.

2.5 Experiment Three: Real World Images

In order to explore how well our findings carry over into real images, we have run two additional experiments with modified photographs as stimulus. The display methodology was based on the same web test as in previous experiments.

2.5 Experiment Three: Real World Images

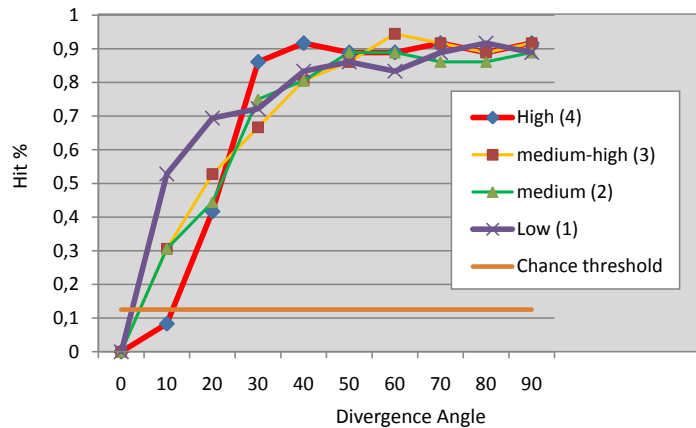


Figure 2.9: Statistics of the responses provided by users in the test, shown by texture frequency.

Experiment 3.1: The first test consists of a simple scene containing a set of eight real objects (see Figure 2.10). The scene was photographed three times: the original scene, plus two more with the angle of the main light source varying 20° and 30° respectively. Two objects from the original image were replaced by their counterparts from the two images with varying light sources. They were composited on top of the original image: the ceramic purple doll and the Venus figurine, both having diffuse and specular components and near-constant albedos. We thus create two "real world" equivalents of objects inconsistently lit, as in our first two experiments: one image with two objects incorrectly lit at 20° and a second one at 30° .



Figure 2.10: Image used in our test, in which the doll and the statue of Venus have been reilluminated. **Left:** The divergence between the lights of the objects reilluminated and the rest is 20° . **Right:** The divergence between the lights of the objects reilluminated and the rest is 30° .

Each image was shown to 25 users (ages 17-62, 14 male and 11 female) which were asked the following question: *In the following image one or two objects have been inserted and they have a different illumination than the rest of the scene. Could you point it/them out?*

In the test, 28% of the users succeeded in spotting one object for the 20° image (see Figure 2.11) whereas, as expected, for 30° of divergence this amount increased up to 36%. Both cases however, are below chance (40, 625%, considering the number of participants that chose one object and the number of participants that chose two). Only one person out of 25 was able to spot both objects, which is slightly above the chance value (3, 125%).

2. THE PERCEPTION OF LIGHT INCONSISTENCIES

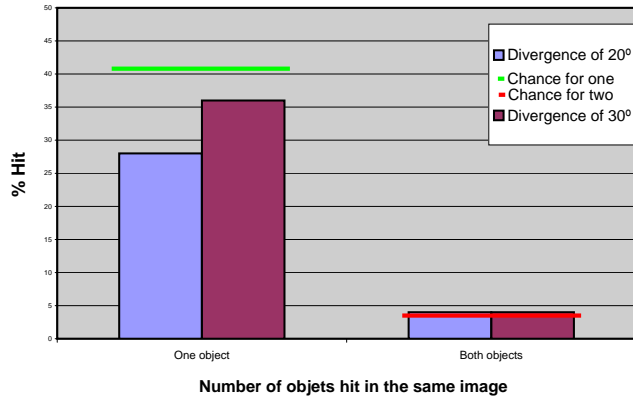


Figure 2.11: Hit ratio by angle of divergence, grouped by users who spotted correctly one (left) and two objects (right) for both 20° and 30° .

Experiment 3.2: The test 3.1 was not intended to be exhaustive, but it was designed to give some insight on how conservative a 20° - 30° threshold may be in a real-world scenario (in the absence of tell-tale shadows). Our results suggest that it may indeed be overconservative for real images. Our next test aims at generalizing a bit more those findings and it includes objects covering additional materials, textures and shapes; additionally, we extend the range of divergence up to 40° .

In this experiment nine versions of a new scene were generated (See Figure 2.12). Four photographs of the same scene were taken at 0° , 20° , 30° and 40° of divergence from a reference direction. They were combined in the same fashion as in the previous test, but in this case three different objects were masked out and only one object was combined at a time thus obtaining nine versions of the same scene (three objects times three divergence degrees). The black background was used to avoid projection of shadows on a parallel surface and the image composition is done with Poisson-based alpha matting. The result is almost seamless as the local environment of the selected object in both images is very similar.

The objects selected for modification cover a wide range of materials, shapes and positions in the scene: the Santa Klaus doll (diffuse material, high frequency geometry, background position), the metallic robot (Highly specular, rightmost foreground position) and the clown doll (multiple albedo, diffuse, leftmost background position). In total, 60 users (ages 18-59, 38 male and 22 female) took the test. Each user was shown three images with a random inconsistently lit object at 20° , 30° and 40° of divergence respectively. The same object was never shown more than once per user.

The results of the test (Figure 2.13) present a similar trend to those from our synthetic experiment, but slightly more conservative: whereas in the synthetic scenes (Experiments One and Two), the detection threshold was somewhere between 20° and 30° , the variety of real world shapes and materials seems to increase that threshold to the 30° - 40° range.



Figure 2.12: **Top:** Original image with all the objects consistently lit. **Bottom:** Example of image used in our experiment. The Santa Klaus doll is lit with a divergence of $\phi = -40^\circ$ from the global light direction.

2.6 Conclusions and Future Work

We have presented the results of four different tests, whose overall goal was to quantitatively measure the accuracy of human vision detecting lighting inconsistencies in images. We have restricted ourselves to the case of inconsistent light direction. The results of our experiments seem to agree with the theories exposed in previous research on illumination perception (OCS05; KvDP04; LMSLG09), but we have extended those to suggest a perceptual threshold for multiple configurations. Additionally, we have shown how that threshold seems to be even larger for real-world scenes. Although we do not claim our experiments to be exhaustive, we do believe they add significant value to the current state of the art.

We can find several possible interpretations to the fact that lighting inconsistencies were harder to detect in real-world images: it may simply be that the combination of multiple visual cues (texture, shading, highlights...) which was richer than in the CG scenes, might have complicated the detection task. But it is also interesting to dig into the influence produced by the different range of naturalness of the images.

In similar contexts (3D shape perception) some authors have related naturalness of stimuli to reduced activation in the visual cortex (V1) (MKO⁺02; GTPO08), which is related to low-level vision. Although the exact relationship between naturalness and the detection process remains unclear, Scott

2. THE PERCEPTION OF LIGHT INCONSISTENCIES

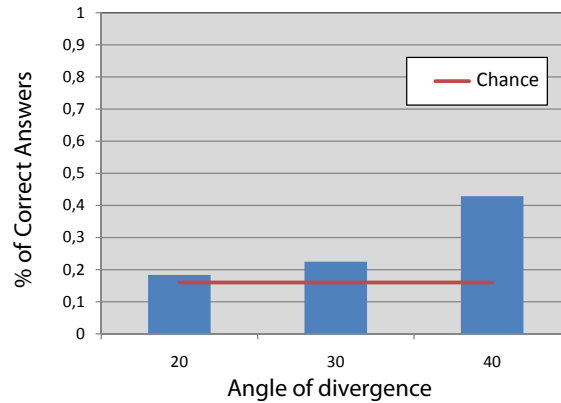


Figure 2.13: Hit ratio by angle of divergence for 20°, 30° and 40°.

et al.(MKO⁺02) suggest that under reduced activity in V1 for grouped elements, isolated or novel elements may be more readily detected. There is an apparent contradiction with our results, which might be due to the fact that prior knowledge of 3D shape and material may reduce accuracy. Also, the degree of visual grouping of objects in the synthetic scene (all objects were semantically the same) could have been greater than in the real images (possibly due to increased visual and semantic complexity of individual objects). This might have augmented the tolerance to illumination differences, but in any case it remains a fascinating problem to study.

We believe that the present work may be of value for those areas of computer graphics and vision that depend on analyzing the lighting environment, including algorithms based on light detection and methods for image synthesis (augmented reality...) analysis (digital forgery detection) and processing (special effects). Given that light detection in an image is an ill-posed problem, being able to work within perceptual error thresholds can make the problem tractable as we propose in the Chapter 3.

References

- [AP96] E. H. Adelson and A. P. Pentland, The perception of shading and reflectance, 409–423, Cambridge University Press, New York, NY, USA, 1996, pp. 409–423. 21
- [GTPO08] Svetlana S. Georgieva, James T. Todd, Ronald Peeters, and Guy A. Orban, *The Extraction of 3D Shape from Texture and Shading in the Human Brain*, Cerebral Cortex (2008), bhn002. 25
- [HTE06] Kenneth Hugdahl, Tormod Thomsenb, and Lars Erslund, *Sex differences in visuo-spatial processing: an fmri study of mental rotation*, Neuropsychologia (2006), no. 3, 15751583. 19
- [JF05] Micah K. Johnson and Hany Farid, *Exposing digital forgeries by detecting inconsistencies in lighting*, MM&Sec '05: Proceedings of the 7th workshop on Multimedia and security (New York, NY, USA), ACM, 2005, pp. 1–10. 15
- [JF07] Micah K. Johnson and Hany Farid, *Exposing digital forgeries in complex lighting environments*, IEEE Transactions on Information Forensics and Security **2** (2007), no. 3, 450–461. 15
- [KKK06] B.G. Khang, J.J. Koenderink, and A.M.L. Kappers, *Perception of illumination direction in images of 3-D convex objects: Influence of surface materials and light fields*, Perception-London **35** (2006), no. 5, 625. 22
- [KvDP04] J. J. Koenderink, A. J. van Doorn, and S. C. Pont, *Light direction from shad(ow)ed random gaussian surfaces*, Perception **33** (2004), no. 12, 1405–1420. 16, 17, 19, 25
- [LB01] M S Langer and H H Bülthoff, *A prior for global convexity in local shape-from-shading*, Perception **30** (2001), 403–410. 17
- [LF06] Pascal Lagger and Pascal Fua, *Using specularities to recover multiple light sources in the presence of texture*, ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition (Washington, DC, USA), IEEE Computer Society, 2006, pp. 587–590. 19
- [LMHRG10] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Compositing images through light source detection*, Computers & Graphics **In press** (2010). 15
- [LMSLG09] Jorge Lopez-Moreno, Francisco Sangorrin, Pedro Latorre, and Diego Gutierrez, *Where are the lights?. measuring the accuracy of human vision*, CEIG '09: Congreso Español de Informática Gráfica, 2009, pp. 145–152. 15, 17, 25
- [LMSSG10] Jorge Lopez-Moreno, Veronica Sundstedt, Francisco Sangorrin, and Diego Gutierrez, *Measuring the perception of light inconsistencies*, Symposium on Applied Perception in Graphics and Visualization (APGV), ACM Press, 2010. 15

REFERENCES

- [MG01] Pascal Mamassian and Ross Goutcher, *Prior knowledge on the illumination position*, *Cognition* **81** (2001), no. 1, B1 – B9. 16
- [MKO⁺02] Scott O. Murray, Daniel Kersten, Bruno A. Olshausen, Paul Schrater, and David L. Woods, *Shape perception reduces activity in human primary visual cortex*, *Proceedings of the National Academy of Sciences of the United States of America* **99** (2002), no. 23, 15164–15169. 25, 26
- [MT86] E. Mingolla and J.T. Todd, *Perception of solid shape from shading*, *Biological Cybernetics* **53** (1986), 137–151. 16
- [OBA08] James P. O’Shea, Martin S. Banks, and Maneesh Agrawala, *The assumed light direction for perceiving shape from shading*, APGV ’08: Proceedings of the 5th symposium on Applied perception in graphics and visualization (New York, NY, USA), ACM, 2008, pp. 135–142. 16
- [OCS05] Yuri Ostrovsky, Patrick Cavanagh, and Pawan Sinha, *Perceiving illumination inconsistencies in scenes*, *Perception* **34** (2005), 1301–1314. 16, 17, 25
- [SP98] Jennifer Sun and Pietro Perona, *Where is the sun?*, *Nature Neuroscience* **1** (1998), no. 3, 183–184. 16
- [TM83] J.T. Todd and E. Mingolla, *Perception of surface curvature and direction of illumination from patterns of shading*, *Journal of Experimental Psychology Human Perception and Performance* **9** (1983), no. 4, 583–595. 16, 19
- [VLD07] Peter Vangorp, Jurgen Laurijssen, and Philip Dutré, *The influence of shape on the perception of material reflectance*, *ACM Trans. Graph.* **26** (2007), no. 3, 77. 17
- [WS02] Y. Wang and D. Samarasinghe, *Estimation of multiple directional light sources for synthesis of mixed reality images*, *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, 2002, pp. 38–47. 15
- [YWAC06] T Yu, H Wang, N Ahuja, and W-C Chen, *Sparse lumigraph relighting by illumination and reflectance estimation from multi-view images*, *Eurographics Symposium on Rendering*, Eurographics Association, 2006, pp. 41–50. 15
- [ZY01] Yufei Zhang and Yee-Hong Yang, *Multiple illuminant direction detection with application to image synthesis*, *IEEE Trans. Pattern Anal. Mach. Intell.* **23** (2001), no. 8, 915–920. 15

Chapter 3

Light Detection in Single Images

This chapter deals with the problem of obtaining the positions and relative intensities of light sources in a scene, given only a photograph as input. This is generally a difficult and under-constrained problem, even if only a single light source illuminates the depicted environment. We present two novel algorithms for multiple light detection that leverage the limitations of the human visual system (HVS) described in the literature and measured by our own psychophysical study. Finally, we show an application of our method to both image compositing and synthetic object insertion.

This research has given rise to two publications: a paper at the conference CEIG 2009 (organized by the Spanish Eurographics Chapter) (LMHRG09) and an article at the Computers & Graphics Journal, indexed Q3 in JCR list (LMHRG10). A third publication, showing our new optimization method and osculating arc shape estimation is planned to be submitted this year to the Computer Graphics Forum journal (JCR listed Q1).

3.1 Introduction

Traditionally, in the field of image based lighting, a lightprobe is used to acquire the illumination condition in the scene. A lightprobe is an object of a known simple geometry and known reflectance properties, which is placed near the subject in the scene that is being photographed. A chrome ball is the most popular lightprobe artifact. Even though the technique is very accurate and effective in capturing the details of the illumination, it is nevertheless an intrusive method.

In this chapter, we would like to address the difficult problem of robustly estimating the illumination in a natural scene, and not have the constraints of a prescribed workflow such as (intrusive) use of the lightprobe. We propose to use a user selected arbitrary subject in a single image, which would be used as a “virtual” lightprobe instead. Our goal is to be able to estimate the illumination in terms of number of few distinct light sources, their directions, possibly their positions and their relative intensities in a scene.

However, the task at hand is highly ill-posed problem; even in the simplified case when only a single light source illuminates the subject. The problem is subject to unknown geometry, unknown albedo and unknown reflectance properties of the virtual lightprobe. In addition, we would like to be able to

3. LIGHT DETECTION IN SINGLE IMAGES

handle the general case of illumination from multiple light sources and peculiarity of illumination from light behind the subject – backlighting. In (GHH01) it is shown that, if an approximate geometry of the subject is known, it is possible to use the subject in the image to estimate the illumination. However, specifying even an approximated geometry of the subject is still fairly labor intensive and we would like to keep the user interaction to a minimum.

Such an ill-posed problem will necessarily lead to an approximated solution. However, our psychophysical experiments, shown in Chapter 2, suggest a threshold for the accuracy with which humans can generally spot flaws in rendered illumination (LMSSG10), (LMSLG09). We will show that our method yields valid illumination estimates that remain within those thresholds. This allows for a wide range of applications related to image compositing, such as image editing and classification, digital forgery and augmented reality. Examples are given in Section 3.8.

3.2 Previous Work

Visual effects, animation and games industry have very successfully used lightprobes to accurately capture the incident lighting in a scene. As part of their standard workflow, they photograph the scene after inserting the light probe at one or multiple key locations. The lightprobe is a simple calibration object of known size and shape with known reflection properties. For instance, a Lambertian sphere inserted in the scene can be analyzed for estimating directions of multiple light sources (HA93; ZY01). Further, multiple specular spheres have been effectively used to triangulate the accurate positions of the lights (PSG01; LF06). It is possible to use a combination of Lambertian and specular spheres (ZK02), or even analyze the reflections in human eye to detect light sources (NN04). Finally, High Dynamic Range images of specular light probe are successfully used in acquiring very detailed illumination environment, which is subsequently used to render synthetic objects – technique known as Image Based Lighting.

However, for the most of the workflows (general photography e.g.), the availability of the lightprobe in the scene is not practical. Detecting lights in this case is difficult and the solutions typically involve making significant and restrictive assumptions about the nature of the scene.

If we assume that the subject in the scene is illuminated by a single light, it significantly simplifies the analysis and we can use the subject itself to detect the incident lighting. In this case, a local analysis of the surface and image derivatives is used to estimate the direction of the light source (Pen82; BH85). Alternatively, occluding contours of a single object (Hor86; NE01) or the texturing (KP03; VZ04) provide clues as to where the light is coming from.

If the geometry of the subject in the scene is known, or can be specified with a certain accuracy, light source positions or directions can be estimated (GHH01). Conversely, if we want to estimate the geometry of the subject in the scene, an ill-posed problem known as Shape from Shading (ZTCS99), we need to know the incident illumination. It is a chicken-and-egg problem. Further, in either case, the reflectance properties of the subject have huge bearing on the results, which is also unknown. One way to overcome the under-constrained nature of the problem is to use a range camera to record the geometry, allowing light sources to be inferred from the combination of the photograph and the range data (MG97). Known geometry can be used to the same effect (WS02; SS199; XW08).

In comparison to the state-of-the-art, our proposed method is free of such restrictions. In particular, there is no need for a calibration object or a subject with a known geometry. Our method is fairly robust with respect to the reflectance properties and the albedo variations. Finally, we are able to detect multiple lights in any complex configuration, including backlighting.

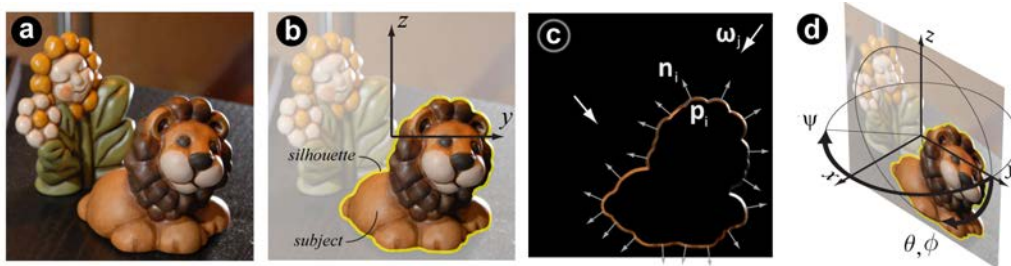


Figure 3.1: a) Input Image, b) Object, c) Silhouette Normals, d) Coordinate System

3.3 Perceptual Framework

Natural illumination in real environments is often complicated, making its analysis by both machines and humans difficult. Natural illumination exhibits statistical regularities that largely coincide with those found for images of natural environments (DLAW01), (PCR10). In particular, the joint and marginal wavelet coefficient distributions, harmonic spectra, and directional derivative distributions are similar. Nonetheless, a complicating factor is that illumination is not statistically stationary because of locally dominant light sources (DWA04). By representing illumination with spherical harmonics, Mury et al. (MPK07) have recently shown that low-order components show significant regularities, whereas the statistical non-stationarity is captured in the higher frequencies. Moreover, variation in the low-frequency representation tends to covary with the geometry rather than with the illumination. This interpretation is consistent with evidence suggesting that human vision assumes a priori the global convexity of object shapes (LB01). Thus, human vision may apply the dark-is-deep paradigm, namely, that globally darker shading values indicate surface points that are further away than lighter values. Natural scenes, however, contain significant high-frequency components, and these complicate analysis. It is possible that human vision ignores these components, and this may help explain why our vision is not accurate in the perception of illumination in cluttered environments (OCS05).

3.4 Estimating Light Sources

For our purposes a directional light source will be defined by its 3D direction (determined by two angles: *azimuth* θ and *zenith* ψ), or by a position if it is a point light. The purpose of our algorithm is to estimate these values, as well as a relative intensity value, for each light source.

In Figure 3.1, we can observe the coordinate system used in this chapter. As depicted in Figure 3.1d, the image plane is assumed to be aligned with the $y-z$ plane, whereas the x -axis points out of the image plane. The origin lies at the center of the image. We also set a polar coordinate system (with angles: *azimuth* θ and *zenith* ψ) such that the equator is aligned with the image plane and the axis is aligned with x -axis.

We estimate the light sources on the basis of a single input image in which the user contours an object which will act as our light probe. Such input can be generated in a short amount of time, even by unskilled users. The selected area is then preprocessed to remove highlights, noise and albedo information (Section 3.5). The subsequent light detection algorithm follows a two-step process.

3. LIGHT DETECTION IN SINGLE IMAGES

In the first step, the contour of the object provides sufficient information to determine the number of light sources, and to detect their position in screen space (Section 3.6). We propose two different approaches for this estimation: One is based in K-means clustering (see Subsection 3.6.1) and the other in light source-fitting optimization (Subsection 3.6.2). The latter method, has shown increased accuracy and robustness at the cost of a more complex implementation. However, we include the K-means approach in this chapter due to the good results achieved in spite of its simplicity. Furthermore, we think that this technique allows for further research which could increase its accuracy.

In the second step, the object's interior is used to infer the zenith (Section 3.7). Likewise, we propose two approaches for its computation; ellipsoid approximation and osculating arc fitting. The first technique has been successfully used to estimate light sources but we have included it in this chapter for the sake of completion, as we recommend the use of the second method because it is able to approximate the shape even if the lightprobe is not globally convex. It suffices if there is a convexity near the contour, widening the range of lightprobes available for our method.

Our methods aim to perform light detection with a wide range of lightprobes commonly found in images with very limited user input. To obtain a feasible solution in such an ill-posed scenario, we make the following assumptions:

- The object's material is assumed to be diffuse (Lambertian).
- The object is globally convex, an assumption that underlies some processing found in the human visual system (LB01). As most objects adhere to this requirement, this is not a strong assumption.
- The estimated lighting environment consists of an unknown number of light sources (point or directional) with unknown intensities.
- Due to the unknown albedo it is not possible to determine the color of the lights without user intervention thus it will be assumed as white by default.
- 3D normals at the contour of the lightprobe are assumed to lie in the screen plane (Hor86).

We argue that this set of assumptions applies to a large enough set of scenes and objects for our algorithms to be practical.

3.5 Pre-processing

Our light detection algorithm assumes that the chosen light probe is made of a diffuse material. To extend its use to dichromatic materials, we preprocess the image to separate the specular component from the shading. For this, we follow the methods proposed for specular removal in Chapter 1, Section 1.2, based in the change of color space suggested by Mallick et al. (MZBK06).

This step is of vital importance, as we assume a constant reflectance term in our computations from this point on. In general, the low frequency term is accurate enough for our purposes but for certain materials (e.g.: a black and white checkerboard texture) the presence of noise and local deviations from the lambertian model is to be expected. In those cases the error yielded by our method is increased progressively but the results are still plausible as we demonstrate in Section 3.8.

3.6 Estimating Azimuth Angles

In the following subsections we propose two different methods to estimate the Azimuth angle. The impact in the final result of choosing one or the other are discussed in the Section 3.8.

3.6.1 K-means approach

The silhouettes of objects have surface normals that are approximately perpendicular to the viewing direction. It is therefore reasonable to assume that the surface normals of the contour of objects lie in the image plane. This assumption is termed *occluding contours*, and has previously been successfully used to detect light sources (Hor86; NE01).

In the following, we are analyzing the contour of an object, as this is where accurate surface normals are given. The problem involves finding either directional or point light sources which lie in the image plane. In Section 3.7 we reintroduce the third dimension by computing the elevation angles for the light sources determined here. The points on the contour are given by \mathbf{p}_i . The pixel values can be converted to luminance, indicated by $L_{\mathbf{p}_i}$. Their surface normals are given by $\mathbf{n}_i = [\cos(\phi_i), \sin(\phi_i)]$. Thus, each surface normal can also be represented by azimuthal angle ϕ_i . If multiple pixels share the same surface normal ϕ_i , we represent this set of pixels with their median luminance value, and therefore run our calculations on fewer pixels. This helps to streamline the optimization process. Finally, the total number of pixels on the contour is assumed to be $N_{\mathbf{p}}$, while the number of contour pixels on which calculations are carried out, is given by N_{ϕ} .

During the estimation process, an estimated light source k is characterized by its direction θ_k and the amount of light that reaches the object's contour L_k^{in} . After rendering the 3D model of the contour, the current set of N estimated light sources gives rise to a set of $N_{\mathbf{p}}$ pixel luminances $L'_{\mathbf{p}_i}$:

$$L_{\mathbf{p}_i} = \sum_{k=1}^N \Omega_{ik} L_k^{\text{in}} \quad (3.1)$$

$$\Omega_{ik} = \Omega(\phi_i, \theta_k) = \begin{cases} 0 & \text{if } \cos(\phi_i, \theta_k) < 0, \\ K_d^i \cos(\phi_i, \theta_k) & \text{if } \cos(\phi_i, \theta_k) \geq 0 \end{cases}$$

where K_d^i is the unknown diffuse reflectivity or albedo of pixel i .

To estimate the lights' direction θ_k , we use an adaptive k-means clustering algorithm. Usually in k-means clustering algorithms, each data point belongs to a certain cluster and affects only to the computation of its corresponding centroid. In our case, a silhouette pixel may be illuminated by more than one light. Thus, we cannot partition the pixels into exclusive clusters. Instead, we devise a

3. LIGHT DETECTION IN SINGLE IMAGES

partial voting scheme based on the Ω function to form 'fuzzy' clusters and to simultaneously compute the corresponding centroids as the lighting directions, as outlined in Algorithm 1.

Require: $L_{\mathbf{p}} \equiv \{L_{\mathbf{p}_i}\}$ {discrete luminances}
Require: $\phi \equiv \{\phi_i\}$ {silhouette normals (characterized by their azimuth angles)}

- 1: $\text{sort}(L_{\mathbf{p}_i}, \phi_i)$ {sort by decreasing luminances}
- 2: $\theta^l \equiv \{\theta_k\} \mid k \in [1 \cdots N]$ {azimuth coordinates of the lights}
- 3: $\text{seed}(\theta^l)$
- 4: $\alpha^\oplus \equiv \{\alpha_k^\oplus\} \mid k \in [1 \cdots N]$ {aggregate of weights per light}
- 5: $\alpha^\oplus \leftarrow \mathbf{0}$
- 6: **repeat**
- 7: **for all** $L_{\mathbf{p}_i} \in L_{\mathbf{p}}$ **do**
- 8: $\Omega_i^\oplus \leftarrow \sum_k \Omega(\phi_i, \theta_k)$ {total weight}
- 9: **for all** $k \in [1 \cdots N]$ **do**
- 10: $\alpha_{ik} \leftarrow L_{\mathbf{p}_i} \Omega(\phi_i, \theta_k) / \Omega_i^\oplus$ {weight of normal i }
- 11: $\theta_k \leftarrow \alpha_k^\oplus \theta_k + \alpha_{ik} \phi_i$ {update direction}
- 12: $\alpha_k^\oplus \leftarrow \alpha_k^\oplus + \alpha_{ik}$
- 13: $\theta_k \leftarrow \theta_k / \alpha_k^\oplus$
- 14: **end for**
- 15: **end for**
- 16: **until** convergence(θ^l)

Algorithm 1: Contour Voting - N lights

In order to perform the normal voting, we go through the list of pixels sorted by luminance (line 7). Notice that each silhouette normal ϕ_i votes for all the N light clusters (lines 10 to 16), according to their luminances $L_{\mathbf{p}_i}$. However, each normal only partially votes for each light cluster, according to the Ω function (line 12). For that, the individual Ω function with respect to each light direction Ω_{ik} was normalized with the aggregate of the Ω functions $\Omega_i^\oplus = \sum_k \Omega(\phi_i, \theta_k)$.

We repeat the voting process (lines 7 to 17) until it converges on the light azimuth angles θ^l (lines 6 and 18). The choice of the initial guess (line 3) for the azimuth angles is important to ensure a speedy and effective convergence. We assign the azimuth of the brightest pixel's normal ϕ_1 to the first light θ_1 . For the successive lights, we set the azimuth angles to $\theta_1 + 2\pi(k-1)/N$.

For the estimation of the number of lights N , our approach subsequently increases the number of lights $N = 1..k$ until either the error is below a given tolerance or the added light source does not improve the result (our stopping criteria). In practice, we find that the number of iterations is usually below $N = 4$. This is due to the quantization associated with the image's finite bit-depth. As the number of opposing lights increases, the variation in the shading over the surface decreases and becomes rather constant.

Although the proposed voting method has built-in resistance to local variations in albedo because of its search of global tendencies, ultimately, the results will be biased if the points in the contour form large clusters with very different luminance values, as shown at the first image of Figure 3.2a.

It is possible to reduce this bias with a second pass, as follows. Once we have a set of N centroids (light directions), we go through all the voting pixels assigned to each k -group, corresponding to a light direction. We then check that the dot product of the normal and the estimated light direction yields a luminance value equal to the original luminance of the pixel, fractioned by its Ω function. If not, we force the fractional albedo of the pixel to be coherent with the fractional luminance of the brightest pixel in the group. Then we repeat the contour voting algorithm. This correction in the albedo values usually produces small shifts (10 to 20 degrees) in the directions in the case of extreme albedo variations (Figure 3.2a).

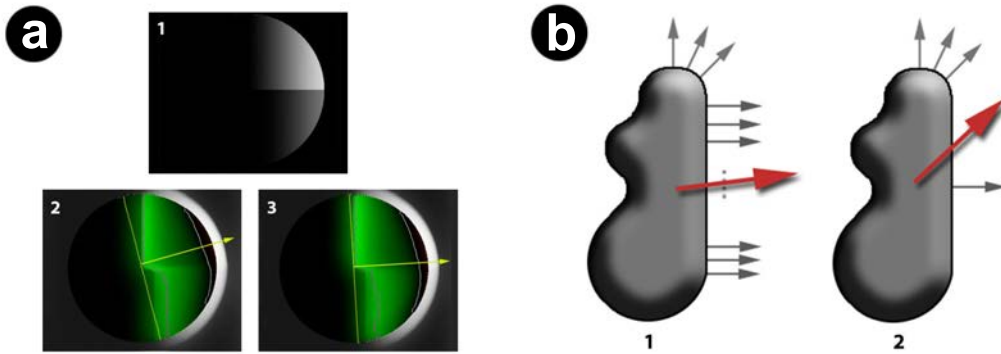


Figure 3.2: a¹) Sphere with a change in the albedo, a²) Initial biased estimation because of a higher albedo, a³) Corrected light direction estimate, b¹) An estimate incorrectly biased because of the geometry of the silhouette, b²) The correct result after eliminating multiple normals.

As in other previous approaches based on contour analysis (YY91; VY94; NE01), the first step will fail if the light is situated around the x -axis; i.e., $\psi \approx \pi/2$. In this case there is no variation in luminances due to shading. This would result in erroneous estimation of the azimuth angles. However, the final direction of the light would be estimated accurately in the second step when we analyze the shading in the interior.

Finally, we correct the potential bias along the direction stemming from the geometry of the silhouette. As depicted in Figure 3.2b, a significant number of silhouette normals are parallel to the y -axis, biasing the resultant light towards that direction. We correct this by eliminating multiple normals. We chose a set of discrete normal directions $\bar{\phi}_i$ and distributed all the silhouette normals into bins. Then, we compute the average luminance for each bin \bar{L}_i and use this set of silhouette normals and luminances instead.

3.6.2 Light Source Fitting Approach

In the following, as with our K-means method, we will analyze the pixels of the contour and their corresponding normals, assumed to be lying in the image plane. For multiple light sources, it would be possible to use known geometry of the surface and to rely on locating *critical points*, which are the points at the boundary of surface areas that are affected by a different combination of lights (ZY01; WS02; BB04). However, these techniques require the surface geometry to be known. In our case, we only have reliable surface normals at the contour, so that these algorithms are less suitable. Furthermore, these algorithms will detect directional lights only, and may become less effective in the presence of noisy input.

Our *K-means* approach overcomes many of these limitations, however, after testing it we found a limitation: when two light sources have overlapping azimuth angles (approximately less than 60 degrees), due to its greedy nature the K-means approach tends to group them into a single light direction, in between both light sources. Figure 3.3 shows a failure case of the *K-means* method. It can be seen how the two top light sources have been collapsed into one, whereas the *Light Source-Fitting* method yields more accurate results.

3. LIGHT DETECTION IN SINGLE IMAGES

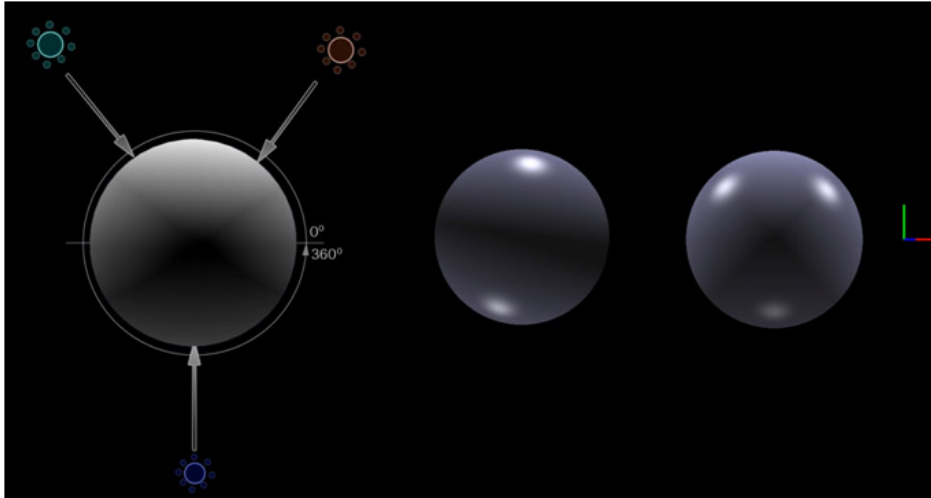


Figure 3.3: Left: Input image. Sphere lit by three light sources. Middle: Wrong result obtained by the *K-means* method, rendered in OpenGL. Right: Result yielded by the *Light Source-Fitting* method.

For these reasons, we have developed an alternative algorithm, which operates under the assumption that the object has constant albedo. We would like to explain the observed image with as few estimated light sources as possible. We therefore consider the silhouette of the object, assume it is Lambertian and begin by lighting it with one directional light source. The best position of the light source is found by optimization. Using the position of the light source, the known surface normals of the contour and the assumed Lambertian material of the object, we can then render an image of the silhouette. The light source is positioned optimally when the difference between the rendered contour and the silhouette in the image becomes minimal.

Then, the result is refined in two ways. First, we test whether a point light source could explain the observed image better than the directional light source would. This is achieved by varying the position of the point light source along the direction of the directional light it replaces. By means of Hooke-Jeeves optimization (HJ61) the best position of the point light source is found.

Second, we split the light source into a pair of sources, to test if the appropriate positioning of two light sources explains the observed contour better than a single light source. This algorithm is iteratively applied, adding more light sources until the process has converged.

3.6.2.1 Finding Light Source Candidates

To find potential light sources that best explain the luminance variation along the contour of an object, we present an iterative algorithm. In the first step we estimate a single directional light source. Then, at each successive iteration more light sources are added, until adding further light sources does no longer improve the results by a large enough amount. Optionally, we can detect whether a directional light source should be replaced with a point light source at a finite distance from the object, as discussed in Subsection 3.6.2.3.

To analyze the luminance variation of the silhouette pixels, we assume that this variation is due to shading, and in particular that the object has a Lambertian material. In that case, the amount

of light reflected depends on the angle between the surface normal and the direction of the first light source (with luminance L_1^{in} and angle θ_1):

$$L'_{\phi_i} = K_d^i L_1^{\text{in}} \cos(\phi_i - \theta_1) \quad (3.2)$$

Here, K_d^i represents the albedo of the pixels represented by surface normal ϕ_i . As our preprocessing step (outlined in Section 3.5) has removed reflectance variations at contour pixels, we simply set this term to 1.0.

For the first light source, we create an objective function O by computing the squared difference between the observed median pixel luminance L_{ϕ_i} and the luminance L'_{ϕ_i} computed by (3.2):

$$O = \underset{\theta_k, L_k^{\text{in}}}{\operatorname{argmin}} \sum_{i=1}^{N_p} (L_{\phi_i} - L'_{\phi_i})^2 \quad (3.3)$$

$$= \underset{\theta_k, L_k^{\text{in}}}{\operatorname{argmin}} \sum_{i=1}^{N_p} (L_{\phi_i} - K_d^i L_k^{\text{in}} \cos(\phi_i - \theta_1))^2 \quad (3.4)$$

This objective function is minimized subjective to a further constraint, which is designed to ensure that the total number of light sources N remains as small as possible (we use this constraint for the first as well as for all subsequent lights that are added to the set). This is achieved by requiring the maximum estimated luminance to equal the observed maximum luminance along the contour:

$$\max L'_{\phi_i} = \max L_{\phi_i} \quad \forall \{p_i \mid \phi_i \in [\theta_k - \pi/2, \theta_k + \pi/2]\} \quad (3.5)$$

To detect the first light source we minimize O subject to the maximum luminance requirement of Equation (3.5), using Hooke-Jeeves optimization.

We then successively add further light sources, without modifying the current set of N_l light sources. We also use a somewhat different optimization strategy because we need to account for the explanatory power of the existing set of light sources, while adding a new light. For each new light, the following objective function is used:

$$O = \underset{\theta_{N_l+1}, L_{N_l+1}^{\text{in}}}{\operatorname{argmin}} \sum_{i=1}^{N_\phi} \omega_i (L_{\phi_i} - L'_{\phi_i})^2 \quad (3.6)$$

$$= \underset{\theta_{N_l+1}, L_{N_l+1}^{\text{in}}}{\operatorname{argmin}} \sum_{i=1}^{N_\phi} \omega_i \left(L_{\phi_i} - K_d^i \sum_{k=1}^{N_l+1} L_k^{\text{in}} \cos(\phi_i - \theta_k) \right)^2 \quad (3.7)$$

where the difference with our first objective function lies in the weight function ω_i , which is given by:

$$\omega_i = \frac{1}{2N_l} \sum_{j=1}^{N_l} 1 - \cos(\phi_j - \phi_i) \quad (3.8)$$

The weight $\omega_i \in [0, 1]$ favors adding new light sources at directions that are maximally different from the directions of existing light sources. This increases the speed of convergence.

Upon this basic scheme, we apply two refinements. In some cases, the most recently added directional light can be replaced with a pair of light sources with directions either side of this light. Such splitting of a light source into two lights may offer a better explanation of the observed luminance profile along the contour of the object than the single directional light. Second, a point light source at a finite distance from the object may provide a better result than the corresponding directional light source. Both refinements are discussed in the following sections.

3. LIGHT DETECTION IN SINGLE IMAGES

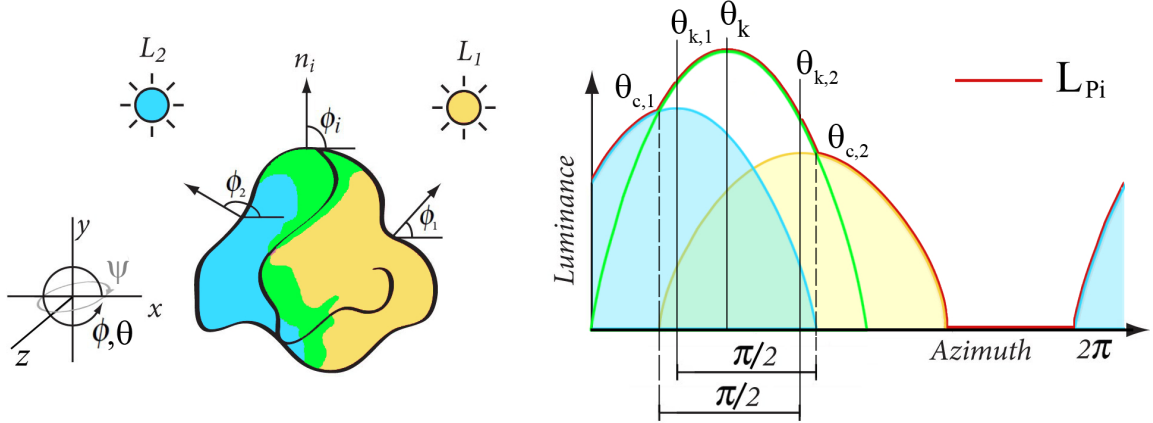


Figure 3.4: Left: Object lit by two light sources. Right: Luminance at the contour, plotted by azimuth angle of the normal at each point.

3.6.2.2 Splitting a light source

If we plot the luminance values along a contour against surface normal we may end up with a plot as shown in Figure 3.4. For a given light source estimated in the previous step, we would have found the peak luminance in this plot, associated with direction θ_i . It may be possible that the angular luminance profile is caused by a pair of directional light sources, rather than a single light source. In that case, we assume that both of these sources can be modeled by a cosine profile, which overlap.

Plotting two overlapping cosine functions results in a profile that has two inflection points, as shown in Figure 3.4. These are known as critical points. The range of directions in between these two critical points denote directions where both light sources contribute to the amount of light reflected. Critical points are therefore useful in helping us determine the luminance of the two light sources $L_{k,1}^{in}$ and $L_{k,2}^{in}$, as well as their directions $\theta_{k,1}$ and $\theta_{k,2}$. Thus, we first detect if two critical points can be found, and if so, we split the directional light source in two, and compute their luminance and angles.

There exist several algorithms to estimate critical points (ZY01; WS02; BB04). Their operation tends to be somewhat vulnerable to noise, and to combat this many samples are necessary. In our case, we only have samples along the contour of an object, which is not enough for these algorithms to produce reliable estimates. We therefore resort to a different approach.

The range of angles which could contain critical points is limited to $\theta_k \pm 90$ degrees. Starting from the direction of our initial light source θ_k , we search for both larger and smaller angles to find a critical point either side of θ_k . For each of these points we evaluate Equation 3.2 and compare against the observed luminance L . If the estimated light at θ_k is due to a pair of differently oriented light sources, then at θ_k and nearby directions the estimated luminance L' will be larger than or equal to the observed luminance. At some point along the contour, this will change. At this point, the over-estimate will become an under-estimate. Thus, the first critical angle is $\theta_{c,1} = \theta_k - \theta_{min}$, where $0 < \theta_{min} < 90$ is the smallest angle for which we have:

$$L_{diff} = L(\theta_k - \theta_{min}) - L'(\theta_k - \theta_{min}) > 0 \quad (3.9)$$

Moreover, we require that in a small neighborhood of directions around this critical point, this difference is larger, i.e.:

$$\int_{-5}^5 (L(\theta_k - \theta_{\min} + \phi) - L'(\theta_k - \theta_{\min} + \phi)) d\phi > L_{\text{diff}} \quad (3.10)$$

The second critical angle $\theta_{c,2} = \theta_k + \theta_{\max}$ is found similarly:

$$L(\theta_k + \theta_{\max}) - L'(\theta_k + \theta_{\max}) > 0 \quad (3.11)$$

$$\int_{-5}^5 (L(\theta_k + \theta_{\min} + \phi) - L'(\theta_k + \theta_{\min} + \phi)) d\phi > L_{\text{diff}} \quad (3.12)$$

where $0 < \theta_{\max} < 90$ degrees.

If critical points are found, then we estimate two new light sources at angles $\theta_{k,1}$ and $\theta_{k,2}$. Given that the directional influence of a light source is 180 degrees, and that a critical point denotes the boundary where a light source begins to contribute, initial estimates of the angles of the two light sources are given by:

$$\theta_{k,1} = \theta_{c,2} - 90 \quad (3.13)$$

$$\theta_{k,2} = \theta_{c,1} + 90 \quad (3.14)$$

The luminances associated with these two light sources are then estimated to be:

$$L_{N_l+i}^{\text{in}} = L_{\theta_{k,j}} = \frac{\cos(\theta_j)}{L(\theta_j)}, \quad \forall j \in [1, 2] \quad (3.15)$$

The estimates for both angles and luminances are then refined by applying a further optimisation using Hooke-Jeeves curve-fitting. This typically produces a small correction on the initial estimates, and comes at a low computational cost.

Finally, we replace the original light source at θ_k with these two new light sources if the error ϵ at direction θ_k , computed with:

$$\epsilon = \left(L_{\theta_k} - \sum_{j=1}^{N_l+2} L_j^{\text{in}} \cos(\theta_k - \theta_j) \right)^2 \quad (3.16)$$

is less for the pair of new lights than for the original light.

3.6.2.3 Detecting point light sources

Each time a directional light k is chosen as candidate, we check that a near point light source is not a better option. If placed infinitely far away, a point light source behaves as a directional light and its corresponding curve in luminance-azimuth space $L(\phi_i)$ corresponds to an scaled cosine (assuming a Lambertian material). However, as the point light source gets closer to the object, this curve changes to a Gaussian-like function (Figure 3.5 left). This behavior is modelled in Equation 3.18 which is inferred from the Lambertian model for a point light at distance d of an object contained in a bounding circle of radius r as illustrated by the following equations:

3. LIGHT DETECTION IN SINGLE IMAGES

$$\begin{aligned}
 \bar{d} &= \bar{r} + \bar{t} \\
 \bar{r} \cdot \bar{t} &= |\bar{t}| |\bar{r}| \cos(\alpha) \\
 \bar{r} &= [r \cos(\phi_i), r \sin(\phi_i)] \\
 \bar{d} &= [d, 0]
 \end{aligned} \tag{3.17}$$

where \bar{t} , \bar{d} and \bar{r} are vectors used to obtain trigonometric ratios and as explained in the Figure 3.5 (right). From Equation 3.17 we infer the following:

$$\begin{aligned}
 L(\phi_i) &= \frac{d \cdot \cos(\phi_i) - r}{\sqrt{r^2 + d^2 - 2d \cdot r \cdot \cos(\phi_i)}} \\
 \text{where } \phi_i &\in [\theta_k - \pi/2, \theta_k + \pi/2]
 \end{aligned} \tag{3.18}$$

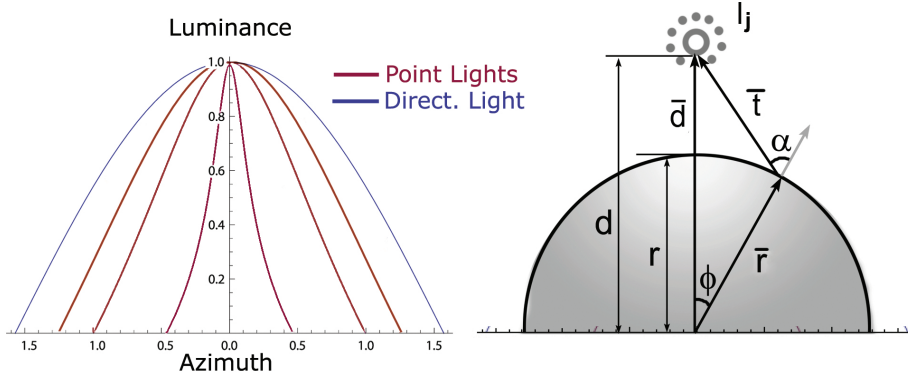


Figure 3.5: **Left:** Shading created on the contour of the lightprobe by a directional light source, plot in angle-luminance space and the corresponding point light sources at different distances (in number of times the radius of the lightprobe's bounding circle). **Right:** Diagram showing the trigonometric ratios between the angles.

Note that we started by assuming a directional light source of intensity L_k^{in} and angle θ_k which is equivalent to a point light source of intensity L_k^{in} and angle θ_k placed at an infinite distance d . Using those values as starting points, we optimize the distance d parameter in Equation 3.18 in order to minimize the error function in Equation 3.4, by means of the Hooke-Jeeves optimization method. The initial value of d is set to 10^3 times the value of the radius. Given that distance, the difference of luminance at any pixel of the image between a directional and a point light source is below the error introduced by the RGB tonemapping of the luminance.

If the directional light assumption is correct the method stops immediately, otherwise after a few iterations the d parameter is estimated. In general the accuracy of the method depends on the size in pixels of the object and the distance of the point light source and it is limited by the radius in pixels of the lightprobe: the smaller it is, the closer it has to be to the light source to be classified as a point light.

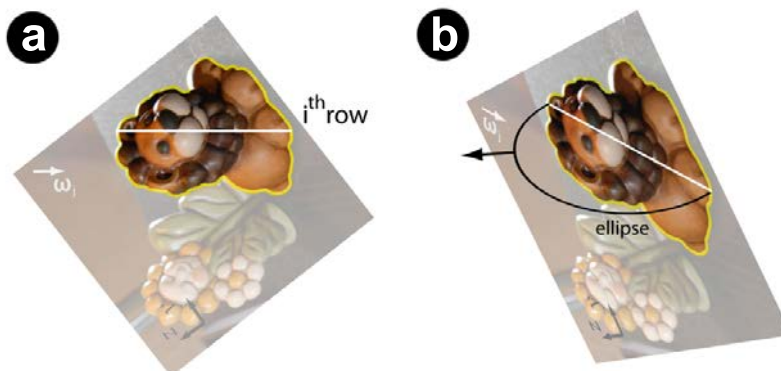


Figure 3.6: Estimating zenith angle: a) Scanning in light direction for highlight or shadow and b) Ellipsoidal geometry

3.7 Estimating Zenith Angles and Intensities

For each of the k light sources previously estimated we have to compute its elevation angle (zenith) ψ to introduce the third dimension to its direction θ_k . In order to estimate this elevation per light we cannot rely on the contour pixels alone as they are assumed to lay in the screen plane (sharing the same $\psi = 0$ angle). Thus, we need to estimate the normal at each point of the surface and we cannot rely on shape-from-shading because of the overlapping of multiple lights.

It is not possible to know a priori which combination of light sources is contributing to a certain point. Furthermore, this is complicated if two given points on the surface of the object are lit by a different and unknown number of light sources. Wang et al. (WS02) developed a technique to determine the number of lights, but they could do this thanks to accurate knowledge of 3D depth and normals. Good solutions for estimating a valid normal at points \mathbf{p}_j^{hi} or \mathbf{p}_j^{lo} in arbitrary images do not exist (ZTCS99). To overcome this limitation we have designed two methods to approximate the normals at the surface of the lightprobe.

3.7.1 Simple Normal Approximation

We propose a simple method (LMHRG10) which locally approximates the geometry of the lightprobe by an ellipse. Let us revert once more to our global convexity assumption and fit an ellipse along the scanline: one of the axes is given by the intersection of such a scanline and the silhouette; the other axis will approximate the object convexity and is a user parameter. By default, both axes are equal (in fact, defining a circumference). The surface normal is subsequently assumed to be the normal of the ellipse at the point under consideration (see Figure 3.6).

However, this shape simplification (ellipse) is prone to accumulate certain error in the zenith estimation by any deviation from global convexity in the surface of the lightprobe. In the following, we introduce the *osculating arc* (see Section 3.7.2) as an alternative for better normal approximation, in order to reduce the surface estimation error. In Section 3.8 we show and discuss our results with both techniques.

3. LIGHT DETECTION IN SINGLE IMAGES

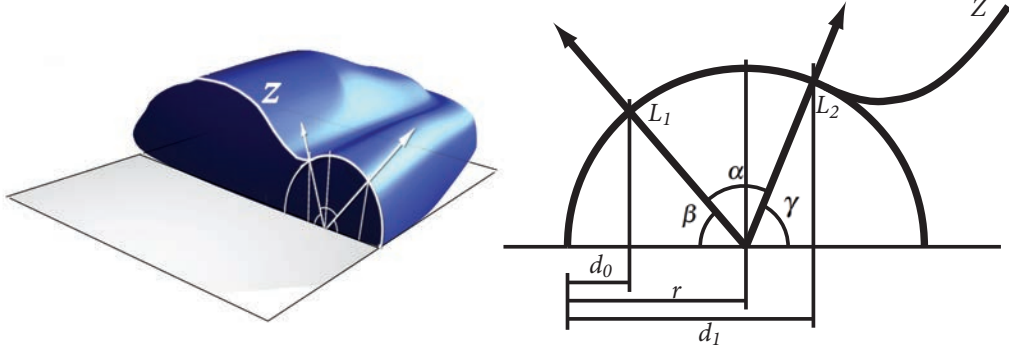


Figure 3.7: Approximation of normals at the given row of the unknown surface \mathbf{Z} by fitting a circle which has the closest gradient of luminance to the original image between the contour and the second change of curvature (the end of a convex section)

3.7.2 Normal Approximation by Osculating Arc

To compute the zenith angle we need to approximate the surface normal at the points of interest (maxima p^{hi} and minima p^{lo} in the variation of the shading $\nabla S(p_i)$). To find a valid solution for such an under-constrained problem, we assume convexity near the silhouette, and fit osculating arcs along the scanline of the figure in the image plane (see Figure 3.7). Each arc (with unknown radius for now) intersects the silhouette at the start of the scanline and fits the curvature of the surface at two points: the first point c_1 represents the first maximum in the luminance gradient for the given scanline (which we assume has its normal in the direction of the light) and it has a luminance value of L_1 . The second is the point c_2 where the luminance gradient changes and has luminance L_2 . The projection of those two points onto the image plane define the distances d_0 and d_1 respectively.

In order to determine the radius r of the semicircle we assume that L , the luminance ratio L_2/L_1 between points c_1 and c_2 , is due only to a variation of shading (Lambertian) and thus is directly related to the cosine of the angle between the corresponding normals. Given this assumption and the known values L_1 , L_2 , d_0 and d_1 from the points c_1 and c_2 we obtain the following constraints:

$$\begin{aligned}
 \alpha + \beta + \gamma &= \pi, 0 < \alpha < \frac{\pi}{2}, 0 < \beta < \frac{\pi}{2} \\
 \cos(\alpha) &= L \\
 r - d_0 &= r \cos(\beta) \\
 d_1 - r &= r \cos(\gamma)
 \end{aligned} \tag{3.19}$$

where α , β and γ are angles used to establish a trigonometric relation between the radius r and the known values L , d_0 and d_1 . From this relation it can be shown that the radius r is then given by:

$$r = \frac{d_0 + d_1 - \sqrt{2} \sqrt{d_0 \cdot d_1 \cdot (L + 1)}}{L - 1} \tag{3.20}$$

In the case of backlighting we use the point at the silhouette and the minima as fitting points c_1 and c_2 respectively.

3.7.3 Zenith estimation

Once we have approximate the local geometry of the lightprobe, we analyze the luminance of the shading $S(p_i)$ at the pixels $\{p_i\}$ enclosed by the contour in the original image. The idea is to find the maxima and minima in the gradient of the luminance; points which provide direct information of the light source's zenith elevation (their surface normals are respectively parallel and perpendicular to the direction of the light). Specifically, for each light source k we march from the silhouette to the interior of the object, following the direction given by θ_k (see Figure 3.6). In the presence of multiple light sources this directional derivative of the luminance is the main indicator of the shading due to a particular light aligned to its direction.

There are two cases of luminance variations in the interior.

Case 1: If the directional derivative $\nabla S(p_i)$ is positive at the silhouette, the light is directed towards the camera from the image ($\psi_k \geq 0$). In this case, the luminances continue to increase as we march along the direction of the light to reach the first local maximum. We denote these maxima as p^{hi} . At these points, the surface normal points in the direction of the light; i.e., $\psi_k - \psi(p^{hi}) = 0$, where $\psi(p^{hi})$ is the theta ψ angle of the surface normal at point p^{hi} . We ignore all the pixels thereafter because the geometry might be self-occluding.

Case 2: At the silhouette, if the directional derivative is negative, this is an indication of back-lighting ($\psi_k < 0$). The luminances will successively decrease as we march along the light direction to reach a singularity. These points are the first self-shadow points p^{lo} and they are marked by either a change of sign in the gradient of the directional derivative $\nabla S(p_i)$ or a minimum value of its luminance L . In this case, the surface normal is perpendicular to the light direction; i.e., $\psi_k - \psi(p^{lo}) = \pi/2$, where $\psi(p^{lo})$ is the theta ψ angle of the surface normal at point p^{lo} . If we detect a change of sign, this will be produced when the contribution to the luminance value at that point by a second light is greater than the contribution of L . Intuitively, this scanline is assuring that up to this point, no minimum was found. Thus the corresponding zenith value is taken into account only if the value is above the average value of ψ for the remaining scanlines.

We can start marching along the light direction from the brightest silhouette point that corresponds to the light. However, in order to minimize the influence of albedo variations, we scan the light direction from multiple silhouette points. One way to realize this scheme is to rotate the image such that the light angle θ_k is aligned with the y-axis and the light on the left, see Figure 3.6. Then, we simply scan each raster line i , starting from the silhouette boundary on the left and moving into the interior. We detect the set of points p^{lo} or p^{hi} , their corresponding the zenith angles ψ_{ki} and their luminances $L(p_i)$. Thus, for the light k , the resultant zenith angle is the weighted sum:

$$\psi_k = \frac{\sum_i \psi_{ki}}{\sum_i L(p_i)} \quad (3.21)$$

Finally, the intensity of each light source, previously estimated by our method along with the azimuth value, is updated by the value of the zenith angle ψ_k as shown in Equation 3.22.

$$L_k^{in} = L_k^{in} \cdot (1.0 + \cos(\psi_k)) \quad (3.22)$$

3. LIGHT DETECTION IN SINGLE IMAGES

3.7.4 Grouping lights and ambient illumination

To avoid overestimating the number of lights, and in order to come up with the simplest possible solution that explains the shading in the image, we perform pairwise comparisons between all the detected candidates; for each pair of light directions on a plane, we collapse them into one direction if the inner angle is less than 15 degrees (an empirical value that works sufficiently well for our purposes). The zenith angles are averaged and their intensities are re-computed by using the Equation 3.22 and the new zenith value.

Once all the light sources have been detected, we add a final term to take into account ambient illumination. Its light contribution is assumed to be constant for all pixels and we simply approximate its intensity by analyzing pixels in the shadow regions (note that we already have detected shadow edges when looking for minima in the zenith estimation, from which shadow regions can trivially be obtained). This ambient intensity estimate is also relative to the previously detected lights.

We average the set of samples along these boundaries. We cannot rely on the regions contained by them as they cannot be assumed to be fully covered in shadows (e.g.: a extruding bump in the middle of a shadowed area can be brightly lit while its surroundings are not).

3.8 Results

We test the accuracy of our method with real (controlled) light configurations. providing a visual validation of our method by using the lights detected in an image for automatic insertion and relighting of synthetic objects. Finally, we show a novel technique of image compositing based on our light detection methods.

Particularly we compare two methods: K-means approach for azimuth estimation with simple normal approximation in zenith computation (henceforth called *K-means method* (LMHRG10)) and the combination of azimuth obtained from critical points and osculating arc normals for zenith estimation (*Light Source-Fitting method*).

3.8.1 Error Analysis

We have tested our algorithms on several images with controlled (known) light configurations to measure the errors in our light detection method. The images include varied configurations (see Figure 3.8): Apple 1, Apple 2 and Apple 3 show a relatively simple geometry under very different lighting schemes (with one or two light sources, plus ambient light). The Guitar and Quilt images show much more complex scenes lit by three and two light sources respectively. The light directions returned by our algorithm show errors usually below 20 degrees for the more restrictive azimuth angle ϕ , which is below the 30-35 degrees limit set by our psychophysical findings (see Chapter 2).

Even for the zenith angle θ , only the second light in the Quilt scene returns a larger error because of the bouncing of that light off the surface on the left. Table 3.1 shows all the data for the input images shown in Figure 3.8. For each light source present in the scene, we show the real measured locations of the light sources, the results output by of our two methods and the corresponding absolute error. The number of directions was acquired automatically. The light probe used in the first three images was the apple; for the other two, we used the head of the guitar player and the Scottish quilt.



Figure 3.8: Input images for the error analysis of Table 3.1. From left to right: Apple1, Apple2 and Apple3, Guitar and Quilt.

The errors shown in Table 3.1 suggest that for globally convex geometries and standard light combinations (up to three main lights, evenly distributed in 3D space) the differences between both methods are small or slightly better in average for the *Light Source-Fitting method*. As we showed in Section 3.6.2 if we have accuracy in mind, the limitations of the *K-means method* (tendency to group light sources into single lights) might tip the scales in the favor of the *Light Source-Fitting method*.

We have further tested the accuracy of the *Light Source-Fitting method* with different geometries and materials. Actual 3D information is used for rendering and validation purposes but not for light detection. In the top row of Figure 3.9 and table 3.2, we can observe how increasing complexity of the geometry of the object has little effect in the accuracy of the method. Similarly, multiple reflectance variations in the object's material are analyzed in the bottom row of Figure 3.9. The rightmost column shows how a failure case (three lights were detected as four by our method) still yields perceptually equivalent results.

The *Light Source-Fitting method* was additionally tested on photographs, captured under known illuminations (measured with a mirror sphere), using multiple objects with very different BRDFs as lightprobes. Our first test consisted of seven objects illuminated by a single light source from five different positions (see Figure 3.10). The setup used to capture the images is shown in Figure 3.11 along with our light detection results. Although certain variations between objects are observable

3. LIGHT DETECTION IN SINGLE IMAGES

		Light 1		Light 2		Light 3	
		θ	ψ	θ	ψ	θ	ψ
Apple1	R	-15.00	40.00	165.00	-40.00	-	-
	E1	1.28	7.36	5.78	2.86	-	-
	E2	20.71	4.69	2.75	24.03	-	-
Apple2	R	90.00	-70.00	-	-	-	-
	E1	5.83	0.98	-	-	-	-
	E2	4.54	4.3	-	-	-	-
Apple3	R	180.00	0.00	0.00	0.00	-	-
	E1	7.63	4.89	2.57	0.00	-	-
	E2	12.50	14.48	0.00	11.31	-	-
Guitar	R	180.00	10.00	30.00	-45.00	260.00	45.00
	E1	2.80	15.96	8.43	14.72	21.49	17.03
	E2	5.71	14.66	4.36	4.19	12.29	3.16
Quilt	R	10.00	-35.00	120.00	-10.00	-	-
	E1	4.96	18.41	9.17	20.73	177.20	10.06
	E2	14.70	16.79	42.25	14.74	-	-

Table 3.1: Real measured light directions (R), error committed by the *Light Source-Fitting method* (E1) and error produced by the *K-means method*(E2) for the zenith ψ and azimuth θ angles for a set of input images (included as additional material). Note how our *Light Source-Fitting method* has detected a third light source in order to explain the light bouncing from the left in the *Quilt* image.

(e.g.:both the shoe and vase figures violate our global convexity assumption) on average the error is below 20 degrees. In several cases a secondary light was detected due to light bouncing from the ground.

Our second test analyzed how the presence of a second light source affects the accuracy of the *Light Source-Fitting method* (see Figure 3.12). Four objects from the previous test were selected and illuminated by a two-light configuration. Although the results show a lower accuracy, the error committed is still below the human perceptual threshold (LMSSG10).

In the third test, we have analyzed the *Light Source-Fitting method* with six spatial combinations of three light sources for three different lightprobes (see Figure 3.13). We can observe for multiple light configurations some light sources are considered as part of the ambient illumination. This tends to happen for low energy light sources which have small influence on the gradient of the shading of the lightprobe.

		Light 1		Light 2		Light 3	
		θ	ψ	θ	ψ	θ	ψ
Shape	O1	2.33	03.81	10.41	5.36	4.90	6.99
	O2	10.41	9.83	3.50	14.15	1.64	7.67
	O3	25.80	11.54	17.70	8.33	0.00	5.82
	O4	3.50	2.04	15.05	12.07	7.43	18.49
Material	O5	23.57	0.30	8.13	15.62	14.83	9.53
	O6	–	–	–	–	–	–
	O7	1.17	27.77	20.92	11.96	16.60	14.50
	O8	26.17	24.27	31.00	4.64	26.14	4.39

Table 3.2: Error measures obtained by our *Light Source-Fitting method* of the same scene with different shape (O1-O4) and reflectance properties (O5-O8) of the lightprobe, corresponding to the images depicted in figure 3.9 in top and bottom rows respectively. In O6 the algorithm yields four lights. Nevertheless, the result is visually equivalent. See Figure 3.9, rightmost column.

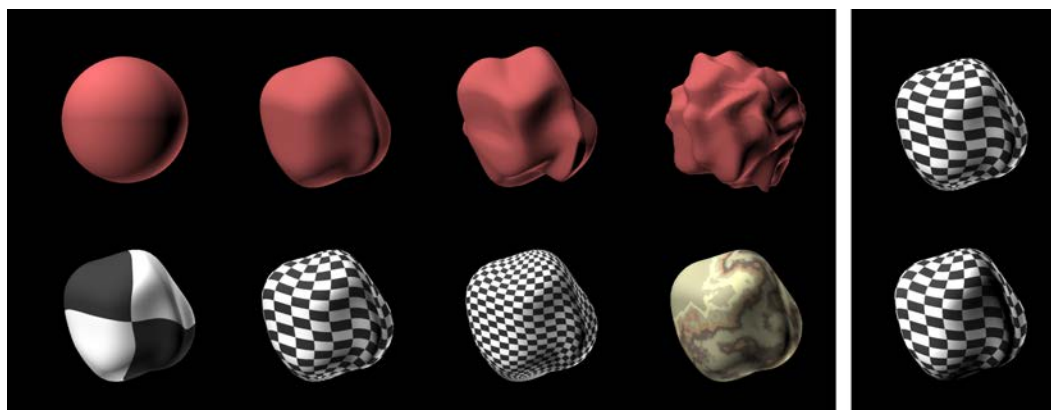


Figure 3.9: Left image: Objects used as lightprobes by our method to generate the results in table 3.2. Light sources are located at $(90^\circ, 0^\circ)$, $(180^\circ, 45^\circ)$ and $(315^\circ, -45^\circ)$. **Top row:** From left to right, the objects have an increasing complexity of the surface (O1-O4 objects in the table). Generated with a combination of fractal and gaussian noise at different spatial scales. **Bottom row:** The objects are textured with different spatial frequencies (O5-O8 objects in the table). The rightmost column shows the object O6 (top) and the result of rendering a new version with the light detected by our method (bottom).

3. LIGHT DETECTION IN SINGLE IMAGES



Figure 3.10: Input images for our single light test. The top row shows different illuminations for one object.

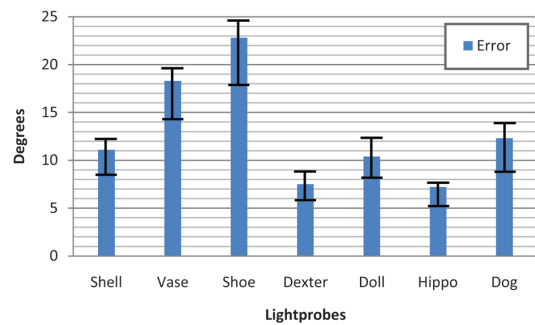


Figure 3.11: Left: Setup used to photograph the light probes used in our tests. Right: Average error committed by our *Light Source-Fitting method* in the single light test.

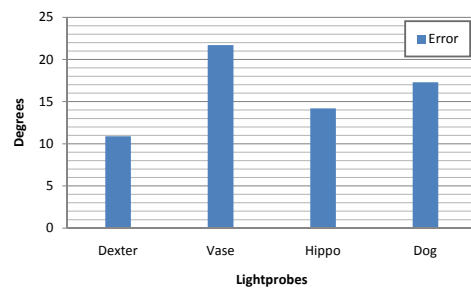


Figure 3.12: Two light sources test. Left: input images. Right: Average error committed by our *Light Source-Fitting method*.

























	Object 1			Object 2			Object 3			Lightprobe
Config 1										
Error	14.92°	42.33°	24.73°	15.16°	Amb	15.02°	12.95°	28.43°	11.19°	
Config 2										
Error	17.12°	Amb	8.70°	29.99°	Amb	10.73°	7.18°	Amb	16.08°	
Config 3										
Error	22.24°	Amb	4.17°	12.85°	Amb	20.83°	18.55°	44.36°	5.32°	
Config 4										
Error	7.34°	17.24°	22.03°	14.73°	32.18°	10.98°	Amb	20.41°	17.27°	
Config 5										
Error	13.67°	Amb	15.91°	22.53°	17.24°	18.11°	6.8°	8.4°	11.9°	
Config 6										
Error	Amb	10.32°	14.37°	Amb	13.51°	8.36°	Amb	11.43°	16.51°	

Figure 3.13: Set of images captured with varying configurations of three light sources and the error committed by our *Light Source-Fitting method*. The value *Amb* indicates that the algorithm considered the contribution of the corresponding source as ambient illumination. Lights sources detected corresponding to light bouncing from the ground have been discarded.

Finally, we have tested the effect of an area light source on the *Light Source-Fitting method* (See Figure 3.14). It can be seen how the algorithm approximates the solution with two light sources at varying distances from one another depending on the size of the area source.

We can select multiple objects (or convex parts of objects) in a single image as light probes, as shown in figure 3.15. In these cases, the analysis returns coherent results for global light sources. Local sources may spatially vary in the image. In both cases (Apollo’s arm and the body of Vulcan’s assistant), the main light shows almost the same direction. This figure also shows the applicability of our method to 2D paintings. In this case we can observe how the artist intended (and was able) to have both characters under consistent lighting.

3. LIGHT DETECTION IN SINGLE IMAGES

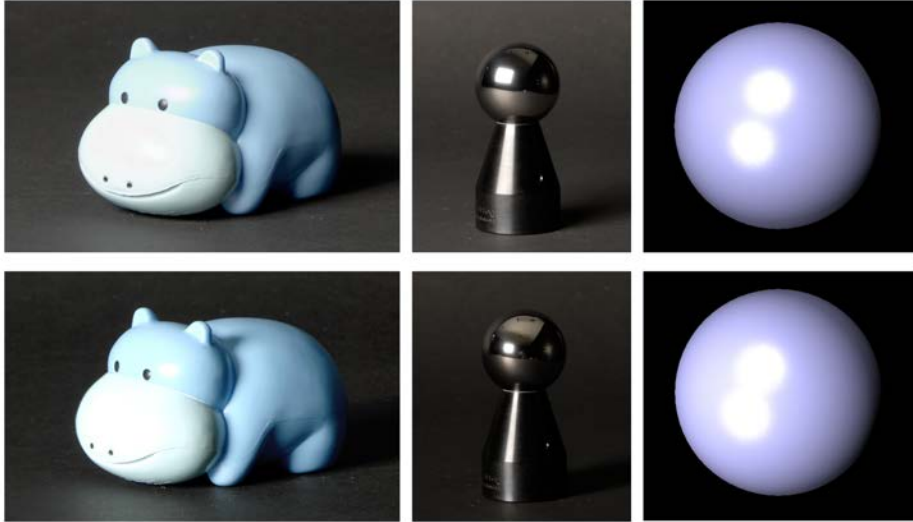


Figure 3.14: Top row: Large area light. Bottom row: small area light. From left to right: input image, ground truth and virtual probe (OpenGL) with the light sources detected by our *Light Source-Fitting method*.



Figure 3.15: Left: Input image, *La fragua de Vulcano* by Diego de Velazquez (1630), oil on canvas. Middle: Areas used as light probes showing the computed horizontal (red) and vertical (green) gradients. Note how the user can select parts of an object, avoiding, for instance, the black albedo of the hair on the head or the shadows in the right leg. Right: A synthetic OpenGL render with the light source detected for the arm. The light direction was estimated with the *K-Method* as $(\phi, \theta) = (139.97, 33.04)$ for the arm and $(\phi, \theta) = (136.17, 39.10)$ for the body.

3.8.2 Visual Validation

We further tested our algorithms on uncontrolled images, depicting scenes with unknown illuminations and varying degrees of diffuse-directional lighting ratios. Given that we obviously cannot provide error measures in those cases, we provide visual validation of the results by rendering a synthetic object with the lighting scheme returned by our algorithm. Figure 3.16, left, shows the original image and an untextured version of the 3D objects to be rendered. The image on the right shows the results of illuminating the 3D objects with the output returned by our *K-means method*. The chosen light probe was one of the mushrooms. Figure 3.17 shows additional examples of uncontrolled input images with synthetic objects rendered into them; the head of the doll and the whole human figure were used as light probes by our *K-means method*, respectively. Note how our methods are robust enough even if the light probe is composed of multiple objects with very different BRDFs (such as the skin, glasses and hair in the doll image). The shadows cast onto the original images were generated by shadow mapping and synthetic planes manually set at approximately the right locations when placing the synthetic objects.



Figure 3.16: Rendering synthetic objects into the images. Left, top: original input image (light probe highlighted). Left, bottom: 3D models lit according to the output of our light detection *K-means method*. Right: final result with the 3D models textured and inserted into the image.



Figure 3.17: Additional examples of synthetic objects rendered into images using the results of our *K-means method*. Left: synthetic teapot. Right: synthetic cone.

3. LIGHT DETECTION IN SINGLE IMAGES

3.8.3 Image Compositing

Finally, we apply the illumination information obtained by our method to a well-known problem in computer graphics: compositing two images with different illumination environments into a single image with coherent illumination. In image compositing, color and intensity can be adjusted with relatively straightforward techniques including Poisson-based approaches (JSTS06) and color transfer algorithms (RAGS01). Although such algorithms go a long way toward matching the color schemes, they do not match the illumination direction on objects. Thus, if strong localized lighting exists in either the source or the target images, the result will look out of place.

For compositing we use the following approach: first, we analyze the background image with our light detection method. Second, we extract a coarse 3D shape of the image to be inserted. Third, we relight this shape using the lights' directions and intensities from the first step and past it in the final image.



Figure 3.18: Two new images relit with our relighting method. Inset: original image

We first need to produce a plausible depth map of every object in the scene to be relit. This can be achieved in a number of ways (IMT99; OCDD01), but we chose to follow a simple method based on the interpretation of luminance as depth values (see Chapter 4, Section 4.2.1). This approach has been successfully used before in the context of image-based material editing (KRFB06) or light transport editing (shown in Chapter 7). A bilateral filter (TM98) is applied to the result to remove high-frequency details. The obtained depth values $D(x, y)$ represent the camera-facing half of the object. For image relighting, we additionally need an approximation of the far side of the object, which aids in the casting of shadows and the computation of diffuse interreflections. As our input does not allow us to infer this geometry with any decent accuracy, we reconstruct this backfacing geometry simply by mirror-copying the front half of the recovered geometry, in accordance with our global convexity assumption. Again, the obvious inaccuracies of this approach are masked by the limitations of our visual perception, as our final results show. To prepare our recovered geometry for relighting, we finally compute a normalized surface normal $n(x, y)$ for each pixel belonging to the object from the gradient field $\nabla z(x, y)$.

Once the 3D shape is known, several rendering approaches are available, and there are no limitations on the complexity of the BRDF employed. For demonstration purposes, we use a combination of Lambert's and Phong's models to represent the surface reflectance (FvFH90). The new texture of the object is generated from the original image using the original hue and saturation channels and the



Figure 3.19: A more dramatic lighting change. Left, original image. Right, the altered version, resembling moonlight as it would possibly be shot by a cinematographer



Figure 3.20: Demonstration of our compositing method. The crumpled papers are chosen as light probe by our *Light Source-Fitting method*. From left to right: Original image. Image of a flute to be inserted. Final composition after light detection and relighting.

high-frequency component of the original luminance channel (extracted by means of a bilateral filter (KRFB06)). Figures 3.18 and 3.19 show examples of the aforementioned relighting technique, which was used in combination with light detection to obtain the composition of the flute in Figure 3.20. As input for the relighting phase and because of the white balance/albedo ambiguity in the lightprobe, the user has to set a base luminance level and a color per light source. The directions and relative intensities are provided by our method. In our experiments we found that this kind of input is feasible for an unskilled user if the tuning is done interactively once the object is inserted with all the lights set as white by default.

Figures 3.21 and 3.22 show an application to image compositing of the *K-means method* and the *Light Source-Fitting method* respectively: in both cases we use an object from the target image as lightprobe, and relight the composited object with the estimated illumination. Note that Figure 3.21 shows a particularly difficult example, given the spatially varying albedo of the Scottish quilt in used to detect lighting directions. Nevertheless, the final composited result is visually plausible, and a significant improvement over naïve insertion of the toy into the scene by simple manipulation of brightness levels.

3. LIGHT DETECTION IN SINGLE IMAGES

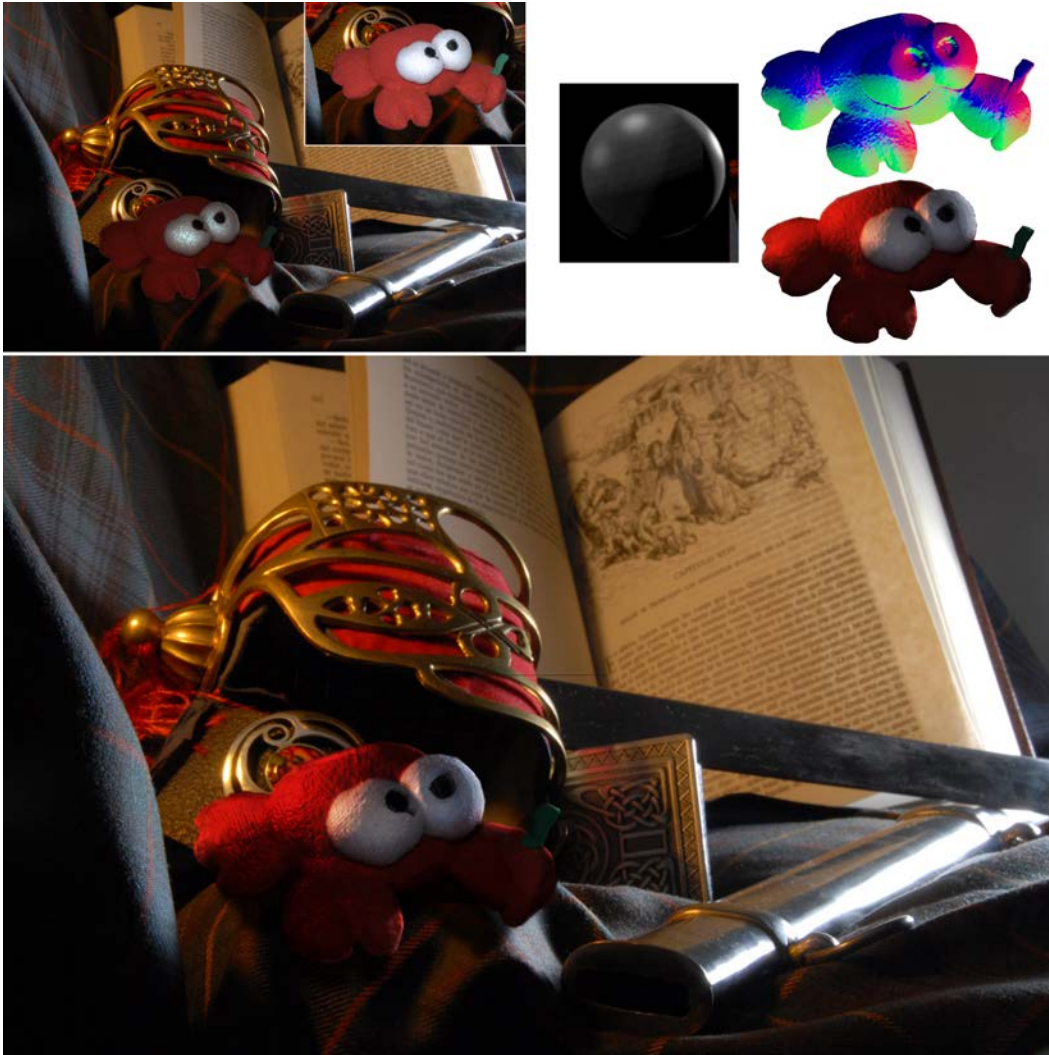


Figure 3.21: Top left: Original background image with a fluffy toy super-imposed manually tonemapped. The inset image shows the original toy. As the lighting on the toy is not corrected, the result looks out-of-place. Top right: A light probe rendered with light sources derived from the detected directions (*Light Source-Fitting method*), a normal map recovered from the toy image, and the resulting crab relit with the recovered illumination. Bottom: Final image with the toy coherently integrated in the image.



Figure 3.22: A result of compositing images through relighting by using the information from our *Light Source-Fitting method*.

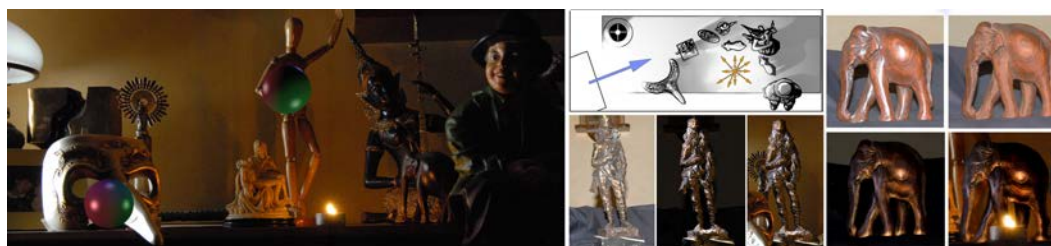


Figure 3.23: The composited objects in the figure 3.22: the soldier and one of the elephants. Left: Original background image, with spheres showing the light directions detected for the two light probes (the Venetian mask and the wooden mannequin). Middle, top: representation of the depths assigned to the objects and the two main light sources detected. Middle, bottom: the soldier as originally photographed, relit by our algorithm and extracted from the composition in the final image. Right: the same sequence of images for the elephant.

3. LIGHT DETECTION IN SINGLE IMAGES

3.9 Discussion and Future Work

This chapter introduces two novel light detection algorithms for single images that only require the silhouette of any object in the image as additional user input. Both of our methods yield a result in less than 4 seconds using a 512x512 version of the original image. Although they work on lower resolution images, higher-resolution images have a smaller effect on the accuracy of the technique. It may seem that the average error of our methods is too high in comparison with previous works in the field; however, compared with those works, we are not limited to detecting just one light source, and no knowledge of the actual 3D geometry is required. Moreover, our psychophysical studies (see Chapter 2) seem to confirm that our results are below a threshold where illumination inconsistencies tend to go unnoticed by human vision.

We have shown good results both with controlled lighting environments (where the light positions were measured and thus numerical data could be compared) and uncontrolled settings (with free images downloaded from the internet and with objects rendered with the results of our algorithm). Both methods provide good (and similar) results for most of the cases, although the *Light Source-Fitting method* seems to be more robust, presenting good results for a wider range of light configurations.

Furthermore, we have introduced a novel image compositing method based on our light detection methods. Our algorithms could help photographers mimic a given lighting scheme inspired by any other shot for which a reduced set of light directions (namely, the typical three-light setup made up of key, fill and rim lights) is preferable.

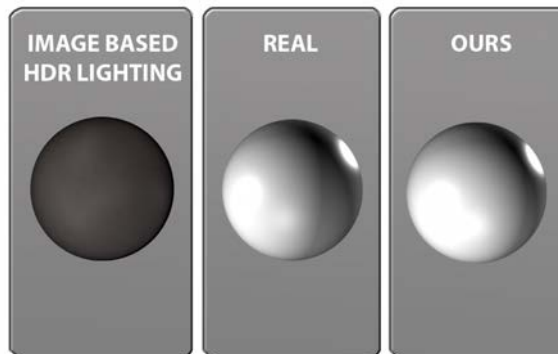


Figure 3.24: Spheres rendered with information from the Guitar image in Figure 3.8. Left: using the image as an environment map. Middle: using the real measured data. Right: using the results of our *K-means method*. Our algorithm provides a much better solution if the light sources are not present in the original image.

It could be argued that because humans are not particularly good at detecting light sources, simpler algorithms that approximate light sources could be employed instead. For instance, in the context of rendering synthetic objects into existing images, one of the most popular recent approaches is to build an environment map from the image. While this approach would provide reasonable results in certain cases (as shown in (KRFB06)), it would fail if the main light sources were actually outside the image. One such example would be the Guitar image in Figure 3.8. If we were to render an object into the image, it would appear unrealistically dark. Figure 3.24 shows a sphere rendered with the actual measured lights for that scene compared with the results from rendering with an environment map and using the lights detected by our *K-means method*.

Several existing applications could benefit from this system, specifically those based on combining pictures from an existing stack to create novel images. These kinds of applications are gaining

popularity because of, among other factors, the existence of huge databases and their accessibility through the internet. Some examples include Photo Clip Art (LHE⁺07), Interactive Digital Photomontage (ADA⁺04) and Photo Tourism (SSS06).

We assume global convexity for the chosen de facto light probes in the images. Although this assumption is true for most objects, the algorithm will return wrong values if a concave object is chosen instead. Our algorithms will also fail in the presence of purely reflective or transparent (refractive) objects chosen as light probes, which break our assumption about shading. In these cases, an approach similar to (NN04) may be more suitable, although previous knowledge about the geometry of the objects in the image would be needed. In Chapter 9 we analyze the application of our method to non lambertian surfaces, specifically to material exhibiting subsurface scattering properties like human skin, milk or marble.

Additionally, the novel compositing method introduced in a previous section has three aspects that need further research. First the recovered 3D shape is obtained by means of a simple shape derived from the shading approach, which might produce wrong and unexpected results with certain light configurations. Given the plausible results we obtained with such a simple method, we intend to test more sophisticated 3D shape recovery algorithms (SM99), (DFS08). Second, regarding the recovered texture of the object to be relit, our approach is valid for images in which the original hue and saturation values are available for most pixels. This assumption works in our examples where shadows are not harsh or cover a small portion of the image (frontal flashlight) or when high dynamic range information is available (hue and saturation values are captured even for pixels in low luminance areas). Hence, for a broader range of scenarios, we plan to use a novel intrinsic images decomposition algorithm (described in Chapter 5) in order to obtain a more accurate separation between the reflectance and the shading of the object before the relighting process. Finally, we intend to validate our composition results by means of additional psychophysical studies.

3. LIGHT DETECTION IN SINGLE IMAGES

References

- [ADA⁺04] Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen, *Interactive digital photomontage*, ACM Transactions on Graphics **23** (2004), no. 3, 294–302. 57
- [BB04] Christos-Savvas Bouganis and Mike Brookes, *Multiple light source detection*, IEEE Trans. Pattern Anal. Mach. Intell. **26** (2004), no. 4, 509–514. 35, 38
- [BH85] M.J. Brooks and B.K.P. Horn, *Shape and source from shading*, Proc. Int. Joint Conf. Artificial Intell., 1985, pp. 932–936. 30
- [DFS08] Jean-Denis Durou, Maurizio Falcone, and Manuela Sagona, *Numerical methods for shape-from-shading: A new survey with benchmarks*, Computer Vision and Image Understanding **109** (2008), no. 1, 22–43. 57
- [DLAW01] R. O. Dror, T. K. Leung, E. H. Adelson, and A. S. Willsky, *Statistics of real-world illumination*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (Kauai, Hawaii), 2001. 31
- [DWA04] R O Dror, A S Willsky, and E H Adelson, *Statistical characterization of real-world illumination*, Journal of Vision **4** (2004), 821–837. 31
- [FvFH90] James Foley, Andries van Dam, Steven Feiner, and John Hughes, *Computer graphics, principles and practice*, 2nd ed. ed., Addison-Wesley, 1990. 52
- [GHH01] S. Gibson, T. Howard, and R. Hubbard, *Flexible image-based photometric reconstruction using virtual light sources*, Computer Graphics Forum **19** (2001), no. 3, C203–C214. 30
- [HA93] D.R. Hougen and N. Ahuja, *Estimation of the light source distribution and its use in integrated shape recovery from stereo shading*, ICCV, 1993, pp. 29–34. 30
- [HJ61] Robert Hooke and T. A. Jeeves, *Direct search solution of numerical and statistical problems*, J. ACM **8** (1961), no. 2, 212–229. 36
- [Hor86] B.K.P. Horn, *Robot vision*, McGraw-Hill, 1986. 30, 32, 33
- [IMT99] T. Igarashi, S. Matsuoka, and H. Tanaka, *Teddy: a sketching interface for 3d freeform design*, SIGGRAPH '99: Proceedings of the 26th annual conference on Computer Graphics and Interactive Techniques (409-416, ed.), 1999. 52
- [JSTS06] Jiaya Jia, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum, *Drag-and-drop pasting*, SIGGRAPH '06: ACM SIGGRAPH 2006 Papers (New York, NY, USA), ACM, 2006, pp. 631–637. 52
- [KP03] J. J. Koenderink and S. C. Pont, *Irradiation direction from texture*, Journal of the Optical Society of America **20** (2003), no. 10, 1875–1882. 30

REFERENCES

- [KRFB06] Erum Arif Khan, Erik Reinhard, Roland Fleming, and Heinrich Bühlhoff, *Image-based material editing*, ACM Transactions on Graphics (SIGGRAPH 2006) **25** (2006), no. 3, 654–663. 52, 53, 56
- [LB01] M. S. Langer and H. H. Bühlhoff, *A prior for global convexity in local shape-from-shading*, Perception **30** (2001), 403–410. 31, 32
- [LF06] Pascal Lagger and Pascal Fua, *Using specularities to recover multiple light sources in the presence of texture*, ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition (Washington, DC, USA), IEEE Computer Society, 2006, pp. 587–590. 30
- [LHE⁺07] Jean-Francois Lalonde, Derek Hoiem, Alexei A. Efros, Carsten Rother, John Winn, and Antonio Criminisi, *Photo clip art*, ACM Transactions on Graphics (SIGGRAPH 2007) **26** (2007), no. 3, See the project webpage at <http://graphics.cs.cmu.edu/projects/photoclipart>. 57
- [LMHRG09] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Light source detection in photographs*, CEIG 2009, Sep 2009, pp. 161–168. 29
- [LMHRG10] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Compositing images through light source detection*, Computers & Graphics **In press** (2010). 29, 41, 44
- [LMSLG09] Jorge Lopez-Moreno, Francisco Sangorrn, Pedro Latorre, and Diego Gutierrez, *Measuring the accuracy of human vision*, CEIG 2009, Sep 2009, pp. 145–152. 30
- [LMSSG10] Jorge Lopez-Moreno, Veronica Sundstedt, Francisco Sangorrin, and Diego Gutierrez, *Measuring the perception of light inconsistencies*, APGV '10: Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization, ACM, 2010, pp. 25–32. 30, 46
- [MG97] Stephen R. Marschner and Donald P. Greenberg, *Inverse lighting for photography*, Fifth IST/SID Color Imaging Conference, 1997, pp. 262–265. 30
- [MPK07] A. A. Mury, S. C. Pont, and J. J. Koenderink, *Light field constancy within natural scenes*, Applied Optics **46** (2007), no. 29, 7308–7316. 31
- [MZBK06] Satya P. Mallick, Todd Zickler, Peter N. Belhumeur, and David J. Kriegman, *Specularity removal in images and videos: A pde approach*, In Proc. of ECCV, 2006, pp. 550–563. 32
- [NE01] Peter Nillius and Jan-Olof Eklundh, *Automatic estimation of the projected light source direction*, CVPR, 2001, pp. I:1076–1083. 30, 33, 35
- [NN04] Ko Nishino and Shree K. Nayar, *Eyes for relighting*, ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH) **23** (2004), no. 3, 704–711. 30, 57
- [OCDD01] B. M. Oh, M. Chen, J. Dorsey, and F. Durand, *Image-based modeling and photo editing*, SIGGRAPH '01: Proceedings of the 28th annual conference on Computer Graphics and Interactive Techniques (433-442, ed.), 2001. 52
- [OCS05] Yuri Ostrovsky, Patrick Cavanagh, and Pawan Sinha, *Perceiving illumination inconsistencies in scenes*, Perception **34** (2005), 1301–1314. 31
- [PCR10] Tania Pouli, Douglas Cunningham, and Erik Reinhard, *Eurographics 2010 star image statistics and their applications in computer graphics*, 2010. 31

-
- [Pen82] A.P. Pentland, *Finding the illuminant direction*, Journal of the Optical Society of America A **72** (1982), no. 4, 448–455. 30
- [PSG01] M.W. Powell, S. Sarkar, and D. Goldgof, *A simple strategy for calibrating the geometry of light sources*, IEEE Transactions on Pattern Analysis and Machine Intelligence **23** (2001), no. 9, 1022–1027. 30
- [RAGS01] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, *Color transfer between images*, IEEE Computer Graphics and Applications **21** (2001), no. 5, 34–41. 52
- [SM99] Dimitrios Samaras and Dimitris Metaxas, *Coupled lighting direction and shape estimation from single images*, ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2 (Washington, DC, USA), IEEE Computer Society, 1999, p. 868. 57
- [SSI99] Imari Sato, Yoichi Sato, and Katsushi Ikeuchi, *Illumination distribution from brightness in shadows: Adaptive estimation of illumination distribution with unknown reflectance properties in shadow regions*, ICCV (2), 1999, pp. 875–882. 30
- [SSS06] Noah Snavely, Steven M. Seitz, and Richard Szeliski, *Photo tourism: Exploring photo collections in 3d*, ACM Transactions on Graphics **25** (2006), no. 3, 835–846. 57
- [TM98] C Tomasi and R Manduchi, *Bilateral filtering for gray and color images*, Proceedings of the IEEE International Conference on Computer Vision, 1998, pp. 836–846. 52
- [VY94] E.V. Vega and Y.-H. Yang, *Default shape theory: with the application to the computation of the direction of the light source*, Journal of the Optical Society of America A **60** (1994), 285–299. 35
- [VZ04] M. Varma and A. Zisserman, *Estimating illumination direction from textured images*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, vol. 1, June 2004, pp. 179–186. 30
- [WS02] Y. Wang and D. Samaras, *Estimation of multiple illuminants from a single image of arbitrary known geometry*, ECCV02, vol. 3, 2002, pp. 272–288. 30, 35, 38, 41
- [XW08] S. Xu and A. M. Wallace, *Recovering surface reflectance and multiple light locations and intensities from image data*, Pattern Recogn. Lett. **29** (2008), no. 11, 1639–1647. 30
- [YY91] Y. Yang and A. Yuille, *Sources from shading*, Computer Vision and Pattern Recognition, 1991, pp. 534–539. 35
- [ZK02] Wei Zhou and Chandra Kambhampettu, *Estimation of illuminant direction and intensity of multiple light sources*, ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part IV (London, UK), Springer-Verlag, 2002, pp. 206–220. 30
- [ZTCS99] R Zhang, P Tsai, J Cryer, and M Shah, *Shape from shading: A survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence **28** (1999), no. 8, 690–706. 30, 41
- [ZY01] Yufei Zhang and Yee-Hong Yang, *Multiple illuminant direction detection with application to image synthesis*, IEEE Trans. Pattern Anal. Mach. Intell. **23** (2001), no. 8, 915–920. 30, 35, 38

REFERENCES

Chapter 4

3D Shape Reconstruction

In this chapter we explore the algorithms available for 3D shape reconstruction, focusing on the most adequate methods for our purposes. Part of Subsection 4.2.1 has been published in JCR listed journals (LMJH⁺11), (LMHRG10), (MELM⁺11), (GLMF⁺08) and conferences (LMJH⁺10), (LMCG08). Our research in light detection and parametric shape from shading (Subsection 4.2.2) is planned to be submitted this year.

4.1 Introduction

The three-dimensional shape reconstruction from a single image is an ill posed problem for which an optimal solution does not exist. In a search for information, several visual cues have been used as input like contours (Joh02), shadows (Sch97), reflections (LBRB08), ambient occlusion (SL97; PJS09) or shading (LR89), (HB89). Due to its ubiquity and strong correlation with geometry, the latter is one of the most studied visual cues. This technique, *Shape From Shading* (SFS), generally tries to get closer to the optimal solution by relying on prior knowledge like the position of the light sources or making assumptions on the material properties (e.g.: assuming lambertian shading). Some approaches even take into consideration perceptual aspects (KRFB06), (GWM⁺08).

Among the algorithms incorporating illumination data, we would like to highlight the work of Wei et al. (WH97), which takes as input the light direction from a single light source. Although it is possible to estimate the properties of the light sources as part of the reconstruction process (SM99), this kind of approaches are computationally expensive and incorporating additional unknowns increases the probability of obtaining a suboptimal solution.

For an exhaustive comparison on SFS methods before 2000, we recommend the reading of the survey made by Zhang et al. (ZTCS99). The most recent approaches, like partial differential equations (PDE), are analyzed and compared in a survey by Durou et al. (DFS08).

4.2 Selecting a Shape From Shading Method

In this thesis we have focused on two approaches, the first, based in the perception of depth and contours (KRFB06), (Joh02), providing fast, plausible results at the cost of accuracy and the second, based in light detection and parametric surface optimization (WH97).

As a general rule, we will consider a coordinate system where the X and Y-axis correspond to the screen’s width and height respectively and the Z-axis is perpendicular to the screen plane and represents the depth of the scene.

4.2.1 Perception-based SFS

Our goal is to devise a depth recovery algorithm which is simple, while allowing the user a certain amount of control. We do not aim to recover accurate depth from a single image, only good enough to retain the main salient features that will still make the edited version plausible. For this, we take a two-layer approach, following the observation that objects can be seen as made up of large features (low frequency) defining its overall shape, plus small features (high frequency) for the details. We thus begin by decomposing the input image into a base layer $B(x, y)$ for the overall shape and a detail layer $D(x, y)$ (BPD06) by means of a bilateral filter (TM98). Note that unlike Bae et al. (BPD06), we do not work in the logarithmic domain. Instead, we compute luminance values on the basis of the RGB pixel input. Assuming that sRGB primaries and white point are used, per-pixel luminance values are computed as $L(x, y) = 0.212R(x, y) + 0.715G(x, y) + 0.072B(x, y)$ (I.T90).

When processing an image with a bilateral filter, the choice of the spatial and intensity kernels σ_1 and σ_2 is crucial in order to produce a robust separation of high frequency details and low frequency features. We have found that good results are achieved by following the recommendations of Bae et al. (BPD06) and setting $\sigma_1 = \min(\text{width}, \text{height})$ and σ_2 to the 90th percentile of the gradient norm of the image, $\sigma_2 = p_{90}(\|\nabla I\|)$. Figure 4.1 shows an example of this separation. Intuitively, the detail layer D can be seen as a bump map for the base layer B. We decouple control over the influence of each layer and allow the user to set their influence in the final image as follows:

$$Z(x, y) = F_b \cdot B(x, y) + F_d \cdot D(x, y) \quad (4.1)$$

where $Z(x, y)$ is the final recovered depth, and F_b and $F_d \in [0, 1]$ are user-defined weighting factors controlling the presence of large and small features in the final image respectively. Obviously, by interpreting input luminance values as depth we are potentially incurring in large errors. However, we leverage the fact that humans tend to perceive objects as globally convex, following the dark-is-deep paradigm (LB01). This assumption made by the human visual system produces a depth hallucination which remains plausible while we do not attempt to change the viewpoint of the images. Shearing in the recovered depth will go unnoticed according to the bas-relief ambiguity (BKY99).

We take into account these two facts by using non-linear spline functions to reshape the base layer $B(x, y)$ and enforce its convexity (KRFB06), producing an inflation analogous to those achievable by techniques like *Lumo* (Joh02) which interpolates the values of the contour. The advantage of our approach over *Lumo* is that salient features of the shape are captured without the need of additional input as shown in Figure 4.2.

The depth map Z serves as input to our algorithms. From this map it is straightforward to derive a normal map if required by any method.



Figure 4.1: Left: Input image (synthetic render of a still life (LMJH⁺11)). Middle: Base layer. Right: Detail layer.

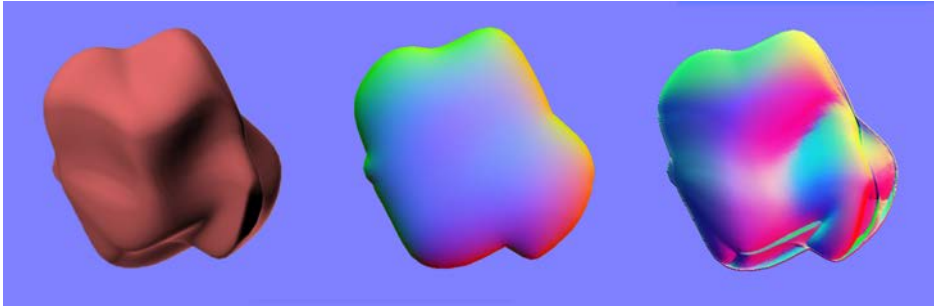


Figure 4.2: Left: Input image (synthetic render of an abstract shape). Middle: Normal map generated with *Lumo* by using the contour of the shape. Right: Normal map generated with our method.

4.2.2 Parametric SFS based on light detection

In most cases, our applications would benefit from a more accurate solution. Our latest research relies on our light detection algorithms in order to apply much more complex shape recovery algorithm: parametric shape from shading based on optimization and radial basis functions (RBFs) (WH97). When published in 1997, this method showed promising results but we found no traces of new implementations or source code available. The authors present a new method of shape from shading by using radial basis functions (Gaussian) to parameterize the object depth. The radial basis functions are deformed by adjusting their centers, widths, and weights such that the intensity errors are minimized. Figure 4.3 shows a sequence of snapshots from a real time capture of our 3D reconstruction tool. We can observe the deformations of the gaussian functions as their parameters are modified by our optimization method in order to create a surface from the input image.

The initial centers and widths are arranged hierarchically (multilevel approach) to speed up convergence and to stabilize the solution. Although a smoothness constraint is used, it can be eventually dropped out without causing instabilities in the solution (see Figure 4.4).

However, the main limitation of this work is that it requires prior knowledge on the illumination properties (direction, energy). Our implementation of their work takes advantage of our light detection algorithm and performs shape approximations at interactive time without code optimization. Instead of a neural-network based implementation, as previous work from the authors seems to suggest (WH96), we opted for our own stochastic gradient optimization. The partial derivatives of the error function are needed to perform the stochastic gradient descent optimization. Unfortunately the equations were not included in the original paper (WH96), so we have added them at the end of this chapter (Annex A). We even think that a parallelized version, GPU or CPU-based, could achieve real time performance.

4. 3D SHAPE RECONSTRUCTION

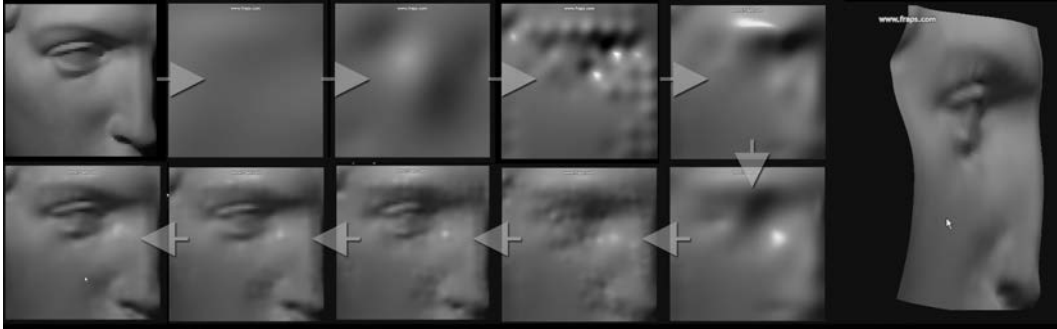


Figure 4.3: Input image and evolution of the parametric surface generated by our method. Real time capture from video footage.

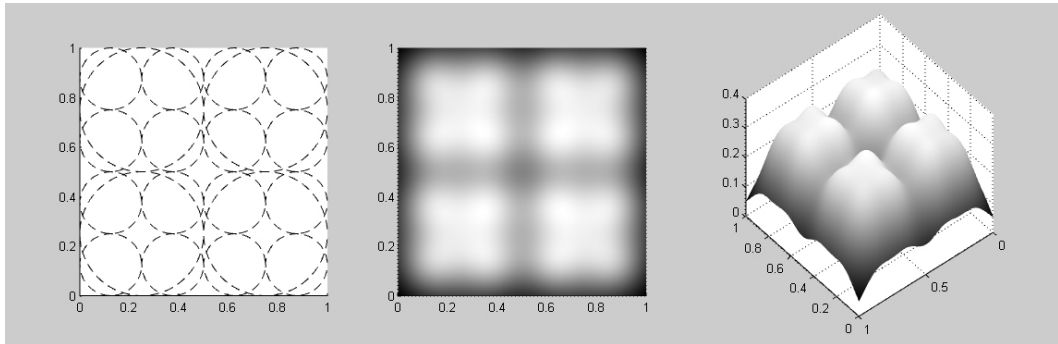


Figure 4.4: Left: View of the radial basis functions (gaussians) representing the 3D surface. Each gaussian is represented by an ellipsoid delimiting its area of influence: 3 times its standard deviation in both axes. Middle: depth map generated by the basis functions. Right: isometric view of the corresponding 3D shape.

One of the key advantages of this method is that it can use additional input in order to improve the results. The system allows for depth, normal, contour and equality constraints. This optimization scheme, by design, is specially suitable for sparse constraint input (such as user strokes) and the system will incorporate this data into the solver, refining the surface in a smooth and continuous fashion.

Depth information might be added from an external source (such as a depth camera like Microsoft KinectTM) or by user strokes. Figure 4.5 shows an example of this kind of input. The constraints are incorporated into the system in a smooth fashion, with their effect in the final result being adjustable by the user. Likewise, the surface normals can be defined as accurately as desired. For instance we can specify if certain region of the image is facing east or west, helping the system into disambiguation convex and concave interpretations of the same result.

Finally, we have designed contour constraints as a combination of depth, equality and normal constraints. We assume that a user-defined contour implies points with equal depth and normals derived from the contour shape and lying on the screen plane.

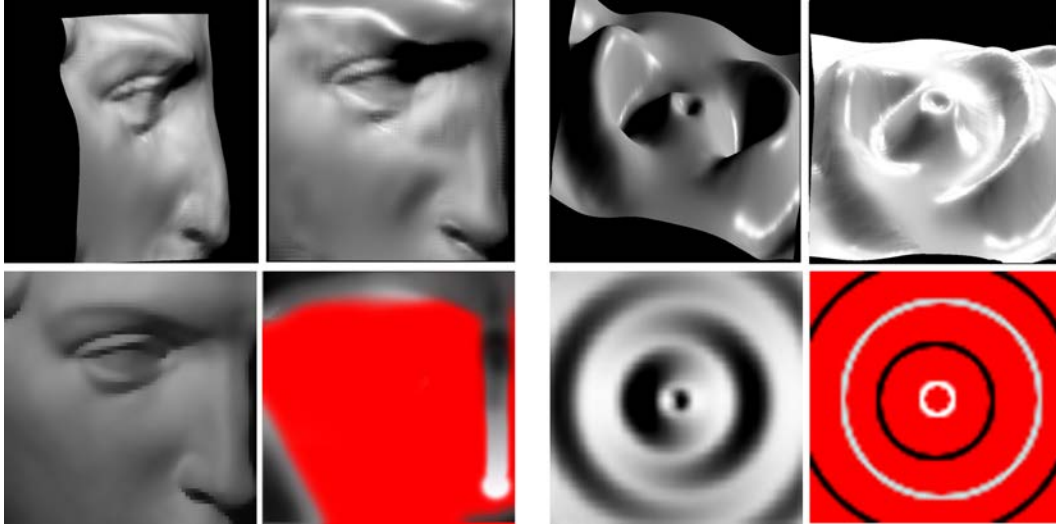


Figure 4.5: Top Row: Left: new 3D views of the surface generated from the constraints and the David image. Right: Novel views of surfaces generated from the sombrero image. The rightmost image was generated including the equality constraints defined in the bottom row. Bottom row: Left: Input image (low resolution fragment of David statue) and user strokes defining depth constraints (dark-is-deep). Right: Input image (low resolution hat) and user strokes denoting similar-height pixels (equality constraint).

4.3 Conclusions and Future Work

In this chapter we have explored several existing shape from shading techniques for depth estimation, implementing novel variations based in the perception of depth which have been extensively used in most of the algorithms shown in this thesis.

By relying in our light source estimation methods it is possible to obtain even more accurate depth estimations. To this end, we have extended the work of Wei et al. (WH97), creating a good depth estimation basis for future research in our single image editing pipeline.

As future work, we plan on incorporating sparse depth maps, like the low resolution depth images acquired by KinectTM(see Figure 4.6, to our RBF shape from shading method in order to capture surfaces with high accuracy.

Likewise, we think that multilevel decomposition of the input images (from low to high frequency) is a very suitable input for the parametric SFS algorithm, which would also benefit from the addition of perceptual cues (like *dark-is-deep* or global convexity).

Additionally, as our light detection approach provides up to four light sources, we plan to extend this work to accept multiple light directions as input, constraining the problem and thus, reducing the range of less optimal solutions for the system.

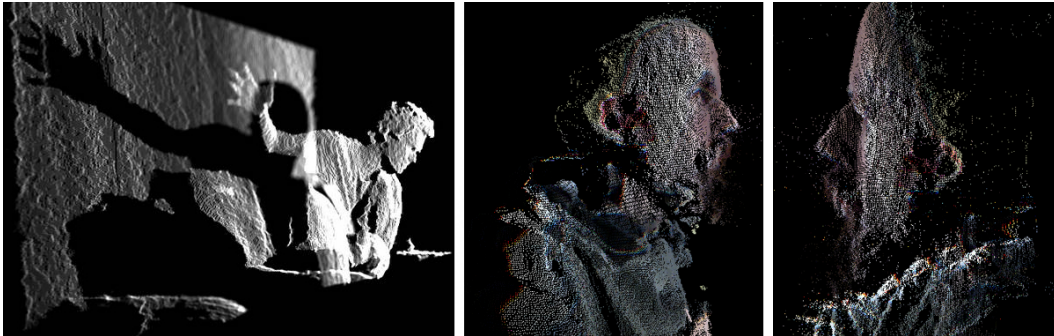


Figure 4.6: Left: Example of depthmap obtained from the Kinect camera. Middle, Right: Additional examples, shown as point clouds, Note the sparsity of the data, very suitable for RBF interpolation. Images from Kyle McDonald. Used by permission (CC-BY-SA-NC).

4.4 Annex A: Derivatives of the Error Function

In this annex we provide the derivatives of the error function which are required in order to implement the stochastic gradient descent optimization method proposed by Wei et al. (WH97).

As show in Figure 4.4, for any given pixel with coordinates (x, y) in the image, its corresponding depth $Z(x, y)$ is given by the sum of gaussian signals at that point:

$$Z(x, y) = \sum_{k=1}^N W_k \phi(x, t_k, s_k) = W_0 \quad (4.2)$$

where W_k is the weighting factor of the function. And each gaussian is defined by the following equation:

$$\phi(x, t_k, s_k) = e^{-\frac{(x-t_x)^2}{\sigma_x^2} - \frac{(y-t_y)^2}{\sigma_y^2}} \quad (4.3)$$

where x is the x,y coordinates of the point to compute, t_k is the center of the gaussian (t_x, t_y) and s_k is the size (standard deviation) in both axes (s_x, s_y) .

In order to find a solution, the stochastic gradient descent method aims to minimize an error function through consecutive iterations in which the result is compared with the goal image and feed back into the solver as input. The error function (difference between the original image and the result generated by the sum of gaussian functions) is composed by E_i , error in luminance values, and E_s , or smoothness error. At each step in the iterative gradient descent, the parameters of the gaussian functions are varied in an amount corresponding to the derivative of the error obtained in the previous iteration. For additional details, please refer to the paper by Wei and Hirzinger (WH97). In the following we provide the derivatives of the error function w.r.t. the five parameters of the radial basis functions, which were not published in the original work. For the sake of clarity we display them in horizontal format, with G representing the function $\phi(x, t_k, s_k)$, R the reflectance map (as defined in the original paper) and Int the input luminance of the pixel.

$$\frac{\partial E_s}{\partial w} = 8w e \left(-\frac{2(X-t_x)^2}{\sigma_x^2} - \frac{2(Y-t_y)^2}{\sigma_y^2} \right) \left(\frac{16S_2^2(X-t_x)^2(Y-t_y)^2}{\sigma_x^4\sigma_y^4} + \frac{S_1^2(\sigma_x^2 - 2(X-t_x)^2)}{\sigma_x^8} + \frac{S_2^2(\sigma_y^2 - 2(Y-t_y)^2)}{\sigma_y^8} \right) \quad (4.4)$$

$$\frac{\partial E_s}{\partial t_x} = \frac{16w^2(X-t_x)e \left(-\frac{2(X-t_x)^2}{\sigma_x^2} - \frac{2(Y-t_y)^2}{\sigma_y^2} \right) (\sigma_y^4(2(X-t_x)^2 - \sigma_x^2)(S_1^2\sigma_x^4(2(X-t_x)^2 - 3\sigma_x^2) + 8S_2^2\sigma_x^4(Y-t_y)^2) + S_3^2\sigma_x^8(\sigma_y^2 - 2(t_y^2 + Y(Y-2t_Y))))^2}{\sigma_x^{10}\sigma_y^8} \quad (4.5)$$

$$\begin{aligned} \frac{\partial E_s}{\partial t_y} &= 16w^2 e \left(-\frac{2(X-t_x)^2}{\sigma_x^2} - \frac{2(Y-t_y)^2}{\sigma_y^2} \right) \frac{\sigma_y^4(Y-t_y)(8S_2^2\sigma_x^4(X-t_x)^2(2(Y-t_y)^2 - \sigma_y^2) + S_1^2\sigma_y^4(\sigma_x^2 - 2(X-t_x)^2)^2)}{\sigma_x^8\sigma_y^{10}} \\ &+ 16w^2 e \left(-\frac{2(X-t_x)^2}{\sigma_x^2} - \frac{2(Y-t_y)^2}{\sigma_y^2} \right) \frac{(S_3^2(\sigma_y^4(Y-3t_y) - 4\sigma_y^2(Y-2t_y)(t_y^2 + Y(Y-2t_Y))) + 4(Y-t_y)(t_y^2 + Y(Y-2t_Y))^2)}{\sigma_x^{10}\sigma_y^{10}} \end{aligned} \quad (4.6)$$

$$\begin{aligned} \frac{\partial E_s}{\partial s_x} &= 16w^2 e \left(-\frac{2(X-t_x)^2}{\sigma_x^2} - \frac{2(Y-t_y)^2}{\sigma_y^2} \right) \cdot \\ &\left(\frac{\sigma_x^4(X-t_x)^2(16S_2^2\sigma_y^4(Y-t_y)^2(X-t_x)^2 - \sigma_x^2)(X-t_x)^2(\sigma_y^2 - 2(t_y^2 - 2Yt_Y + Y^2))^2}{\sigma_y^8} + S_1^2(-12\sigma_x^2(X-t_x)^4 + 7\sigma_x^4(X-t_x)^2 + 4(X-t_x)^6 - \sigma_x^6) \right) \frac{\sigma_x^{11}}{\sigma_x^{11}} \end{aligned} \quad (4.7)$$

$$\begin{aligned} \frac{\partial E_s}{\partial s_y} &= 16w^2 e \left(-\frac{2(X-t_x)^2}{\sigma_x^2} - \frac{2(Y-t_y)^2}{\sigma_y^2} \right) \frac{S_2^2(2t_y^2 + 2Y(Y-2t_Y) - \sigma_y^2)(\sigma_y^2(2Yt_y - 5t_y^2 + Y(8t_Y - 5Y)) + 2(Y-t_y)^2(t_y^2 + Y(Y-2t_Y)) + \sigma_y^4)}{\sigma_y^{11}} \\ &+ 16w^2 e \left(-\frac{2(X-t_x)^2}{\sigma_x^2} - \frac{2(Y-t_y)^2}{\sigma_y^2} \right) \frac{\sigma_y^4(Y-t_y)^2(16S_2^2\sigma_x^4(X-t_x)^2(Y-t_y)^2 - \sigma_y^2) + S_1^2\sigma_y^4(\sigma_x^2 - 2(X-t_x)^2)^2}{\sigma_x^8\sigma_y^{11}} \end{aligned} \quad (4.8)$$

4. 3D SHAPE RECONSTRUCTION

(4.9)

$$\begin{aligned}
 E'_i &= -2(\text{Int} - R) \cdot R' \\
 &= 4G \left(2G l_3 w \left(\sigma_y^4 (X - t_x)^2 + \sigma_x^4 (Y - t_y)^2 \right) + l_1 \sigma_x^2 \sigma_y^4 (t_x - X) + l_2 \sigma_x^4 \sigma_y^2 (t_y - Y) \right) \left(\text{Int} - \frac{2Gw \left(\frac{l_1(X-t_x)}{\sigma_x^2} + \frac{l_2(Y-t_y)}{\sigma_y^2} \right) + l_3}{\sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1}} \right) \\
 &= \frac{\sigma_x^4 \sigma_y^4 \left(4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1 \right)^{3/2}}{4G \left(2G l_3 w \left(\frac{l_1(X-t_x)}{\sigma_x^2} + \frac{l_2(Y-t_y)}{\sigma_y^2} \right) + l_3 \right)} \\
 &= -4Gw \left(\text{Int} - \frac{2Gw \left(\frac{l_1(X-t_x)}{\sigma_x^2} + \frac{l_2(Y-t_y)}{\sigma_y^2} \right) + l_3}{\sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1}} \right) .
 \end{aligned}$$

(4.10)

$$\begin{aligned}
 \frac{\partial R}{\partial t_x} &= \frac{(2l_2 \sigma_x^2 \sigma_y^2 (X - t_x) (Y - t_y) (2G^2 w^2 + \sigma_x^2) - 4Gw \sigma_x^4 (Y - t_y)^2 (G l_1 w + l_3 (X - t_x)) + \sigma_y^4 (\sigma_x^2 - 2(X - t_x)^2) (2G l_3 w (X - t_x) - l_1 \sigma_x^2))}{\sigma_x^6 \sigma_y^4 \left(4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1 \right)^{3/2}} \\
 &= -4Gw \left(\text{Int} - \frac{2Gw \left(\frac{l_1(X-t_x)}{\sigma_x^2} + \frac{l_2(Y-t_y)}{\sigma_y^2} \right) + l_3}{\sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1}} \right) . \\
 \frac{\partial R}{\partial t_y} &= \frac{(2l_1 \sigma_x^2 (X - t_x) (Y - t_y) (2G^2 w^2 + \sigma_y^2) + l_2 (2\sigma_x^4 (Y - t_y)^2 - \sigma_y^4 (4G^2 w^2 (X - t_x)^2 + \sigma_x^4))) + 2G l_3 w (Y - t_y) (\sigma_y^4 (\sigma_x^2 - 2(Y - t_y)^2) - 2\sigma_y^4 (X - t_x)^2)}{\sigma_x^4 \sigma_y^6 \left(4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1 \right)^{3/2}} \\
 &= -4Gw \left(\text{Int} - \frac{2Gw \left(\frac{l_1(X-t_x)}{\sigma_x^2} + \frac{l_2(Y-t_y)}{\sigma_y^2} \right) + l_3}{\sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1}} \right) .
 \end{aligned}$$

(4.11)

$$\begin{aligned}
 \frac{\partial R}{\partial x_x} &= 8Gw \sigma_y^2 (X - t_x) \left(-\text{Int} \sigma_x^2 \sigma_y \sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1} + 2Gw \left(l_1 \sigma_y^2 (X - t_x) + l_2 \sigma_x^2 (Y - t_y) \right) + l_3 \sigma_x^2 \sigma_y^2 \right) \cdot \\
 &= \frac{(l_2 \sigma_x^2 \sigma_y^2 (X - t_x) (Y - t_y) (4G^2 w^2 + \sigma_x^2) - 2Gw \sigma_x^4 (Y - t_y)^2 (2G l_1 w + l_3 (X - t_x)) - \sigma_y^4 ((X - t_x)^2 - \sigma_x^2) (2G l_3 w (X - t_x) - l_1 \sigma_x^2))}{\sigma_x \left(4G^2 w^2 \sigma_y^2 (X - t_x)^2 + \sigma_x^2 \left(4G^2 w^2 (Y - t_y)^2 + \sigma_y^2 \right) \right)^{3/2}} \\
 &= 8Gw \sigma_x^2 (Y - t_y) \left(-\text{Int} \sigma_x^2 \sigma_y \sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1} + 2Gw \left(l_1 \sigma_x^2 (X - t_x) + l_2 \sigma_y^2 (Y - t_y) \right) + l_3 \sigma_x^2 \sigma_y^2 \right) \cdot \\
 &= \frac{(\sigma_y^2 (l_1 \sigma_x^2 (X - t_x) (Y - t_y) (4G^2 w^2 + \sigma_y^2) + l_2 (\sigma_x^4 (Y - t_y)^2 - \sigma_y^4 (4G^2 w^2 (X - t_x)^2 + \sigma_x^4))) + 2G l_3 w (Y - t_y) (\sigma_x^4 (\sigma_y^2 - (Y - t_y)^2) - \sigma_y^4 (X - t_x)^2))}{\sigma_y \left(4G^2 w^2 \sigma_x^4 (X - t_x)^2 + \sigma_x^2 \left(4G^2 w^2 (Y - t_y)^2 + \sigma_y^2 \right) \right)^{3/2}}
 \end{aligned}$$

(4.12)

$$\begin{aligned}
 \frac{\partial R}{\partial x_y} &= 8Gw \sigma_x^2 (Y - t_y) \left(-\text{Int} \sigma_x^2 \sigma_y \sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1} + 2Gw \left(l_1 \sigma_x^2 (X - t_x) + l_2 \sigma_y^2 (Y - t_y) \right) + l_3 \sigma_x^2 \sigma_y^2 \right) \cdot \\
 &= \frac{(\sigma_y^2 (l_1 \sigma_x^2 (X - t_x) (Y - t_y) (4G^2 w^2 + \sigma_y^2) + l_2 (\sigma_x^4 (Y - t_y)^2 - \sigma_y^4 (4G^2 w^2 (X - t_x)^2 + \sigma_x^4))) + 2G l_3 w (Y - t_y) (\sigma_x^4 (\sigma_y^2 - (Y - t_y)^2) - \sigma_y^4 (X - t_x)^2))}{\sigma_y \left(4G^2 w^2 \sigma_x^4 (X - t_x)^2 + \sigma_x^2 \left(4G^2 w^2 (Y - t_y)^2 + \sigma_y^2 \right) \right)^{3/2}}
 \end{aligned}$$

(4.13)

$$\begin{aligned}
 \frac{\partial R}{\partial x_y} &= 8Gw \sigma_x^2 (Y - t_y) \left(-\text{Int} \sigma_x^2 \sigma_y \sqrt{4G^2 w^2 \left(\frac{(X-t_x)^2}{\sigma_x^2} + \frac{(Y-t_y)^2}{\sigma_y^2} \right) + 1} + 2Gw \left(l_1 \sigma_x^2 (X - t_x) + l_2 \sigma_y^2 (Y - t_y) \right) + l_3 \sigma_x^2 \sigma_y^2 \right) \cdot \\
 &= \frac{(\sigma_y^2 (l_1 \sigma_x^2 (X - t_x) (Y - t_y) (4G^2 w^2 + \sigma_y^2) + l_2 (\sigma_x^4 (Y - t_y)^2 - \sigma_y^4 (4G^2 w^2 (X - t_x)^2 + \sigma_x^4))) + 2G l_3 w (Y - t_y) (\sigma_x^4 (\sigma_y^2 - (Y - t_y)^2) - \sigma_y^4 (X - t_x)^2))}{\sigma_y \left(4G^2 w^2 \sigma_x^4 (X - t_x)^2 + \sigma_x^2 \left(4G^2 w^2 (Y - t_y)^2 + \sigma_y^2 \right) \right)^{3/2}}
 \end{aligned}$$

References

- [BKY99] Peter N. Belhumeur, David J. Kriegman, and Alan L. Yuille, *The bas-relief ambiguity*, Int. J. Comput. Vision **35** (1999), no. 1, 33–44. 64
- [BPD06] Soonmin Bae, Sylvain Paris, and Frédo Durand, *Two-scale tone management for photographic look*, ACM Trans. Graph. **25** (2006), no. 3, 637–645. 64
- [DFS08] Jean-Denis Durou, Maurizio Falcone, and Manuela Sagona, *Numerical methods for shape-from-shading: A new survey with benchmarks*, Computer Vision and Image Understanding **109** (2008), no. 1, 22–43. 63
- [GLMF⁺08] Diego Gutierrez, Jorge Lopez-Moreno, Jorge Fandos, Francisco Seron, Maria Sanchez, and Erik Reinhard, *Depicting procedural caustics in single images*, ACM Transactions on Graphics (Proc. of SIGGRAPH Asia) **27** (2008), no. 5, 120:1–120:9. 63
- [GWM⁺08] Mashhuda Glencross, Gregory J. Ward, Francho Melendez, Caroline Jay, Jun Liu, and Roger Hubbard, *A perceptually validated model for surface depth hallucination*, ACM Trans. Graph. **27** (2008), 59:1–59:8. 63
- [HB89] B.K.P. Horn and M.J. Brooks, *Shape from shading*, MIT Press, Cambridge, MA., 1989. 63
- [I.T90] I.T.U., *Basic parameter values for the hdtv standard for the studio and for international programme exchange*, 1990. 64
- [Joh02] Scott F. Johnston, *Lumo: illumination for cel animation*, NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering (New York, NY, USA), ACM, 2002, pp. 45–ff. 63, 64
- [KRFB06] Erum Arif Khan, Erik Reinhard, Roland Fleming, and Heinrich Bühlhoff, *Image-based material editing*, ACM Transactions on Graphics (SIGGRAPH 2006) **25** (2006), no. 3, 654–663. 63, 64
- [LB01] M S Langer and H H Bühlhoff, *A prior for global convexity in local shape-from-shading*, Perception **30** (2001), 403–410. 64

REFERENCES

- [LBRB08] J. Lellmann, J. Balzer, A. Rieder, and J. Beyerer, *Shape from specular reflection and optical flow*, no. 2. 63
- [LMCG08] Jorge Lopez-Moreno, Angel Cabanes, and Diego Gutierrez, *Image-based participating media*, CEIG 2009, Sep 2008, pp. 179–188. 63
- [LMHRG10] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Compositing images through light source detection*, Computers & Graphics **In press** (2010). 63
- [LMJH⁺10] Jorge Lopez-Moreno, Jorge Jimenez, Sunil Hadap, Erik Reinhard, Ken Anjyo, and Diego Gutierrez, *Stylized depiction of images based on depth perception*, NPAR '10: Proceedings of the 8th international symposium on Non-photorealistic animation and rendering, ACM, 2010. 63
- [LMJH⁺11] Jorge Lopez-Moreno, Jorge Jimenez, Sunil Hadap, Erik Reinhard, Ken Anjyo, and Diego Gutierrez, *Non-photorealistic, depth-based image editing*, Computers & Graphics **In press** (2011). 63, 65
- [LR89] C.H. Lee and A. Rosenfeld, *Improved methods of estimated shape from shading using the light source coordinate system*, Shape from shading (B.K.P. Horn and M.J. Brooks, eds.), MIT Press, 1989, pp. 323–569. 63
- [MELM⁺11] Adolfo Muñoz, Jose I. Echevarria, Jorge Lopez-Moreno, Francisco Serón, Mashhuda Glencross, and Diego Gutierrez, *Bssrdf estimation from single images*, Computer Graphics Forum (Proc. of EUROGRAPHICS) (2011). 63
- [PJS09] Emmanuel Prados, Nitin Jindal, and Stefano Soatto, *A Non-Local Approach to Shape From Ambient Shading*, 2nd International Conference on Scale Space and Variational Methods in Computer Vision (SSVM'09) SSVM '09: Proceedings of the Second International Conference on Scale Space and Variational Methods in Computer Vision, Springer-Verlag, 2009, pp. 696–708. 63
- [Sch97] Karsten Schlüns, *Shading based 3d shape recovery in the presence of shadows*, In: Proc. Intern. Conf. on Digital Image & Vision Computing, 1997, pp. 10–12. 63
- [SL97] A. James Stewart and Michael S. Langer, *Toward accurate recovery of shape from shading under diffuse lighting*, IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997), 1020–1025. 63
- [SM99] Dimitris Samaras and Dimitris N. Metaxas, *Coupled lighting direction and shape estimation from single images*, ICCV, 1999, pp. 868–874. 63
- [TM98] C Tomasi and R Manduchi, *Bilateral filtering for gray and color images*, Proceedings of the IEEE International Conference on Computer Vision, 1998, pp. 836–846. 64
- [WH96] Guo-Qin Wei and G. Hirzinger, *Learning shape from shading by a multilayer network*, Neural Networks, IEEE Transactions on **7** (1996), no. 4, 985–995. 65

REFERENCES

- [WH97] Guo-Qing Wei and Gerd Hirzinger, *Parametric shape-from-shading by radial basis functions*, IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997), 353–365. 63, 64, 65, 67, 68
- [ZTCS99] R Zhang, P Tsai, J Cryer, and M Shah, *Shape from shading: A survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence **28** (1999), no. 8, 690–706. 63

REFERENCES

Chapter 5

Intrinsic Images Decomposition

This chapter describes our proposal for *intrinsic images decomposition*: a novel algorithm which in opposition to previous approaches, requires no user input and can work at interactive rates (our non-optimized implementation computes an average-size image in less than a minute). In following sections we will show the potential of this method as a pre-processing step for image-based 3D extraction algorithms like our *Shape from Shading* approach (see Figure 5.2).

Most of the contents of this chapter are to be submitted next March, 2011 to the International Conference on Computer Vision, ICCV (which has a *CiteSeer* impact factor ranking in the top 5% of all Computer Science journals and conferences).

5.1 Introduction

In the last few years the field of *computational photography* (RT10) has concentrated research from such different areas as computer graphics, photography and computer vision. In this chapter, we are interested in the *computational photography* sub-field of image editing in single images. As we stated in previous chapters, several techniques such as relighting, 3D depth extraction, material edition, etc., rely on the disambiguation of texture (inherent albedo of the surface) and shading (produced by illumination and geometry). Furthermore, this kind of texture extraction might benefit many applications in computer vision based in feature and pattern recognition.

The problem of illumination and material extraction in a single image is an open challenge since Barrow and Tenenbaum (BT78) formulated the problem in 1978 with the name of *intrinsic images decomposition*. This decomposition consists of separating an image in two components (images): one representing the reflectance of the object(texture) and the other containing the shading (interaction of illumination and geometry). We can observe an example in Figure 5.1.

5. INTRINSIC IMAGES DECOMPOSITION

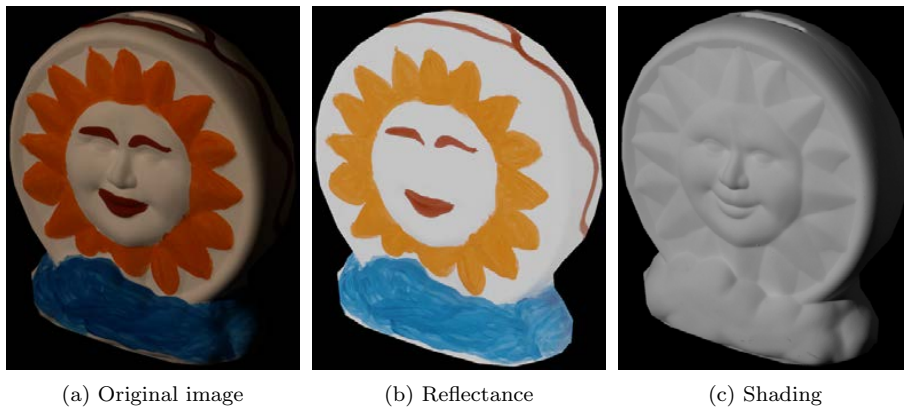


Figure 5.1: *Example of intrinsic images decomposition. Image from (BPD09).*

However, in a single image, reflectance and illumination are coupled in a very complex manner and, as multiple combinations may produce the same outcome it is impossible to extract the actual pair which originated a given image. For instance, how could we possibly know if certain blue-colored area of an image is a white material lit by a blue light source or a blue material lit by a white light? The human visual system (HVS) deals with such an ill-posed problem in two ways: First by applying previously learned knowledge regarding the object properties, and second, by eliminating the color differences produced by differences in the illumination. This property of HVS is known as *color constancy* (EHL71) and it is a very desirable feature in image processing algorithms.

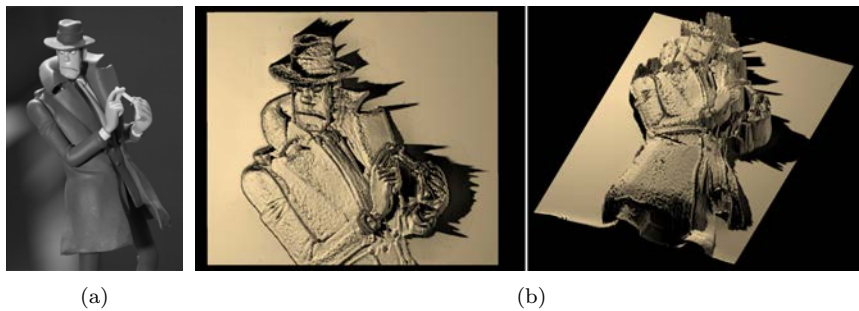


Figure 5.2: **Shape from Shading** algorithms reconstruct 3D shape (b) from the shading component (a). Some of them rely on general assumptions such as the global convexity and the dark is deep prior (LMJH⁺ 11).

5.1.1 Image Generation

As mentioned before, the idea of separating reflectance and illumination was introduced by Barrow and Tenenbaum (BT78) who denoted the problem as *intrinsic images decomposition*. The reflectance describes how an object reflects the light and is also known by the name of *albedo*. The illumination

component corresponds with the amount of incoming light at a given point (or pixel) and depends on the surface's orientation and the light's properties (orientation, energy,...). Although this component is also called *shading* it also includes, apart from shadows, effects such as indirect illumination. A simplified formulation for the problem is given by the following: being $I(x, y)$ the input image, its decomposition in intrinsic images is given by the following equation,

$$I(x, y) = S(x, y) \times R(x, y) \quad (5.1)$$

where $S(x, y)$ is the illumination image and $R(x, y)$ is the image of the reflectance (see a visual example in Figure 5.3).

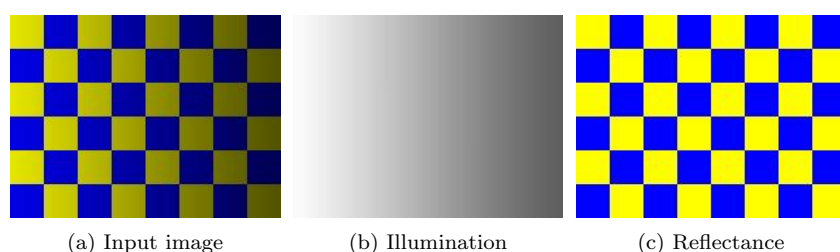


Figure 5.3: *Decomposition.* The input image (a) is composed by its intrinsic images (b) and (c)

Our goal is to obtain $S(x, y)$ and $R(x, y)$. We can observe the ill-posed nature of this problem in the previous formula: as we have double the number of incognita than equations, it is impossible to disambiguate the solution. In the last few years, due to the proliferation of image-editing techniques, several approaches to this problem have been developed from very different points of view. The following section introduces the most relevant.

5.2 Previous Work

This section focuses in the concept of *intrinsic images* and their decomposition, describing the most relevant techniques proposed to date.

5.2.1 State of the Art

Weiss (Wei01) proposes a novel method to acquire intrinsic images by using a large sequence of images of the same scene, where the reflectance remains constant and illumination varies in time. This approach was extended by Liu et al. (LWQ⁺08) to any sequence of uncontrolled images for a given scene in order to colorize black and white photographs. However, these techniques require too many input images to be useful in a wide range of cases.

5. INTRINSIC IMAGES DECOMPOSITION

Due to the lack of constraints, the single image decomposition problem cannot be solved without any prior knowledge on the scene. Horn (Hor86) circumvented this problem by relying in Retinex theory (EHL71) and assuming that, while the reflectance remains constant by segments, illumination (shading) varies smoothly. This heuristic yields the reflectance of an image by thresholding the small gradients, as they are considered to be part of the illumination. Tappen et al. (TFA05) take a step forward and use classifiers trained with the derivatives of an image to disambiguate reflectance and shading. Despite these advanced techniques, several configurations of illumination and reflectance remain very difficult to decompose and additional techniques like Markov Random Fields (MRF) and Belief Propagation (BP) are necessary in order to yield good solutions (Figure 5.4). Following this line of work, Shen et al. (STL08) propose to enrich these approaches with global texture constraints. Starting from a *Retinex* algorithm, they force the pixels with the same texture to have the same reflectance.

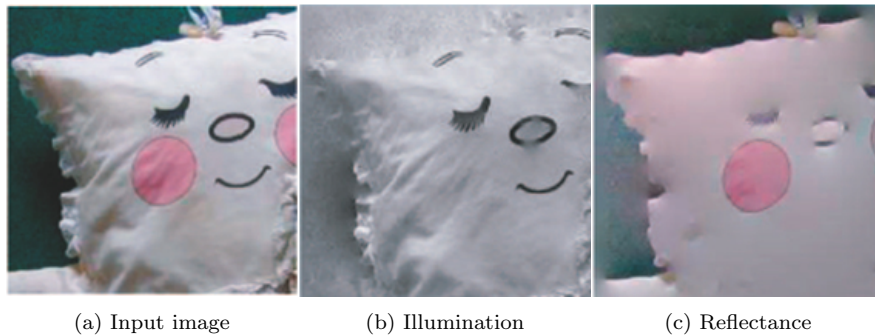


Figure 5.4: *Error in decomposition (TFA05). The variation of white over black in the eye and smile painted in the cushion, being part of the texture, can be misconstrued as shading.*

It is worth mentioning the work by Bousseau et al. (BPD09), which shows very good decompositions by assuming that reflectance shows low range variations at a local level. However, their approach relies heavily on skilled user input, trained in their tools, as the use of their brushes is far from intuitive (see Figure 5.5).

Shadow removal, being part of the intrinsic image decomposition problem, has also a vast literature on the topic, both automatic (GDFL04; FHL06) and based on user input (MTC07; WTBS07). The common idea of both approaches consists of identifying the shadowed pixels by boundary detection or image segmentation. Once the shadows are delimited, they can be eliminated by color correction or gradient filters. However, these methods are focused in detecting cast shadows, which are characterized by well differentiated boundaries and even different chromaticity. However, we aim to eliminate also smooth gradients, where the limits between light and shadows cannot be easily demarcated. We should highlight that, although the approach of Finalyson et al. (GDFL04) estimates an invariant chromaticity image which is a good guide map to identify constant reflectance areas, this image does not represent actual reflectance and shading is not a byproduct of this decomposition.

The intrinsic images decomposition is closely related to multi-level decomposition. Automatic

5.3 Reflectance and Illumination Decomposition

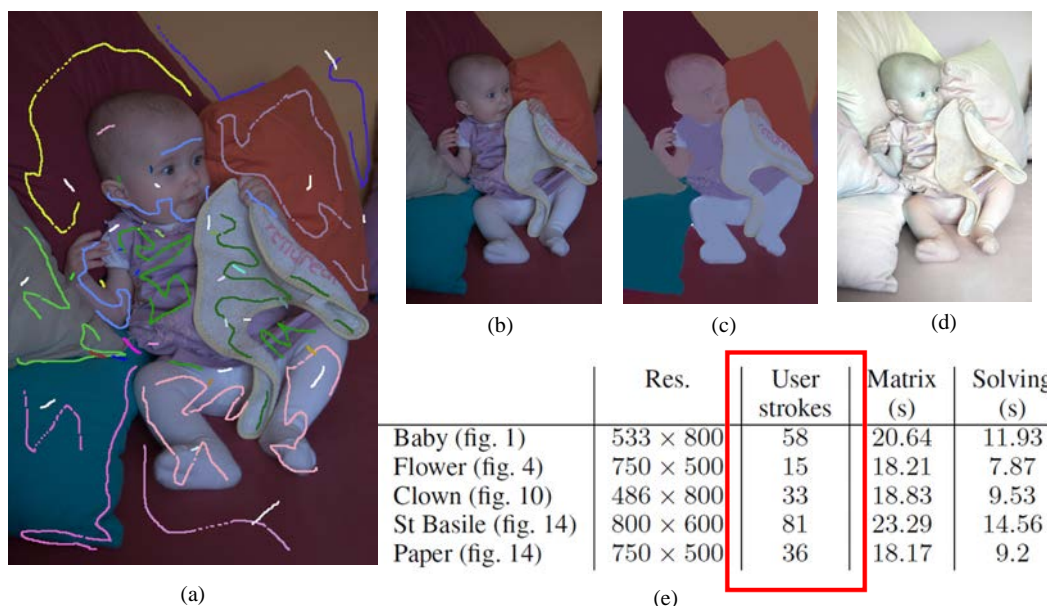


Figure 5.5: User strokes required by the tool proposed by Bousseau et al. (BPD09). In (a) we can observe the strokes needed to obtain the decomposition of the image (b) in reflectance (c) and illumination (d). Table (e) shows the total number of strokes used as input by their method for a given set of images. Image from (BPD09)

techniques which capture different levels of detail at different scales (SSD09; FAR07; FFLS08) can modify, or even isolate, the illumination component of an image in certain cases. For instance, the work by Subr et al. (SSD09), captures global features of the illumination in a scene at its smaller detail levels. Some of these algorithms such as *fast bilateral filter* (CPD07) have been used as a basis to extract the shading component for 3D reconstruction methods like *Shape from Shading* (KRFB06).

5.3 Reflectance and Illumination Decomposition

Our decomposition algorithm is an automatic process (user input is not required) which takes as input the original image (and optionally, a black and white mask defining the area to decompose). It consists mainly of two modular steps. An overview of our system is shown in Figure 5.6:

In the first step we divide the image in small fragments (pixel clusters) of constant color (albedo). For this, we use a graph-based segmentation method (FH04a) modified to work in Lab color space (more details in Section 5.4). These segments (or pixel clusters) together with the image representing the perceptual luminance will serve as input to the second step of our method.

In the second step (Section 5.5), we build a linear equation system which describes the relationships between pairs of neighboring pixel clusters in the image. As a result, we obtain luminance ratios for

5. INTRINSIC IMAGES DECOMPOSITION

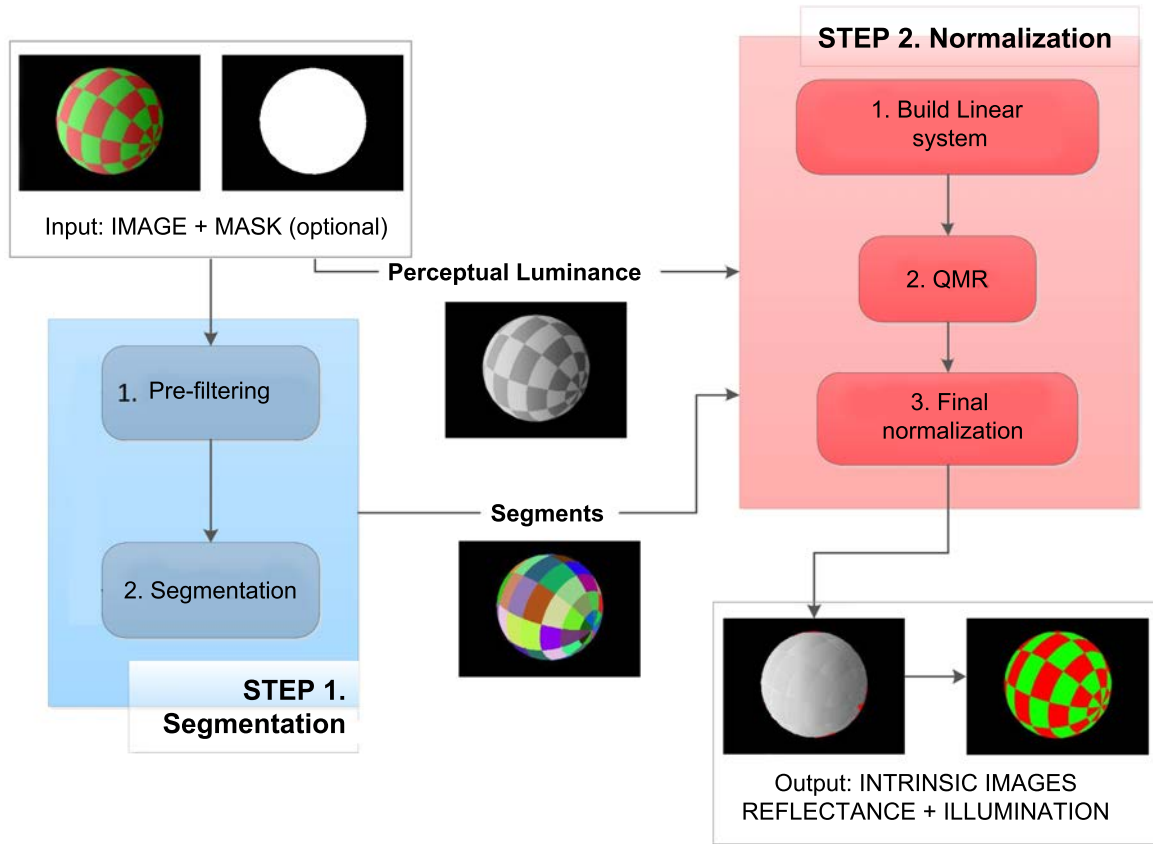


Figure 5.6: Overview of our system.

each cluster. With these ratios we compute a new image without reflectance variations, namely a *normalized* image. From the normalized image containing illumination, we compute the corresponding reflectance image. This pair of images are the intrinsic decomposition yielded by our method.

5.4 Step 1: Image Segmentation

The process of image segmentation decomposes an image in its different parts. This might look simple, but the quality of a segmentation is very subjective and depends in great manner on the goal of our application. In order to evaluate the existing techniques we devised the following initial requirements: the process has to be unsupervised (as the initial number of clusters and their characteristics are unknown), texture and/or color information has to be considered and low computation time costs are highly desired in order to embed our image decomposition technique in interactive tools like PhotoshopTM.

Bearing this in mind, we analyzed algorithms based in Markov Random Fields (MRF), clustering methods and graph-based methods.

The **Markov Random Fields** are specially suitable for local analysis and propagations of values, as they set that the conditional probability of a pixel having a certain value depends on the value of its neighboring pixels. There are several techniques and problems solved by MRF (SZS⁺08). Some of the most advanced and recent are *Belief Propagation* (YFW03; FH04b) and *Graph Cuts* (BVZ01).

However, these methods require prior knowledge about the number of regions in the image or, at least, an energy function which relates the variables and the observations in order to define similarity functions. We find this line of work interesting, but already explored, or in exploration by other researchers in the field, without any conclusive results so far.

The **clustering** techniques such as *mean shift* (CM02) are widely used in computer vision algorithms. This technique clusters data by searching similar values in a multi-feature space (without taking into account spatial relationships between pixels), assuming that the image is constant when considered at a segment level. *Mean shift* smoothes the image by grouping similar pixels into clusters characterized by their most significant color. This technique yields very good results but the election of the initial parameters depends in great manner on the input image and the type of clustering that we expect (UPH07).

The **Graph-based** techniques represent an image as a weighted undirected graph, where each pixel represents a node and the weight is given by a function modeling the relationship between the connected pixels (e.g.: the difference in luminance). The segmentation of this kind of structure is given by minimum cuts which minimize the similitude between the pixels to separate. One of the most relevant of such techniques is *normalized cuts* (SM00), which is able to capture both local and global features in an image. However, the computational cost of this algorithm is beyond our time constraints.

After considering the aforementioned methods, we selected the graph-based segmentation algorithm by Felzenszwalb and Huttenlocher (FH04a), used in multiview applications to cluster the most similar pixels of an image into *superpixels* (MK10). This method is specially suitable to cover our needs of both speed and accuracy.

The key idea behind the efficacy of this method is the use of an automatic adaptative threshold for clustering. Our implementation of the original algorithm along with the modifications needed to fit our purposes, are detailed in the following subsection.

5.4.1 Graph-based Segmentation

The algorithm starts with an undirected graph $G = (V, E)$ composed by a set of vertices $v_i \in V$, corresponding to the pixels of the image to be segmented, and a set of edges $(v_i, v_j) \in E$ connecting

5. INTRINSIC IMAGES DECOMPOSITION

pairs of neighboring pixels. Each edge has a weight $w((v_i, v_j))$ which represents the degree of similarity between the two connecting pixels. Felzenszwalb (FH04a), proposed two different graph structures: one based on a 8-neighbor grid (*GRID* graph) using the eight nearest screen-space positions, and the other based in the K nearest neighbors (*KNN* graph), mapping each pixel in a N -dimensional space of features. Both the number K of connections per pixel and the N features can be freely defined.

In the case of a *GRID* graph, the function defining the similitude between two pixels connected by an edge, is given by their differences in color. We use the Euclidean distance L_2 ,

$$w((v_i, v_j)) = \|C(v_i) - C(v_j)\| = \sqrt{\sum_{t=1}^N (C(v_i)_t - C(v_j)_t)^2} \quad (5.2)$$

where $C(v)$ is the color vector of the vertex v , being $C(v) = \{r, g, b\}$ in *RGB* space and $C(v) = \{0.5L, a, b\}$ in *Lab* space (see section 5.4.2 for additional details on color spaces).

For *KNN* graphs, each vertex is mapped in a space $\{x, y, C(x, y)\}$, where (x, y) is the location of the vertex in the image and $C(x, y)$ is the color of the corresponding point, which depends on the color model employed. In the same way as with *GRID* graphs, we use the Euclidean distance L_2 to set the weights of the edges. However, in this case, the position of the pixels in the image are also considered for the weighting factor. The advantage of *KNN* over *GRID* is twofold: first, we can select a variable number of neighbors and second, the similitude function considers both the color and the spatial position per pixel, allowing the creation of connections between separated regions of the image with similar color values, in opposition to the locality of the *GRID* approach.

In order to segment the image, the algorithm localizes the boundaries between regions with different albedo by comparing two quantities: the first based in the luminance level differences between neighboring regions and the second based in the inner luminance variation of each region. Intuitively, the luminance difference between two regions is perceptually relevant if it is greater than the luminance inner variation of, at least, one of the regions. In the segmentation process the pixels are distributed, clustering into different regions which are subsequently modified until the system converges to a steady state and the inner cohesion between the pixels of each cluster is high enough (in our experiments, no connections can have a similitude value above 50, the half of *Lab* scale).

5.4.2 The influence of color space: RGB and Lab

The original work by Felzenszwalb and Huttenlocher (FH04a) performs the image segmentation in *RGB* space. Although the results are compelling, they are not suitable for our purposes. If we consider absolute differences *RGB* space as our similitude function we are not taking into account that in a region with constant reflectance, although the pixels share a similar chromaticity, their luminance values can be altered by shading (FDB91) resulting in very different *RGB* values. Hence, we use the

Lab color model ¹ which allows us to minimize the influence of shading variations by working with chrominance. In Figure 5.7 we can observe how a surface with constant albedo regions and shading produced by a horizontal light source, is better segmented in Lab space. Notice how the erroneous clusters in the RGB version follow vertical areas of constant luminance.

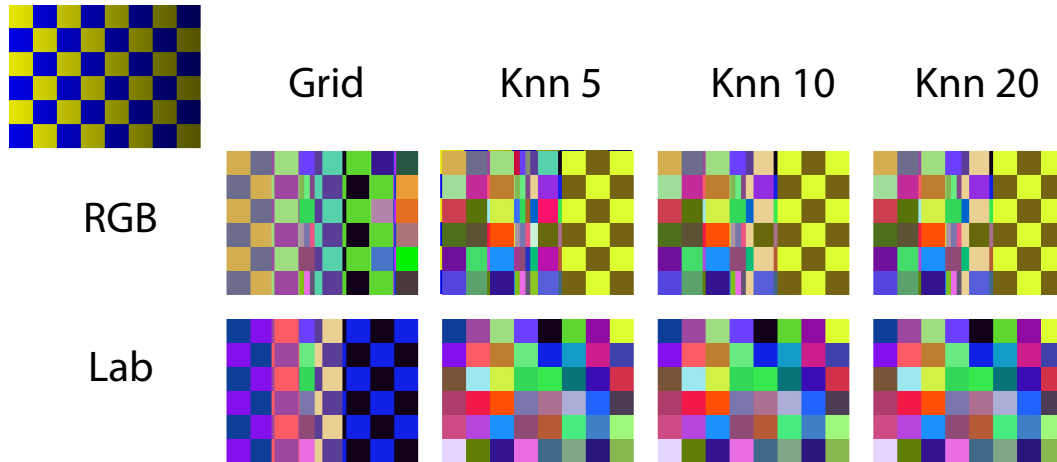


Figure 5.7: *RGB Vs Lab comparison. For any type of graph (Grid (8-neighbors) and KNN with 5, 10 and 20 neighbors are shown), the best segmentations of the upper left corner image are obtained in Lab space.*

5.4.3 Filtering and Segmentation Refinement

The results of the segmentation can be further refined (increasing the inner coherence of the clusters) by iteratively re-segmenting them after a filtering process. This filter consists of a median 2x2 filtering which reduces the color mix produced by the discretization in pixels of the region boundaries. This minimizes the misclassification of those mixed pixels. We can observe an example in Figure 5.8 of pixels wrongly segmented due to this effect.

This segmentation refinement might be avoided if the image is previously preprocessed by an edge-aware filter such as a fast bilateral filter (CPD07) in order to avoid fuzzy values at the boundaries.

¹Lab is a perceptual color model which consists of three dimensions (or channels), the first, L , represents luminance, while a y b are the color chromaticity channels.

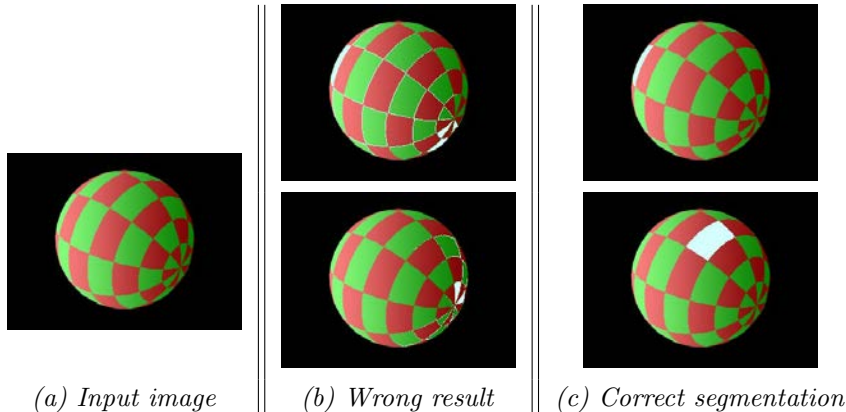


Figure 5.8: *Segmentation examples. White pixels represent an area classified as unique cluster. In (b) we can observe how boundary pixels are wrongly selected as a large cluster of pixels due to the mix of colors between adjacent regions.*

5.4.4 Segmentation Results

In order to evaluate our approach we performed a series of experiments with *GRID* and *KNN* graphs (in the latter, varying the number of neighbors from five to thirty). Likewise, we also tested the Lab and RGB color spaces in both synthetic and real images.

Our experiments showed that the Lab color model allows for a better classification of clusters by albedo than the RGB model (see Figure 5.7). Likewise, KNN graphs showed higher accuracy in the capture of similitude relationships between near pixels in the image.

Additionally, and for the sake of automatization, we analyzed the parameters of the original segmentation algorithm in order to set a default configuration and avoid user interaction. The best results were obtained for a KNN with five neighbors and a value for the parameter k (see the original paper (FH04a)) of 50. We can see some examples in Figure 5.9.

5.5 Step 2: Normalization

Our goal is to *normalize* the luminance of the input image, by removing those luminance variations produced by texture(albedo) while maintaining those originated by the geometry of the object (shading). We start by computing the initial luminance image from the original RGB values with the following equation $L(x, y) = 0.212R(x, y) + 0.715G(x, y) + 0.072B(x, y)$ (I.T90). The resulting image approximates the luminance of the original colors as perceived by human vision, not the actual physical luminance.

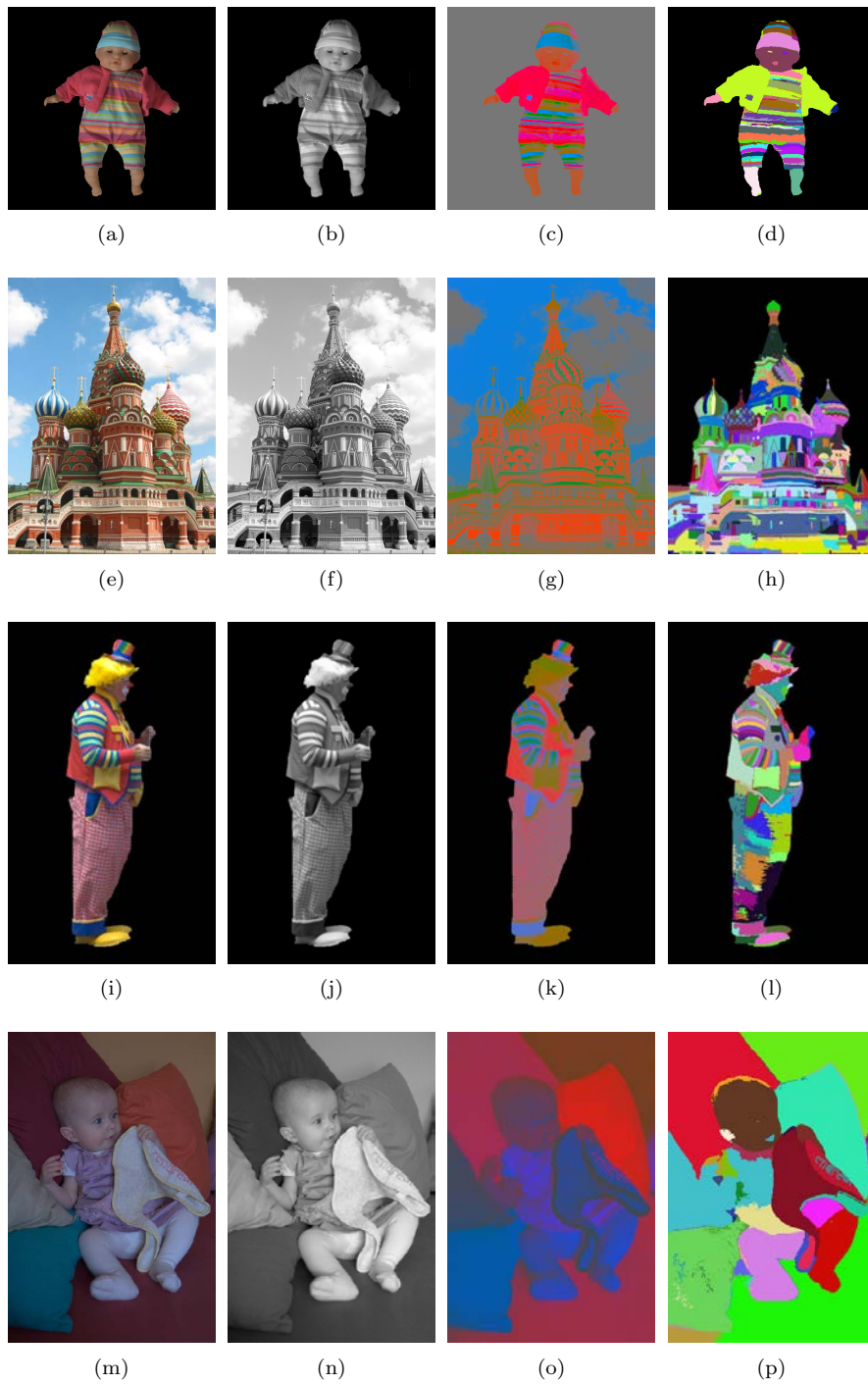


Figure 5.9: Segmentation examples. From left to right, the columns represent: the input image (a)(e)(i)(m), the perceptual luminance channel L (b)(f)(j)(n), the chromatic channels a and b (c)(g)(k)(o) and the resulting segmentation (d)(h)(l)(p).

5. INTRINSIC IMAGES DECOMPOSITION

In this context, *normalizing* a set of albedo clusters with different luminance levels, consists of finding the factors and ratios between neighboring clusters such that by multiplying them to the original luminance image we obtain a constant albedo image showing only shading variations (intuitively, we *normalize* all the albedo values to one of them).

In order to avoid the inherent ambiguity of shading and reflectance when computing these factors, we assume that the illumination (shading) produces smooth variations of luminance (Hor86) and we analyze only the pixels at the boundaries of the clusters and their local neighbors. This locality allows us to assume that the luminance variations are produced only by reflectance differences (we work with a radius of three pixels). Although it is possible to have an abrupt geometry variation like a crease, there is not a corresponding change of chromaticity.

Likewise, we have discarded the fact that under the presence of multiple light sources with different colors, a change of illumination might imply a change of chromaticity. This kind of problem has already been solved in literature (HMP⁺08) and is beyond the scope of this research.

Based in this local analysis of pixels, in the following subsection we define an equation system to find the factors or ratios which relate the luminance between each pair of neighboring clusters.

5.5.1 Linearizing the Problem

Given a set C of pixel clusters, we aim to find the factors F_c which multiply each cluster of the input luminance image L to yield the normalized luminance image L_n ,

$$L_n(x, y) = F_c L(x, y) \quad (5.3)$$

for $c \in C$ y $(x, y) \in c$.

Given that we want to equalize the luminance value of the pixels at the cluster boundaries, we have an equation per each pair of neighboring clusters expressing the following equality,

$$F_{c_i} L_m(c_i)_{c_j} - F_{c_j} L_m(c_j)_{c_i} = 0 \quad (5.4)$$

where $L_m(c_i)_{c_j}$ represents the average luminance of the pixels in the cluster c_i , which are located in the frontier with the cluster c_j . The factors F_{c_i} and F_{c_j} are the values by which each cluster c_i and c_j has to be multiplied in order to equal their luminances.

The set of equations formed by each pair of neighboring clusters defines a linear system of M equations y N unknowns, being M the total number of adjacent clusters and N the total number of clusters in the image. As we can observe, with more equations than unknowns, this is an overdetermined system and has the trivial solution: $F_{c_1} = F_{c_2} = F_{c_N} = 0$. To avoid this solution, we add a new equation which conserves the overall energy (luminance) of the image,

$$\sum_{i=1}^N F_{c_i} L_{Me}(c_i) = \sum_{i=1}^N L_{Me}(c_i) \quad (5.5)$$

where L_{Me} is average luminance per cluster, considering all the pixels within. We denominate this equation as *conservation of energy* equation, as it forces the system to keep a balance of luminance w.r.t. the original image.

The equations 5.4 and 5.5 form the equation system $AX = B$ for N clusters M pairs of neighboring clusters. Each row a_i of A is given by,

$$\forall i \in 1..M, a_i = \begin{cases} \exists k, l \in 1..N \ni a_{ik} = L_m(c_k)_{c_l}, a_{il} = -L_m(c_l)_{c_k} \\ \text{con } k < l \wedge c_k \text{ is neighbor of } c_l \\ a_{ih} = 0, \forall h \in 1..N \ni h \neq k \wedge h \neq l \end{cases} \quad (5.6)$$

$$i = M + 1, \forall j \in 1..N, a_{ij} = L_{Me}(c_j)$$

We define X and B by,

$$X_{N \times 1} = \begin{pmatrix} F_{c_1} \\ F_{c_2} \\ \vdots \\ F_{c_N} \end{pmatrix} \quad B_{(M+1) \times 1} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \sum_{i=1}^N L_{Me}(c_i) \end{pmatrix} \quad (5.7)$$

We solve the equivalent system $(A^T A)X = (A^T B)$ by using the Quasi-Minimal Residual method (QMR) (BBC⁺94). As we can observe in Figure 5.10c, the results show an undesirable bias or polarization in the distribution of the luminance energy. The reason is that, while the equation 5.5 maintains the global energy of the system constant, it is not suitable to keep an even distribution of energy over the image. For our iterative solver, at a local level, the ratios converge to 1.0 for most of the pairs, compensating the fact that a few of them (those between dark and bright areas) obtained really poor ratios (below 0.5). Additionally, considering the problem from a mathematical point of view, as B is mainly composed by zeroes the solver tends to yield the trivial solution. In order to overcome this problem, in the following subsection we have established a similitude between our problem and the steady state computation for thermodynamics and fluids.

5.5.2 Looking for the Luminance Steady State

Our system starts with an unstable initial state: there is a luminance imbalance between the clusters, this is, the transition (in luminance) from the pixels of one cluster to the adjacent pixels of neighboring clusters is not negligible. The goal of the system is to reach the steady state, a smooth image without sharp luminance variations between albedo clusters. To achieve this, each cluster tries to balance

5. INTRINSIC IMAGES DECOMPOSITION

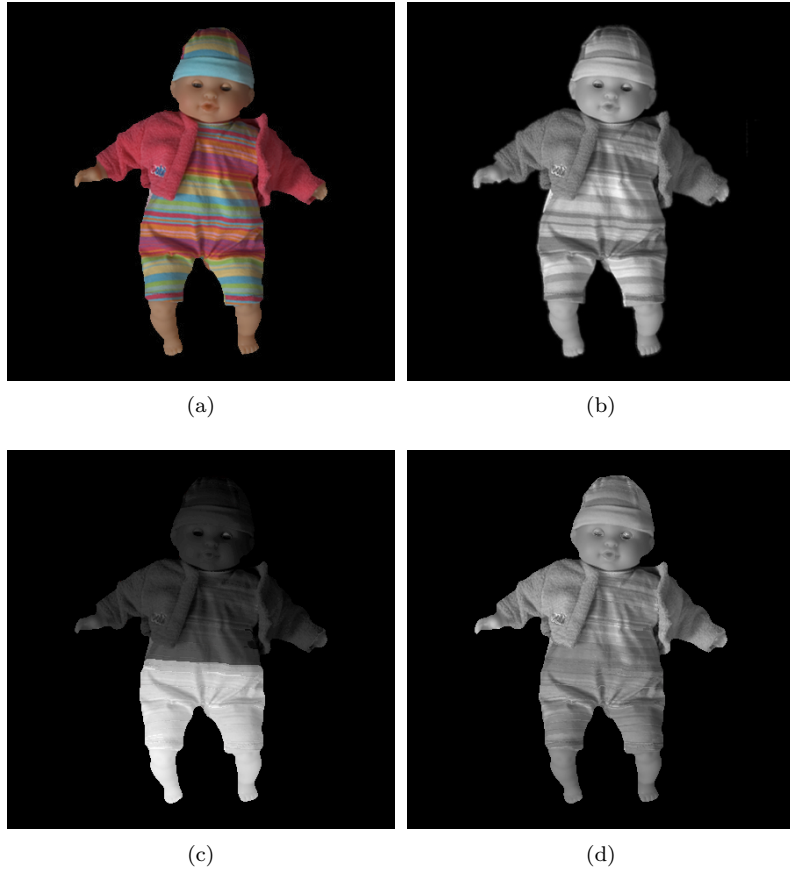


Figure 5.10: *Luminance Normalization.* (a) is the input image, (b) is the perceptual luminance image, (c) is the luminance yielded by our method with the energy system (Equations 5.6 and 5.7) and (d) is the luminance with the flow system (Equations 5.10 and 5.11)

itself by transferring luminance with adjacent regions. This incoming or outgoing luminance transfer is denominated *flow* φ . To devise the new linear system we take logarithms in the previous system, turning the products into additions. Equation 5.4 is therefore modified:

$$\ln(L_m(c_i)_{c_j}) - \ln(L_m(c_j)_{c_i}) = \varphi_j - \varphi_i \quad (5.8)$$

where $\varphi_i = \ln(F_{c_i})$ and $\varphi_j = \ln(F_{c_j})$.

The equation of energy conservation 5.5 is transformed into a flow conservation equation in this manner,

$$\sum_{i=1}^N \varphi_i = 0 \quad (5.9)$$

With this approach we rewrite our previous equation system $AX = B$. As in prior equations, A is a matrix with N columns and $M + 1$ rows, being M the number of adjacent clusters and N the number of clusters in the image. In this case, each row a_i of A is given by,

$$\forall i \in 1..M, a_i = \begin{cases} \exists k, l \in 1..N \ni a_{ik} = 1, a_{il} = -1 \\ \text{with } k < l \wedge c_k \text{ is adjacent to } c_l \\ \forall h \in 1..N, a_{ih} = 0, \text{ si } h \neq k \wedge h \neq l \end{cases} \quad (5.10)$$

$$i = M + 1, \forall j \in 1..N, \quad a_{ij} = 1$$

With X and B defined by,

$$X_{N \times 1}^T = (\varphi_1 \quad \varphi_2 \quad \dots \quad \varphi_N) \quad (5.11)$$

$$B_{(M+1) \times 1}^T = (b_1 \quad \dots \quad b_M \quad 0) \text{ where,} \quad (5.12)$$

$$\forall i \in 1..M, \quad \exists k, l \in 1..N \ni b_i = \ln(L_{Me}(c_l)) - \ln(L_{Me}(c_k)) \\ \text{with } k < l \wedge c_k \text{ is adjacent to } c_l$$

Now, the vector B is composed by non-zero values, which helps in converging to a better solution. The results (see Figure 5.10d) improve in great manner those shown in the previous case. However, we have observed that for certain images with very complex interactions among clusters (highly textured, not globally convex and with large areas covered by self-cast shadows) the results remain biased: although locally correct, there is still a luminance unbalance between regions of the image (see Figure 5.11c).

One way to circumvent this problem is by relying in image statistics: forcing the result to share certain statistical attributes (e.g.:mean and standard deviations in histogram) with the input image. In the following equations, we introduce a strong constraint in the system which consists of forcing each cluster to have a final luminance equal to the average of the global system:

$$\forall j \in 1..N, \quad \frac{1}{N} \sum_{i=1}^N \varphi_i + \ln(L_{Me}(c_i)) = \varphi_j + \ln(L_{Me}(c_j)) \quad (5.13)$$

This is analogous to adding a similar constraint in our previous system by equaling the luminance of each cluster to the geometric mean of the image,

$$\forall j \in 1..N, \quad \left(\prod_{i=1}^n F_{c_i} L_{Me}(c_i) \right)^{1/n} = F_{c_j} L_{Me}(c_j) \quad (5.14)$$

5. INTRINSIC IMAGES DECOMPOSITION

Finally by adding Equation 5.13 to our flow system 5.10 we obtain a *balanced* flow system. The matrix A will have $M + N + 1$ rows and N columns, where $M + 1$ rows are defined by Equation 5.10 and the subsequent N rows are given by:

$$\forall j \in 1..N, \quad a_{M+1+j} = \begin{cases} \exists k \in 1..N \ni a_{M+1+j,k} = -1, \text{ si } k = j \\ \forall h \in 1..N, \quad a_{M+1+j,h} = \frac{1}{N}, \text{ if } h \neq j \end{cases} \quad (5.15)$$

The vector X remains unchanged and vector B is obtained as follows:

$$B_{(M+N+1) \times 1}^T = \left(b_1 \quad \dots \quad b_M \quad 0 \quad b'_1 \quad \dots \quad b'_N \right) \text{ where,} \quad (5.16)$$

$$\forall i \in 1..N, \quad b'_i = \ln(L_{Me}(c_i)) - \frac{1}{N} \sum_{j=1}^N \ln(L_{Me}(c_j))$$

In Figure 5.11d we can observe how the result improves significantly. However, we find that this kind of image-statistics approaches depend in great manner in the good choice of image metrics and, with robustness in mind, should be avoided as much as possible. We propose an alternative solution, which consists of grouping the clusters which were successfully normalized with our QMR solver and feeding them back into our system for additional iterations until convergence. In our experience our method yields satisfactory results in four-five iterations (which are faster due to the reduced number of clusters). The results of this alternative are shown in the next section.

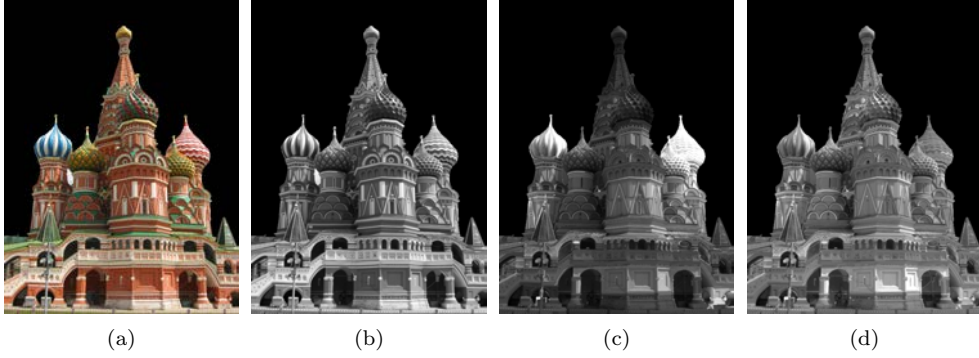


Figure 5.11: *Luminance Normalization* (a) is the input image, (b) is the perceptual luminance image, (c) is the resulting image with the flow system (Equations 5.10 and 5.11) and (d) is the final luminance with the balanced flow system (Equation 5.13)

5.5.3 Solving the System

As previously mentioned, the linear system is solved by means of a Quasi-Minimal Residual method (BBC⁺94). This method was chosen due to its fast convergence. The solver yields the values φ_c . By

a change of variables $F_c = \exp^{\varphi_c}$ we obtain the ratios by which we have to multiply the clusters of the input image in order to obtain the corresponding constant-reflectance image.

In order to accelerate the convergency our method, we have introduced a *Jacobi* preconditioner, that is, a matrix M where $M = D = \text{diag}(A)$ which transforms the original system $Ax = b$ into an equivalent system $\tilde{A}x = \tilde{b}$ by multiplying both A and b . In this fashion we reduce the number of iterations to less than its half.

Additionally we have tested more customized preconditioner matrices which introduce prior knowledge into the system in order to reduce the instability of the data and speed up the solving process. For instance, we added weight to those connections between clusters which share more pixel connections or to those clusters which had more pixels w.r.t. the total area of the image. Therefore, the preconditioner matrix W would be built as follows:

$$W = \begin{pmatrix} w_1 & 0 & \dots & 0 \\ 0 & w_2 & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & \dots & 0 & w_{M+1} \end{pmatrix} \quad (5.17)$$

$$\forall k \in 1..M \quad w_k = \text{Area}_{c_i} \frac{\text{nconex}_{ij}}{\text{nconex}_i} + \text{Area}_{c_j} \frac{\text{nconex}_{ij}}{\text{nconex}_j} \quad (5.18)$$

$$k = M + 1 \quad w_k = \frac{\sum_{i=1}^M w_i}{M}$$

where c_i and c_j are the connected clusters at the row a_k of the matrix A , nconex_{ij} represents the total number of connecting edges between clusters c_i and c_j , and nconex_i is the total number of connections which has the cluster c_i with the remaining clusters of the image.

Unfortunately, so far our preconditioners have not improved the results of our system and we are still working on new priors which might accelerate our convergence rates.

5.6 Results

We have tested our decomposition method in a varied set of images. Some examples of our intrinsic images decomposition are shown in Figures 5.13 , 5.12 and 5.20. In Figures 5.14, 5.17 and 5.18 we compare our results with the most relevant automatic (a single image as input) techniques in the field: the method proposed by Tappen et al. (TFA05) and Shen's algorithm (STL08). Additionally, we compare our approach with algorithms which require additional input: Weiss' method (Wei01), which uses multiple images from the same scene acquired under different illumination conditions, and the approach proposed by Bousseau et al. (BPD09), which require user interaction in the form of brush strokes.

5. INTRINSIC IMAGES DECOMPOSITION

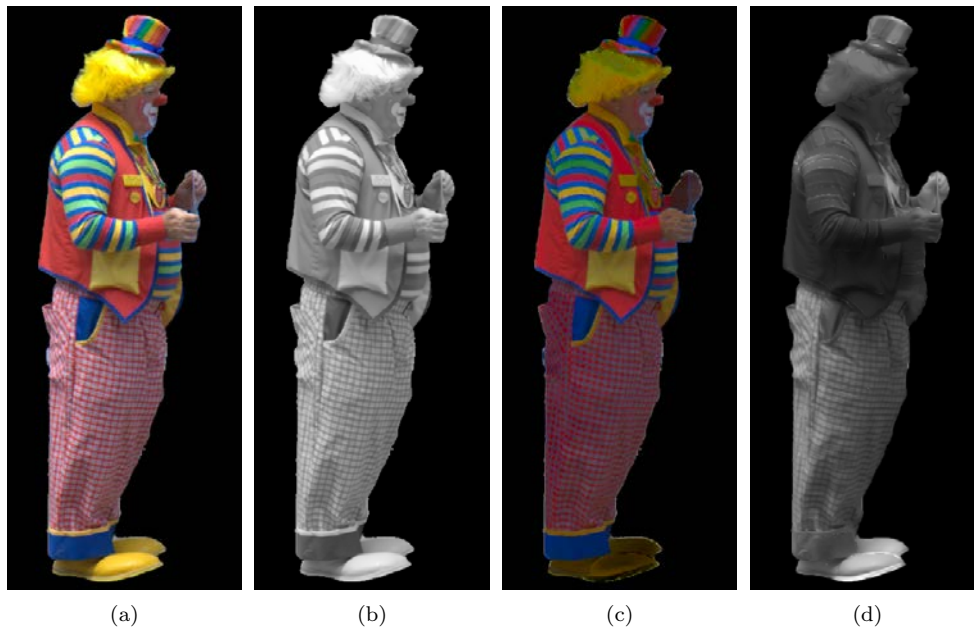


Figure 5.12: *Intrinsic images obtained by our method. (a) Input image. (b) Perceptual luminance. (c) Reflectance. (d) Illumination.*

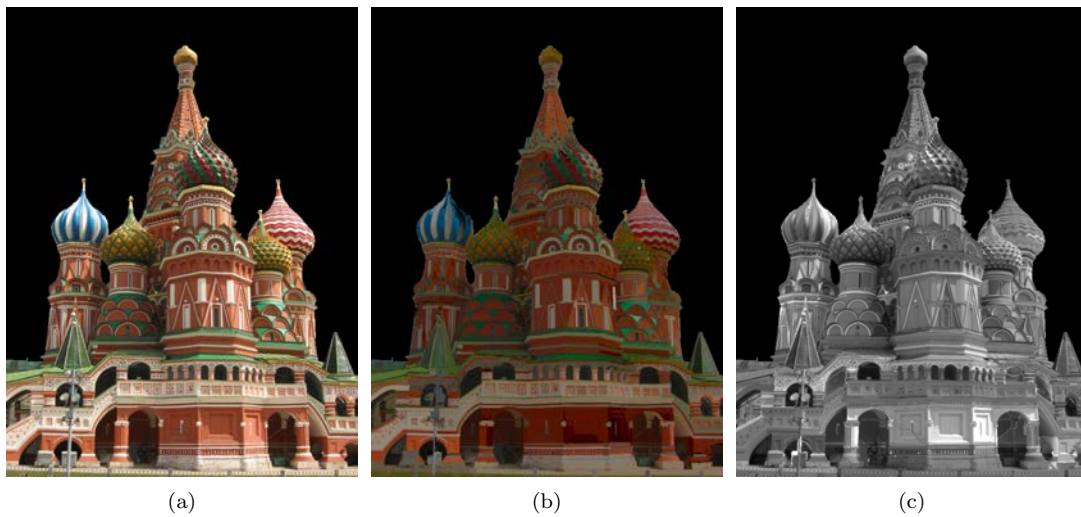


Figure 5.13: *Intrinsic images obtained by our method. (a) Input image. (b) Reflectance. (c) Illumination. Original image authored by Captain Chaos, flickr.com*

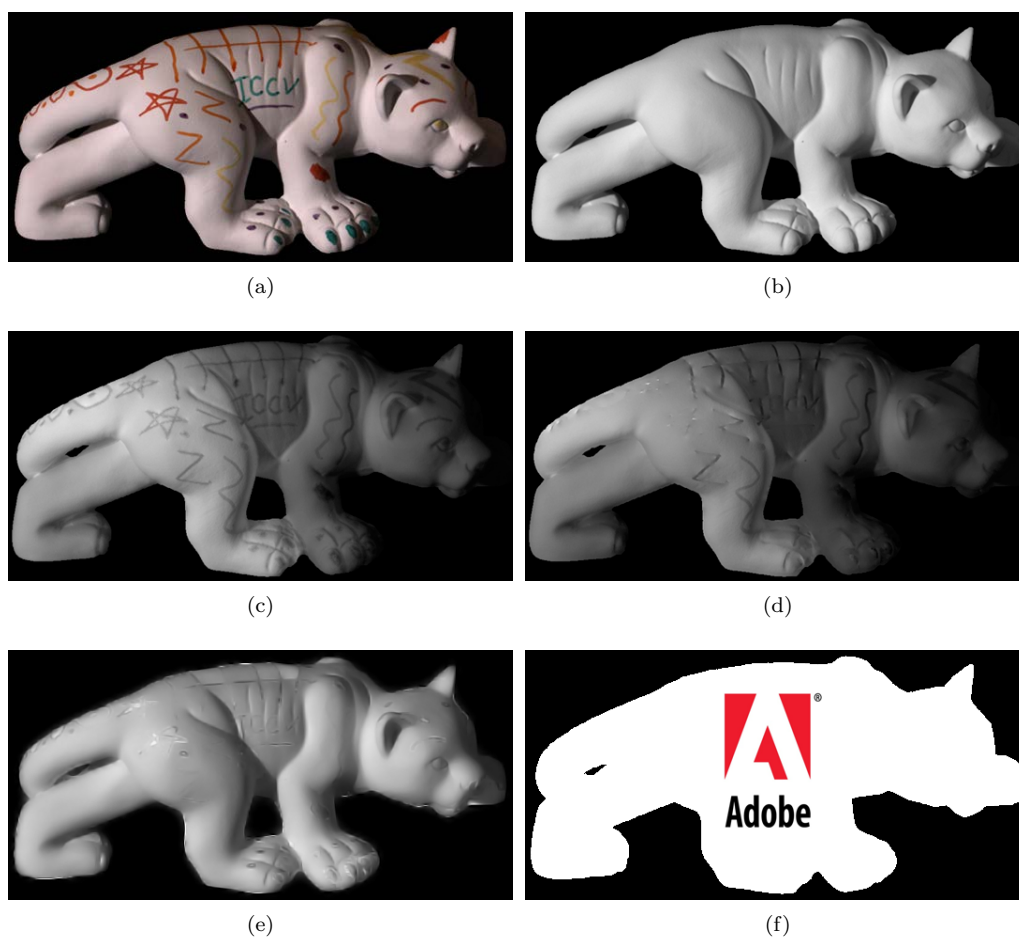


Figure 5.14: *Comparison of decomposition: illumination component. (a) Input image. (b) Ground truth. (c) Our solution is closer to ground truth (b), than (c) Shen (STL08) or (d) Tappen (TFA05). In (f) we show additional results of a method developed by Adobe Systems not shown due to copyright issues. It will be shown in the defense of this thesis.*

5. INTRINSIC IMAGES DECOMPOSITION

The results show that our method equals or improves those obtained by all the aforementioned automatic methods, being on par in several cases with the methods which require additional input.

If we observe the reflectance images in Figures 5.13b and 5.20b, we can see how these look plain, with almost no trace of shading. On the contrary, we can find the illumination variations in the shading images. A good example of the correct behavior of our approach is shown in Figure 5.20c, where the letters in the bib have been completely removed.

In Figure 5.12d we can observe that the overall illumination of the scene has been correctly captured. For instance, the stripes pattern on the sleeves has been almost completely removed. However, our method failed in capturing the high (spatial) frequency of the checker pattern in the trousers. In future research we aim to overcome this limitation by working at different levels of detail. Figure 5.15 shows another example of illumination edit beyond the capabilities of an histogram-based tonemapping like those provided by PhotoshopTM.

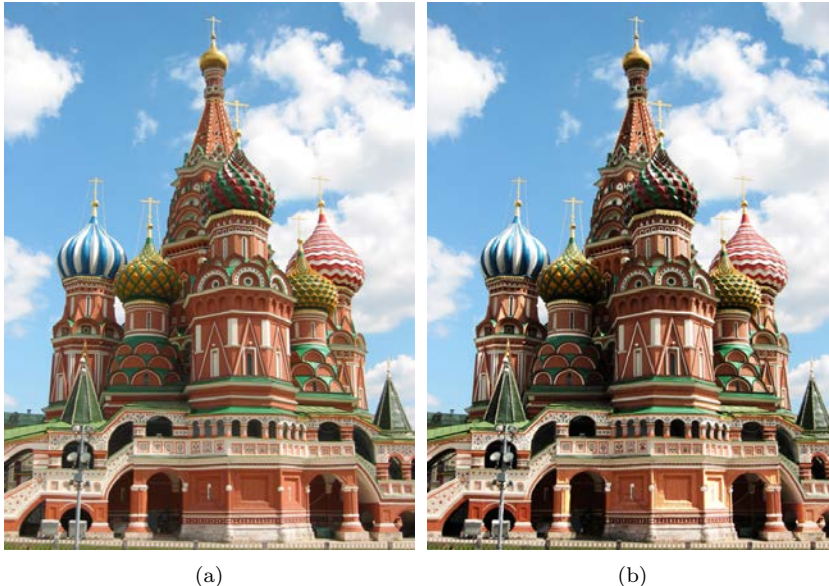


Figure 5.15: *Example of reillumination by our method. (a) Input image. (b) Image where the illumination energy levels have been enhanced. Note how the result conserves a natural look and feel.*

In Figures 5.14, 5.17 and 5.18 we compare our method with the most representative algorithms in the field. We can observe how these methods could not perform a satisfactory removal of the paintings in Figure 5.14. Our method, however, was able to extract almost all the texture information (although some clusters were left unaltered due to their small pixel size). In Figure 5.18 our results are on par with those of Shen et al. (STL08) and Bousseau et al. (BPD09), but without the need of user interaction (see Figure 5.18f) or additional images.

Finally, in Figure 5.22 we show the result of applying our method in an iterative fashion, by using

as input for each iteration the resulting illumination of the previous computation. As described in Subsection 5.5.2, when the number of clusters is considerable (hundreds) our system may globally converge while conserving certain local clusters without normalized ratios (< 0.9). This local error, depending on the connectivity of the graph may result in an unbalanced normalization (like in Figure 5.10c). However if we iterate our method, the number of clusters is reduced by grouping normalized clusters, altering the topology of the graph and helping in the propagation of correct ratios among the set of clusters. In our experience, we find that up to five iterations of the method, in general suffices to refine the results and reach both global and local small errors.

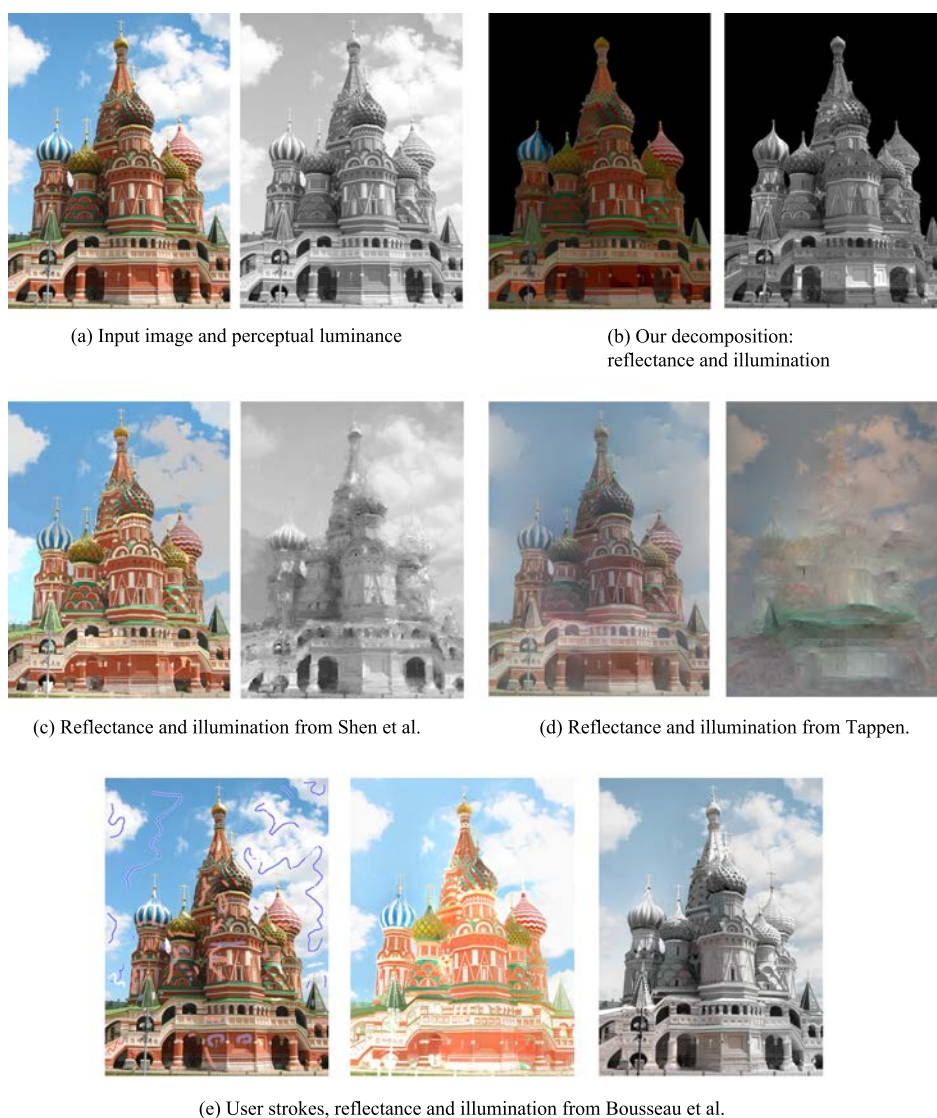


Figure 5.16: *Comparison with other decomposition methods.*

5. INTRINSIC IMAGES DECOMPOSITION

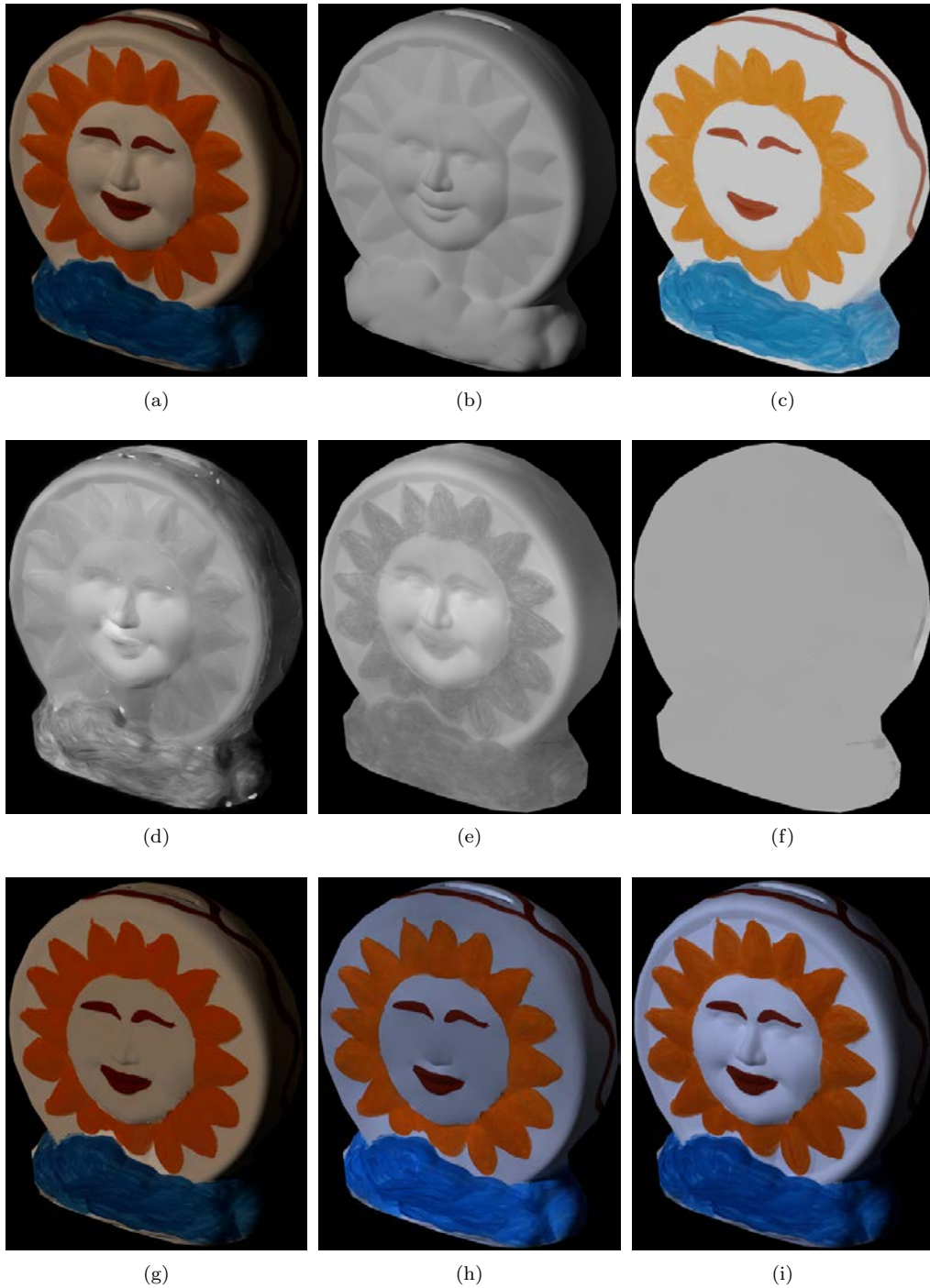


Figure 5.17: Comparison with other decomposition methods. (a) Input image. (b) Ground truth shading. (c) Ground truth reflectance. (d) and (g) shading and reflectance with our method. (e) and (h) shading and reflectance by Shen et al. (STL08). (f) and (i) shading and reflectance by Tappen et al. (TFA05).



Figure 5.18: Comparison with ⁹⁷other decomposition methods.

5. INTRINSIC IMAGES DECOMPOSITION

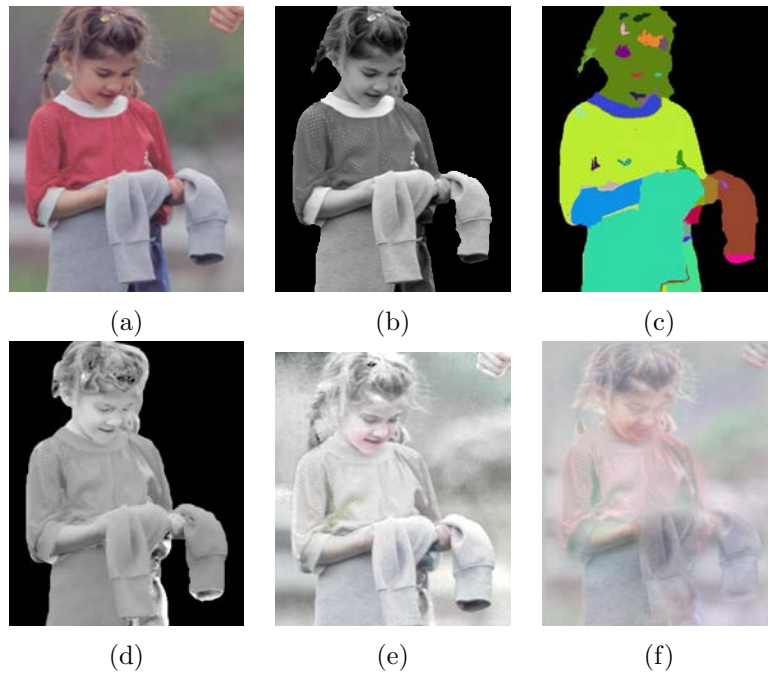


Figure 5.19: *Comparison with other decomposition methods.* (a) Input image. (b) Original luminance. (c) Segmentation obtained by our method. (d) Illumination yielded by our method. (e) Illumination from Bousseau et al. (BPD09) method. (f) Illumination by Tappen et al. (TFA05) method.

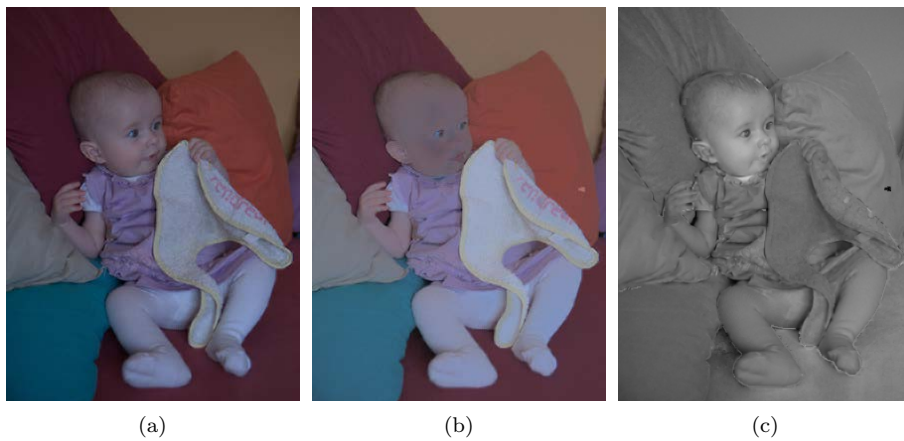
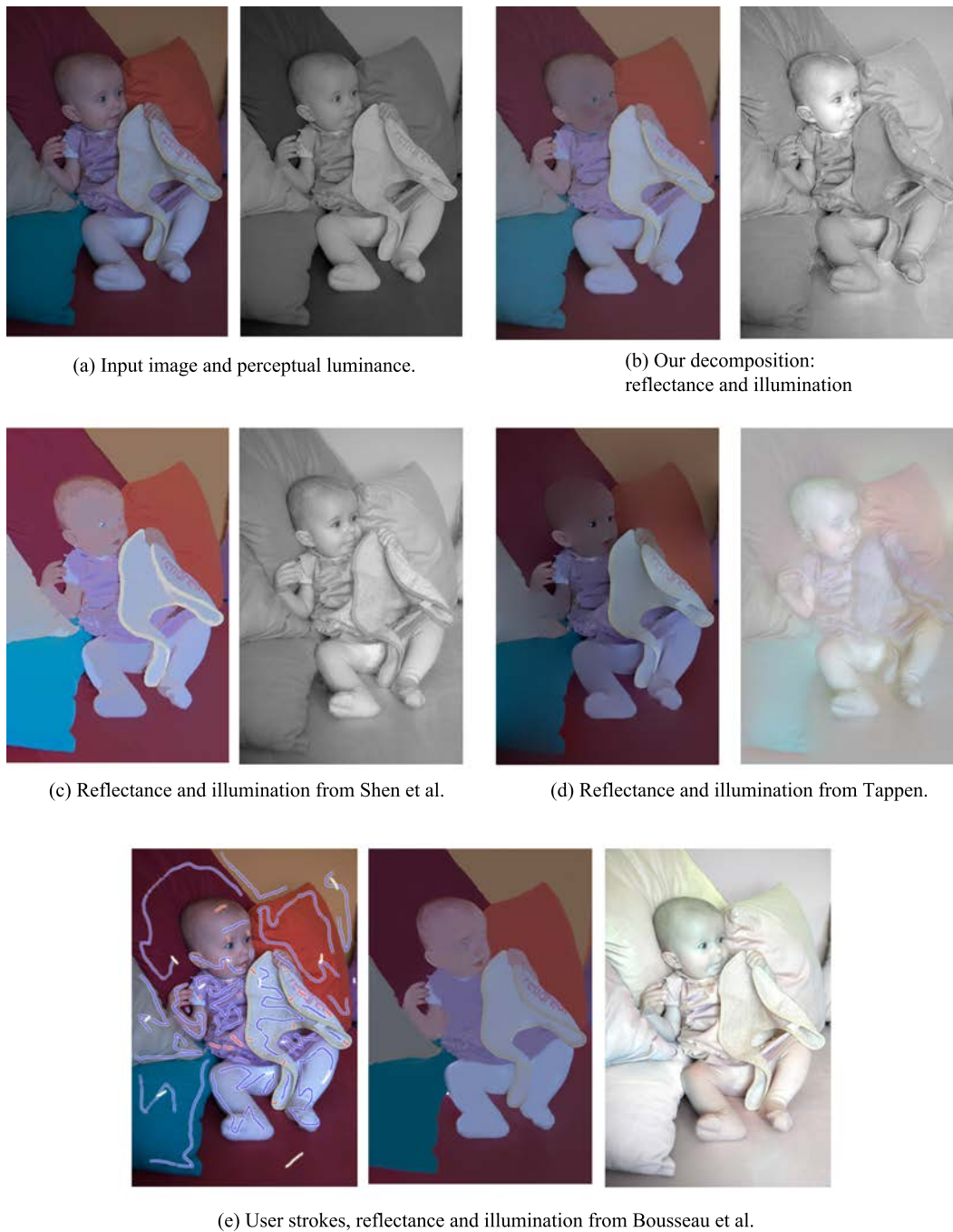


Figure 5.20: *Intrinsic images obtained by our method.* (a) Input image. (b) Reflectance. (c) Illumination.

Figure 5.21: *Comparison with other decomposition methods*

5. INTRINSIC IMAGES DECOMPOSITION

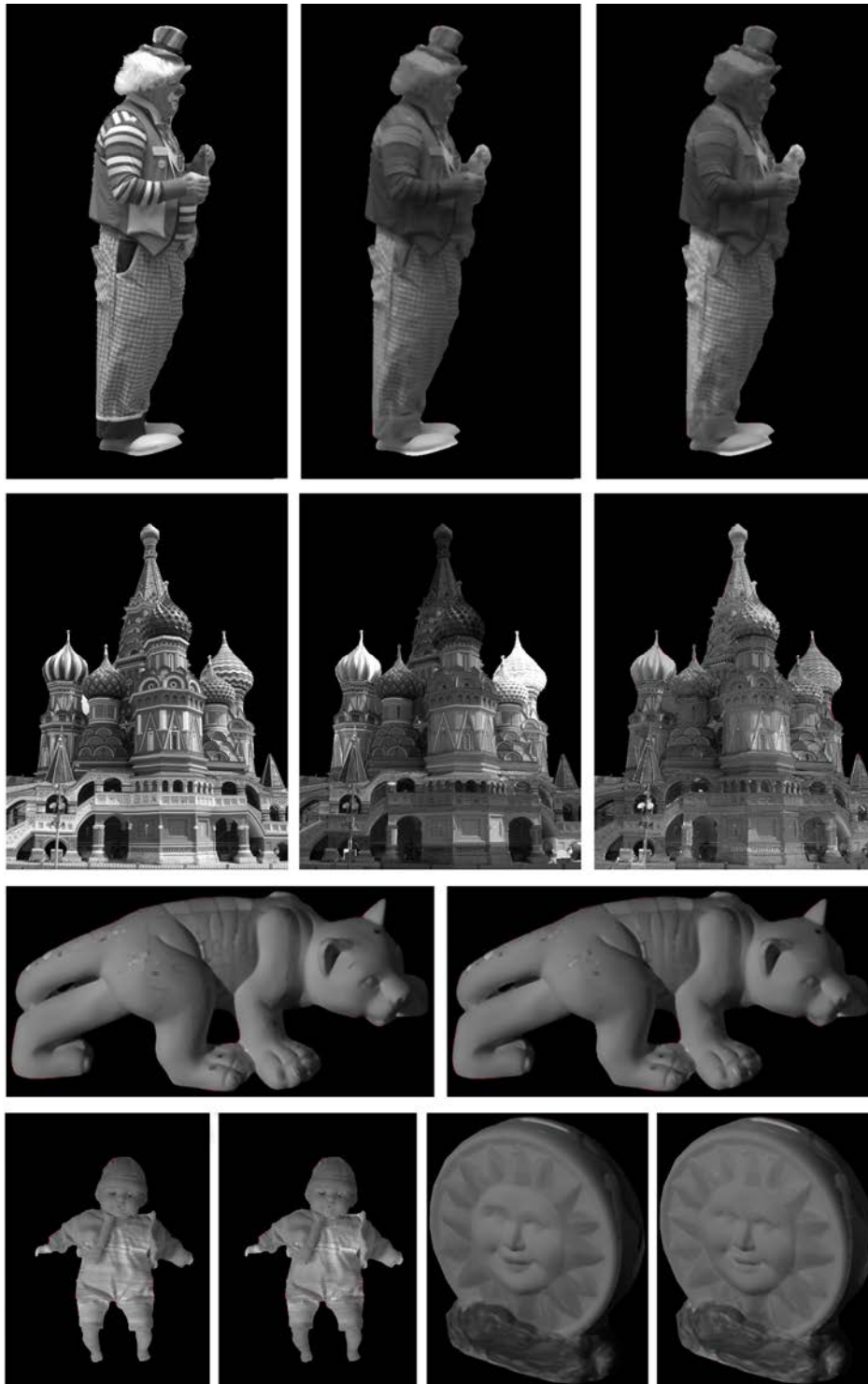


Figure 5.22: Top row and second row, from left to right: Input luminance, result of one iteration of our algorithm and result after five iterations grouping clusters and normalizing the result. Likewise, the pairs of images shown at the third and fourth rows are the result obtained after one iteration and five iterations respectively.

5.7 Conclusions

At this point we can conclude that most of the initial objectives have been achieved. We have presented a novel *intrinsic images* decomposition method which surpasses in most aspects the most relevant approaches in the field (TFA05; STL08), being on par with algorithms that require more information than a single image (Wei01; BPD09). Our method does not require user interaction and works at interactive rates. We have adapted an existing segmentation method (FH04a) to detect areas of constant albedo. Our method has been used as pre-processed input for other applications such as image relighting. These results are promising in order to improve the accuracy of techniques such as light detection or 3D shape reconstruction.

5.8 Limitations and Future Work

Although the results obtained are compelling and suggest that this line of research has great potential as a robust intrinsic image decomposer, our work is still in progress and we are working in the following three aspects: First, we are considering incorporating the knowledge of the direction of illumination, based on the work in light source detection in single images of Lopez-Moreno et al. (LMHRG10). By knowing the orientation of shading gradients we could reduce its influence and disambiguate shading from reflectance in our segmentation process, specially considering images without color information (where chromaticity cannot be used as weighting factor). Second, our initial experiments seem to confirm that pre-filtering the images with techniques such as the bilateral filter or mean shift, improve the accuracy of the segmentation process, but an exhaustive study is still needed.

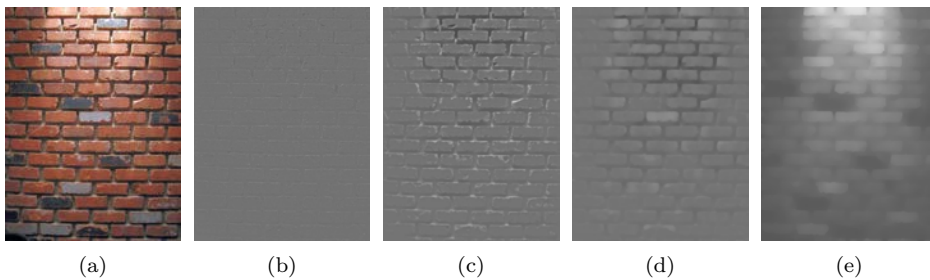


Figure 5.23: **Multi-level decomposition (SSD09)**. In (a) we show the input image. In (b)-(e) we can see different layers, from the finer to the coarser level of detail. It can be observed how the illumination component is mostly captured at level (e).

Finally, we are exploring the use of our method in a multi-scale decomposition framework (SSD09; FAR07; FFLS08). Specifically, we are working with the technique by Subr et al. (SSD09) (an example is shown in Figure 5.23). We hope that this approach will help us to work with images which contain (simultaneously) very high and low spatial frequency textures (e.g.:the checker pattern of the trousers in the clown shown in Figure 5.12).

5. INTRINSIC IMAGES DECOMPOSITION

References

- [BBC⁺94] R. Barret, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, R. Pozo, V. Eijkhout, H. Van der Vorst, and C. Romine, *Templates for the solution of linear systems: Building blocks for iterative methods, 2nd edition*, SIAM, 1994. 87, 90
- [BPD09] Adrien Bousseau, Sylvain Paris, and Frédo Durand, *User assisted intrinsic images*, ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2009) **28** (2009), no. 5. 76, 78, 79, 91, 94, 98, 101
- [BT78] H.G. Barrow and J.M. Tenenbaum, *Recovering intrinsic scene characteristics from images*, Computer Vision Systems (1978), 3–26. 75, 76
- [BVZ01] Yuri Boykov, Olga Veksler, and Ramin Zabih, *Fast approximate energy minimization via graph cuts*, IEEE Transactions on Pattern Analysis and Machine Intelligence **23** (2001), 2001. 81
- [CM02] D. Comaniciu and P. Meer, *Mean shift: a robust approach toward feature space analysis*, Pattern Analysis and Machine Intelligence, IEEE Transactions on **24** (2002), no. 5, 603–619. 81
- [CPD07] Jiawen Chen, Sylvain Paris, and Frédo Durand, *Real-time edge-aware image processing with the bilateral grid*, SIGGRAPH '07: ACM SIGGRAPH 2007 papers (New York, NY, USA), ACM, 2007, p. 103. 79, 83
- [EHL71] John Edwin H. Land and J. Mccann, *Lightness and retinex theory*, Journal of the Optical Society of America (1971), 1–11. 76, 78
- [FAR07] Raanan Fattal, Maneesh Agrawala, and Szymon Rusinkiewicz, *Multiscale shape and detail enhancement from multi-light image collections*, SIGGRAPH '07: ACM SIGGRAPH 2007 papers (New York, NY, USA), ACM, 2007, p. 51. 79, 101
- [FDB91] Brian V. Funt, Mark S. Drew, and Michael Brockington, *Recovering shading from color images*, ECCV-92: Second European Conference on Computer Vision, Springer-Verlag, 1991, pp. 124–132. 82

REFERENCES

- [FFLS08] Zeev Farbman, Raanan Fattal, Dani Lischinski, and Richard Szeliski, *Edge-preserving decompositions for multi-scale tone and detail manipulation*, ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2008) **27** (2008), no. 3. 79, 101
- [FH04a] Pedro F. Felzenszwalb and Daniel P. Huttenlocher, *Efficient graph-based image segmentation*, International Journal of Computer Vision **59** (2004), 2004. 79, 81, 82, 84, 101
- [FH04b] P.F. Felzenszwalb and D.R. Huttenlocher, *Efficient belief propagation for early vision*, Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, 2004. 81
- [FHL06] Graham D. Finlayson, Steven D. Hordley, Cheng Lu, and Mark S. Drew, *On the removal of shadows from images*, IEEE Transactions on Pattern Analysis and Machine Intelligence **28** (2006), 59–68. 78
- [GDFL04] Mark S. Drew Graham D. Finlayson and Cheng Lu, *Intrinsic images by entropy minimization*, Proc. 8th European Conf. on Computer Vision, Prague, 2004, pp. 582–595. 78
- [HMP⁺08] Eugene Hsu, Tom Mertens, Sylvain Paris, Shai Avidan, and Frédo Durand, *Light mixture estimation for spatially varying white balance*, ACM Trans. Graph. **27** (2008), 70:1–70:7. 86
- [Hor86] B. K. Horn, *Robot vision*, MIT Press, 1986. 78, 86
- [I.T90] I.T.U., *Basic parameter values for the hdtv standard for the studio and for international programme exchange*, 1990. 84
- [KRFB06] Erum Arif Khan, Erik Reinhard, Roland Fleming, and Heinrich Bühlhoff, *Image-based material editing*, ACM Transactions on Graphics **25** (2006), no. 3, 654–663. 79
- [LMHRG10] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Compositing images through light source detection*, Computers & Graphics **In press** (2010). 101
- [LMJH⁺11] Jorge Lopez-Moreno, Jorge Jimenez, Sunil Hadap, Erik Reinhard, Ken Anjyo, and Diego Gutierrez, *Non-photorealistic, depth-based image editing*, Computers & Graphics **In press** (2011). 76
- [LWQ⁺08] Xiaopei Liu, Liang Wan, Yingge Qu, Tien-Tsin Wong, Stephen Lin, Chi-Sing Leung, and Pheng-Ann Heng, *Intrinsic colorization*, ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2008) (2008), 1–9. 77
- [MK10] Branislav Micusik and Jana Koseck, *Multi-view superpixel stereo in urban environments*, International Journal of Computer Vision **89** (2010), 106–119, 10.1007/s11263-010-0327-9. 81
- [MTC07] Ankit Mohan, Jack Tumblin, and Prasun Choudhury, *Editing soft shadows in a digital photograph*, IEEE Comput. Graph. Appl. **27** (2007), no. 2, 23–31. 78

-
- [RT10] R. Raskar and J. Tumblin, *Computational photography: Mastering new techniques for lenses, lighting, and sensors*, A K Peter, 2010. 75
- [SM00] Jianbo Shi and J. Malik, *Normalized cuts and image segmentation*, Pattern Analysis and Machine Intelligence, IEEE Transactions on **22** (2000), no. 8, 888–905. 81
- [SSD09] Kartic Subr, Cyril Soler, and Frédo Durand, *Edge-preserving multiscale image decomposition based on local extrema*, , Annual Conference Series, ACM, ACM Press, dec 2009. 79, 101
- [STL08] Li Shen, Ping Tan, and Stephen Lin, *Intrinsic image decomposition with non-local texture cues*, Computer Vision and Pattern Recognition, IEEE Computer Society Conference on **0** (2008), 1–7. 78, 91, 93, 94, 96, 101
- [SZS⁺08] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, *A comparative study of energy minimization methods for markov random fields with smoothness-based priors*, Pattern Analysis and Machine Intelligence, IEEE Transactions on **30** (2008), no. 6, 1068–1080. 81
- [TFA05] Marshall F. Tappen, William T. Freeman, and Edward H. Adelson, *Recovering intrinsic images from a single image*, IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005), 1459–1472. 78, 91, 93, 96, 98, 101
- [UPH07] R. Unnikrishnan, C. Pantofaru, and M. Hebert, *Toward objective evaluation of image segmentation algorithms*, Pattern Analysis and Machine Intelligence, IEEE Transactions on **29** (2007), no. 6, 929–944. 81
- [Wei01] Yair Weiss, *Deriving intrinsic images from image sequences*, Computer Vision, IEEE International Conference on **2** (2001), 68. 77, 91, 101
- [WTBS07] Tai-Pang Wu, Chi-Keung Tang, Michael S. Brown, and Heung-Yeung Shum, *Natural shadow matting*, SIGGRAPH '07: ACM SIGGRAPH 2007 papers **26** (2007), no. 2, 8. 78
- [YFW03] Jonathan S. Yedidia, William T. Freeman, and Yair Weiss, *Understanding belief propagation and its generalizations*, Exploring artificial intelligence in the new millennium (2003), 239–269. 81

REFERENCES

Chapter 6

Application 1: Light Transport in Participating Media. An Image Editing Approach

In this chapter, we introduce a novel method for image-based simulation of light transport in participating media (fog). Although the scope of this research is quite limited in comparison with the remaining applications shown in this thesis, we consider it a good example of how to reduce a physically complex problem with multiple dimensions into a sequence of image processing operations. This research has been published at the conference CEIG 2008 (organized by the Spanish Eurographics Chapter) (LMCG08).

6.1 Introduction

Participating media like fog or smoke have a great influence in the light transport of a scene. Its presence implies a series of complex interactions which greatly affect how objects are perceived. More precisely, light transport through participating media is affected by the following phenomena (SKSU05) (Figure 6.1):

- Emission: Radiance is increased by the photons emitted by the participating medium itself.
- Absorption: Radiance decreases when photons are absorbed by the particles composing the participating medium.

6. APPLICATION 1: LIGHT TRANSPORT IN PARTICIPATING MEDIA. AN IMAGE EDITING APPROACH

- In-scattering: Radiance increases due to photons scattered in the direction of the considered path.
- Out-scattering: Radiance decreases due to photons scattered out of the direction of the considered path.

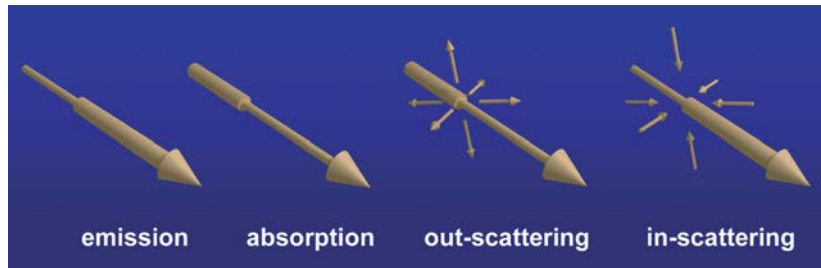


Figure 6.1: The four types of interaction of light in participating media (after (PPS97)).

This phenomena makes simulating light transport in participating media a computationally expensive process, often requiring previous 3D knowledge of the scene. Rather than attempting to provide a physically-based simulation, which would require complete knowledge of the scene’s properties, such as dimensions, optical thickness of the medium or reflectance properties of the objects, we aim to simulate its effects in image-space, starting with a single high dynamic range (HDR) image as input. Given the underconstrained nature of the problem, a physically accurate solution is obviously impossible to achieve. However, Ramanarayanan and colleagues (RFB07) showed that physical inaccuracies in an image come largely undetected in some situations, and therefore a perceptually plausible solution can be achieved that will be perceived as correct by a human observer. In that regard, our research is similar in spirit to the work by Khan et al.(KRF06), which provides an algorithm for image-based material editing by exploiting the limitations of the human visual system. The purpose of this research is to extend the current capabilities of image-editing tools (such as PhotoshopTM), for which intuitive interactivity and short computational times are required.

The rest of the chapter is structured as follows: In section 6.2 we discuss previous work similar to ours. In section 6.3, we analyze the natural process: the physical interpretation and its influence on the perception of scenes. Based on this analysis, we implement a processing pipeline capable of simulating the presence of participating media in a single HDR image, yielding visually realistic results. Finally we validate our results in section 6.4 by means of two psychophysical tests, and by measuring our rendering times against several artists’ renditions using commercial image editing software (like PhotoshopTM).

6.2 Previous Work

Our research is closely related to the image-based work by Nayar and colleagues (SNN01),(NN01), (NN03b),(NN03a). A method to remove haze based on the partial polarization of airlight (defined as the ambient light scattered towards the viewer) is introduced by Schechner et al. (SNN01). This method requires two images taken with polarization filters, preferably at parallel and perpendicular orientations. Narasimhan et al. (NN01) restore partially the contrast in foggy images; although the method does not require any prior information, it needs several images of the same scene as input. This restriction is subsequently lifted in posterior research (NN03b), in exchange of some user input. The results in all these works are compelling, contributing to the problem of removing undesirable effects from images, caused by light transport in participating media. However, it is not clear if the processes could be reversed to *add* those effects to clean input images.

By contrast, Narasimhan and Nayar simulate multiple scattering of single point light sources by means of a point-spread function (NN03a). Unfortunately with this method, only the light transport originated by the light sources which are visible in the original image can be simulated, and the results are constrained to a limited subset of cases: light sources in almost completely dark scenes (e.g. lamps in a misty night). We overcome these limitations by presenting a method to simulate participating media in images relying in image processing techniques. We show that the underconstrained nature of the problem can be solved with little unskilled user input. The algorithms use a single HDR image as input, and no previous knowledge of the scene is required.

An important difference with previous work is our overall goal: whilst Nayar and co-workers focus in computer vision related problems, our goal is to extend the available repertoire of image editing tools. We believe that the progressive establishment of an HDR imaging pipeline opens up new possibilities for these kind of applications, creating image editing techniques that were not possible before due to the quantization and loss of data in traditional low dynamic range images. An example of this is the work by Khan et al. (KRFB06), which shows how extreme material edits can be performed in perceptual space, leveraging the wealth of information available in HDR format. We thus aim at producing simulated participating media that can be seen as perceptually plausible, a claim we support by means of psychophysical validations. Sundstedt et al.(SGA⁺07) already proved the convenience of such an approach for participating media: the authors were able to drastically cut down rendering times by taking advantage of the limitations of human perception, producing images indistinguishable from ground-truth, Monte Carlo based renderings at a fraction of the time. While the aforementioned authors used a traditional 3D rendering approach (with a complete description of the scene and the medium), we use a single HDR image as input.

6.3 Light in Participating Media

In this section we analyze light-medium interactions and how they influence the visual perception of the image (Subsections 6.3.1 and 6.3.3). Then, we simplify the physical model, thus circumventing the underconstrained nature of working in image space while still producing plausible results (Subsection 6.3.2). Finally we present a processing pipeline capable of simulating the complex interactions inside the participating medium by means of a sequence of simple image filters (Subsection 6.3.4).

6.3.1 Assumptions

Starting with a single HDR image, we follow the approach introduced by Debevec and co-workers for image-based lighting (Deb98) and interpret every pixel as a light source. We limit ourselves to isotropic light sources and homogeneous participating media. Further, we will show how non-homogeneous media can be simulated by simply adding Perlin noise to our algorithm.

6.3.2 Simplifying the Physical Model

We describe the physical process in terms of the Radiance Transfer Equation (RTE) (Gla95). Marching along a ray, we can find the total change in radiance per unit distance t as follows:

$$\begin{aligned}
 (\vec{w} \cdot \nabla)L(t, \vec{w}) &= \alpha(t)L_e(t, \vec{w}) \\
 &+ \sigma(t) \int_{\Omega} p(t, \vec{w}', \vec{w})L_i(t, \vec{w}')d\vec{w}' \\
 &- k(t)L(t, \vec{w})
 \end{aligned} \tag{6.1}$$

Where the term $\alpha(t)L_e(t, \vec{w})$ adds energy due to emission, $\sigma(t) \int_{\Omega} p(t, \vec{w}', \vec{w})L_i(t, \vec{w}')d\vec{w}'$ represents in-scattering events and $k(t)L(t, \vec{w})$ subtracts energy due to absorption and out-scattering. $\alpha(t)$, $\sigma(t)$ and $k(t)$ are the emission, in-scattering and extinction coefficients respectively. We can simplify this equation by assuming a homogeneous medium, therefore the three coefficients become constant values for each differential step of t : α , σ and k . We further consider an isotropic L_i term, therefore $L_i(t, \vec{w}')$ can be reduced to $L_i(t)$. Furthermore, in many cases we can dismiss the term representing the emission of light, as most of the participating media, like fog, do not have light-emitting particles.

Given that the phase function is considered to be isotropic and the medium is homogeneous, we get $p(t, \vec{w}', \vec{w}) = 1/4\pi$, and we can further simplify the in-scattering integral term. Equation 6.1 now can be written as:

$$(\vec{w} \cdot \nabla)L(t, \vec{w}) = \sigma \frac{L_i(t)}{4\pi} \int_{4\pi} d\vec{w}' - kL(t, \vec{w}) \tag{6.2}$$

By integrating the in-scattering term in the whole sphere Ω we substitute it by a constant value $InScat$. This value will be given by the user as a parameter:

$$(\vec{w} \cdot \nabla)L(t, \vec{w}) = \sigma InScat - kL(t, \vec{w}) \quad (6.3)$$

As we are working in image space, we need to integrate the radiance along each ray, thus obtaining:

$$L(x_s, y_s) = \sigma InScat \Delta t + L_0 e^{-k \Delta t} \quad (6.4)$$

where (x_s, y_s) represent pixel coordinates after integration in t , L_0 is the original luminance value without participating media (given by each pixel value in the image) and Δt is the estimated per-pixel depth of the scene (given by the user as discussed in Section 6.3.4). In the following, we show how to simulate in-scattering, out-scattering and extinction phenomena in images by using this simplified equation coupled with some unskilled user input.

6.3.3 Perception of the Natural Process

The visual inspection of images with participating media (see figures 6.2 and 6.3) allows us to identify the main telltale cues that reveal the presence of participating media from a perceptual perspective. In the absence of existing literature on this topic, we propose the following, which will work well enough for our purposes:

- **De-saturation**, contrast reduction and loss of apparent volume. When surrounded by participating media, shadows are softened and colors lose intensity, due to multiple scattering phenomena.
- **Attenuation of highlights** both from light sources and in the objects of the scene, also due to multiple scattering phenomena.
- **Airlight**. Added luminosity due to in-scattering effect originated by light sources located inside or outside of the medium(e.g.: the sun).
- **Extinction** of the original pixel luminosities, due to out-scattering and absorption in the medium.
- **Blur and detail loss** due to multiple scattering.

We now show how our method applies these visual cues to simulate the effects of participating media in an image.

6. APPLICATION 1: LIGHT TRANSPORT IN PARTICIPATING MEDIA. AN IMAGE EDITING APPROACH



Figure 6.2: Some photographs of real participating media, showing combinations of the perceptual cues enumerated in Section 6.3.3.



Figure 6.3: Highlights attenuation due to scattering of light sources.

6.3.4 Image Processing

From the aforementioned observations of the natural processes we derive an image processing pipeline: First the original image is relit as if it were inside the participating media. As we cannot perform an actual relighting of the scene due to unknown geometry and reflectance properties, we simulate its effects of the medium through simple filters: desaturation of colors, contrast reduction, increased luminance in shadows, and highlight attenuation. This is achieved by means of histogram manipulation. Extinction and airlight effects are subsequently simulated by following equation (6.4). Finally, blur and detail loss is simulated by defining a point spread function for the image.

6.3.4.1 Depth estimation

Before going into detail describing the process, we will discuss one of the main user inputs to the image processing pipeline: the approximated depth information of the scene. Given that our input is a single image, we lack any depth information associated to its pixels but the perception of the objects in a scene with participating media is highly dependent on how far they are with respect to the sensor, and thus this depth information needs to be approximated somehow. For our purposes there is no need for great accuracy: the human visual system is not a perfect light meter and thus great discrepancies from a physically accurate solution go undetected (as our psychophysical tests will show). We propose two different approaches, *depth simplification* and *shape from shading*.

Depth simplification: In terms of composition, almost any image can be decomposed in up to three planes: close-up plane, middle plane (optional) and background. Therefore, an user-made segmentation of the scene is enough to create a discrete depth map, capable of creating visually plausible depth perceptions in our media simulations. However, in certain cases perspective issues make this discrete plane segmentation not good enough (a ground receding into the distance, for instance). In those cases it is necessary to create a smooth depth gradient instead. We generate those gradients by hand simply by dragging the mouse over the region of interest and defining a perspective plane as in the work by Oh and colleagues (OCDD01), as illustrated by Figure 6.4.

Shape from shading: When the surface of an object is too complex for a depth simplification, *Shape From Shading* (SFS) techniques could be used (ZTCS99; EP06) to recover the shape of an object in an image by analyzing shading variations across its projected surface.

As it is discussed in (KRFB06), the drawback of these algorithms is that they are greatly constrained to the conditions where the image was taken (presence of textures, self shadows, highlights,...) and usually show poor results for arbitrary images. Other techniques (Kan98), (OCDD01) depend greatly on the quality and amount of user input to infer depth in the scene.

To avoid these problems, we again leverage the limitations of human perception in order to obtain an approximation which suffices for our purposes. In particular, we follow the approach by Khan et al. (KRFB06), which is based on a surprisingly simple assumption: the brighter a pixel, the closer it is to the camera. Thus, darker values are interpreted as points far from the observer. This is clearly not true for a vast amount of cases, but it has been proved to be one of the basic assumptions of the human visual system (KvDS96), (LB00). We use this SFS-based segmentation approach when required by the complexity of the scene. Although the ideal would be to perform this step without any user input, we find that hand-made segmentation of the regions of interest is still the best option: constraining the users to an algorithm default outcome would reduce artistic criteria, thus limiting its flexibility as an image editing tool. Furthermore, the mental framework of the artists usually includes the concept of layers to separate image areas in depth by its visual importance¹.

¹As confirmed by interviews with artists using our system.

6. APPLICATION 1: LIGHT TRANSPORT IN PARTICIPATING MEDIA. AN IMAGE EDITING APPROACH

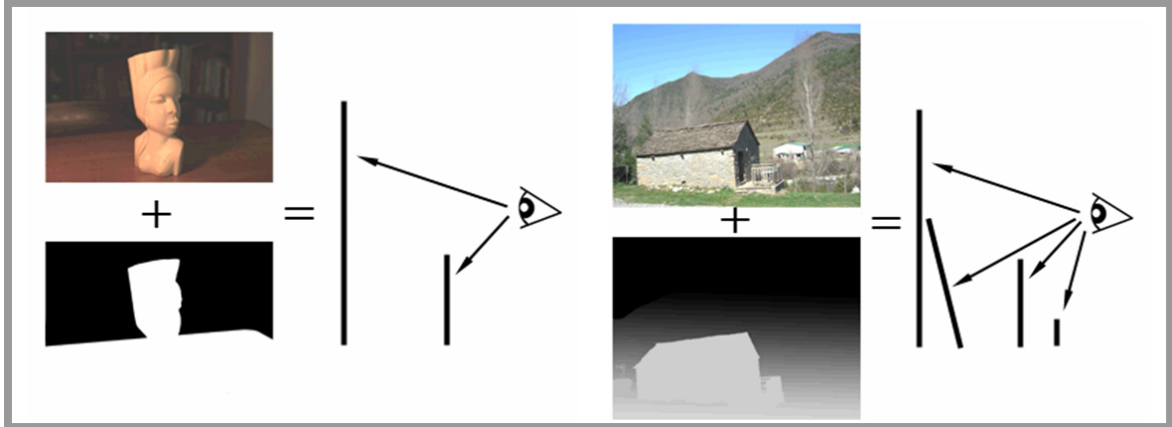


Figure 6.4: Binary depth information (left) and detailed depth information using a multiple level z-buffer image (right).

6.3.4.2 Image Processing Pipeline

The image processing pipeline starts with the transformation of the color space from RGB space to HLS space (Hue-Luminance-Saturation). We subsequently process the image, simulating the light transport from the light sources to the pixels of the image. This is done in a two-steps fashion: First, we detect the highlights by finding the minimum in the derivative of the image histogram. This minimum is usually a reasonable start for a highlight as shown in (KRFB06). Second, we compute per-pixel attenuation of the luminance channel L for the corresponding pixels forming a highlight. Given the high dynamic range of the input image, it is possible to recover the original colors in most of the cases (see Figure 6.5) without the need of more sophisticated methods.

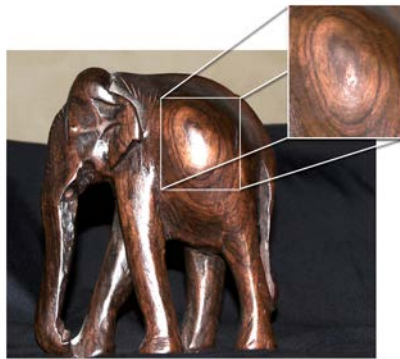


Figure 6.5: Example of highlight attenuation by histogram analysis and manipulation.

Once the highlights have been processed, we simulate the rest of the light transport in the scene by directly increasing the luminance in shadowed areas (defined as pixels below five f-stops in the histogram) and modifying the hue and saturation values in proportion to the σ and k parameters. We

illustrate the process in Algorithm 2:

Data: I_o : Original image, I_f : Final image, Z : z-buffer, σ : attenuation index, $InScat_H$:
 Airlight hue, S_R : Shadow reduction coefficient (0...1, default=0.5)
for each pixel x_s, y_s in the image **do**
 $I_f(x_s, y_s)_S \Leftarrow [I_o(x_s, y_s)_S \cdot (1 - k)];$
 $I_f(x_s, y_s)_H \Leftarrow [I_o(x_s, y_s)_H \cdot (1 - \sigma) + InScat_H \cdot \sigma];$
if $I_o(x_s, y_s)_L < (Max_L - Min_L)/5$ **then**
 $I_f(x_s, y_s)_L \Leftarrow [I_o(x_s, y_s)_L + \frac{(Max_L - Min_L) \cdot S_R}{5}];$
end
end

Algorithm 2: Preprocessed image relighting.

Attenuation due to extinction and out-scattering is computed again manipulating the luminance channel, following the second term of equation (6.4), as shown in Algorithm 3:

Data: I_i : Input image, I_f : Final image, Z : depth z-buffer, K : extinction index
for each pixel x_s, y_s in the image **do**
 $I_f(x_s, y_s)_L \Leftarrow [1 - I_i(x_s, y_s)_L \cdot e^{-K \cdot Z(x_s, y_s)}];$
end

Algorithm 3: Extinction due to out-scattering.

Next, we add the airlight typical of a participating medium, by increasing the luminance and hue channels for each pixel (see Algorithm 4), as suggested in equation (6.4). For implementation purposes, algorithms 3 and 4 are actually computed in the same loop.

Data: I_i : Input image, I_f : Final image, Z : z-buffer, σ : attenuation index, $InScat_L$: airlight
 luminance
for each pixel x_s, y_s in the image **do**
 $I_f(x_s, y_s) \Leftarrow [I_i(x_s, y_s) + \sigma \cdot InScat_L \cdot Z(x_s, y_s)];$
end

Algorithm 4: Airlight due to in-scattering.

We simulate the blur and detail loss due to the multiple scattering of the light by means of an atmospheric point spread function (APSF). The concept of the APSF was first described by Narasimhan and Nayar (NN03a) to simulate the effects of multiple light scattering without the cost of ray tracing techniques. Thus, it can be seen as an extension of traditional point spread functions, which model the response of any optical system in the presence of a point light source. For our functions, we use the same values as described by Narasimhan and Nayar (NN03a), as shown in Figure 6.7. We simulate the characteristic blur of a participating media by applying a convolution of the luminance channel of the image with the APSF.

When multiple levels of depth are present in the scene, we cannot apply the same APSF to all of them, as the kernel size of the function is determined by the very nature of the participating medium

6. APPLICATION 1: LIGHT TRANSPORT IN PARTICIPATING MEDIA. AN IMAGE EDITING APPROACH

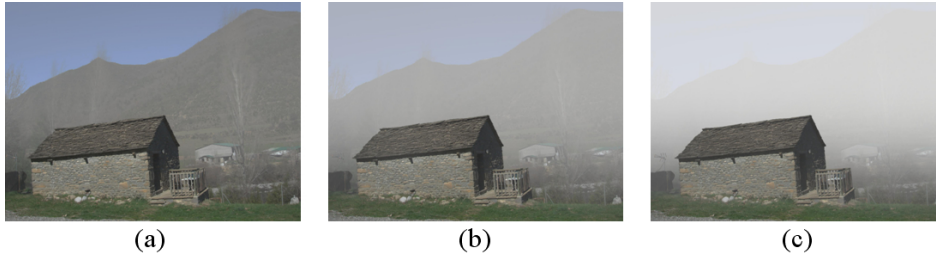


Figure 6.6: Images obtained varying the scattering and absorption coefficients (σ, \mathbf{k}) : (a) = $(0.2, 0.2)$, (b) = $(0.44, 0.5)$ y (c) = $(0.75, 0.9)$.

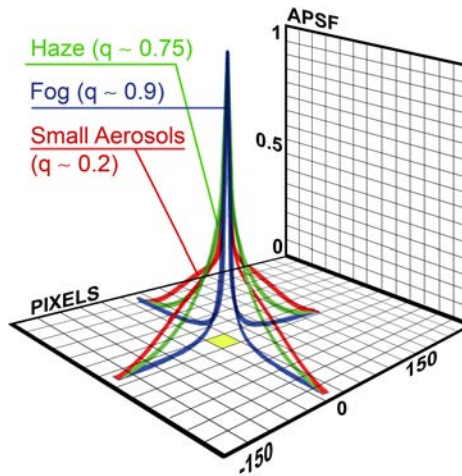


Figure 6.7: Graphical representations of some APSFs. After Nayar et al. (NN03a)

and it is fixed to a determined depth. Therefore we need to resize it according to the user's previous segmentation of the image. We simply interpolate the values of each kernel for intermediate distances as shown in Figure 6.8.

Finally, Perlin noise (Per02) can be added to achieve the visual appearance characteristic of non-homogeneous media. Perlin noise is a procedural pseudo-random noise which takes two parameters as input for 2D images: Period and number of octaves (amplitude is equal to 1 in our case). An octave is the number of noise functions which, when added together, yield the final noise. Each noise function doubles the frequency of its predecessor (see fig. 6.9). Perlin noise divides the image into a grid with the size of the cell side equal to *period*. If the pixel corresponds to a vertex of the grid, the value of the noise function is returned. Otherwise, an interpolation of its four-neighbors is performed. Then for each octave the process is repeated dividing the period by two. The final Perlin noise image is used as a density function for the participating media simulation. To summarize, the user input required by our method is as follows:

- **Approximate Z-buffer:** depth information for each pixel.

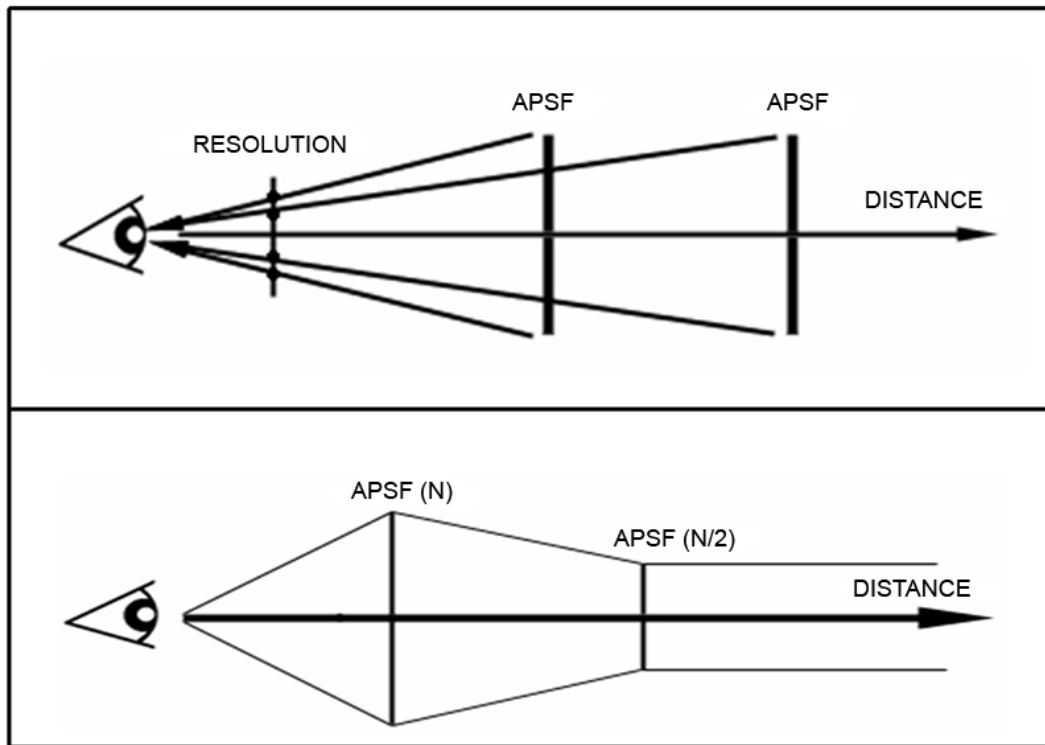


Figure 6.8: We can see how maintaining the kernel size of the APSF constant requires decreasing the size of the kernel of its projection in the screen(top). We interpolate between decreasing kernel sizes at three different depths.(bottom)

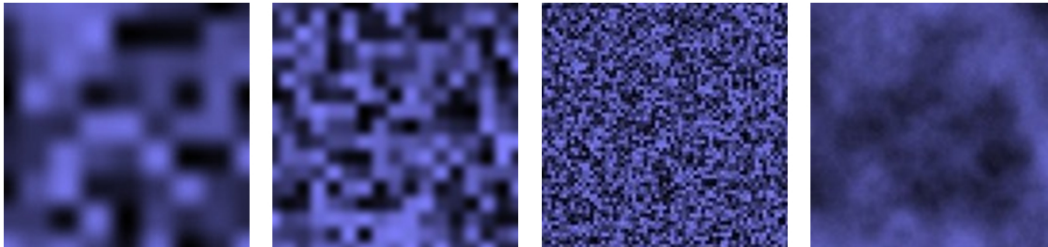


Figure 6.9: Noise functions in 2D with different frequency and amplitude. The last one is the total sum: Perlin noise.

- **Scattering** and **absorption** coefficients, which characterize the medium.
- ***InScat* value**: which defines airlight intensity.
- **APSF Threshold**: Used to divide the depth of the scene in three regions, as in Figure 6.4.
- **Perlin noise parameters**: Octaves and period. Used to configure the number of signals to shape the noise function.

6.4 Validation

Our motivation is to develop a method capable of computing a visually plausible simulation of participating media. As there is no accurate method to measure the degree of realism achieved by our system, we rely on the psychophysical analysis of the perceptions induced by our results. First (Section 6.4.1), we add participating media to a photograph with our method (we restrict ourselves to fog); then we ask the participants to recreate similar media in other images, using commercial image editing software (we use PhotoshopTM). Second (Section 6.4.2), we compare the outcome of our system with the artists renditions from a perceptual point of view.

6.4.1 Adding participating media

6 individuals took part in this part of the experiment. Two of them, from now on called 'average users', had low or medium-low expertise with the tool. The remaining four, henceforth called 'artists' had a great knowledge of both the tool and its potential applications.



Figure 6.10: (a) Original image used by participants as input. (b) Image with fog computed by our method, used as sample.

The participants were asked to work with a clean image (fig. 6.10 (a)), and manipulate it so that it looked as if contained the same kind of fog as the image generated by our method (fig. 3.1 (b)). They were instructed that the meaning of *same kind of fog* is subjective, in order to avoid hindering the artistic expression. The participants were given unlimited time to finish the images. The images generated by the participants are shown in Figure 6.12. These images, together with the result of our method (using the same parameters for the fog as in the sample image; See fig. 6.11) are be used as input for the psychophysical test. If we analyze Figure 6.12 qualitatively, we can observe that both average users (u1, u2) have not considered how depth affects the opacity and luminance of the fog. Furthermore (u2) has tried to simulate the non homogeneity of the medium adding a disproportionate

amount of noise. The artists (a1, a2, a3, a4) have taken into account depth attenuation effects, although some seem to have added too much noise to the fog (a1,a2). (a3) created the closer outcome to our image although extinction effects are not visible, producing a global over-illumination of the image. Finally (u4) overexposes certain areas too much, obtaining a very artificial finish.



Figure 6.11: Result generated by our model.



Figure 6.12: Images created by average users (u1,u2) and artists (a1...a4)

In Figure 6.13 we show the times needed by each participant and by our method. The data show that our model generates results in less than a quarter of the time needed in average by any user. Furthermore, our model does take into account all the telltale visual cues from participating media, whereas as we have seen, most of the participants failed to capture at least some of them (qualitatively speaking).

6. APPLICATION 1: LIGHT TRANSPORT IN PARTICIPATING MEDIA. AN IMAGE EDITING APPROACH



Figure 6.13: Images created by average users (u1,u2) and artists (a1...a4)

6.4.2 Psychophysical test

Once we have shown that our algorithm produces faster results, we now want to shed some light on two other important questions: does our simulation look real? And, is it comparable to what an artist's rendition? The image set used in the test is composed by the five best renditions of the previous step (we discarded the image by (u2)) and the outcome of our model (image 4 in Figure 6.14). A gender-balanced total of 20 individuals took part in the experiment, all of them having reported normal or corrected-to-normal vision. About half of them had previous knowledge of computer graphics in general.

The images were shown in a 22" LCD DELL monitor. The test had two parts. First, each individual was exposed to a random sequence of the 6 images (fig. 6.14) without any possible user interaction. They were simply asked the following question:

"Please indicate if the fog in the image corresponds to a real photograph or if it was digitally processed."

Participants could give only a yes-or-no answer. The time to observe and answer was limited to 20 seconds per image.

Although useful to detect preference trends in the participants this question might introduce a bias. In order to disambiguate and measure this degree of preference we performed a second experiment showing the 6 images at the same time (a 'stimulus sextuple') while asking the participants to rank them (1: less realistic,... 6: most realistic), as suggested in (MMS06). The display was the same as in the previous experiment, but no time limits were imposed for this task.

The results are shown in Figure 6.15. (a) shows that the result from our algorithm has the highest ratio of high scores (5 or 6 on a 6-point scale) assigned by the users (52,94%). (b) shows the average scores, where our algorithmic result competes with images 3 and 5 (but has been generated at a fraction of the time as shown before). In particular, our image obtained an average score of 4, only 0.05 points below Image 3 and 0.05 points over Image 5. Finally, (c) shows that 70,59% of the

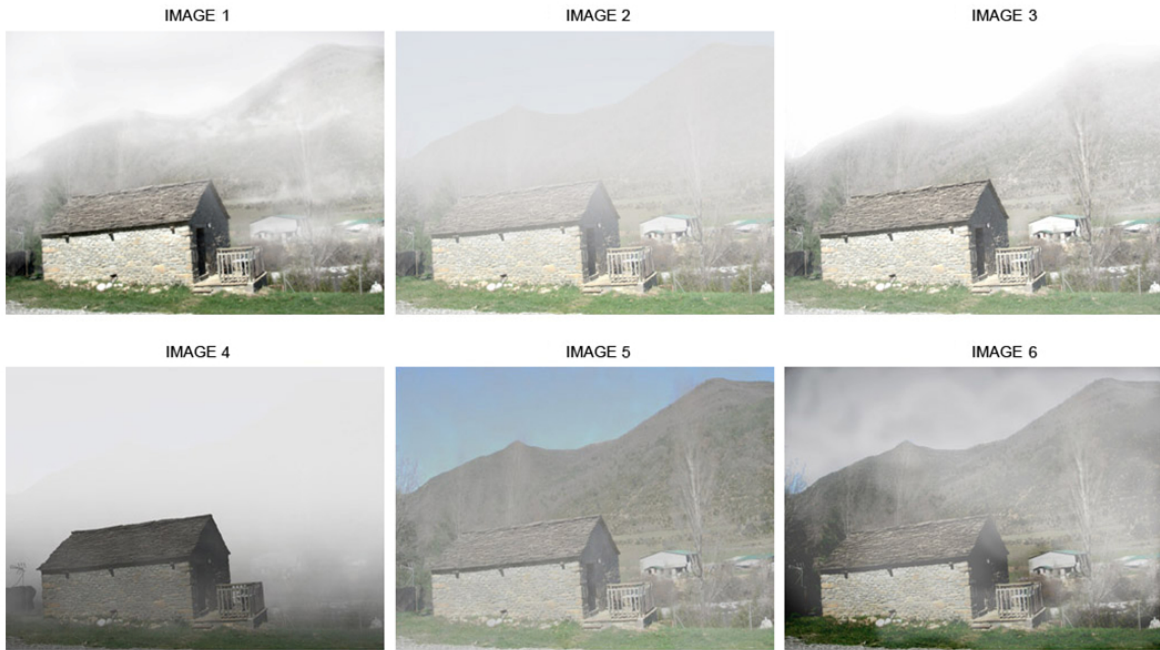


Figure 6.14: Images shown in the test.

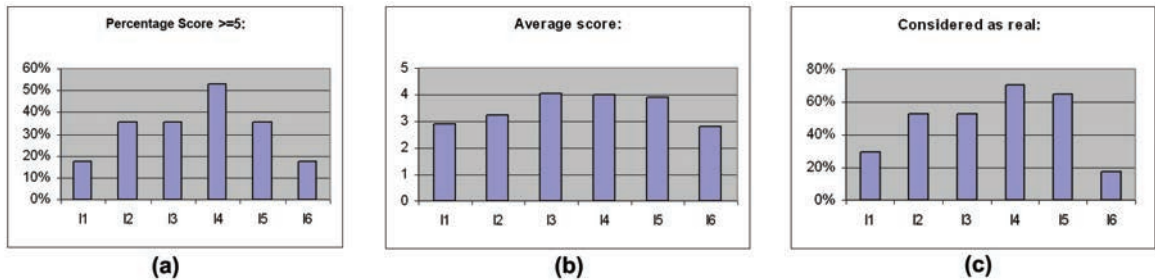


Figure 6.15: Graphs representing: (a) Percentage of participants evaluating the image with high scores (5 or 6). (b) Average score assigned to each image (1..6). (c) Percentage of individuals who considered the image as real. participants perceived our result as a real, photographed scene without digital editing, a percentage not equaled by any of the artists' renditions.

6.5 Conclusions and Future Work

In this chapter have presented an image based method to simulate plausible participating media in 2D images, using an HDR image as input. The underconstrained nature of the problem is circumvented by means of unskilled user input. We believe the amount of user input is reasonable, given that our results are four times faster than the average artist's time using a conventional, image editing tool. We show our results on some images and the parameters used in Figure 6.16 and Table 6.1.

6. APPLICATION 1: LIGHT TRANSPORT IN PARTICIPATING MEDIA. AN IMAGE EDITING APPROACH



Figure 6.16: First column: Original images. Second, third and fourth columns: Processed images using the parameters shown in table 6.1.

Table 6.1: Parameter data of images shown in figure 6.16. $InScat_H$ is constant.

Image	Parameters					
	k	σ	$InScat_L$	$InScat_S$	Octaves	Period
House (left)	1.2	0.8	0.6	0.2	5	250
House (center)	0.6	0.35	0.25	0.2	6	450
House (right)	0.7	0.5	0.5	0.3	3	300
Forest (left)	1.0	0.8	0.4	0.35	6	250
Forest (center)	0.5	0.35	0.15	0.35	5	400
Forest (right)	0.9	0.65	0.5	0.35	4	400
Statue (left)	0.5	0.3	0.01	0.025	6	900
Statue (center)	0.65	0.45	0.15	0.025	4	1500
Statue (right)	0.75	0.5	0.025	0.04	5	750

After analyzing the results of our psychophysical tests, we can state that our model is able to simulate participating media with, at least, the same degree of realism and accuracy as an artist using an image editing tool like PhotoshopTM. Which is more, observers tend to show preference for our image in detriment to artist paintings.

In terms of computing time cost, our model generates an image in less than five minutes (without

GPU support or parallel implementation) against the 20 minutes needed in average by any user. We have to remark that most of the computing time is devoted to automatic processing without user input. Even unskilled users could match artist renditions in a quarter of the time.

Thus far, in terms of time, the amount of user input needed to create the depth maps is the greatest bottleneck of our method. We believe that our method would be highly improved if the users were provided with computational tools in order to set (with feedback) the depth parameters. Some interesting future work lies ahead: The application of our RBF shape form shading (see Chapter 4) could increase the accuracy of depth maps without user intervention and regarding outdoors scenes, Saxena et al. (SCN08) introduced a depth acquisition method based in machine learning through the analysis of multiple sets of images and their corresponding actual depths, obtained by laser scan. In this manner, the system is able to infer a likely depth for each pixel based on its previous experience. Although it lacks of great accuracy, the flexibility and robustness of this method makes it a very good candidate to feed depth information to our algorithm, ameliorating the need for user-guided segmentation of the scene.

**6. APPLICATION 1: LIGHT TRANSPORT IN PARTICIPATING MEDIA. AN
IMAGE EDITING APPROACH**

References

- [Deb98] Paul E. Debevec, *Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography*, SIGGRAPH, 1998, pp. 189–198. 110
- [EP06] Olivier Faugeras Emmanuel Prados, *Handbook of Mathematical Models in Computer Vision*, ch. Shape From Shading, pp. 375–388, Springer, 2006, pp. 375–388. 113
- [Gla95] A. Glassner, *Principles of digital image synthesis*, Morgan Kaufmann, San Francisco, California, 1995. 110
- [Kan98] S. Kang, *Depth painting for image-based rendering applications*, 1998. 113
- [KRFB06] Erum Arif Khan, Erik Reinhard, Roland Fleming, and Heinrich Buelthoff, *Image-based material editing*, ACM Transactions on Graphics, Proceedings of SIGGRAPH 06 **25** (2006), no. 3, 654–663. 108, 109, 113, 114
- [KvDS96] Jan J. Koenderink, Andrea J. van Doorn, and Marigo Stavridi, *Bidirectional reflection distribution function expressed in terms of surface scattering modes*, ECCV '96: Proceedings of the 4th European Conference on Computer Vision-Volume II (London, UK), Springer-Verlag, 1996, pp. 28–39. 113
- [LB00] Michael S. Langer and Heinrich H. Bulthoff, *Depth discrimination from shading under diffuse lighting*, Perception **29** (2000), 649–660. 113
- [LMCG08] Jorge Lopez-Moreno, Angel Cabanes, and Diego Gutierrez, *Image-based participating media*, CEIG 2009, Sep 2008, pp. 179–188. 107
- [MMS06] Rafal Mantiuk, Karol Myszkowski, and Hans-Peter Seidel, *A perceptual framework for contrast processing of high dynamic range images*, ACM Trans. Appl. Percept. **3** (2006), no. 3, 286–308. 120
- [NN01] Srinivasa G Narasimhan and Shree K Nayar, *Removing weather effects from monochrome images*, Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, June 2001, pp. 186 – 193. 109

REFERENCES

- [NN03a] S.G. Narasimhan and S.K. Nayar, *Shedding Light on the Weather*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. I, Jun 2003, pp. 665–672. 109, 115, 116
- [NN03b] Srinivasa G Narasimhan and Shree Nayar, *Interactive deweathering of an image using physical models*, IEEE Workshop on Color and Photometric Methods in Computer Vision, In Conjunction with ICCV, October 2003. 109
- [OCDD01] Byong Mok Oh, Max Chen, Julie Dorsey, and Frédo Durand, *Image-based modeling and photo editing*, SIGGRAPH 2001, Computer Graphics Proceedings (Eugene Fiume, ed.), ACM, 2001, pp. 433–442. 113
- [Per02] Ken Perlin, *Improving noise*, ACM Trans. Graph. **21** (2002), 681–682. 116
- [PPS97] Frederic Perez, Xavier Pueyo, and Francois X. Sillion, *Global illumination techniques for the simulation of participating media*, Rendering Techniques '97 (Proceedings of the 8th Eurographics Workshop on Rendering) (NY) (Julie Dorsey and Phillip Slusallek, eds.), Springer Wien, 1997, pp. 309–320. 108
- [RFBW07] Ganesh Ramanarayanan, James Ferwerda, Bruce Walter, and Kavita Bala, *Visual equivalence: towards a new standard for image fidelity*, SIGGRAPH '07: ACM SIGGRAPH 2007 papers (NY, USA), ACM, 2007, p. 76. 108
- [SCN08] Ashutosh Saxena, Sung H. Chung, and Andrew Y. Ng, *3-d depth reconstruction from a single still image*, Int. J. Comput. Vision **76** (2008), no. 1, 53–69. 123
- [SGA⁺07] V. Sundstedt, D. Gutierrez, O. Anson, F. Banterle, and A. Chalmers, *Perceptual rendering of participating media*, ACM Transactions on Applied Perception **4** (2007), no. 3, 1–22. 109
- [SKSU05] László Szirmay-Kalos, Mateu Sbert, and Tamás Umenhoffer, *Real-time multiple scattering in participating media with illumination networks*, Rendering Techniques, 2005, pp. 277–282. 107
- [SNN01] Yoav Y Schechner, Srinivasa G Narasimhan, and Shree K Nayar, *Instant dehazing of images using polarization*, Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, June 2001, pp. 325 – 332. 109
- [ZTCS99] Ruo Zhang, Ping-Sing Tsai, James Edwin Cryer, and Mubarak Shah, *Shape from shading: A survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence **21** (1999), no. 8, 690–706. 113

Chapter 7

Application 2: Procedural caustics

In this chapter we introduce a novel algorithm to alter the light transport (in the form of caustics) on the basis of a single image. We show that for simple geometric configurations the caustics obtained with our algorithm are perceptually equivalent to the physically correct solution. Our results are validated by means of psychophysical tests, comparing our renderings both with ground-truth, photon-mapped caustics and renditions produced by professional artists. This research was presented at the international conference (ERA Rank: A) *Siggraph Asia 2008* and published (GLMF⁺08) in the *Transactions on Graphics* journal, indexed first of 86 journals at the JCR Software Engineering list.

7.1 Introduction

It is a well-known fact that the human visual system is not a simple linear light meter. By taking advantage of this fact, in graphics applications we can sometimes get away with imperfect simulations. The challenge is to understand what type of inaccuracies tend to go unnoticed, and which ones are easily spotted. We are interested in extending the set of tools available to artists to effect high level changes in single images, at much reduced labor costs, compared with painstakingly painting over all pixels. We have already seen very interesting advances in this field, such as retexturing objects with arbitrary textures (FH04; ZFGH05; FH06), creating translucent materials or objects re-rendered with arbitrary BRDFs (KRFB06), or image editing in general (OCDD01). We focus on altering light transport on the basis of a single image which, to our knowledge, has not been attempted before.

We specifically consider the effect some extreme material edits have on their environment and on human visual perception. In particular, changing an object to transparent during an image edit would have an effect on light transport: nearby diffuse surfaces would exhibit caustics. While their exact calculation is expensive, several approaches exist to approximate the solution and obtain faster

7. APPLICATION 2: PROCEDURAL CAUSTICS

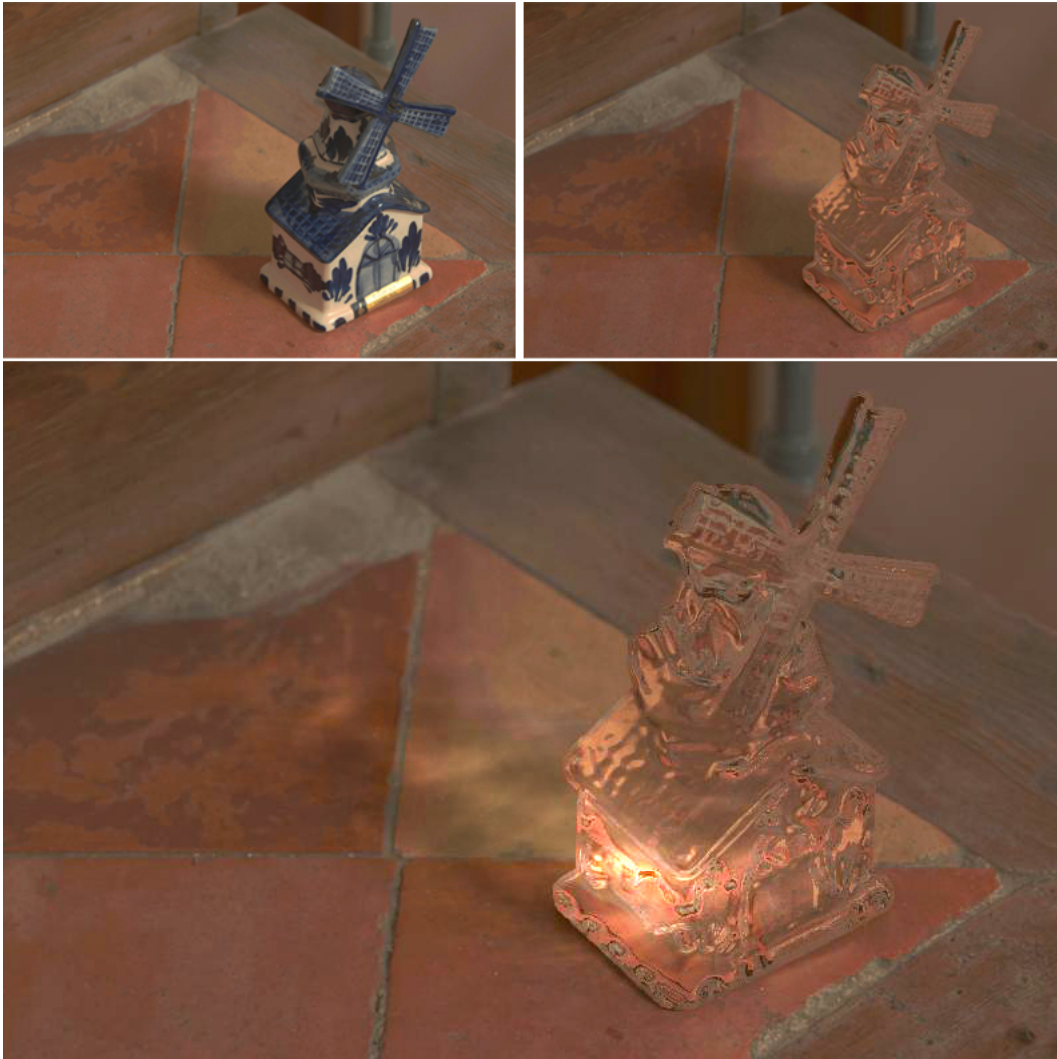


Figure 7.1: Example of light transport editing. Top left, original image. Top right, transparent mill following the approach in (KRFB06). Notice the absence of caustics. Bottom: final result, with caustics added with our algorithm.

frame rates, usually taking advantage of the GPU (SKALP05; SKP06; Wym05; Wym07). Most of the existing approaches, however, build a caustics map in 3D space, where the geometry of the objects and the position of the light sources are known. Such caustics maps are generally computed in three distinct steps (Wym08). In the first step, photons are emitted from the light source, passed through transparent objects, and deposited onto non-transparent surfaces. The second step then uses these photon locations as point primitives, rendering them into the caustic map. The third step projects the caustic map onto the scene. Several different variations have been proposed, including minimizing the number of photons (SKALP05), efficient schemes to collect photons in a caustic map (WD06),

or computing caustics for each object, rather than for each light source (WK07). Various techniques which improve quality and performance are also known (KBW06; WD08; Wym08).

In this work we limit ourselves to the more difficult case of single-image inputs. To effectively simulate plausible caustics, the challenge lies in the fact that 3D shape will have to be estimated from the image itself, an inherently under-constrained problem. While multi-camera and video-based solutions would enable us to extract depth more accurately, we envisage our algorithms to find utility in image editing programs such as PhotoshopTM.

To account for the reduced accuracy with which we can estimate the geometry of the environment depicted in the image, we rely heavily on the limitations of human visual perception. By means of a psychophysical study, we show that while humans are adept at detecting caustics, they are very inaccurate at predicting their shape. We therefore follow the rationale that perceptually plausible rather than physically accurate solutions are both desired and sufficient in our case.

The contributions of this thesis about this topic are as follows. First, we introduce a novel algorithm that can produce light transport edits on a single image, in the form of caustics. We show that for simple geometric configurations the caustics obtained with our algorithm are perceptually equivalent to the physically correct solution. Second, with the aid of psychophysics we show that for more complex objects our algorithm produces caustics that are perceived as perceptually equivalent to ground-truth, photon-mapped caustics. Third, we demonstrate that our caustics are on par with output produced by professional artists, but at a fraction of the time.

In the following, we outline the reasoning behind our approach in Section 7.2. Our algorithm is then described in Section 7.3, with results shown and validated in Sections 7.4 and 7.5. Conclusions are drawn in Section 7.6.

7.2 Motivation

Let us consider a homogeneous transparent object, having a constant index of refraction. Since light propagation at our scale of interest is rectilinear, the occurrence of caustics is determined by the shape of the refracting geometry and the placement of light sources. A narrow beam of rays may enter and exit a transparent volume at points P_1 and P_2 , causing refraction according to Snell's law.

Assuming that the dielectric boundaries at entry and exit points (P_1 and P_2) are locally smooth, we may view this pair of surface areas to constitute a small segment of a thick lens. Dependent on the orientation of the surface normals at P_1 and P_2 , the lens segment will be either converging or diverging according to a limited number of configurations¹.

¹The three possible converging lenses are biconvex, plano-convex and concave-convex; the three possible diverging lenses are biconcave, plano-concave and convex-concave (BW99).

7. APPLICATION 2: PROCEDURAL CAUSTICS

Similarly, each pair of surface points on the transparent object forms a separate segment of a thick lens. If the local curvature around surface points is consistent with the global curvature, then all surface points form part of the *same* thick lens, resulting in a very simple caustic (see the real sphere in Figure 7.5). In the limit the global curvature is identical to that of a thick lens.

Conversely, with increasing complexity of surface curvature, the object will cease to resemble a single lens, but can be thought of as a collection of segments belonging to a set of different thick lenses (Figure 7.2, left). The number of thick lenses that together would create the same caustic as the object itself, is indicative of the complexity of the caustic. However, we treat here a heavily under-constrained problem, with only the approximate shape of the camera-facing surface of the object available to us (Section 7.3.1). As a consequence, we have no knowledge of the back-facing surface. Nevertheless, Khan et al (KRFB06) showed that this has little influence on the identification as a transparent object. We assume that this result extends to caustic rendering (an assumption further backed by our psychophysical analysis in Section 7.5), and therefore ignore the backface in preference of analyzing the frontface of the object only. Thus, we simplify our thick lens approach and interpret the recovered surface as a collection of *thin* lens segments, which refract incoming light and thus generate caustics (Figure 7.2, right).

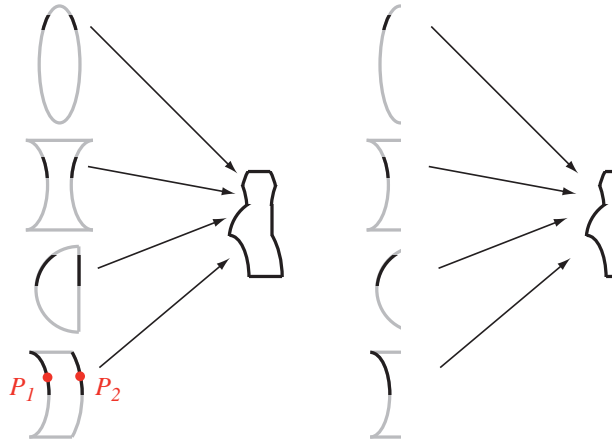


Figure 7.2: Left: a simple object constructed from thick lens segments. Right: our thin lens simplification.

A convex thin lens is circularly symmetric, which gives rise to light being focused at a single point, as shown in Figure 7.3 (left). If the symmetry were broken, for instance by replacing the thin lens with an arbitrary surface, then the amount of residual symmetry would determine how much light is focused along the line of interest, shown in Figure 7.3 (right), while the remainder of the light diverges into different directions. This is similar to how photons would be refracted by the surface, distributing their energy along the line of interest; in a photon-mapping approach, caustics would then be obtained by estimating radiance. In our method, we obtain a map representing the caustic pattern that would be cast by an object by computing the amount of symmetry present for each point of that object.

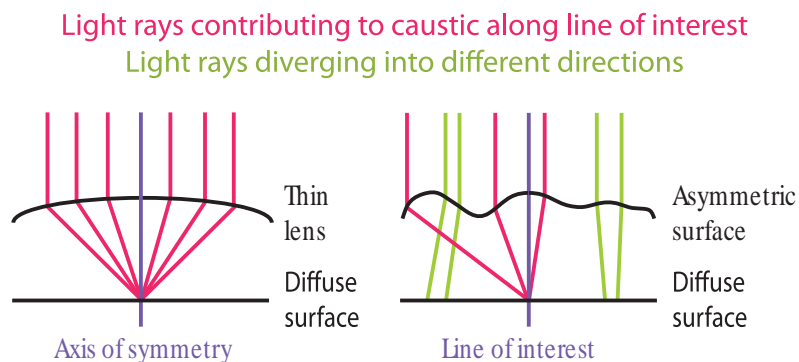


Figure 7.3: Perfect symmetry of a theoretical thin lens (left) causes light to converge at the focal point, where a diffuse surface is placed. If the lens were replaced with an arbitrary surface (right), the residual symmetry in the line of interest will contribute to a caustic at the same focal point.

Ideally, we would like to detect symmetry with respect to the position of the light source. However, with only one image at our disposal, we are limited to detecting the degree of symmetry with respect to the viewpoint. For a spherically symmetric object our approach will therefore be accurate, while for asymmetric objects the physical error could be large. However, we demonstrate in Section 7.5 that perceptual equivalence can be maintained even for large discrepancies between the camera and the light positions. We speculate that this is due in part to humans' inability to predict the shape of both caustics and light directions (tPP05).

Various techniques exist to detect symmetry in images. Morphological approaches such as median-axis transformation or thinning can only be applied to binary images, and the outlines of the object usually need to be smoothed. Intensity gradients tend to be sensitive to contrast in addition to geometry (see (Tyl96) for a review). We are interested in finding a robust measure which requires no previous knowledge or pre-processing of the image. We find such measure in the frequency domain, where local geometric symmetries can be found in an image by analyzing its phase information (Kov97; WY05; XHMW05).

Phase symmetry appears to play a role in human vision, which perceives features at points where the phase information is highly ordered (MB88; WBG06), potentially pre-attentively enhancing the recognition and reconstruction of shapes and objects (Wag95; Zab93). Phase symmetry is also used in computer applications ranging from segmentation (Ros86) and feature detection (Kov96; YS05) to image understanding (OL81; PC82). On this basis, we argue that phase symmetry may help simulate plausible caustics. The results of our psychophysics tests in Section 7.5 confirm that this is a viable approach.

7.3 Simulating Caustics

The problem of adding a caustic to an image can be split into several stages. First, the image is preprocessed to obtain a depth map, serving as a rough representation of the object’s geometry. Second, the recovered geometry is analyzed to establish likely caustic patterns that such an object may cast. As previously mentioned, this analysis takes the form of symmetry detection, for which we employ an algorithm that works in frequency space and makes minimal assumptions on its input. Finally, the luminance channel of the image is varied according to the projected caustic patterns. These steps are discussed in the following sub-sections.

7.3.1 Depth Recovery

Given that global illumination is an inherently three-dimensional process, we must first approximate the 3D object depicted in the image. We rely on the depth-map recovery algorithm by Khan et al (KRFB06). Depth recovery starts by applying a bilateral filter (TM98) to the luminance values of the object’s pixels, obtaining the signal $D(x, y)$. This signal is then reshaped to produce the final depth values (KRFB06) (for additional details please see Chapter 4.

This approach is based on the idea of "dark-is-deep" which can be seen as one (of possibly several) components of human depth perception (LB00). We demonstrate here that it can also be used to produce procedural, perceptually-plausible caustics, relying on two key insights. First, we will produce a caustic from the perspective of the view-point, given that this is the only view available from a single image. While physically inaccurate, statistical symmetries of the transparent object ensure that for our purposes, in most cases this is a reasonable approximation. Second, with this approach, this depth map is both created and used from the same perspective, so that systematic errors introduced by the depth extraction algorithm do not become perceptually distracting.

7.3.2 Phase Symmetry

To detect symmetries in the recovered depth map, we follow the approach of Kovesei (Kov96; Kov97), which has the desirable property that no assumptions on the input are required. However, while Kovesei uses the intensity values of the image as input, thus providing a low-level view of symmetry, we use the depth map instead. This allows us to identify higher level structures based on the recovered geometry. The phase of the depth map at each location is obtained by decomposing it into its different frequency components: we convolve it by even-symmetric (sine) and odd-symmetric (cosine) wavelet filters operating at different scales. We use log Gabor filters, which have the desirable property of having a Gaussian transfer function on the logarithmic frequency scale, consistent with the characteristics of our visual system. Symmetry appears as large absolute values of the even-symmetric filter and small

absolute values of the odd-symmetric filter (Kov97). A weighted average combines information over multiple scales n and multiple orientations θ_i , yielding the following symmetry map $S(x, y)$:

$$S(x, y) = \frac{\sum_i \sum_n [A_{n,\theta_i}(x, y) B - T_{\theta_i}]}{\sum_i \sum_n A_{n,\theta_i}(x, y) + \epsilon} \quad (7.1a)$$

$$B = |\cos(\Theta_{n,\theta_i}(x, y))| - |\sin(\Theta_{n,\theta_i}(x, y))| \quad (7.1b)$$

where A and Θ represent amplitude and phase respectively and T is an estimate of the signal noise. Details of the implementation are provided in the appendix.

The two parameters in this equation are the angular interval between filter orientations θ_i (which defines the number of directions d where symmetry is searched for) and the angular spread of each filter (which is a Gaussian with respect to the polar angle around the center). Ideally, we seek the minimal necessary angular overlap to achieve approximately even spectral coverage (Kov99); angular overlap is given by the ratio of the angular interval between filter orientations and the standard deviation of the angular Gaussian function used to construct filters in the frequency plane θ/σ_θ . Our experience indicates that good results are achieved with $\theta/\sigma_\theta = 1.2$, which is the value used for all the images in this chapter. The number of directions d varies between 1 and 20 (see Table 7.4), and is the only user-defined parameter of the symmetry detection algorithm. Direction $d = 1$ is defined as the direction yielding the highest symmetry for a given object, for which an initial search is performed at one-degree increments over the full angular space, a process that takes only a few seconds. Successive directions specified by the user are then defined according to this reference.

Intuitively, increasing the number of search directions will create a progressively more complex pattern, given that more symmetries will be detected, thus yielding more complex combined patterns. The degree to which this happens depends on the geometrical complexity of the object. Very simple objects like the sphere in Figure 7.5 are relatively invariant to changes in d , but the resulting caustics are very similar to the physically-correct ones. The influence of d on more complex objects will be analyzed in Section 7.4.

7.3.3 Luminance Adjustment

To apply the caustic map $S(x, y)$, we first obtain its projection $S'(x, y)$ onto a user-defined quadrilateral projection area. This is achieved by means of a simple perspective transform. In general, shadows cast by the opaque object provide a reasonable first indicator of a suitable quadrilateral projection region (see Figure 7.5, left and middle).



Figure 7.4: From left to right: segmented mill from Figure 7.1, recovered depth map (KRFB06) and two maps with 1 and 20 orientations respectively.

By analysing the silhouette of the shadow, in combination with the silhouette of the shadow-casting object, it may be possible to infer the orientation of the underlying plane. However, we are not aware of robust solutions to this problem. Moreover, in the case of non-planar surfaces, further depth map extraction would be required to determine how the caustic map should be projected.

To avoid these complications, we assume the caustic to be mapped onto a planar surface, adopting a simpler user-assisted approach similar to Mohan et al’s (MTC07), whereby the user specifies the vertices of the projection region by just clicking four points located approximately around the shadow region. An additional advantage to this solution is that the user implicitly and naturally accounts for the fact that the transparent object may be some distance away from the surface that exhibits the caustic.

We then modify the original image according to the following operation on the luminance channel:

$$L_c(x, y) = L(x, y) + \alpha S'(x, y) \quad (7.2)$$

where α represents a weighting factor to control its apparent brightness, and $L_c(x, y)$ is the luminance channel of the final image (see Figure 7.5 (right)).

7.4 Results

The choice of the number of search directions in the phase symmetry has an impact on the appearance of the resulting caustic, as shown in Figure 7.6. Fewer directions in general yield simpler, more focused

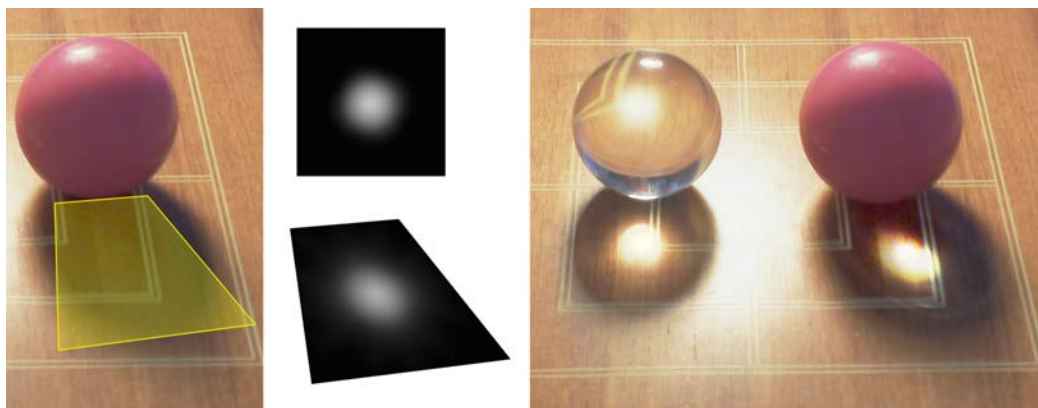


Figure 7.5: From left to right: Detail of the original picture, with user-defined projection area. Original focused caustic, and its projected version. Final result, shown next to a real transparent sphere for comparison purposes.

caustics, whereas increasing the number of directions creates more complex patterns. Note that the apparent degree of sharpness in the mapped caustics w.r.t. the number of directions analyzed depends on the specific object and the corresponding ratio defining $S(x)$ in Equation 7.8. Usually, it is desirable to have a mixture of both focused and complex patterns to better simulate the appearance of real-world caustics. Several caustics maps can be combined in those cases using:

$$L_c(x, y) = L(x, y) + \sum_i \alpha_i S'_i(x, y) \quad (7.3)$$

However, our experiments revealed that combining up to two symmetry maps usually suffices in producing plausible imagery. Table 7.4 shows the number of caustics maps and directions d for each image in this chapter.

Object	Maps	d_1	d_2	Object	Maps	d_1	d_2
Mill	2	1	20	Phone	2	1	4
Can	1	2		Sphere	1	4	
Horse	1	4		Skull	2	4	20
Elephant	1	20		Vertebrae	2	4	20
Vase	2	1	12	Dolphin	2	4	20
Doll	2	1	12	Bull	2	4	20
Car	2	1	12				

Table 7.1: Number of caustics maps and directions d for the images in the chapter.

Figure 7.7 shows three real-world objects and their caustics computed with our algorithm. The real objects have not been changed to transparent for demonstration purposes. It can be seen that, for simple objects such as the soda can, the algorithm yields results very similar to those obtained in real life (as in the case of the sphere in Figure 7.5 and the vase in Figure 7.8 (left)). As the object

7. APPLICATION 2: PROCEDURAL CAUSTICS



Figure 7.6: The influence of the number of directions. From left to right, caustics obtained searching for symmetries in 1, 2, 4, 12 and 20 directions respectively. The complexity of the caustic pattern increases accordingly.

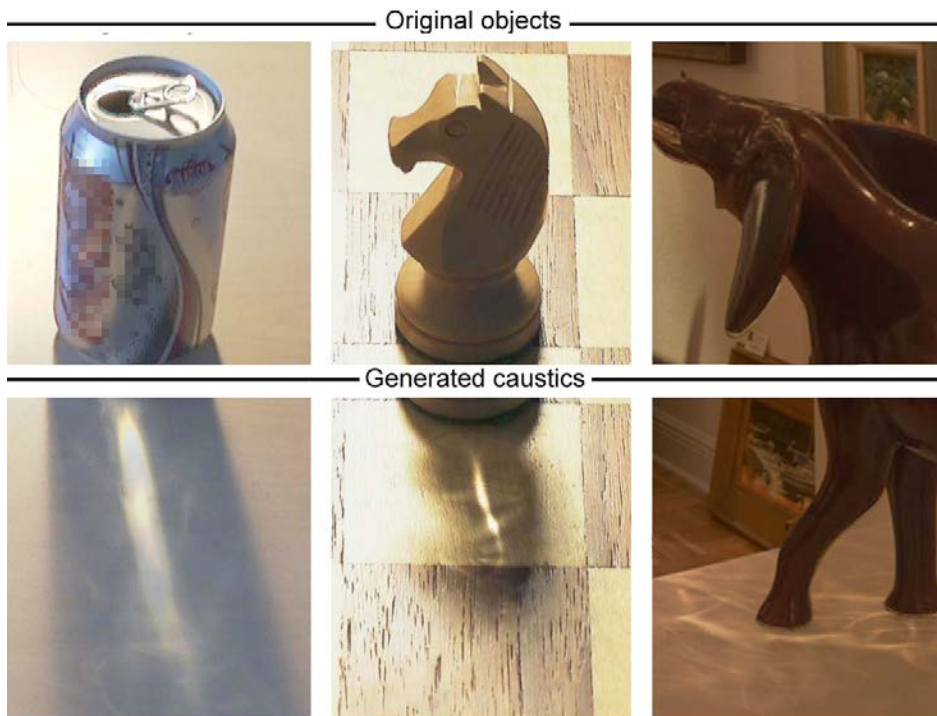


Figure 7.7: Real objects with the caustics obtained with our algorithm. For simple objects such as the soda can, the caustics obtained accurately resemble those that would occur in real transparent objects. For more complicated objects, it starts diverging from the real solution but still produces plausible results.

becomes progressively more complex, like the chess piece and the elephant figurine, the caustics become more complicated and less predictable for an observer. Nonetheless, the caustics produced by our algorithm continue to be commensurate with the expected visual complexity, thereby remaining plausible (Figure 7.8 (right)). This will be validated by means of psychophysical studies in Section 7.5, while further results are shown in Figure 7.9.



Figure 7.8: Two full results, showing transparent objects casting caustic patterns near their base (transparency achieved using (KRFB06)). The shape of the caustic for the vase is relatively simple due to the high degree of symmetry of the object, whereas for the elephant is more complex. Both produce perceptually plausible results. Insets: original images.

7.5 Psychophysics

We claim that the human visual system cannot reliably predict caustics for relatively complex geometries. A very simple test suggests that this is so: Figure 7.10 shows two images of crystal figurines. One image has photon-mapped caustics, which we take as ground-truth; the other has caustics painted by a digital artist. We then asked 36 participants which one they thought was real. Even though both images present clear differences in the shape and concentration of caustics, none was chosen above chance: 17 people chose the photon-mapped image, compared to 19 people who chose the artist's impression.

Does our algorithm perform as well as this artist? To find out, we performed two experiments, described below. The first assesses the level of tolerance that humans exhibit with respect to errors in caustics, while supporting our choice of algorithm to simulate them. The second experiment is then a ranking of our algorithm against several images on which artists have painted their impression of caustics. We have taken this specific approach since the only way to produce caustics in existing images is currently by painting over pixels.

A set of 44 participants took part in our first study, and 87 different observers partook in the second, all of them having reported normal or corrected to normal vision. They were naïve as to the design and goals of the experiments, and included computer graphics graduate students as well as non-experts.

7.5.1 Experiment 1: Validation against 3D Rendering

In this experiment, the first question answered is whether our algorithm produces images which are visually as plausible as a full 3D photon mapping simulation. For this, we employ four different 3D

7. APPLICATION 2: PROCEDURAL CAUSTICS



Figure 7.9: Additional results adding caustics to the doll, car and phone images.

opaque objects of increasing geometric complexity: skull, vertebrae, dolphin and bull (Figure 7.11). For each one, on the one hand, the algorithm described in this work was applied: phase symmetry was computed in image-space from the opaque renders, then composited into a similar image with a transparent version of the object, thus simulating caustics. Note that no 3D information was used to derive the caustics at this stage. On the other hand, regular photon mapped caustics were rendered for the transparent versions, taken advantage of the true 3D information of the objects. The stimuli were then used in a paired comparisons test.

The second question is whether a simpler algorithm would also produce plausible caustics. If so, then this would indicate that our proposed algorithm is overly complicated, and a simpler solution would suffice. In particular, one might reconstruct approximate geometry from the image, and then render them directly with photon mapping. One of the simplest approaches to generate geometry is to assume that objects are globally convex, thus enabling their silhouettes to be revolved. This approach was added to the paired comparisons test.



Figure 7.10: Computer generated crystal figurines. Left: photon-mapped caustics. Right: caustics painted by an artist.

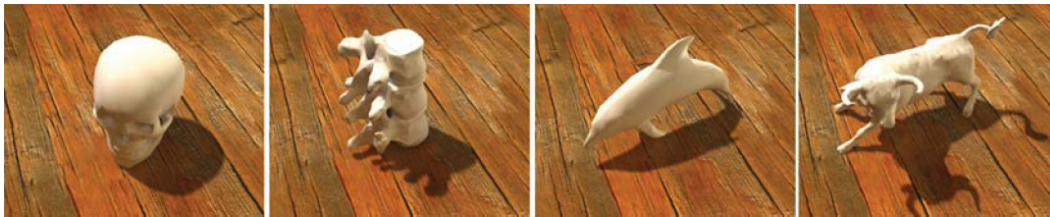


Figure 7.11: The four objects used in our first psychophysical test. From left to right: skull, vertebrae, dolphin and bull.

Finally, we assess whether knowledge of the light direction in the scene is important for constructing a believable caustic. To this end, each stimulus was recreated for 4 different light positions, with one of the light directions coinciding with the viewpoint. This test allows us to determine if the error introduced by our algorithm (it generates the caustic from the viewpoint, rather than from the light source) in any way harms visual impression. Figure 7.12 shows the complete set of stimuli for the skull and bull objects.

For each object and light position, we employed a balanced paired comparison test, for a total of 48 pairs ($4 \text{ scenes} \times 4 \text{ light positions} \times 3 \text{ rendering algorithms}$), shown side-by-side in random order. The display is a calibrated 21" TFT LCD monitor (1800×1600 resolution, 60 Hz refresh rate) with an approximately 150:1 contrast ratio. The participants had to perform a two-alternative forced-choice (2AFC) to answer the question *Which image contains the caustics that look more real to you?*. Upon request, the concept of caustics was explained to each participant individually. All the participants were informed that all the images were computer generated, and that there was not a right or wrong answer. They were also told that the images in each pair were identical except for the caustics. They were previously trained with a different set of images until they felt confident with both the question and the procedure.

As a paired comparisons test is an indirect way to infer a rank order of the three algorithms, it is possible that circular triades occur. For instance a participant may indicate the following preference order: $A_1 > A_2 > A_3 > A_1$, which signifies an inconsistency. The presence of inconsistencies can be

7. APPLICATION 2: PROCEDURAL CAUSTICS

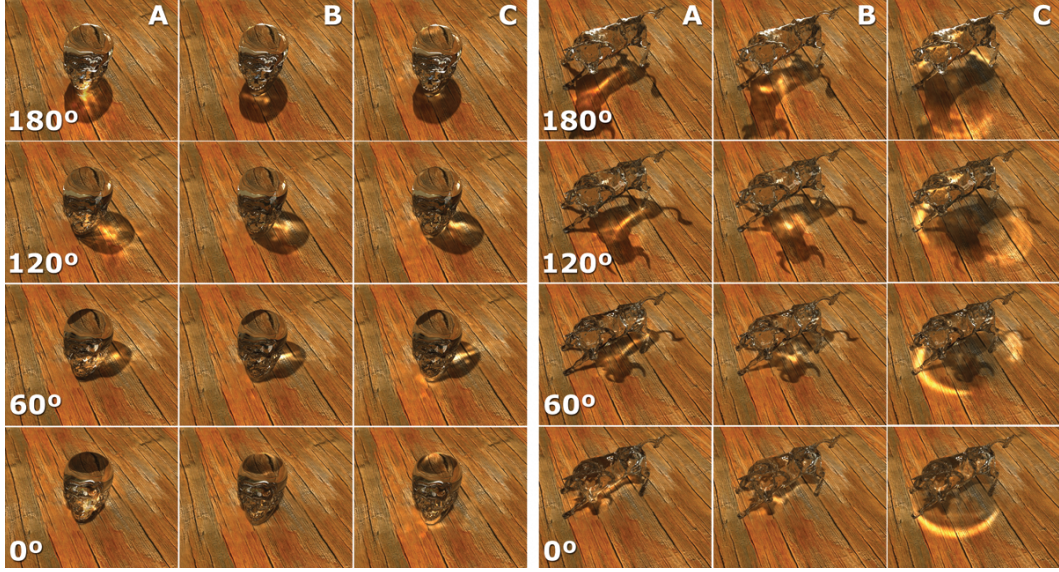


Figure 7.12: The complete set of stimuli for the skull and bull objects. Columns A, B and C show the results of our algorithm, photon mapping and the alternative algorithm respectively. Rows indicate light position (degrees) w.r.t the camera. Details are given in the text.

measured with the coefficient of consistency ξ (KB40). Its value will tend to 1 the more consistent the results are. Values for each scene and for each light direction (angle) are given in Table 7.2, showing that consistency is overall very high.

Scene	ξ	u	Angle	ξ	u
Skull	0.790	-0.068	0	0.903	0.040
Vertebrae	0.903	-0.047	60	0.903	0.044
Dolphin	0.966	0.240	120	0.909	0.021
Bull	0.972	0.249	180	0.914	0.020

Table 7.2: *Coefficient of consistency ξ and coefficient of agreement u per scene and per angle.*

The coefficient of agreement u , also shown in Table 7.2, measures whether the three algorithms received equal preference (low scores) or could be discerned based on preference (high scores). We see that for simple geometries (Skull, Vertebrae), participants found it difficult to indicate a preferred algorithm, whereas complicated geometries, with associated complex caustics, lead to more pronounced viewer preference.

These results are consistent over all angles tested, showing that the position of light sources is of little influence, as evidenced by the low values of u shown on the right side of Table 7.2. We therefore conclude that the error we make by computing the caustic with respect to the viewpoint, rather than with respect to the light source, does not impair our ability to generate a plausible caustic.

Finally, as complicated geometries lead to larger differences in preference ratings, we carried out a significance test of score differences, which allows us to assess which algorithms belong to the same group. Two algorithms belong to different groups if the difference in scores R is below $\lceil R_c \rceil$. Thus, we would like to compute R_c such that:

$$P(R \geq \lceil R_c \rceil) \leq \alpha \quad (7.4)$$

where α is the significance level. It can be shown that in the limit R will be identical to the distribution of the range $W_{t,\alpha}$ of a set of t normally distributed random variables with variance $\sigma = 1$ (Dav88). This enables us to compute R_c using (SL04; LCTS05):

$$P\left(W_{t,\alpha} \geq (2R_c - 0.5)/\sqrt{nt}\right) \quad (7.5)$$

where n is the number of participants (44 in our case) and t is the number of algorithms we compare ($t = 3$). The value of $W_{t,\alpha}$ can be interpolated from tables provided by Pearson and Hartley (PH66). For $\alpha = 0.01$, we find that $W_{3,0.01} \approx 4.125$, so that $\lceil R_c \rceil = 24$. The resulting groupings per scene are given in Table 7.3. At the 0.01 confidence level, our algorithm is always in the same group as the photon mapping approach, and can therefore not be distinguished from the ground truth. For simple geometric shapes this is true also for the method which revolves the silhouette. However, for more complex geometries, this technique is too simple and is reliably distinguished from the ground truth. We therefore conclude that in cases where true 3D geometry is unavailable, our phase symmetry approach can be effectively employed.

Skull:	<table border="1"><tr><td>K</td><td>PM</td><td>R</td></tr></table>	K	PM	R	
K	PM	R			
Vertebrae:	<table border="1"><tr><td>K</td><td>PM</td><td>R</td></tr></table>	K	PM	R	K = Kovesi Phase Symmetry
K	PM	R			
Dolphin:	<table border="1"><tr><td>K</td><td>PM</td><td>R</td></tr></table>	K	PM	R	PM = Photon Mapping
K	PM	R			
Bull:	<table border="1"><tr><td>K</td><td>PM</td><td>R</td></tr></table>	K	PM	R	R = Revolution Method
K	PM	R			

Table 7.3: *Grouping of algorithms per scene.*

This experiment provides insight into our algorithm as compared with a full 3D simulation, showing that the results are visually equivalent. Moreover, for complex geometry an obvious simpler approach falls short, whereas the phase symmetry algorithm continues to produce plausible caustics.

7.5.2 Experiment 2: Validation against Direct Painting

In addition to assessing the performance of our algorithm with respect to 3D rendering, which establishes a ground truth, we are interested whether direct painting using an image editing program (such as Adobe PhotoshopTM) would produce visually comparable results. We expect that the success of direct painting depends on the skill of the artist, as well as the amount of time expended to generate the image.

We therefore asked five digital artists with different backgrounds and styles to paint caustics in two images which were manipulated to create transparency without caustics using Khan et al's

7. APPLICATION 2: PROCEDURAL CAUSTICS



Figure 7.13: Detail of some of the artists' depictions of the caustics for the vases and elephant images (images 1, 2 and 4 for the vases; 2, 3 and 4 for the elephant, as numbered in the tests).

method (KRFB06). One image has a highly symmetric object (a vase) which presumably would yield a symmetric caustic that may be predicted more easily. The other contains an asymmetric object (an elephant figurine) which would produce more complicated caustics. Some of the results are shown in Figure 7.13, whereas the output of our algorithm is given in Figure 7.8. One of the artists failed to deliver the vase image. Each of the eleven resulting images was printed using a professional sublimation printer at 20×15 cm.

Each participant was informed that the only variation between each set of images were the caustics, and was asked to order the images from more to less real (from 1 to 5 in the vase image; 1 to 6 in the elephant image), according to his or her own standards. No previous training was performed, other than an explanation of what caustics are. The order of the images was randomized within each set for each subject.

Since our goal is to determine if our algorithm produces results comparable to what can be achieved by using image-editing software, rank data is sufficient for our analysis. Figure 7.14 shows mean rankings for all the images in each series (lower ranking means higher perceived realism) with $p < 0.05$. Our algorithm performed slightly better than the best of the artists images in the case of the vase series, and significantly better in the elephant series.

Tables 4 and 5 show normal fit data for all images. Our algorithm has the lowest mean (higher perceived realism) of all the tested images. The artists had no time limitations to paint the caustics. They ended up spending between five and fifty minutes to produce the images, while our algorithm

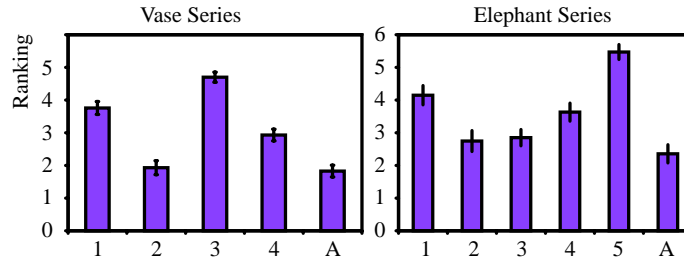


Figure 7.14: Mean intervals for all the images in the vase and elephant series, along with the 0.95 confidence interval.

Image	1	2	3	4	A
Mean	3.759	1.931	4.701	2.931	1.828
StDev	0.939	1.021	0.733	0.860	0.865

Table 7.4: Normal fit data (vase series).

Image	1	2	3	4	5	A
Mean	4.149	2.747	2.851	3.632	5.471	2.356
StDev	1.402	1.527	1.186	1.313	1.087	1.329

Table 7.5: Normal fit data (elephant series).

runs in approximately two minutes for the images shown in this chapter. We therefore conclude that our algorithm produces results significantly faster than an artist, while obviating the need for skilled input. Moreover, our results are perceived to be more realistic than artists' best efforts.

7.6 Conclusions

In this chapter we have demonstrated the feasibility of rendering perceptually plausible caustics into existing images. We have shown that although humans find it easy to detect the presence of caustics, they are much less adept at predicting the shape of caustics. We have leveraged this feature of human vision to produce an image editing tool that enables, for the first time, aspects of global illumination to be simulated on the basis of a single photograph. There are several advantages to this approach. First, the required user input is unskilled, making the algorithm straightforward to apply. Second, the results are at least on a par with those produced by skilled artists, as evidenced by the second validation study reported in this chapter. Third, the time required to render a caustic is only a fraction of the time that a skilled artist would need to paint over all pixels. Our approach could potentially be used in combination with a traditional 3D rendering algorithm, avoiding the need to compute costly caustics and approximating them in image-space. Accurate object depth could be used instead of shape-from-shading information.

7. APPLICATION 2: PROCEDURAL CAUSTICS

Extending this work to video is also possible. For the simplest case of camera movement only, the caustics shape is not expected to change, given that the light is fixed with respect to the object. The projected caustics map for the first frame simply needs to be tracked over successive frames. For more general dynamic scenes with moving objects and/or lights, we can leverage the fact that the shape from shading approach used (from which phase symmetries are obtained) does not introduce temporal artifacts (KRFB06).

7.7 Annex A. Phase symmetry

The phase symmetry algorithm is based on a log Gabor filter bank. We present the phase symmetry algorithm in 1D first, and then show how it is applied to the 2D signal. In 1D, a signal $D(x)$ is convolved by even-symmetric (cosine) wavelet filters M_n^e and odd-symmetric (sine) wavelet filters M_n^o which operate at scale n . The even-symmetric and odd-symmetric responses to such a quadrature pair of filters at scale n is given by $e_n(x)$ and $o_n(x)$ respectively (Kov99):

$$(e_n(x), o_n(x)) = (D(x) \otimes M_n^e, D(x) \otimes M_n^o) \quad (7.6)$$

where \otimes denotes a convolution. Wavelets have a limited spatial extent, which is determined by the chosen scale n . A filter bank analyzing different frequencies can therefore be constructed by repeating this computation for different scales. The $e_n(x)$ and $o_n(x)$ values represent the real and imaginary components of the local frequencies present in the signal around the location of interest x . The amplitude $A_n(x)$ and phase $\Theta_n(x)$ are then given by¹:

$$A_n(x) = \sqrt{e_n^2(x) + o_n^2(x)} \quad (7.7a)$$

$$\Theta_n(x) = \tan^{-1} \left(\frac{e_n(x)}{o_n(x)} \right) \quad (7.7b)$$

Given that symmetry appears as large absolute values of the even-symmetric filter and small absolute values of the odd-symmetric filter, we can subtract both values and produce a weighted average to combine information over multiple scales. This measure of symmetry $S(x)$ corresponds to (Kov97):

$$S(x) = \frac{\sum_n [A_n(x) (|\cos(\Theta_n(x))| - |\sin(\Theta_n(x))|) - T]}{\sum_n A_n(x) + \epsilon} \quad (7.8)$$

Here, ϵ is a small constant to avoid division by zero (we use 0.01), and T is an estimate of the signal noise, and is included to remove spurious responses. This estimate can be computed by first

¹Note that to determine in which quadrant $\Theta_n(x)$ lies, it is effectively computed with $\text{atan2}()$.

considering the energy vector $E(x)$:

$$E(x) = \sqrt{\left(\sum_n e_n(x)\right)^2 + \left(\sum_n o_n(x)\right)^2} \quad (7.9)$$

Assuming that the noise has a Gaussian distribution with random phase and a standard deviation of σ_G , then it can be shown that the noise distribution of the magnitude of the energy vector has a Rayleigh distribution with mean μ_R and variance σ_R^2 given by (Kov99):

$$\mu_R = \sigma_G \sqrt{\frac{\pi}{2}} \quad (7.10a)$$

$$\sigma_R^2 = \frac{4 - \pi}{2} \sigma_G^2 \quad (7.10b)$$

With a scale factor k chosen to be 2 or 3, a good value for T is then:

$$T = \mu_R + k \sigma_R \quad (7.11)$$

The one-dimensional symmetry computation $S(x)$ can be extended to two dimensions by repeating (7.8) for different directions in the frequency domain. Using polar coordinates, the filter in the radial direction is given by $S(x)$, whereas in the angular direction θ filters $G(\theta)$ with Gaussian cross-sections are chosen:

$$G(\theta) = \exp\left(-\frac{(\theta - \theta_0)^2}{2\sigma_\theta^2}\right) \quad (7.12)$$

Here, θ_0 is the orientation angle of the filter, and σ_θ is the standard deviation chosen for the Gaussian filter. In addition to summing over all scales, we now have to sum over all orientations θ_i as well, yielding equation (7.1a).

7. APPLICATION 2: PROCEDURAL CAUSTICS

References

- [BW99] Max Born and Emil Wolf, *Principles of optics: Electromagnetic theory of propagation, interference and diffraction of light*, 7th ed., Cambridge University Press, Cambridge, UK, 1999. 129
- [Dav88] H A David, *The method of paired comparisons*, Charles Griffin & Company, London, 1988. 141
- [FH04] Hui Fang and John C Hart, *Textureshop: Texture synthesis as a photograph editing tool*, ACM Transactions on Graphics **23** (2004), no. 3, 354–359. 127
- [FH06] Hui Fang and John C Hart, *Rototexture: Automated tools for texturing raw video*, IEEE Transactions on Visualization and Computer Graphics **12** (2006), no. 6, 1580–1589. 127
- [GLMF⁺08] Diego Gutierrez, Jorge Lopez-Moreno, Jorge Fandos, Francisco Seron, Maria Sanchez, and Erik Reinhard, *Depicting procedural caustics in single images*, ACM Transactions on Graphics (Proc. of SIGGRAPH Asia) **27** (2008), no. 5, 120:1–120:9. 127
- [KB40] M G Kendall and B Babington-Smith, *On the method of paired comparisons*, Biometrika **31** (1940), no. 3/4, 324–345. 140
- [KBW06] J Kruger, K Burger, and R Westermann, *Interactive screen-space accurate photon tracing*, Proceedings of the Eurographics Symposium on Rendering, 2006, pp. 319–329. 129
- [Kov96] Peter Kovesi, *Invariant measures of image features from phase information*, Ph.D. thesis, The University of Western Australia, 1996. 131, 132
- [Kov97] Peter Kovesi, *Symmetry and asymmetry from local phase*, 10th Australian Joint Conference on Artificial Intelligence, 1997, pp. 2–4. 131, 132, 133, 144
- [Kov99] Peter Kovesi, *Image features from phase congruency*, Videre: Journal of Computer Vision Research **1** (1999), no. 3, 2–26. 133, 144, 145
- [KRFB06] Erum Arif Khan, Erik Reinhard, Roland Fleming, and Heinrich Bülthoff, *Image-based material editing*, ACM Transactions on Graphics **25** (2006), no. 3, 654–663. 127, 128, 130, 132, 134, 137, 142, 144

REFERENCES

- [LB00] Michael Langer and Heinrich H Bülthoff, *Depth discrimination from shading under diffuse lighting*, Perception **29** (2000), no. 6, 649–660. 132
- [LCTS05] P Ledda, A Chalmers, T Troscianko, and H Seetzen, *Evaluation of tone mapping operators using a high dynamic range display*, ACM Transactions on Graphics **24** (2005), no. 3, 640–648. 141
- [MB88] M. C. Morrone and D. C. Burr, *Feature detection in human vision: A phase-dependent energy model*, Proceedings of the Royal Society of London B **235** (1988), no. 1280, 221–245. 131
- [MTC07] A. Mohan, J. Tumblin, and P. Choudhury, *Editing soft shadows in a digital photograph*, IEEE Computer Graphics and Applications **27** (2007), no. 2, 23–31. 134
- [OCDD01] B M Oh, M Chen, J Dorsey, and F Durand, *Image-based modeling and photo editing*, SIGGRAPH '01: Proceedings of the 28th annual conference on Computer Graphics and Interactive Techniques, 2001, pp. 433–442. 127
- [OL81] A V Openheim and J S Lim, *The importance of phase in signals*, Proceedings of the IEEE **69** (1981), no. 5, 529–541. 131
- [PC82] L N Piotrowski and F W Campbell, *A demonstration of the visual importance and flexibility of spatial-frequency amplitude and phase.*, Perception **11** (1982), no. 3, 337–346. 131
- [PH66] E S Pearson and H O Hartley, *Biometrika tables for statisticians*, 3rd ed., vol. 1, Cambridge University Press, 1966. 141
- [Ros86] A Rosenfeld, *Axial representations of shape*, Computer Graphics and Image Processing **33** (1986), no. 2, 156–173. 131
- [SKALP05] László Szirmay-Kalos, Barnabás Aszódi, István Lazányi, and Mátyás Premecz, *Approximate ray-tracing on the GPU with distance impostors*, Computer Graphics Forum **24** (2005), no. 3, 695–704. 128
- [SKP06] M. Shah, J. Konttinen, and S. Pattanaik, *Caustics mapping: an image-space technique for real-time caustics*, IEEE Transactions on Visualization and Computer Graphics **13** (2006), no. 2, 272–280. 128
- [SL04] I Setyawan and R L Lagendijk, *Human perception of geometric distortions in images*, Proceedings of SPIE, Security, Steganography and Watermarking of Multimedia Contents VI, 2004, pp. 256–267. 141
- [TM98] C Tomasi and R Manduchi, *Bilateral filtering for gray and color images*, Proceedings of the IEEE International Conference on Computer Vision, 1998, pp. 836–846. 132
- [tPP05] S F te Pas and S C Pont, *Estimations of light source direction depend critically on material brdfs*, Perception. Supplement ECVP05 **34** (2005), 212. 131

-
- [Tyl96] C W Tyler (ed.), *Human symmetry perception and its computational analysis*, VSP International Science Publishers, Utrecht, 1996. 131
- [Wag95] J Wagemans, *Detection of visual symmetries*, *Spatial Vision* **9** (1995), no. 1, 9–32. 131
- [WBG06] Felix A Wichmann, Doris I Braun, and Karl R Gegenfurtner, *Phase noise and the classification of natural images*, *Vision Research* **46** (2006), no. 8/9, 1520–1529. 131
- [WD06] Chris Wyman and Scott Davis, *Interactive image-space techniques for approximating caustics*, *Proceedings of the ACM Symposium on Interactive 3D Graphics and Games*, 2006, pp. 153–160. 128
- [WD08] Chriss Wyman and Carsten Dachsbacher, *Improving image-space caustics via variable-sized splatting*, *Journal of Graphics Tools* **13** (2008), no. 1, 1–17. 129
- [WK07] H Wei and Q Kaihuai, *Interactive approximate rendering of reflections, refractions, and caustics*, *IEEE Transactions on Visualization and Computer Graphics* **13** (2007), no. 3, 46–57. 129
- [WY05] Jun Wu and Czhao-Xuan Yang, *Detecting image symmetry based on phase information*, *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, 2005, pp. 5150–5153. 131
- [Wym05] Chris Wyman, *An approximate image-space approach for interactive refraction*, *ACM Transactions on Graphics* **24** (2005), no. 3, 1050–1053. 128
- [Wym07] Chris Wyman, *Interactive refractions and caustics using image-space techniques*, 2007, pp. 359–371. 128
- [Wym08] Chris Wyman, *Hierarchical caustic maps*, *Proceedings of the ACM Symposium on Interactive 3D Graphics and Games*, 2008, pp. 163–171. 128, 129
- [XHMW05] Zhitao Xiao, Zhengxin Hou, Changyun Miao, and Jianming Wang, *Using phase information for symmetry detection*, *Pattern Recognition Letters* **26** (2005), no. 13, 1985–1994. 131
- [YS05] Xiaoyan Yuan and Pengfei Shi, *Iris feature extraction using 2d phase congruency*, *Third International Conference on Information Technology and Applications*, vol. 2, 2005, pp. 437–441. 131
- [Zab93] H Zabrodsky, *Computational aspects of pattern characterization. continuous symmetry*, Ph.D. thesis, Hebrew University in Jerusalem, 1993. 131
- [ZFGH05] Steve Zelinka, Hui Fang, Michael Garland, and John C Hart, *Interactive material replacement in photographs*, *Proceedings of Graphics Interface*, 2005, pp. 227–232. 127

REFERENCES

Chapter 8

Application 3: Image Stylization and Non Photorealist Rendering

This chapter presents a set of stylization techniques that deals with a single photograph as input. We have applied our processing pipeline to the design of novel non-photorealistic stylization techniques. By leveraging well-known characteristics of human perception along with a simple depth approximation algorithm (shown in Chapter 4, we explore six novel stylization methods based on different rendering techniques; halftoning, multitone, lambertian shading, ambient occlusion and global illumination proving the versatility of our approach, and validate our assumptions and simplifications by means of a user study. As proof of concept we have developed an interactive (real-time) editing interface which complements the edition of lighting by providing the user with full artistic control over the generation of color, shading and shadows.

This research has given rise to two new publications: a paper awarded as *Best Paper* at the NPAR 2010 international conference (LMJH⁺10) and an article in the Computers & Graphics Journal, indexed Q3 in JCR list (LMJH⁺11).

8.1 Introduction

Whether the goal is to convey a specific mood, to highlight certain features or simply to explore artistic approaches, non-photorealistic rendering (NPR) provides an interesting and useful set of techniques to produce computer-assisted stylizations. Most of those techniques either leverage 3D information from a model, work entirely in 2D image space, or use a mixed approach (typically by means of a Z- or G-buffer) (Dur02). We are interested in exploring new possibilities for stylized depiction using a single image as input, while escaping traditional limitations of a purely 2D approach. For instance,

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING

the design of lighting schemes is crucial to communicate a scene’s mood or emotion, for which depth information is required.

Our key observation is the fact that a single photograph or painting has richer information than we might expect. In particular, we ask ourselves what layers of information present in an image may have been usually overlooked by stylized depiction techniques? And what would the simplest way to access that ”hidden” information be, in a way that allows dramatic manipulation of the look of an image?

It is well known that, when it comes to stylized depiction, human perception is able to build complex shapes with very limited information, effectively filling in missing detail whenever necessary, as illustrated in Figure 8.1 (left). The power of suggestion and the influence of light and shadows in controlling the emotional expressiveness of a scene has also been extensively studied in photography and cinematography (KCCP96; Alt45): for instance, carefully placed shadows can turn a bright and cheerful scene into something dark and mysterious, as in Figure 8.1 (right).

With this in mind, we propose a new class of methods for stylized depiction of images based on approximating significant depth information at local and global levels. We aim to keep the original objects recognizable while conveying a new mood to the scene. While the correct recovery of depth would be desirable, this is still an unsolved problem. Instead, we show that a simple methodology suffices to stylize 3D features of an image, showing a variety of 3D lighting and shading possibilities beyond traditional 2D methods, without the need for explicit 3D information as input. An additional advantage of our approach is that it can be mapped onto the GPU, thus allowing for real-time interaction.

Within this context, we show stylized depictions ranging from simulating the *chiaroscuro* technique of the old masters like Caravaggio (Civ06) to techniques similar to those used in comics. In recent years, both the movie industry (Sin City, A Scanner Darkly, Renaissance etc.) and the photography community (more than 4000 groups related to comic art on Flickr) have explored this medium. The goal of obtaining comic-like versions of photographs has even motivated the creation of applications such as Comic Life¹.

8.2 Previous Work

Our work deals with artistic, stylized depictions of images, and thus falls under the NPR category. This field has produced techniques to simulate artistic media, create meaningful abstractions or simply to allow the user to create novel imagery (SS02; GG01). In essence, the goal of several schools of artistic abstraction is to achieve a depiction of a realistic image where the object is still recognizable but where the artist departs from the accurate representation of reality. In this departure, the object

¹<http://plasq.com/comiclife-win>



Figure 8.1: Left: The classic image of "The Dog Picture", well known in vision research as an example of emergence: even in the absence of complete information, the shape of a dog is clearly visible to most observers (original image attributed to R. C. James (Mar82)). Right: Example of dramatically altering the mood of an image just by adding shadows.

of depiction usually changes: a certain mood is added or emphasized, unnecessary information is removed and often a particular visual language is used.

In this chapter, we explore what new possibilities can be made available by adding knowledge about how the human visual system (HVS) interprets visual information. It is therefore similar in spirit to the work of DeCarlo and Santella (DS02) and Gooch et al. (GRG04). DeCarlo and Santella propose a stylization system driven by both eye-tracking data and a model of human perception, which guide the final stylized abstraction of the image. Their model of visual perception correlates how interesting an area in the image appears to be with fixation duration, and predicts detail visibility within fixations based on contrast, spatial frequency and angular distance from the center of the field of view. Although it requires the (probably cumbersome) use of an eye-tracker, as well as specific per-user analysis of each image to be processed, the work nevertheless shows the validity of combining perception with NPR techniques, producing excellent results.

Instead, we apply well-established, general rules of visual perception to our model, thus freeing the process from the use of external hardware and individual image analysis. The goals of both works also differ from ours: whilst DeCarlo and Santella aim at providing meaningful abstraction of the input images, we are predominantly interested in investigating artistic possibilities.

Gooch and colleagues (GRG04) multiply a layer of thresholded image luminances with a layer obtained from a model of brightness perception. The system shows excellent results for facial illustrations. It is noted that in their approach some visual details may be difficult (or even impossible) to recover. Although in the context of facial stylization this counts as a benefit, it might not be desirable for more general imagery.

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING

Depth information has previously been used to aid the generation of novel renditions. For instance, ink engravings can be simulated by estimating the 3D surface of an object in the image, and using that to guide strokes of ink (Ost99). This method is capable of producing high-quality results, although it requires the user to individually deform 3D patches, leading to a considerable amount of interaction. The algorithms proposed by Oh et al. (OCDD01) cover a wide range of image scenarios with specific solutions to extract 3D data for each one, but also come at the expense of considerable manual input. Okabe and colleagues (OZM⁺06) present an interactive technique to estimate a normal map for relighting, whereas in (YCLL08), painterly art maps (PAMs) are generated for NPR purposes. While both works show impressive results, they again require intensive, skilled user input, a restriction we lift in our system.

In their work, Raskar and colleagues (RTF⁺04) convey shape features of objects by taking a series of photographs with a multi-flash camera strategically placed to cast shadows at depth discontinuities. Akers et al. (ALK⁺03) take advantage of relighting to highlight shape and features by combining several images with spatially-varying light mattes, while in (RBD06) details are enhanced in 3D models via exaggerated shading. In contrast, our approach operates on single off-the-shelf images, allows for new, artistic lighting schemes, and requires at most a user-defined mask to segment objects, for which several sophisticated tools exist (LSTS04; RKB04).

In the field of halftone stylization based on 3D geometry we should mention the recent work of Buchholz et al. (BBDA10), which incorporates information from shading, depth and geometry in order to generate boundaries between black and white regions which run along important geometric features for shape perception (like creases).

Bhat et al. (BZCC10) proved the potential of gradient-based filtering in the design of image processing algorithms like painterly rendering or subtle image relighting.

A 2.5D approach has been explored in the context of video stylization (SZKC06), aiding the production of hatching and painterly effects. This method, however, requires the specific calibrated capture of the 2.5D video material to be processed, which is still either cumbersome or expensive.

8.3 Perceptual Background

At the heart of our algorithm, which will be described in the next section, lies the extraction of *approximate* depth information from the input image. Since we do not have any additional information other than pixel values, we obviously cannot recover depth accurately, and therefore the result will potentially contain large errors. However, given that we are interested in stylized depictions of images, we will show that we do not require physical accuracy, but only approximate values which yield pleasing, plausible results. Our depth approximation algorithm leverages some well-known characteristics of the human visual system. Although the inner workings of human depth perception are not yet fully

understood, there exist sufficient indicators of some of its idiosyncracies that enable us to approximate a reasonable depth map for our purposes. In particular we rely on the following observations:

1. Belhumeur et al. (BKY99) showed that for unknown Lambertian objects, our visual system is not sensitive to scale transformations along the view axis. This is known as the *bas-relief ambiguity*, and due to this implicit ambiguity large scale errors along the view axis such as those produced in many single view surface reconstruction methods tend to go unnoticed.
2. Human vision tends to reconstruct shapes and percepts from limited information, for instance filling in gaps as shown in Figure 8.1, and is thought to analyse scenes as a whole rather than as a set of unconnected features (Lof; EZ96).
3. Causal relationships between shading and light sources are difficult to be detected accurately (OCS05). The visual system does not appear to verify the global consistency of the light distribution in a scene (LZ97). Directional relationships tend to be observed less accurately than radiometric and spectral relationships.
4. There is evidence that human vision assumes that the angle between the viewing direction and the light direction is 20-30 degrees above the view direction (OBA08).
5. In general, humans tend to perceive objects as globally convex (LB00).

In the following three sections we describe our algorithm and its applications while, in Section 8.7 we will show the results of a user test validating our assumptions.

8.4 Algorithm

We rely on prior knowledge about perception, summarized above, to justify the main assumptions of our depth approximation algorithm. In particular, the bas-relief ambiguity (Observation 1) implies that any shearing in the recovered depth will be masked by the fact that we will not deviate from the original viewpoint in the input image (KDKT01); in other words, we assume a fixed camera. The second and third observations suggest that an NPR context should be more forgiving with inaccurate depth input than a photorealistic approach, for instance by allowing the user more freedom to place new light sources to achieve a desired look, as we will see. Finally, the combination of the first, fourth and last observations allows us to infer approximate depth based on the dark-is-deep paradigm, an approach used before in the context of image-based material editing (KRFB06) and simulation of caustics (Chapter 7).

The outline of the process is as follows: first the user can select any object (or objects) in the image that should be treated separately from the rest. Usually the selection of a foreground and a background suffices, although this step may not be necessary if the image is to be manipulated as a

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING

whole. We assume that such selection is accomplished by specifying a set of masks using any existing tool (LSTS04; RKB04).

In the last step of the process, the user can specify new light sources as necessary (for which object visibility will be computed), and choose from a variety of available styles.

8.4.1 Depth Recovery

The contents of this section have already been covered in Chapter 4, Section 4.2.1. We include it here for the sake of completeness and may be skipped if the reader is familiar with the aforementioned chapter.

As we described in Chapter 4, our goal is to devise a simple depth recovery algorithm which works well in an NPR context and offers the user real-time control for stylized depiction. We aim to approximate the main salient features without requiring a full and accurate depth reconstruction. We take a two-layer approach, following the intuition that objects can be seen as made up of large features (low frequency) defining its overall shape, plus small features (high frequency) for the details. This approach has been successfully used before in several image editing contexts (BPD06; MG08; RBD06), and has recently been used to extract relief as a height function from unknown base surfaces (ZTS09). We begin by computing luminance values on the basis of the (sRGB) pixel input using $L(x, y) = 0.212 \cdot R(x, y) + 0.715 \cdot G(x, y) + 0.072 \cdot B(x, y)$ (I.T90). Then we decompose the input object in the image into a base layer $B(x, y)$ for the overall shape as well as a detail layer $D(x, y)$ (BPD06), by means of a bilateral filter (TM98). Additionally, as the methods based on the dark-is-deep assumption tend to produce depth maps biased towards the direction of the light, we smooth this effect by filtering $B(x, y)$ with a reshaping function (KRFB06) which enforces its convexity, producing an inflation analogous to those achievable by techniques like *Lumo* (Joh02).

The detail layer D can be seen as a bump map for the base layer B . We decouple control over the influence of each layer and allow the user to set their influence in the final image as follows:

$$Z(x, y) = F_b \cdot B(x, y) + F_d \cdot D(x, y) \tag{8.1}$$

where $Z(x, y)$ is interpreted as the final, approximate depth, and F_b and F_d are user-defined weighting factors to control the presence of large and small features in the final image respectively, both independent and in the range $[0, 1]$. Figure 8.2 shows the results of different combinations of the base and detail layer of the teddy bear image, using the halftoning technique described in Section 8.5. The depth Z is stored in a texture in our GPU implementation (lower values meaning pixels further away from the camera).

The depth map Z serves as input to the relighting algorithm. Although a normal map could be derived from the depth map, it is not necessary for our purposes (except for the color relighting effect explained in Section 8.5). Figure 8.3 shows 3D renderings of the recovered depth for an input image;



Figure 8.2: Different combinations of the detail and base layer yield different depictions (here shown for the halftoning technique). From left to right: original image, base and detail layers, plus different depictions with a fixed $F_b = 1.0$ and increasing F_d from 0 to 1 in 0.25 increments.

it can be seen how depth inaccuracies are more easily noticed if the viewpoint changes, while they remain relatively hidden otherwise.



Figure 8.3: Recovered depth from a given image. Errors remain mostly unnoticed from the original viewpoint (left), but become more obvious if it changes (right). Light and shadows have been added for visualization purposes.

8.4.2 Computing Visibility for New Light Sources

The user can now adjust the lighting of the scene by defining point or directional light sources, to obtain a specific depiction or mood of the image. In the following, we assume a point light source at $\mathbf{p} = (p_x, p_y, p_z)^T$. There are no restrictions on where this light source can be placed.

Visibility is then computed on the GPU (in a similar fashion as other techniques such as parallax mapping (Tat06)): for each pixel in the framebuffer $\mathbf{q} = (x, y, z(x, y))^T$ belonging to an object we wish to relight, the shader performs a visibility test for the light (see Figure 8.4), by casting a ray towards its position. The pixels visited between \mathbf{q} and \mathbf{p} are given by Bresenham's line algorithm. The z -coordinate of the ray is updated at each step. Visibility is determined by querying the corresponding texels on the depth map. This information will be passed along to the specific NPR stylization techniques (see Section 8.5). Once a pixel visibility has been established, we can apply different NPR techniques to produce the desired stylized depiction of the image.

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING

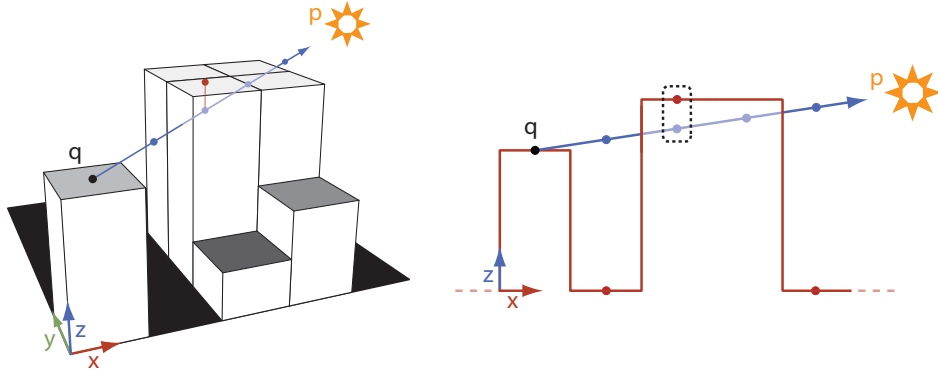


Figure 8.4: 3D and lateral views of the visibility computations for each texel.



Figure 8.5: From left to right: Input image. Output yielded by halftoning as described in (MG08) (both images courtesy of D. Mould). Result lit by a close point light. Another result lit by a directional light.

8.5 Stylization examples

We show a variety of examples which are currently implemented in our system. In each case, the defining difference over existing NPR work is the ability to relight the original image on the basis of the recovered 2.5D depth information. This adds versatility and artistic freedom. The different effects can be combined in layers for more complex looks, as some of our results show.

Halftoning: By simply mapping pixels visible from a light source to white and coloring all other pixels black, a halftoned rendition of the image is achieved. Figure 8.5 shows two examples of new relighting from an original input. Starting from a single image, we first create a halftoned version similar to what can be achieved with other systems (we use the implementation described in (MG08), where the authors present a method based on segmentation from energy minimization). The remaining sequence of images in this figure shows the application of two novel lighting schemes that leverage the recovered depth information, thereby extending the capabilities of previous approaches. In the first one, a point light source has been placed at (165, 240, 450) (in pixel units), whereas the second is lit by a directional light in the x direction. The weighting between detail and base layers is $(F_b, F_d) = (1.0, 0.9)$ for both images.

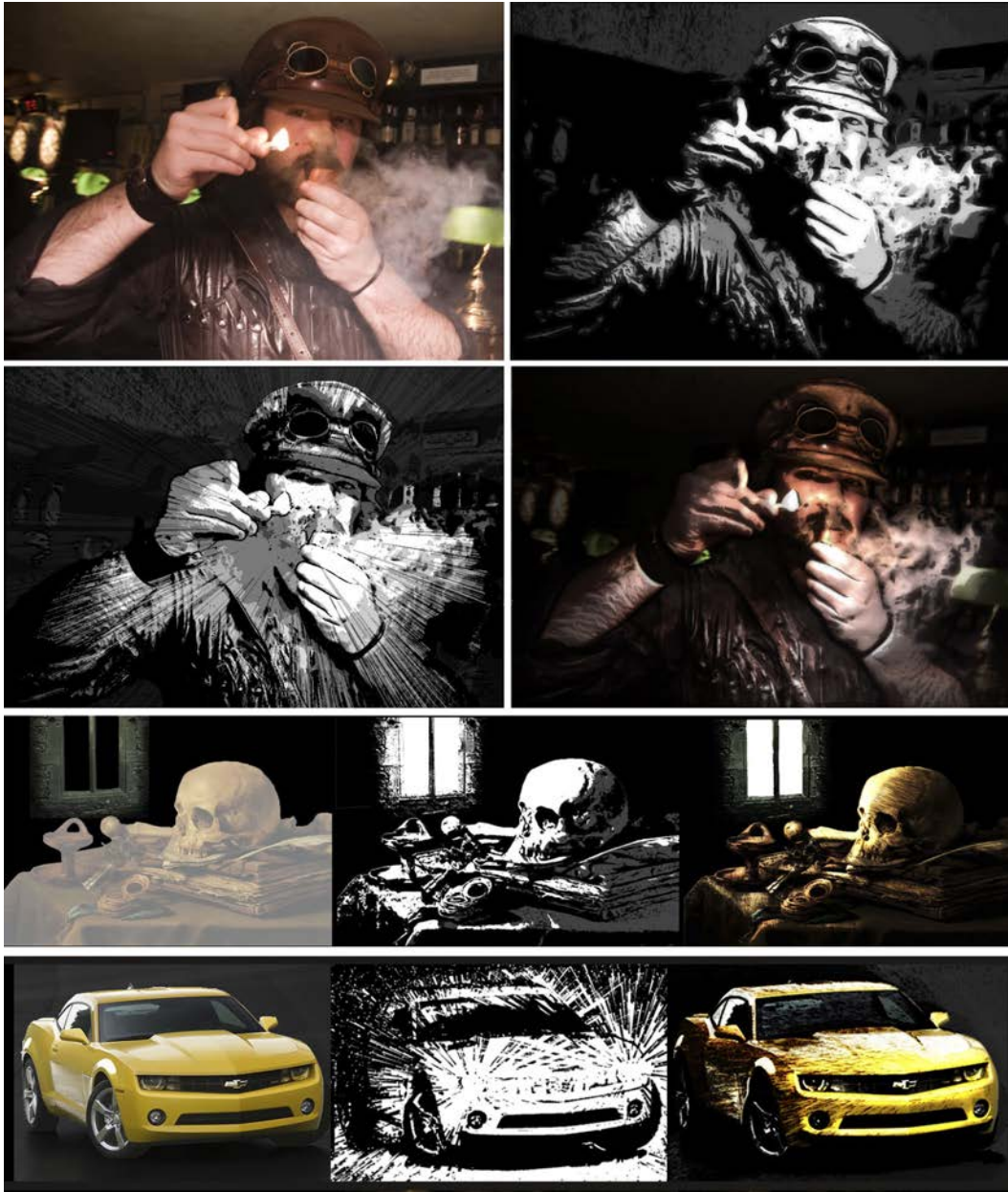


Figure 8.6: Stylized results achieved with our method. Top row, left: Original input image. Top row, right: Multitoned depiction with two point light sources at $(506,276,1200)$ and $(483,296,900)$, and using $(F_b, F_d) = (0.5, 0.8)$. Second row, left: Multitoned image with two layers of dynamic lines added, generated from the same light at $(500,275,1000)$. Second row, right: Result of multiplying color relighting with the multitoned version. Third row, from left to right: Mask with foreground objects (window painted manually for artistic effect and motivate subsequent relighting), multitone depiction of *Vanitas*, and result of multiplying two layers of color relighting and five layers of dynamic lines (please refer to the supplementary material to see the individual layers). Fourth row, from left to right: Original input image, *Dynamic lines* version placing a light source at both headlights, and a multilayer combination similar to *Vanitas* figure above.

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING

Multitoning: The spatial modulation of more than two tones (such as the black and white used in halftoning, plus several shades of gray) is known as multitoning or multilevel halftoning. In our implementation the user sets the position of a light source, after which a set of new lights with random positions located nearby the original is automatically created (the number of new lights is set by the user). This approach creates visually appealing renditions without having to place all light sources manually. Visibility is then computed separately for each light, and the results are combined in a single output by setting the value of each pixel in the final image to the average of the corresponding pixels in each layer. Results are shown in the second and sixth images in Figure 8.6 (in reading order) and the middle image of Figure 8.19 for three different inputs.

Dynamic Lines: When sketching, an artist may draw lines towards the light source to add a more dynamic look to the scene. We can emulate a similar technique just by direct manipulation of the depth map. We randomly select a set of object pixels; the probability of choosing a specific pixel is set to be inversely proportional to the Euclidean distance to the position of the considered light source. The depth values of the selected pixels are altered, effectively changing the results of the visibility computations in the image and casting shadows which are perceived as lines. The third and ninth image in Figure 8.6 show final examples using this technique.

Color relighting: For each pixel belonging to the object, we compute a normalized surface normal $\vec{n}(x, y)$ from the gradient field $\nabla z(x, y)$ (KRFB06):

$$\vec{g}_x(x, y) = [1, 0, \nabla_x z(x, y)]^T \quad (8.2)$$

$$\vec{g}_y(x, y) = [0, 1, \nabla_y z(x, y)]^T \quad (8.3)$$

$$\vec{n}(x, y) = \vec{g}_x \times \vec{g}_y / \|\vec{g}_x \times \vec{g}_y\| \quad (8.4)$$

Using this normal map as well as the 3D position of a light source, it is straightforward to modify pixel luminances or shading as function of the angle between the normals and the lights. Figures 8.6, 8.19 and 8.20 show examples with Gouraud shading. The color is extracted from the original image RGB values, converted to its corresponding value in *Lab* space and its luminance is set to a middle constant value. The initial albedo is obtained by combining the RGB original value with this luminance-attenuated value. The user can control this mixing, which is limited to pixels originally not clamped to black or white (where chromatic information is not available). The result is used as multiplying albedo by the color stylization methods.

Ambient occlusion: Local render methods like Phong shading fail to achieve the visual quality obtained by global illumination techniques. A crude yet effective method of approximating global illumination is the usage of ambient occlusion. It allows us to take into account attenuation of light due to occlusion of near surfaces. Occlusion is calculated by casting rays in the upper hemisphere of the rendered point, which allows us to obtain a binary value that describes whether the ray is occluded by a surface or if it is able to reach the background, usually referred to as the *sky*. An average is performed on these binary values, obtaining a visibility value. This visibility value is then usually multiplied with the ambient term of the lighting equation. In our case we multiply it with the

output of the color relighting shader, as we are looking for an stylized result. In the following lines, the method used for computing ambient occlusion will be described, which is based on the Starcraft II approach (FM08): it is one of the most elaborated methods, and already proven to work in a general, non-controlled environment.

However, casting rays in every direction of the hemisphere rules out real-time manipulation. Solutions to this problem have been presented in the form of screen-space methods that approximate occlusion by using simple depth comparisons. Thus, instead of casting rays in each direction, a randomized n -set of (x, y, z) offsets are used to query depth at different positions. Then, a depth value $Z(x, y)$ is compared with the z component of the corresponding offset; if z is greater than $Z(x, y)$, it is assumed that there is no geometry blocking at that offset. The result of this comparison is a binary value that is averaged similarly to the ray casting approach, which yields an approximated visibility term for a pixel (x, y) :

$$V(x, y) = \sum_{i=0}^n \frac{Z(x_i, y_i) < z_i}{n}. \quad (8.5)$$

To achieve real-time rendering, only a few samples can be used, usually between 8 and 32. Sampling uniformly using such low sample counts leads to banding artifacts. To improve image quality, randomized sample positions are used. An 8×8 randomized texture containing normalized vectors is tiled in screen-space, giving each pixel its own random vector r . However, a set of n random offsets is required for each pixel. They are passed as fragment shader constants and reflected using each pixel's unique random vector r , effectively giving a semi-random set of n offsets for each screen pixel. To avoid self-occlusion, offset vectors are flipped when they point inwards with respect to the surface normal (which is obtained in the same way as in the color relighting shader).

Randomizing the sampling position trades banding for noise. It yields better results, but by itself it is unable to produce high quality results. To deal with the resulting noise, a smart Gaussian blur is performed that takes into account differences in depth, which enables the removing of noise from the calculated visibility while avoiding visibility bleeding along object edges.

An important piece of a screen-space ambient occlusion shader is the attenuation function. It must be chosen with care, in order to prevent far away objects from generating occlusion among themselves. Instead of simply comparing depth with the z component of the offset, a delta $e = z - Z(x, y)$ is calculated. This delta is then modified by the attenuation function. As done by Filion and McNaughton (FM08), a linearly stepped attenuation function is used, where delta values less than an artist-chosen constant c give an occlusion of 0, whereas values higher than c are modified using $a \cdot \text{abs}(e)^b$. All the images used in this work have empirically fixed values of $c = 0.05$, $a = 10.0$ $b = 2.0$. The right image in Figure 8.7 shows the result of multiplying three passes of ambient occlusion with different attenuation values ($b = 2.0, 4.0$ and 8.0) to obtain a pencil-style depiction of a photograph.

The most correct approach for sampling is to convert depth values to camera space, add the randomized offsets, then project to screen space. However, we cannot transform depth values to eye-

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING



Figure 8.7: Ambient occlusion effect. Left: Input image. Right: The result of combining three different attenuation values. By increasing b (2.0, 4.0 and 8.0) we obtain local occlusion (detail) and medium-range occlusion (smooth shading). The depth map was generated with $F_b = 1.0$ and $F_d = 0.3$.

space positions as the projection matrix of an image is not known. Therefore, a simpler approach is used, where sampling is entirely done in screen space (Kaj09).

For more details about screen-space ambient occlusion we refer the reader to the existing bibliography (FM08; Kaj09; GR10).

Global illumination: A natural extension to ambient occlusion is the inclusion of an indirect bounce of global illumination (GR10). The scene must be modified first using color relighting, and stored in the direct radiance texture L . Then, in a second pass, ambient occlusion and global illumination are calculated together. For each sample position given by the randomized offset vectors, the radiance contribution $L(x, y)$ from sampled point A to current point P is calculated taking into account both the normal at the sampling position A , the normal at current point P and the attenuation produced as the light travels between the two points (see Figure 8.8):

$$L_{ind}(x, y) = \sum_{i=0}^n \frac{L(x_i, y_i) \cdot \cos(\theta_{s,i}) \cdot \cos(\theta_{r,i})}{s_i^2}, \quad (8.6)$$

where $\theta_{s,i}$ and $\theta_{r,i}$ are the angles between the transmission direction and the sender and receiver normals respectively, and s is the distance between the points P and A .

The final pixel value, using both ambient occlusion and global illumination is given by the following equation:

$$P(x, y) = ((1 - \alpha) + \alpha \cdot V(x, y)) \cdot L(x, y) + \beta \cdot L_{ind}(x, y), \quad (8.7)$$

where α and β control the strength of the ambient occlusion and global illumination effects, respectively. The parameter α has a valid range of values of $[0..1]$. On the other hand, floating point values

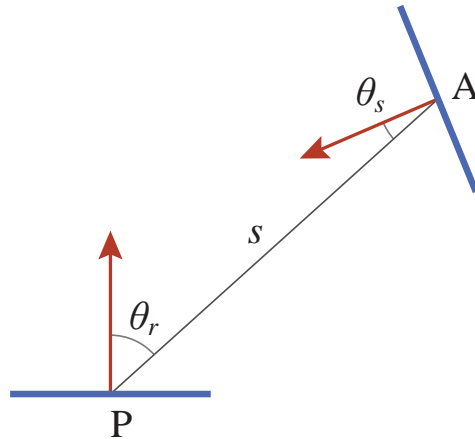


Figure 8.8: Radiance is transmitted from sender point A to receiver point P . The distance between both points is used to calculate the attenuation term $1/s^2$. On the other hand, the angles θ_s and θ_r are used to compute how much radiance is arriving at point P , as it depends on the orientation of both surfaces. *Figure adapted from (GR10).*

greater than or equal to zero are appropriate for the β parameter. Figure 8.9 shows another relighting example with our user interface. With three dials, the user can control α and β values and the range for the offset of the samples taken. See Figure 8.10 for some additional examples of the global illumination effect.

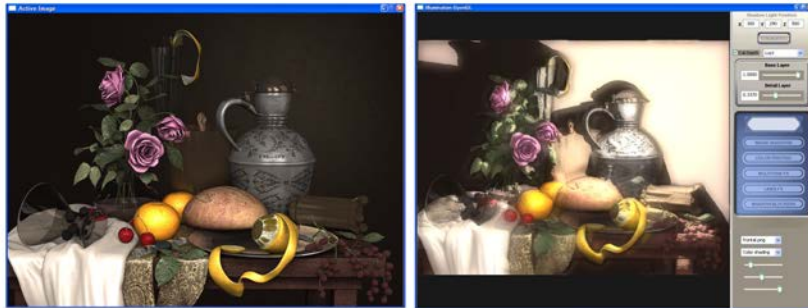


Figure 8.9: Example of the global illumination user interface. The dials (at the bottom of the right panel) are set to (offset) = 0.15 (maximum screen offset to take samples), $\alpha = 0.5$ and $\beta = 1.0$.

8.6 Image retouching interface

In order to incorporate local control over the stylization process we have developed a real time interactive brush. The artist can paint directly over the image with the mouse to alter the underlying geometry of the image thus altering the resulting stylization: modify the shading, set how shadows are

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING



Figure 8.10: Some examples of global illumination effect. From left to right: input image, relighting with $\alpha = 1.0$ and $\beta = 1.0$ and light source at $(80,1000,500)$, relighting with $\alpha = 1.0$ and $\beta = 2.0$ and light source at $(570,500,597)$. In this case the offset is set to 0 to over illuminate the image, producing an interesting glow effect. Finally, relighting with two light sources at $(50,920,230)$ and $(315,400,438)$. α and β are set to $(1.0,0.8)$. Note the color bleeding (red) produced at the jaw.

cast, highlight areas, etc. Our tool allows for edits like those shown by Todo and colleagues (TABI07) in stylized depictions of 3D models. However our work is based on a depth map without an associated implicit 3D surface therefore this kind of edition fits in the same category as approaches like gradient painting (MP08) or depth painting (Kan98). This tool is motivated both by the increased degree of artistic control it provides and the inherent inaccuracy of automatic depth map generation. In most cases, the automatically generated depth maps produce perceptually plausible depictions. However, in some scenarios this method yields results which may be non-plausible at certain regions of the image. This can be due to number of reasons such as the limitations of the shape from shading technique, the violation of our assumptions about the input (materials, global convexity,...), or even when reconstructing well-known geometries like a human face.

Shadow blockers: To further enhance artistic control over the generation of specific shadows, the user can paint directly over the image with the mouse, and the depth associated to the corresponding pixels is modified to block light and thus cast shadows. Figure 8.11 shows an example of a projected pattern and user-defined cast shadows. Note that these can be colored as well.

Depth sculpting tools: We have implemented the basic depth painting operations described by Kang (Kan98): shift depth by addition and subtraction (carve) and both global and local bump (see Figure 8.12). Both bump effects have an area of influence which is inversely weighted by the distance to the central pixel in the screen plane. However, in the case of the local bump the difference in depth is also considered. Additionally we have developed a smoothing brush which performs a gaussian convolution of Z values (see Figure 8.13 for an example of use).

Albedo painting: For color relighting techniques, the user can modify the albedo color of the image without affecting its 3D shape. The initial albedo is combined with the color of the brush in Lab space.



Figure 8.11: Adding mystery with shadows, cornerstone of the *noir* genre. Left: Original image. Right: Output yielded by a simple blocker which simulates light coming through blinds. $(F_b, F_d) = (0.6, 0.9)$

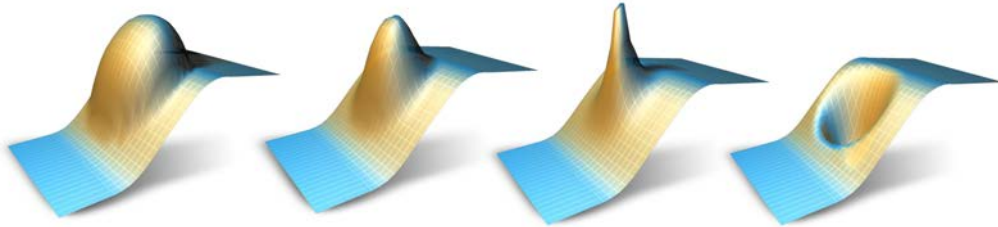


Figure 8.12: Results of applying different brushes to the depth map. The artistic control is given by the parameters of a gaussian function centered at the brush. From left to right, the degree of decay is increased (*pinch effect*) with the rightmost figure showing a carving example (depth subtraction).

Lighten/Darken: This tool allows the user to freely add localized light and shadows to an object in a manner that is consistent and seamlessly integrated with the current light environment and it is inspired by the work of Todo et al. (TAB107) which shows how to add intentional, even unrealistic, shade and light edits in NPR cartoon stylization. Intuitively, they force the shade and light boundaries to follow the user strokes as much as possible while yielding a plausible solution. To do so, they establish a set of boundary constraints based on the user strokes and try to find a displacement function for the underlying surface which, taking into account the light direction, yields the desired shade/light boundary. In order to make their strategy computationally tractable at interactive rates, they represent the offset function with a sum of Radial Basis Functions (RBF) and solve the linear problem for the desired curvature and boundary restrictions.

In our case, rather than affect shade/light boundaries, we intended to lighten or darken a local area by modifying its shading while keeping boundary coherency with the rest of the surface. To achieve this we have to shift their normals towards the light's direction (and do the opposite to darken it). Our approach is based on the convolution of the depth map with a gaussian function; the

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING



Figure 8.13: Example of depth editing. From left to right: Input image and relighting result with light source at $(550,400,460)$, image obtained with automatic depth map generation and after being edited by an artist with our tools for 5-10 minutes. The retouching tools helped in both correcting noticeable mistakes from depth generation (the emboss effect of the sunglasses) and creating a more interesting combination of shading and shadows (nose, lips, cheeks, jaw, ...).

brush has a radial area where the influence of the brush decays exponentially having a value equal to zero in its boundary. Additionally each user's stroke has only a delta addition/subtraction to the depthmap values, subsequently shifting the normals towards the light direction in a small quantity. This behavior is analogous to the RBF technique in the sense that there will be a smooth blending between the modified area (sum of gaussian radial functions produced by multiple strokes) and the original depthmap of the image.

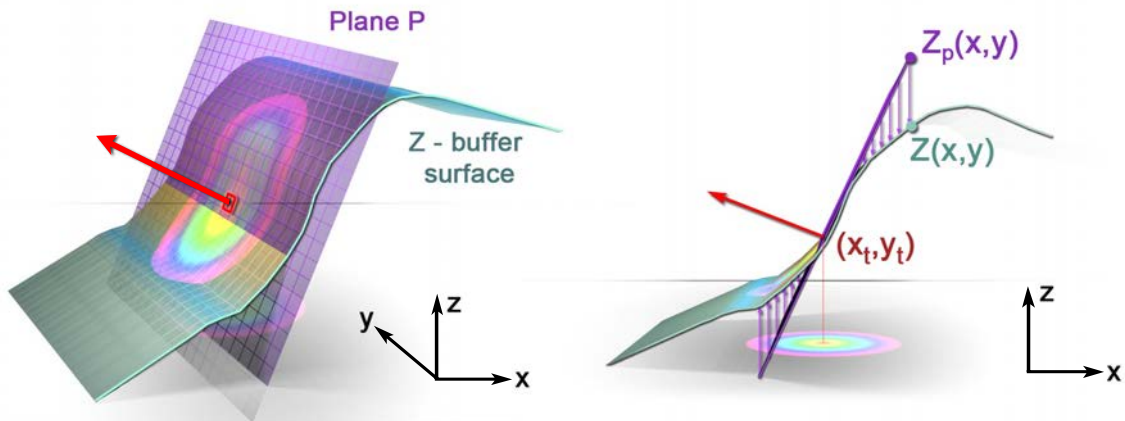


Figure 8.14: Two views of the depth map Z showing the virtual plane P used to shift the normals in the area of the brush (centered at pixel (X_t, Y_t)). The false radial colors illustrate the decay of the effect applied by the brush, which is adding $(Z_p(x, y) - Z(x, y))$ to the depth value of each pixel $Z(x, y)$.

To force the local normals to be oriented in a particular direction, we built a plane defined by that direction and the 3D position of the pixel corresponding to the center of the brush. We then modified each of the neighboring pixel's depth $Z(x, y)$ in direct relation to their distance to the plane P (see Figure 8.14). The computed variation of depth per pixel is weighted by its distance s to the center of the brush (t_x, t_y) in the screen plane (See Equation 8.8). The distance is computed by using a gaussian distribution with a scale λ and a standard deviation σ set by the user. A minimum value of one third of the brush's radius for σ ensures a smooth interpolation near the boundaries.

$$\begin{aligned}
s &= \lambda \cdot e^{-\frac{(x-t_x)^2+(y-t_y)^2}{\sigma^2}} \\
Z(x, y) &= Z(x, y) + s \cdot (Z_p(x, y) - Z(x, y))
\end{aligned} \tag{8.8}$$

Where $Z_p(x, y)$ is the depth value of the plane P at pixel (x, y) .

All the aforementioned techniques can be applied to both base and detail layers independently or in a combined way. In this fashion the artist has control over the range of the tool, editing the overall shape (base) and/or the local bumps (detail).

8.7 Evaluation

In order to test our algorithm and the assumptions it relies on, we devised a psychophysical experiment to objectively measure how inaccurate the recovered depth is, compared to how well these depth maps work in an NPR context. The test is designed as follows: we take a rendered image of a 3D scene of sufficient diversity, having both complex and simple shapes, and a wide range of materials including transparent glass. Since it is a synthetic scene, its depth information is accurate and known, and we can use it as ground-truth. We then generate two additional depictions of the same scene, changing the lighting conditions. The original image has the main light falling in front of the objects at an angle from right-above; we thus create two very different settings, where light comes a) from the camera position (creating a very flat appearance) and b) from behind the objects. Together, the three lighting schemes (which we call original, front and back) plus the variety of shapes and materials in the scene provide an ample set of conditions in which to test our algorithm. Figure 8.15, top, shows the three resulting images.

We then compare the ground-truth depth map of the 3D scene with each of the approximate depths recovered using our image-based algorithm (with $F_b = 1.0$ and $F_d = 0.3$ according to Equation 8.1). Figure 8.15 (middle and bottom rows) shows the four depth maps, the alpha mask used to define foreground and background, and the base and detail layers for each approximate depth map. Note that the ground-truth depth is the same for the three images, whereas our approximated depth is different since it depends on pixel values.

Table 8.1 shows the results of the L_2 metric and correlation coefficient (considering depth values pixel by pixel): our algorithm cannot recover precise depth information from just a single image, but the correlation with the ground truth is extremely high. Additionally, we also compare with a gray-scale version of the Lena image and with gray-level random noise (with intensity levels normalized to those of the 3D scene render), in both cases interpreting gray levels as depth information; both metrics yield much larger errors and very low, negative correlation. These results suggest that our simple depth extraction method approximates the actual depth of the scene well (from the same point

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING

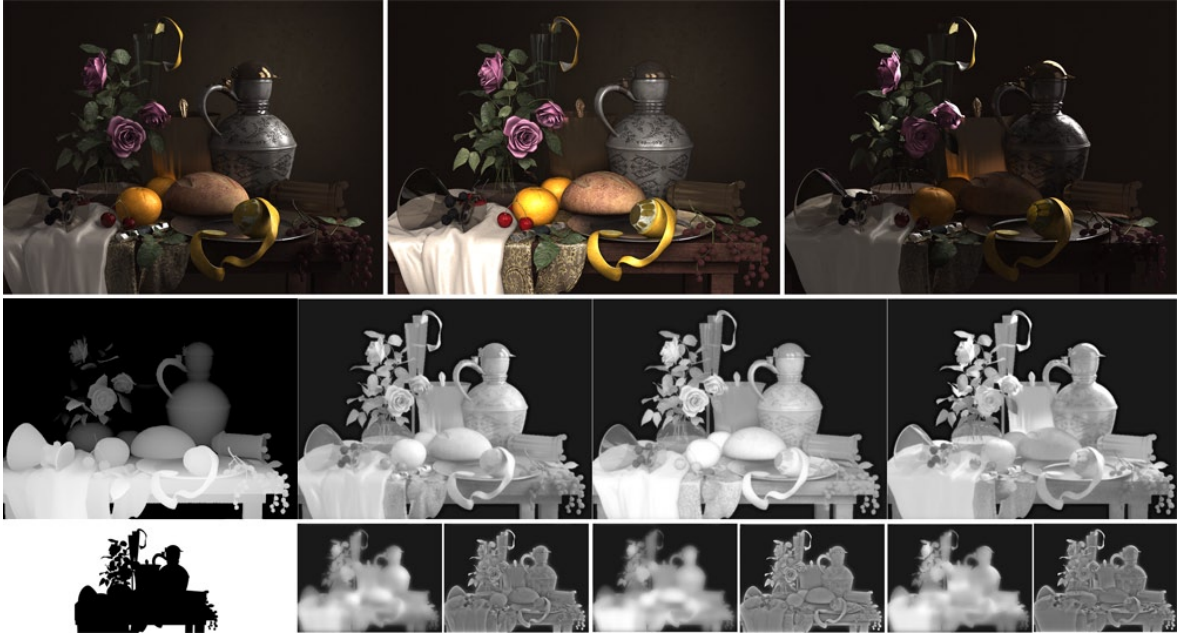


Figure 8.15: First row: The three rendered images used as input in our test, lit by the original, frontal and back illumination schemes respectively. Second row: Ground truth depth map obtained from the 3D information of the scene (bumpmaps not included), plus approximate depths recovered for each of the input images. Third row: alpha mask, plus the base and detail layers of each image, used to obtain the corresponding depth maps.

Input image	L_2	$Corr$
Original	100.16	0.93
Front	120.47	0.952
Back	121.66	0.925
Lena	383.92	-0.138
Random noise	524.74	-0.00075

Table 8.1: Results of the L_2 metric and correlation coefficient comparing the ground-truth depth of the 3D scene with the approximate depth extracted from each input image, plus a gray-scale version of the Lena image and gray-level random noise (interpreting gray levels as depth).

of view, since we are dealing with static images). The question we ask ourselves now is, is this good enough for our purposes? In other words, is the error obtained low enough to achieve our intended stylized depictions of the input image, without a human observer perceiving inconsistencies in the results?

One of the main advantages of our approach over other image-based stylization techniques is the possibility of adding new light sources. We thus explore that dimension as well in our test: for each of the three input images, we create two new lighting schemes, one with slight variations over the original scheme, and one with more dramatic changes. Finally, for each of the six resulting images, we create halftoning, multitoning and color relighting depictions, thus yielding a total of eighteen images.

Given that the ultimate goal of our test is to gain some insight into how well our recovered depth performs compared to real depth information, for each of the eighteen stimuli we create one version using real depth and another using recovered depth. We follow a two-alternative forced choice (2AFC) scheme showing images side-by-side, and for each pair we ask the participants to select the one that looks better from an artistic point of view. A gender-balanced set of sixteen participants (ages from 21 to 39 years old) with normal or corrected-to-normal vision participated in the experiment. All participants were unaware of the purpose of the study, and had different areas of knowledge and/or artistic backgrounds. The test was performed through a web site, in random order, and there was no time limit to complete the task (although most of the users reported having completed it in less than five minutes). Figure 8.16 shows some examples of the stimuli, comparing the results using real and approximate depth, for the three stylized depictions.

Figure 8.17 summarizes the results of our test, for the three styles (halftoning, multitone and color relighting) and two light variations (similar, different). The bars show the percentage of participants that chose the depiction using our method over the one generated with real depth (ground truth). Despite the relatively large errors in the approximate depth (as the metrics from Table 8.1 indicate), the results lie very closely around the 50-percent mark. We run a significance test on our results. Our hypothesis is that, despite the sometimes obvious differences in the depictions due to the different depths employed, there is no significant difference in the participants' choices when judging the resulting artistic stylizations. The differences in preference percentage for each of the aforementioned techniques are 0,04762, 0,09524 and 0,02439, which is in all the cases below 0,15121, the standard error for a 95% of confidence. Therefore, we can assure that there is no significant preference for actual depth over approximated depth in our test.

8.8 Discussion

We have shown results with a varied number of styles, all of which have been implemented on the GPU for real-time interaction and feedback, including relighting¹. Our simple depth approximation model works sufficiently well for our purposes, while allowing for real-time interaction, which more complex algorithms may not achieve. On a GeForce GTX 295, and for a 512×512 image and a single light source, we achieve from 110 to 440 frames per second. Performance decays with the number of lights: in our tests, real-time operation can be maintained with up to 5 light sources on average.

Our approach has several limitations. If the convexity assumption is violated, the depth interpretation of our method will yield results which will be the opposite to what the user would expect them to be. For small features it usually goes unnoticed, but if the object is not *globally* convex the results may not be plausible. Wrong depth interpretations from the dark-is-deep paradigm, such as the teddy bear's nose in Figure 8.2, can also be taken as intrusive regions; thus, expected cast shadows and relighting may look wrong in that area. Our method also assumes relatively Lambertian surface

¹Please refer to the video.

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING



Figure 8.16: Examples of the stimuli used in our user test, for the halftoning (top row), multitone (middle row) and color relighting styles (bottom row). Left and right columns were obtained with approximate and real depths respectively.

behavior: while highlights could be removed through thresholding or hallucination techniques, our assumptions on the perception of depth are broken in the case of highly refractive or reflective objects. In the latter case, shape-from-reflection techniques could be investigated. Also, since we do not attempt to remove the original shading from the image, our technique could potentially show artifacts if new lights are placed in the same direction of existing shadows (see Figure 8.18). However, our re-

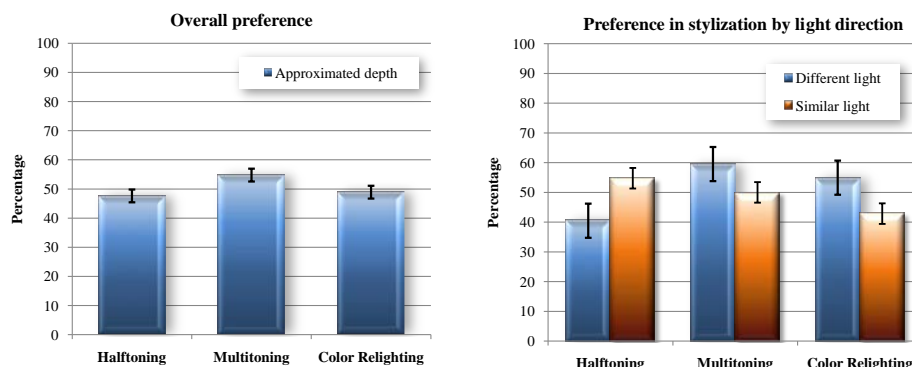


Figure 8.17: Percentage of participants that chose the depiction using approximate depth over the one generated with real depth, for the three styles (halftoning, multitoning and color relighting). Left: Average preference for all the images used in the test. Right: Preference in stylization considering the light direction: similar and differing from the original light source in the relighted images.

sults confirm that quite large shading inaccuracies tend to go unnoticed in a NPR context. We think that future research with different shape from shading techniques could clarify if simpler methods (sTS94) can still produce plausible depictions or even if more sophisticated techniques might extend the applicability to photorealistic image editing. Finally, since we recover only depth information from camera-facing object pixels, completely accurate shadows cannot be produced.



Figure 8.18: Artifacts due to original shadows in the image. Left: Detail of the original image depicted in Figure 8.6. Right: Relighting with a light source at (510, 520, 740) wrongly illuminates the shadowed areas.

Our method could potentially be used for video processing, for which temporal coherence should be considered. For the dynamic lines stylization technique proposed here, this could be very complicated since it would most likely require tracking features at pixel level. Video segmentation is also a difficult task that would be necessary to address (although as some of the images in this chapter show, compelling results can also be achieved in certain cases by processing the image as a whole). Finally, we expect that advances in the fields of perception and shape-from-shading will provide more exciting new grounds for artistic depiction of images and video.

8.9 Conclusions

We have presented a new methodology to develop NPR techniques based on the recovery of information about the depth from input images. Relying on known characteristics of human visual perception, our work offers more flexibility and artistic freedom than previous approaches, including the possibility of extreme relighting of the original image. Accurate extraction of depth information from a single image is still an open, ill-posed problem for which no solution exists. In this work we have shown that while our recovered depth is not accurate enough for certain applications, non-photorealistic stylization of images provides a much more forgiving ground, masking possible inconsistencies and leaving the abstraction process unhampered. Our results have been published at NPAR 2010 (LMJH⁺10)(*Best Paper*) and the Computers & Graphics Journal (LMJH⁺11).

The fact that the algorithm also works well with a painted image (*Vanitas*) is quite interesting: a human artist painting the scene performs inaccurate depth recovery and very coarse lighting estimation, and the perceptual assumptions made by our algorithm seem to correlate well with the human artistic process. Future work to develop a system that mimics this process more closely can give us valuable insight and become a very powerful NPR tool.

Our 2.5D interpretation of objects in images yields an appropriate basis for appealing visual effects. We have shown several applications for this approach, such as halftoning, multitoneing, dynamic lines, color relighting, ambient occlusion and global illumination, but many more effects could be devised (e.g.: relighting with non-Lambertian reflectance models). Furthermore, we have developed a set of real-time tools which allows the user to overcome the limitations of our automatic depth acquisition, providing full artistic control over the generation of color, shadows and shading. The work by Bhat et al. (BZCC10) could be combined with our approach in order to produce a wider range of visual effects. We think that future computer-aided 2D image editing techniques will benefit from a similar combination of underlying geometry (automatically generated and/or user-made) and the knowledge of the related human perception processes.



Figure 8.19: Application of our method to a very diffusely lit image. In this example we aim to obtain different moods by changing the light environment and the degree of stylization. Left: Original input image. Middle: A very stylized and dark version of the input by multitone depiction with four point light sources at $(140,400,300)$, $(140,400,350)$, $(140,400,400)$ and $(140,400,900)$ and using $(F_b, F_d) = (1.0, 0.2)$. Right: Less stylized depiction obtained by combination of multitone and color relighting effects with lights at $(134,530,290)$, $(115,15,270)$, $(315,695,350)$, $(100,400,1000)$ and $(589,325,325)$. No mask was used for these depictions.

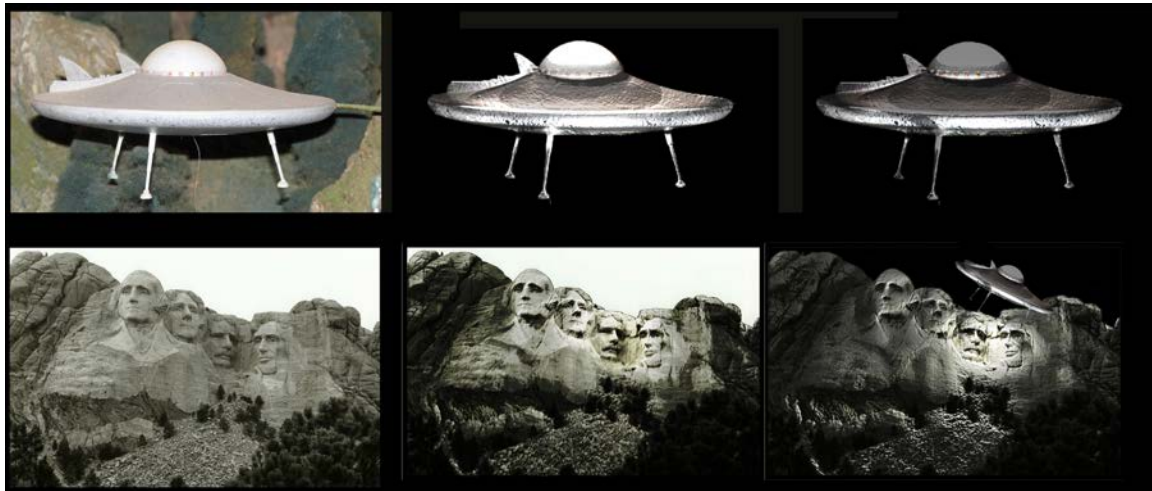


Figure 8.20: Composition of results. Top row, left: Original input image. Top row, middle: Color relighting with five point light sources: two from above at $x = 480, y = 520, z = (500, 250)$ and three surrounding the disk at $x = (50, 550, 100), y = 400, z = 1000$, and using $(F_b, F_d) = (1.0, 0.1)$. Top row, Right: result of multiplying a shadow layer created by a light source at $(580, 0, 500)$ and the relighted image (middle). Second row, from left to right: Original input image, stylized depiction by combination of color relighting and halftone, and result of compositing the relighted UFO from top row and a new relit version of the input image.

8. APPLICATION 3: IMAGE STYLIZATION AND NON PHOTOREALIST RENDERING



Figure 8.21: Top row: Example of relighting with ambient occlusion and global illumination effects (with a light source placed at $(570,320,710)$). The iron figure in the right was masked out from the input image and was affected by two additional light sources at $(468,535,420)$ and $(376,200,500)$ to produce highlights in the body and illuminate the shadowed area of the head respectively. Middle row: Input image and the result of combining multitone rendering with global illumination from three light sources (one placed in front of each eye and a third centered in the mouth). α was set to 1.0 and β to 2.0 to overexpose the original colors, producing a watercolor-comic book effect. Bottom row: from left to right: input image, color relighting with a top-left light source at $(146,1000,532)$ and global illumination relighting ($\alpha = 1.0$ and $\beta = 1.0$) with a bottom light at $(334,65,464)$. Note the effect of the light bouncing in the area marked by the white rectangle.

References

- [ALK⁺03] David Akers, Frank Losasso, Jeff Klingner, Maneesh Agrawala, John Rick, and Pat Hanrahan, *Conveying shape and features with image-based relighting*, VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03) (Washington, DC, USA), IEEE Computer Society, 2003, p. 46. 154
- [Alt45] John Alton, *Painting with light*, Berkeley: University of California Press, 1945. 152
- [BBDA10] Bert Buchholz, Tamy Boubekeur, Doug DeCarlo, and Marc Alexa, *Binary shading using geometry and appearance*, no. 6, 1981–1992. 154
- [BKY99] Peter N. Belhumeur, David J. Kriegman, and Alan L. Yuille, *The bas-relief ambiguity*, Int. J. Comput. Vision **35** (1999), no. 1, 33–44. 155
- [BPD06] Soonmin Bae, Sylvain Paris, and Frédo Durand, *Two-scale tone management for photographic look*, ACM Trans. Graph. **25** (2006), no. 3, 637–645. 156
- [BZCC10] Pravin Bhat, Larry Zitnick, Michael Cohen, and Brian Curless, *Gradientshop: A gradient-domain optimization framework for image and video filtering*, ACM Trans. Graph., vol. 29, 2010, pp. 1–14. 154, 172
- [Civ06] Giovanni Civardi, *Drawing light and shade: Understanding chiaroscuro (the art of drawing)*, Search Press, 2006. 152
- [DS02] Doug DeCarlo and Anthony Santella, *Stylization and abstraction of photographs*, ACM Trans. Graph. **21** (2002), no. 3, 769–776. 153
- [Dur02] Frédo Durand, *An invitation to discuss computer depiction*, NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering (New York, NY, USA), ACM, 2002, pp. 111–124. 151
- [EZ96] James H. Elder and Steven W. Zucker, *Computing contour closure*, In Proc. 4th European Conference on Computer Vision, 1996, pp. 399–412. 155
- [FM08] Dominic Filion and Rob McNaughton, *Starcraft II: Effects & Techniques*, Advances in Real-Time Rendering in 3D Graphics and Games Course (N. Tatarchuk, ed.), 2008. 161, 162

REFERENCES

- [GG01] B. Gooch and A. Gooch, *Non-photorealistic rendering*, 2001. 152
- [GR10] Thorsten Grosch and Tobias Ritschel, *Screen-space directional occlusion*, GPU Pro (Wolfgang Engel, ed.), A.K. Peters, 2010, pp. 215–230. 162, 163
- [GRG04] B. Gooch, E. Reinhard, and A. Gooch, *Human facial illustrations: creation and psychophysical evaluation*, ACM Trans. Graph. **23** (2004), no. 1, 27–44. 153
- [I.T90] I.T.U., Basic Parameter Values for the HDTV Standard for the Studio and for International Programme Exchange, ch. ITU-R Recommendation BT.709, Formerly CCIR Rec. 709, Geneva, 1990. 156
- [Joh02] Scott F. Johnston, *Lumo: illumination for cel animation*, NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering (New York, NY, USA), ACM, 2002, pp. 45–ff. 156
- [Kaj09] Vladimir Kajalin, *Screen-space ambient occlusion*, ShaderX⁷ (Wolfgang Engel, ed.), Charles River Media, 2009, pp. 413–424. 162
- [Kan98] Sing Bing Kang, *Depth painting for image-based rendering applications*, U.S. Patent no. 6,417,850, granted July 9, 2002, 1998. 164
- [KCCP96] J. Kahrs, S. Calahan, D. Carson, and S. Poster, *Pixel cinematography: A lighting approach for computer graphics*, ACM SIGGRAPH Course Notes, 1996, pp. 433–442. 152
- [KDKT01] J.J. Koenderink, A.J. Van Doorn, A. Kappers, and J. Todd, *Ambiguity and the mental eye in pictorial relief*, Perception **30** (2001), no. 4, 431–448. 155
- [KRFB06] Erum Arif Khan, Erik Reinhard, Roland Fleming, and Heinrich Bühlhoff, *Image-based material editing*, ACM Transactions on Graphics (SIGGRAPH 2006) **25** (2006), no. 3, 654–663. 155, 156, 160
- [LB00] Michael Langer and Heinrich H Bühlhoff, *Depth discrimination from shading under diffuse lighting*, Perception **29** (2000), no. 6, 649–660. 155
- [LMJH⁺10] Jorge Lopez-Moreno, Jorge Jimenez, Sunil Hadap, Erik Reinhard, Ken Anjyo, and Diego Gutierrez, *Stylized depiction of images based on depth perception*, NPAR '10: Proceedings of the 8th international symposium on Non-photorealistic animation and rendering, ACM, 2010. 151, 172
- [LMJH⁺11] Jorge Lopez-Moreno, Jorge Jimenez, Sunil Hadap, Erik Reinhard, Ken Anjyo, and Diego Gutierrez, *Non-photorealistic, depth-based image editing*, Computers & Graphics **In press** (2011). 151, 172
- [Lof] Gunter Loffler. 155
- [LSTS04] Y. Li, J. Sun, C-K. Tang, and H-K Shum, *Lazy snapping*, Siggraph (Los Angeles, California), ACM, 2004, pp. 303–308. 154, 156

-
- [LZ97] M.S. Langer and S.W. Zucker, *Casting light on illumination: A computational model and dimensional analysis of sources*, Computer Vision and Image Understanding **65** (1997), 322–335. 155
- [Mar82] D. Marr, *Vision*, W. H. Freeman and Company, New York, 1982. 153
- [MG08] David Mould and Kevin Grant, *Stylized black and white images from photographs*, NPAR '08: Proceedings of the 6th international symposium on Non-photorealistic animation and rendering (New York, NY, USA), ACM, 2008, pp. 49–58. 156, 158
- [MP08] James McCann and Nancy S. Pollard, *Real-time gradient-domain painting*, ACM Trans. Graph. **27** (2008), no. 3, 1–7. 164
- [OBA08] James P. O'Shea, Martin S. Banks, and Maneesh Agrawala, *The assumed light direction for perceiving shape from shading*, ACM Applied Perception in Graphics and Visualization (APGV), 2008, pp. 135–142. 155
- [OCDD01] Byong Mok Oh, Max Chen, Julie Dorsey, and Frédo Durand, *Image-based modeling and photo editing*, SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques (New York, NY, USA), ACM, 2001, pp. 433–442. 154
- [OCS05] Yuri Ostrovsky, Patrick Cavanagh, and Pawan Sinha, *Perceiving illumination inconsistencies in scenes*, Perception **34** (2005), 1301–1314. 155
- [Ost99] Victor Ostromoukhov, *Digital Facial Engraving*, SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques (New York, NY, USA), ACM, 1999, pp. 417–424. 154
- [OZM⁺06] Makoto Okabe, Gang Zeng, Yasuyuki Matsushita, Takeo Igarashi, Long Quan, and Heung yeung Shum, *Single-view relighting with normal map painting*, In Proceedings of Pacific Graphics 2006, 2006, pp. 27–34. 154
- [RBD06] Szymon Rusinkiewicz, Michael Burns, and Doug DeCarlo, *Exaggerated shading for depicting shape and detail*, SIGGRAPH '06: ACM SIGGRAPH 2006 Papers (New York, NY, USA), ACM, 2006, pp. 1199–1205. 154, 156
- [RKB04] C. Rother, V. Kolmogorov, and A. Blake, *GrabCut: Interactive foreground extraction using iterated graph cuts*, Siggraph (Los Angeles, California), ACM, 2004, pp. 309–314. 154, 156
- [RTF⁺04] Ramesh Raskar, Kar-Han Tan, Rogerio Feris, Jingyi Yu, and Matthew Turk, *Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging*, SIGGRAPH '04: ACM SIGGRAPH 2004 Papers (New York, NY, USA), ACM, 2004, pp. 679–688. 154
- [SS02] T. Strothotte and S. Schlechtweg, *Non-photorealistic computer graphics*, 2002. 152

REFERENCES

- [sTS94] Ping sing Tsai and Mubarak Shah, *Shape from shading using linear approximation*, Image and Vision Computing **12** (1994), 487–498. 171
- [SZKC06] Noah Snavely, C. Lawrence Zitnick, Sing Bing Kang, and Michael Cohen, *Stylizing 2.5-d video*, NPAR '06: Proceedings of the 4th international symposium on Non-photorealistic animation and rendering (New York, NY, USA), ACM, 2006, pp. 63–69. 154
- [TABI07] Hideki Todo, Ken Anjyo, William Baxter, and Takeo Igarashi, *Locally controllable stylized shading*, SIGGRAPH '07: ACM SIGGRAPH 2007 papers (New York, NY, USA), ACM, 2007, p. 17. 164, 165
- [Tat06] Natalya Tatarchuk, Shader X5, ch. Practical Parallax Occlusion Mapping, pp. 75–105, Charles River Media, 2006, pp. 75–105. 157
- [TM98] C Tomasi and R Manduchi, *Bilateral filtering for gray and color images*, Proceedings of the IEEE International Conference on Computer Vision, 1998, pp. 836–846. 156
- [YCLL08] Chung-Ren Yan, Ming-Te Chi, Tong-Yee Lee, and Wen-Chieh Lin, *Stylized rendering using samples of a painted image*, IEEE Transactions on Visualization and Computer Graphics **14** (2008), no. 2, 468–480. 154
- [ZTS09] Rony Zatzarinni, Ayellet Tal, and Ariel Shamir, *Relief analysis and extraction*, ACM Transactions on Graphics, (Proceedings of SIGGRAPH ASIA 2009) **28** (2009), no. 5, 1–9. 156

Chapter 9

Application 4: BSSRDF Estimation from Single Images

In this chapter we present a novel method to obtain an *approximation* of the Bidirectional Subsurface Scattering Reflectance Distribution Function (BSSRDF) of translucent, homogeneous objects from a single image, based on the diffusion approximation (JMLH01). Under unknown lighting conditions and assuming no previous knowledge of the scene, this is a very ill-posed problem, which makes it impossible to recover the exact BSSRDF.

This work has been presented in Eurographics 2011 and published in the journal Computer Graphics Forum (MELM⁺11), which is indexed in Q1 at the JCR list for Software Engineering. Part of the key contributions of this research (see Section 9.3) have been published in the thesis of Dr. Muñoz (Muñ10).

9.1 Introduction

Rendering algorithms have evolved considerably over the past decades, which in turn has motivated new acquisition methods of reflectance data from real-world objects. While this is still an active area of research (WLL⁺08; GJJD09), the ability to estimate the reflectance characteristics of materials from a single image remains a considerable challenge. Given sparse photographic input, it is impossible to infer the exact geometry and lighting captured in a photograph, which are necessary for an accurate capture. Thus, additional hardware and multiple images are usually employed to obtain that information.

This work has two parts: First, a novel acquisition method is introduced to estimate the BSSRDF of translucent, homogeneous objects. This method is designed to be more robust than previous

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES



Figure 9.1: Starting with a single image, and without any other prior information, we capture an approximation of the subsurface scattering properties of objects with varying degrees of translucency. Then, we use the estimated BSSRDFs to render objects made of similar materials. From left to right: grape, orange soap and wax. The source photos are shown in the insets.

approaches and requires an input image with known geometry and illumination. Second, the algorithm is extended to work with a single image as input, inferring both geometry and illumination by applying the techniques exposed in this thesis.

Likewise, the acquisition algorithm can be divided in two steps: First, we approximate the diffusion profile as a linear combination of piecewise constant functions, an approach that enables a linear system minimization and maximizes robustness in the presence of suboptimal input data inferred from the image. We then fit to a smoother monotonically decreasing model, ensuring continuity on its first derivative. We show the feasibility of our approach and validate it in controlled environments, comparing well against physical measurements from previous works. We would like to refer the readers to the thesis of Dr. Muñoz (Muñ10) for details on the acquisition algorithm, as our main contribution to the application shown in this chapter is the process of approximating both geometry and illumination in order to extend the method to single images.

In the following sections we explore the performance of the acquisition method in uncontrolled scenarios, where neither lighting nor geometry are known. We show that these can be roughly approximated from the corresponding image by making two simple assumptions: that the object is lit by a distant light source and that it is globally convex, allowing us to capture the visual appearance of the photographed material. Our method yields a physically plausible function that captures the appearance of the material and can be used for rendering. Figures 9.1, 9.13 and 9.14 show some of our results.

Compared with previous works, our technique offers an attractive balance between visual accuracy and ease of use, allowing its use in a wide range of scenarios including off-the-shelf, single images, thus extending the current repertoire of real-world data acquisition techniques.

9.2 Previous Work

A wide range of methods for measuring reflectance properties from real-world samples exists. These typically use specialized equipment such as a gonireflectometer and/or photographic input obtained over a range of known viewing and lighting directions, e.g. (LKG⁺03; ST06). Single image approaches that require prior knowledge about the shape of the object have also been developed (BG01). These methods usually aim at capturing a representation of the BRDF of opaque objects; we refer the reader to the excellent existing literature for a more comprehensive description (DRS07; WLL⁺08).

Capturing and modeling the BSSRDF of *translucent* materials is a harder problem that generally requires the use of special measuring setups and long capture sessions (see for instance (JMLH01; GLL⁺04; WMP⁺06)). Camera-projector systems have also been used to measure reflectance of small material samples (PvBM⁺06; TGL⁺06). More recent approaches aim to capture BSSRDF models using more practical camera equipment. Donner and colleagues (DWd⁺08) use multi-spectral images to measure skin reflectance, requiring samples to be taken in front of their capture setup. Another approach exploits cross-polarization photography and uses 20 photographs from a single viewpoint to acquire a layered reflectance model of skin (GHP⁺08). The final example in this kind of approaches requires sampling a cube of the material to be captured, constraining the position of the camera and light source (WZT⁺08). Other alternative approaches aim to separate the subsurface scattering component of objects in an image, either by adding a set of diffuse priors (WT06) or using high-frequency patterns of illumination in a set of images (NKGR06). No specific reflectance model parameters are estimated, and thus using the results in a different context remains an open problem. The recently published SubEdit system (STPP09) includes the possibility of hallucinating a BSSRDF from two inputs: a single photograph under fixed lighting, plus previously acquired data from one or more different BSSRDFs. The user assigns scattering profiles from the measured data set to representative points in the image, and the effect is propagated across the surface. Our approach does not require the user to mark corresponding scattering functions and does not require the use of previously measured data. The Lit Sphere user-guided appearance transfer approach (SMGG01) transfers shading information from an image of a lit sphere to a complex object. In contrast to our work, this approach requires user interaction and would not allow relighting of the original material. It is also unclear how such approximation could be extended for translucent materials.

Recently, there have been two works that focus on estimating translucency properties from single images (MSY09; MMTG09). Both propose methods that approximate scattering properties of objects under controlled settings, based on the dipole approximation. In contrast to our approach, they require the 3D location of the camera, the lighting configuration of the scene and the geometry of the target object to be known a-priori. Additionally, the method by Mukaigawa et al. (MSY09) require the use of manually-rotated polarizing filters and light-absorbing black sheets during the capture. As acknowledged in their paper, their approach is quite unstable despite this dedicated hardware; this limits the applicability of the method, as their reduced set of results suggests.

The following section is part of the thesis of Dr. Muñoz (Muñ10), added for the sake of complete-

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

ness, as our main contributions are on the side of perception analysis and shape and light estimation for the extension of the method to the ill-posed scenario of single images (Section 9.4).

9.3 BSSRDF Estimation

Our BSSRDF estimation is based on the diffusion approximation (JMLH01) and is performed in two steps. First, the diffusion profile is expressed as a linear combination of piecewise constant basis functions, resulting in a linear system that can be efficiently solved applying the Quasi-Minimal Residual method (BBC⁺94). This increases the robustness of the method in the presence of suboptimal input derived from our ill-posed, uncontrolled scenarios. The second step performs a smoothing over the profile, eliminating discontinuities on the first derivative and ensuring physical plausibility. A reasonable option for the algorithm design would be a single-step non-linear optimization. However, preliminary tests (MMTG09) show that due to the ill-posed and underconstrained nature of the problem, this usually reaches local minima, yielding no plausible results.

In this section we introduce an approach for controlled environments, where both the geometry and the main light direction in the scene are known. This allows us to demonstrate the validity of our BSSRDF estimation algorithm as a previous step to its generalization for single images (Section 9.4).

9.3.1 Algorithm

We take as input a photo of a translucent object. As we aim to capture subtle reflectance variations, we avoid quantized data by using the RGBE high dynamic range format. Given an alpha matte \mathbb{O} of the object in the image, we first discard pixels representing highlights by simply assuming that the minimum of the derivative of the histogram of the input image indicates the start of the highlight (KRFB06). This defines $\mathbb{I} \subseteq \mathbb{O}$ as the set of object pixels from which we will estimate subsurface light transport information¹. We subsequently minimize the effect of indirect lighting by finding the pixel in \mathbb{O} with the lowest luminance, and subtracting that value from the pixels in \mathbb{I} . These simple operations help increase the accuracy of the input data.

Our BSSRDF estimation process leverages the fact that within optically thick materials, single scattering effects are negligible (JB02). Light distribution can be considered isotropic and thus we can expect the dipole diffusion approximation to hold. This allows us to express multiple subsurface scattering as:

$$L(x_{out}, \omega_{out}) = \frac{1}{\pi} F_t(\eta, \omega_{out}) \int_A R_d(\|x_{out} - x_{in}\|) E(x_{in}) dA(x_{in}) \quad (9.1)$$

¹Alternatively, the user can manually define a more specific suitable region. All the results shown in this paper, however, have been computed with our default definition of \mathbb{I}

where $L(x_{out}, \omega_{out})$ refers to the outgoing radiance at a specific point x_{out} in a specific direction ω_{out} , $F_t(\eta, \omega)$ is the Fresnel transmission coefficient and η represents the relative index of refraction. $R_d(\|x_{out} - x_{in}\|)$ is called the *diffuse reflectance function*, and depends on the distance between the incident and outgoing points and the properties of the corresponding translucent material (e.g. absorption coefficient, scattering coefficient, albedo or phase function). $E(x_{in})$ is the irradiance at a given point on the surface, expressed as:

$$E(x_{in}) = \int_{\Omega} F_t(\eta, \omega_{in}) L(x_{in}, \omega_{in}) |n_{in} \cdot \omega_{in}| d\omega_{in} \quad (9.2)$$

where $L(x_{in}, \omega_{in})$ represents incident radiance from direction ω_{in} . Given that we have roughly eliminated highlights and indirect illumination from the object matte, we assume that the outgoing radiance is mainly due to subsurface scattering. So the pixel values in \mathbb{I} are taken as a good estimator for the radiance L in Equation 9.1.

The two terms in Equations 9.1 and 9.2 that define the properties of the translucent material are the index of refraction η and the diffuse reflectance function $R_d(\|x_{out} - x_{in}\|)$. We use a standard value of $\eta = 1.3$ (XGL⁺07; WZT⁺08). Consequently, the only unknown in our model is $R_d(\|x_{out} - x_{in}\|)$. Different formulations for this function have been previously proposed. Note that our method is independent of the specific definition of this function. From Equation 9.2, and assuming directional light sources, we build the *front irradiance map* E , similar to the Translucent Shadow Maps technique (DS03). Different from TSM, we also define the *back irradiance map* E_b , in order to approximate the whole light transport through the object. Notice that this is just a separation of the surface, and that this information is not present (but approximated) from the photograph. The irradiance maps are defined per color channel in RGB space, and our algorithm is applied to each channel independently.

	Piecewise constant			Piecewise linear (MSY09)			Zero-mean gaussian			Hermite polynomials			Legendre polynomials		
Number of functions	10	20	30	10	20	30	10	20	30	10	20	30	10	20	30
Estimation time	24 s	31 s	37 s	32 s	45 s	59 s	10 m	20 m	31 m	86 s	-	-	91 s	198 s	14 m
Condition number	$2.9 \cdot 10^3$	$2.2 \cdot 10^4$	$5.9 \cdot 10^4$	$1.6 \cdot 10^4$	$1.9 \cdot 10^6$	$6.3 \cdot 10^6$	$1.7 \cdot 10^7$	$7.4 \cdot 10^7$	$2.3 \cdot 10^8$	$1.9 \cdot 10^{12}$	-	-	$4.6 \cdot 10^7$	$2.1 \cdot 10^9$	$1.9 \cdot 10^{10}$
Error	1.13	0.69	0.86	2.70	1.86	5.24	134.23	356.84	1708.74	3.47	-	-	5.23	5.41	4.85

Table 9.1: Results from our basis functions tests for the skull made of whole milk material (JMLH01) from Figure 9.3. For an increasing number of basis functions, the table shows estimation time, condition number of the matrices and error of the resulting diffusion profile (defined as $\int_0^1 [R_d(r) - \sum_{h=1}^m \hat{w}_h B_h(r)]^2 dr$, where R_d is the original diffusion profile). For more than 20 Hermite polynomials the system does not converge. For piecewise linear representation, the first row refers to the number of points of the piecewise linear representation.

Assuming an orthogonal projection, the view vector c for each point p is $c = (0, 0, 1)$. Considering $\omega_{out} = c$ in Equation 9.1, this yields $L_i = L(p_i, c)$ for each pixel in \mathbb{I} . Therefore we can now express Equation 9.1 in terms of depth, surface normals, camera and irradiance maps as follows:

$$L_i = \frac{1}{\pi} F_t(\eta, c) \sum_{j \in \mathbb{O}} (R_d(r) E_j \Delta A + R_d(r_b) E_{b,j} \Delta A_b) \quad (9.3)$$

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

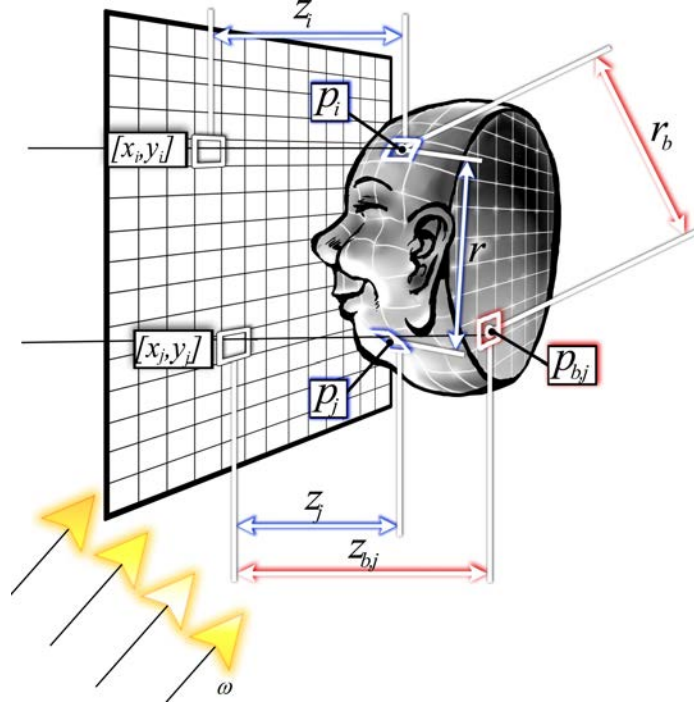


Figure 9.2: Definition of some of the parameters used in our algorithm.

where L_i represents the color of a pixel, $\Delta A = |c \cdot n_i|^{-1}$ is a factor related to the screen-space projection of the area of the object in a single pixel (similarly for ΔA_b), and r and r_b represent Euclidean distances in 3D space from point p_i on the front of the object to points p_j and $p_{b,j}$ (see Figure 9.2).

Approximating the diffuse reflectance function: The only unknown in Equation 9.3 is the diffuse reflectance function R_d , which defines the properties of a translucent material. As we have seen before, standard minimization-optimization algorithms could be used to estimate it. However such algorithms would be very time consuming, would require defining a specific model for the R_d function and might not converge to a plausible solution.

We thus opt for an efficient, robust two-step method. We first approximate R_d by a linear combination of a set of basis functions. This linear combination enables us to apply Equation 9.3 for each pixel $i \in \mathbb{I}$. We first rewrite Equation 9.3 as:

$$L_i = \sum_{j \in \mathbb{O}} (K_j R_d(r) + K_{b,j} R_d(r_b)) \quad (9.4)$$

where $K_j = \pi^{-1} F_i(\eta, c) E_j \Delta A$ (with a similar definition for $K_{b,j}$). Next, we estimate R_d by a linear

combination of m basis functions:

$$R_d(r) \approx \sum_{h=1}^m \hat{w}_h B_h(r) \quad (9.5)$$

where $B_h(r)$ represents the basis functions (discussed at the end of this section) and \hat{w}_h are the weights assigned to each basis function. Equation 9.4 now yields:

$$L_i = \sum_{j \in \mathbb{O}} \left(K_j \sum_{h=1}^m \hat{w}_h B_h(r) + K_{b,j} \sum_{h=1}^m \hat{w}_h B_h(r_b) \right) \quad (9.6)$$

This equation applies to every pixel $i \in \mathbb{I}$, so the complexity of this algorithm is $O(p^2)$ (where p is the number of pixels of the image). However, we have found that downscaling \mathbb{I} to a resolution of around 200x200 (preserving the aspect ratio of the input image) yields valid approximations for R_d while greatly reducing computation times. Applying the equation to each pixel of the scaled \mathbb{I} we get a linear system defined by the matrix product $\mathbf{A} \cdot \mathbf{X} = \mathbf{B}$, for n pixels and m basis functions, with:

$$a_{ih} = \sum_{j \in \mathbb{O}} (K_j B_h(r) + K_{b,j} B_h(r_b)) \quad (9.7)$$

$$\mathbf{X}_{m \times 1}^T = \begin{pmatrix} \hat{w}_1 & \hat{w}_2 & \dots & \hat{w}_m \end{pmatrix} \quad (9.8)$$

$$\mathbf{B}_{n \times 1}^T = \begin{pmatrix} L_1 & L_2 & \dots & L_n \end{pmatrix} \quad (9.9)$$

Resolution method: To solve the equivalent system $(\mathbf{A}^T \mathbf{A})\mathbf{X} = (\mathbf{A}^T \mathbf{B})$ we note that some columns in \mathbf{A} may contain values close to zero. This leads to a highly ill-conditioned matrix, while the related basis functions have negligible influence in the final solution. We thus set the associated weights \hat{w}_h to 0 and remove the corresponding columns from \mathbf{A} . Although this approximation reduces the condition number, the system is still ill-conditioned; we improve it further by using a Jacobi pre-conditioner for $(\mathbf{A}^T \mathbf{A})$, and solve the system using the Quasi-Minimal Residual (QMR) method (BBC⁺94).

Basis functions: In order to choose an appropriate set of basis functions, we rendered translucent objects using measured materials (JMLH01): in their work, the authors obtain scattering parameters by illuminating the surface of a translucent sample with focused white light and photograph it using a 3-CCD video camera. We then used the resulting renderings along with known geometry and lighting as input to approximate their diffusion profiles testing different options: uniformly distributed piecewise constant functions, zero-mean gaussians (inspired by the work of d'Eon et al (dLE07)), Hermite

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

and Legendre polynomials. Another option that has been previously used to represent diffusion profiles are piecewise linear polynomials (XGL⁺07; MSY09). To be able include them in our tests we use the more recent formulation by Mukaigawa et al (MSY09).

Zero-mean gaussian functions, Hermite and Legendre polynomials show high condition numbers, thus leading to unstable linear systems (see Table 9.1). Hermite polynomials do not even converge for 20 basis functions or more, while gaussian functions show very high errors. On the other hand, the condition number of piecewise linear functions (MSY09) is two orders of magnitude higher, and the error between two and six times larger than piecewise constant functions, which show the best overall behavior while being the fastest to compute. We thus choose to represent diffusion profiles with these basis functions in the first step of our algorithm. A good compromise between detail in the estimation and system stability is reached by using between 20 and 30 basis functions.

This difference between the stability of piecewise constant functions and the other presented options becomes very relevant in the case of inaccurate inputs, which is always the case when generalizing to uncontrolled single images (see Section 9.4). We found that, in those cases, more unstable bases such as Legendre polynomials or piecewise linear functions lead to higher condition numbers and the QMR method does not often converge to a solution.

	Reduced albedo									Reduced extinction (mm^{-1})								
	(JMLH01)			Estimated			Error			(JMLH01)			Estimated			Error		
	R	G	B	R	G	B	R	G	B	R	G	B	R	G	B	R	G	B
Apple	0.9987	0.9986	0.9772	0.9969	0.9985	0.9686	0.18%	0.01%	0.88%	2.2930	2.3934	2.0160	2.2428	2.3216	2.0202	2.19%	3.00%	0.21%
Cream	1.0000	0.9995	0.9949	1.0000	1.0000	0.9967	0.00%	0.05%	0.18%	7.3802	5.4728	3.1663	7.4580	5.9233	3.4267	1.05%	8.23%	8.22%
Marble	0.9990	0.9984	0.9976	1.0000	1.0000	1.0000	0.10%	0.16%	0.24%	2.1921	2.6241	3.0071	2.3543	2.7351	3.0359	7.40%	4.23%	0.96%
Potato	0.9965	0.9873	0.8209	1.0000	0.9999	0.9145	0.35%	1.27%	11.40%	0.6824	0.7090	0.6700	0.6690	0.6806	0.5651	1.97%	4.00%	15.65%
Skim milk	0.9980	0.9980	0.9926	0.9898	1.0000	0.9981	0.82%	0.20%	0.56%	0.7014	1.2225	1.9142	0.6875	1.2602	1.8943	1.99%	3.08%	1.04%
Whole milk	0.9996	0.9993	0.9963	1.0000	1.0000	0.9818	0.04%	0.07%	1.46%	2.5511	3.2124	3.7840	2.4968	3.1725	3.7553	2.13%	1.24%	0.76%

Table 9.2: Comparison between the measured properties of several materials (JMLH01) and the estimated properties resulting from our method, fitted to the dipole model.

Smoothing: In our second step, we fit this piecewise constant profile to a continuous, differentiable, monotonically decreasing function. This helps to eliminate noise and avoid discontinuities in the renderings, while keeping the function physically plausible. Our algorithm does not impose a particular model for this function, although the logical option would be to fit both scattering and absorption of the dipole model (JMLH01). However, working with a single image, it is not possible to deduce the physical size of the object nor the power of the light source, both necessary to obtain the corresponding dipole diffusion profile.

Thus, we propose a piecewise cubic polynomial $\hat{R}_d(r)$ instead, using Hermite interpolation. This model is generic and not associated to any physically-based BSSRDF model, which makes the method more flexible. The set of points and derivatives of this function is obtained by using a Simulated Annealing algorithm to minimize the following energy function:

$$E = w_d \int_0^1 \left(\hat{R}_d - \sum_{h=1}^m w_h B_h \right)^2 dr + w_p \int_0^1 \left(\hat{R}_d' \right)^2 \delta_{\hat{R}_d'}(\mathbb{R}^+) dr + w_s \int_0^1 \left(\hat{R}_d'' \right)^2 dr \quad (9.10)$$

where δ represents the Dirac measure function and w_d , w_p and w_s represent the weights of each term (which we experimentally set to 1, 10 and 10^{-4} , respectively). The first term is related to the difference between the smoothed function and the linear combination; the second term preserves the physical plausibility of the profile by penalizing positive derivatives, and the third term preserves the smoothness of the function. The dependencies on r have been omitted for the sake of clarity.

Validation: In order to validate our BSSRDF estimation algorithm independently of the accuracy of the input data, we first test it under known geometry and lighting (which allows us to use the dipole model): we again rendered objects with different measured material parameters (JMLH01) and then used the resulting images as input to our algorithm. To derive reduced albedo and extinction coefficients and thus provide an accurate numerical comparison, the estimated piecewise constant diffusion profiles were fitted in this case to the dipole model. Note that, as stated before, this fitting to the dipole is not possible for uncontrolled environments, and is introduced here for validation purposes only. For the rest of the paper, we use the piecewise cubic polynomial previously introduced.

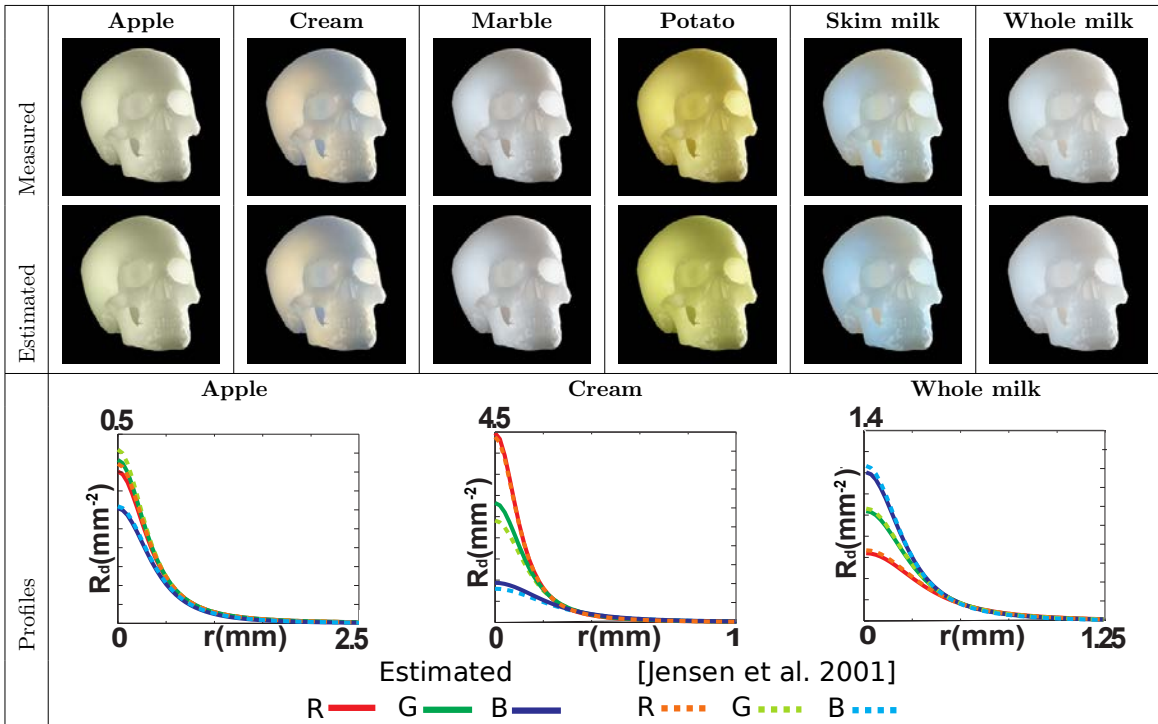


Figure 9.3: Top two rows: Comparison between renderings using physically measured materials (JMLH01) and our estimated diffusion profiles. Bottom row: Comparison of diffusion profiles. Please refer to (MELM⁺11) for the whole set of profiles.

Table 9.2 compares our results with the original physically measured data (JMLH01); it can be seen how our method yields very small residual error for most materials. As a result, both the profiles and the overall look of the images rendered with them are very similar to the ground truth (see Figure 9.3). The differences are due to the coarse modeling of the R_d function by a limited number of basis

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

functions, given the intrinsic trade-off between this number and the conditioning of the linear system.

In Section 9.4 we extend this method for the ill-posed case of single images, showing how to leverage rough estimates of both shape and light direction.

9.4 Estimation from Uncontrolled Single Images

We have demonstrated the suitability of our method in controlled environments. In this section we extend our approach to a much more challenging scenario: approximating diffusion profiles from uncontrolled single images. This is a heavily ill-posed problem, given that neither the light direction nor the geometry are known in this case. Therefore, instead of trying to recover an exact physically-based BSSRDF (which is obviously impossible), we aim to estimate a plausible representation that yields results similar to the material depicted in the input image.

We leverage the findings by Fleming and colleagues (FB05), who conclude that humans do not understand translucency through accurate inverse optics, but instead perceive the overall look of translucent materials based on simple image heuristics. This suggests that a suitable *approximation* of both the shape of the object and incident light direction may suffice for our purposes. We extend the usability of existing techniques, originally devised for opaque objects, and show that they can still yield plausible results when complying with our initial assumptions of global convexity and distant light sources.

Estimating shape: Estimating shape from a single image of an opaque object is an under-constrained problem by itself. Our work in depth estimation (see Chapter 4), however, has shown how rough approximations can work well in the context of image compositing (LMHRG10) or the simulation of caustics (GLMF⁺08) (see Chapters 3 and 7). We note that this estimation is even harder if the object is translucent, given the softening effects of subsurface scattering; we aim to find a similar approximation that works well for our purposes.

We base our estimation on three sources of information: pixels in the contour (which we assume to lie on the image plane at $Z=0$), shading information across its surface and the assumption of global convexity (LB00). Inspired by previous approaches (KRFB06; Joh02), we reconstruct the depth map Z of an object as the weighted sum of a base layer (which encodes global convexity) and a detail layer (which encodes high frequency), both obtained by means of the bilateral filter. We use values of $\sigma_{spatial} \in [0.08..0.1]$ and $\sigma_{intensity} \in [0.3..0.5]$ for the bilateral filter, while the weights for adding the base and detail layers are usually 0.8 and 0.2 respectively (thus favoring global convexity over details). We rely on additional non-linear spline functions to reshape the base layer and boost its apparent "inflation" (KRFB06). Given the inherent bass-relief ambiguity, we reverse the resulting signal if necessary to comply with our global convexity assumption, which yields our final depth map Z . A normal map N is subsequently computed from Z . Additionally, a *back depth map* Z_b plus the corresponding *back normal map* N_b are generated. We make the simplifying assumption that the back

of the object can be approximated by mirroring Z . While this is a strong simplification to circumvent the fact that we do not have information about the back portion of the object in the image, this straightforward operation suffices to produce good results when the object is not strongly illuminated from its back side. In fact, note that the heart-shaped soaps from Figure 9.14 and the mouse-shaped soap from Figure 9.1 are not symmetrical (their back face is plain) but still yield plausible profiles.

It could be argued that a simpler depth-recovery technique could be used instead, but in our experiments (see Section 9.5) this approach showed a good compromise between quality of the results and ease of use. We nevertheless restrict our estimations to simple geometries in order to minimize the impact of this error on the BSSRDF estimation, leaving the field of depth estimation from complex translucent geometries still open for further research. In the future, more accurate techniques could be trivially included at this stage.

Estimating light direction: Several existing methods can estimate light source directions from a single image, but usually at the expense of assuming some previous knowledge or including a calibration object in the scene (ZY01; WS02). In contrast, our goal is to obtain the dominant light direction starting with a single, off-the-shelf image, and thus we cannot impose such restrictions to our inputs.

We apply our *K-means* light detection method, published (LMHRG10) and described in Chapter 3, which performs a two-step analysis of the luminance channel of an object: first, the pixels of the contour \mathcal{O}' are clustered by a k-means algorithm to identify the number of light sources in the scene, as well as their azimuth θ_i direction (in image-space) and relative intensities. Second, zenith angles ϕ_i are approximated for each light direction by analyzing gradients in the interior of the object. The pair (θ_i, ϕ_i) defines the recovered 3D direction for each light.

Note that the original light detection algorithm was designed for opaque objects. In order to assess how well it extends to translucent objects, we tested it in controlled scenes with incident lights at specific directions over different objects with varying degrees of translucency. In our tests with different degrees of translucency, the error of the algorithm was always less than 20° , which has been found to be below perceptual threshold (see Chapter 2). The complete test with the different geometries, levels of translucency and light positions, plus another test of the behavior of the BSSRDF estimation algorithm when the input light directions are not accurate, can be found in the supplementary material.

Size of the object: Automatic estimation of the actual size of an object from a single photograph is not possible. Given that the diffusion profile R_d is a function of distance, we use a normalized unit distance equal to the width of the object in the image, and distribute all the piecewise constant basis functions in the range $[0, 1]$. In order to change the relative apparent size of the new rendered objects, it is possible to scale the diffusion profile as follows (STPP09):

$$R'_d(r) = \frac{1}{s^2} R_d\left(\frac{r}{s}\right) \tag{9.11}$$

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

where s is the scaling factor and R'_d is the scaled diffusion profile. Figure 9.4 shows the effect of this scaling.

9.5 Results and Discussion

Figure 9.5 shows the complete validation of the whole pipeline. We first rendered a heart-shaped object with three different measured materials (potato, marble and apple). We then used the rendered images as the only input to our algorithm (no geometry nor lighting are known) and approximated the BSSRDF from them. Finally, we re-rendered the same object with the resulting function. As it can be seen, the estimated materials achieve a very good visual match when compared to the original renderings.



Figure 9.4: Relative sizes for the same material. Left: blue soap from Figure 9.14. Right: grape from Figure 9.1

We solve the linear system in 30-40 seconds on a Dual Opteron @2.2 GHz with 4 GB of RAM, using between 20 and 30 basis functions for our representation. The smoothing step takes around 20 additional seconds. The recovered BSSRDF for the different materials can be directly used for rendering with no restrictions: for different geometries and under different illumination conditions. Figures 9.1 and 9.14 show several results for a wide range of translucent materials, including wax, soap, milk, ketchup, orange juice, detergent, grape and human skin. Our method works well even for extremely complex materials like skin, although it obviously cannot reproduce the subtleties of light transport in its multi-layered structure. Note that the renderings include additional specular highlights (Phong model) not captured with our method. The lighting in those figures has been set up to match the source image for easy direct comparison: more results under different lighting conditions and geometries can be seen in Figure 9.13, and with different relative sizes for the same material in Figure 9.4.

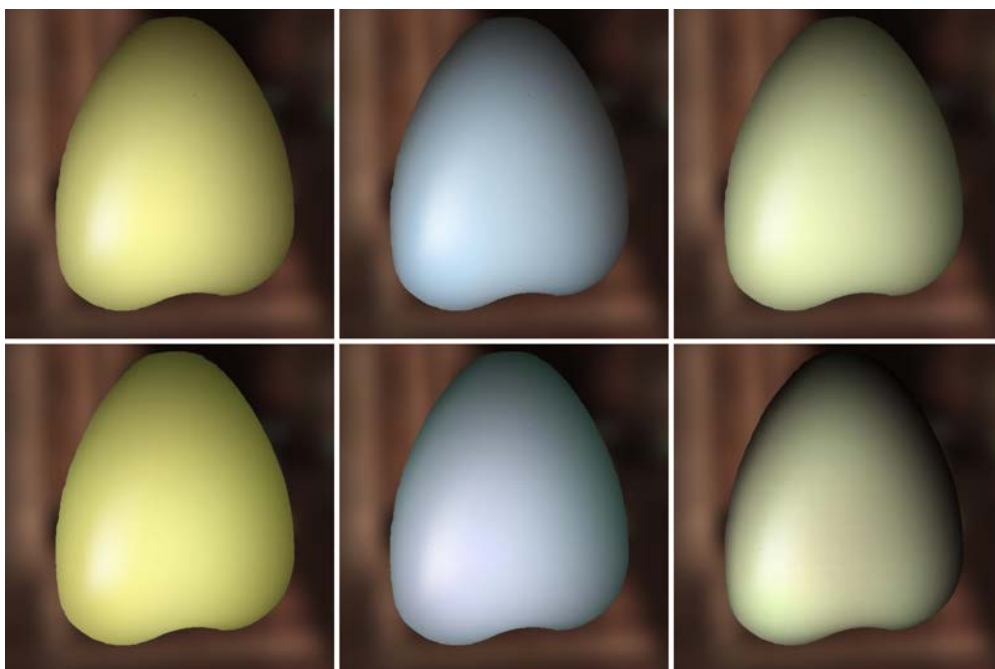


Figure 9.5: Validation of the whole algorithm. Top row: render of measured materials (JMLH01). Bottom row: our resulting estimations without any prior information. From left to right: potato, marble and apple.

As the results show, our method is fairly robust to inaccurate inputs, although it presents some limitations. In the case of uncontrolled images, large errors in depth or light estimations may of course lead to larger errors in the results. Figure 9.7 shows the validation of our two-layer shape from shading method. In Figure 9.8 we can observe how the light detection method proposed in Chapter 3 behaves with increasing values of translucency and geometric complexity. In Figure 9.9 we can observe the effect for even more light source positions. Finally, in Figure 9.10 we show the impact of the light detection error into the BSSRDF estimation error.

We are, therefore, bound by the current state of the art in depth and light approximation algorithms from single images, which in practice means that the algorithm works better with images showing simple, convex shapes lit from one direction. Furthermore, our approximation of the geometry of the back side prevents us from estimating the material from objects that present a strong illumination from its back.

Our algorithm works only with the information that is present in the source image. It is therefore expected to be less accurate with sub-optimal input data when estimating parts of the diffusion profile that are not represented in the source image and thus sub-optimally represented in the captured profile. Figure 9.6 (top row) shows our captured potato material from Figure 9.5 rendered over different geometry and light directions; the bottom row depicts the equivalent results using the physically measured material (JMLH01) for comparison purposes. Our algorithm, handles this lack of information pretty well when geometry or lighting change substantially from the original image. However, when

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

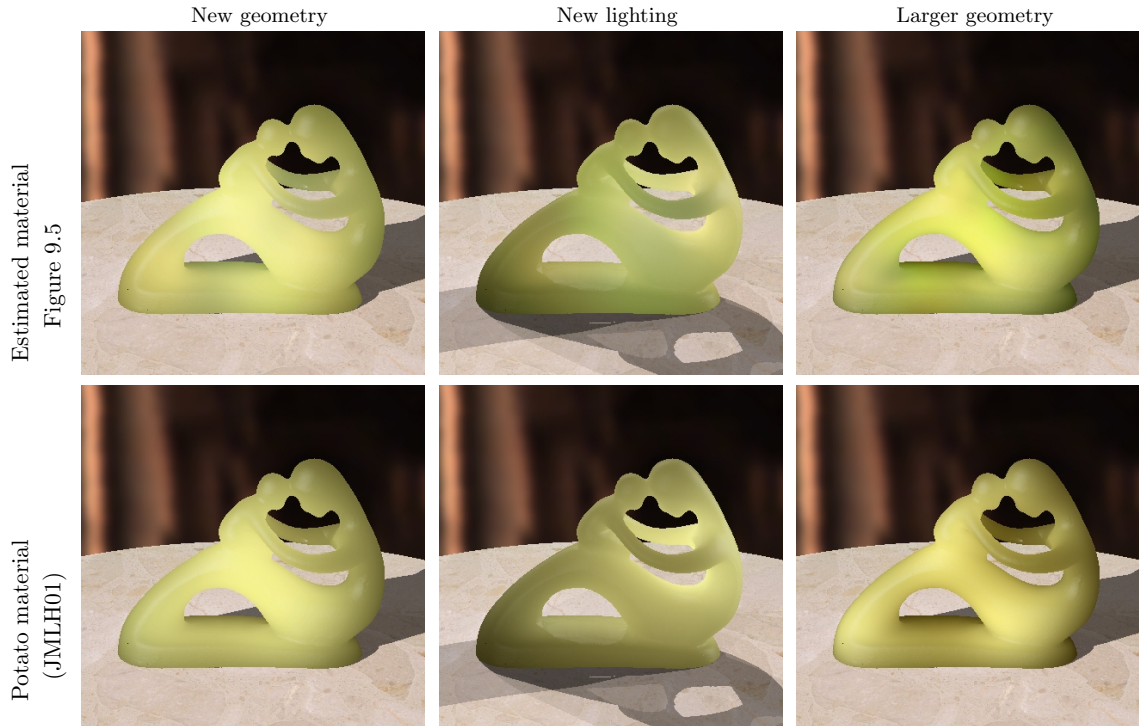


Figure 9.6: Comparison between our estimated potato material from Figure 9.5 and the source potato material from (JMLH01). Left column: Applying both materials to a new geometry. Middle column: Applying both materials to new geometry under new illumination conditions. Right column: Larger size of the geometry

the size of the geometry changes, the final rendering may deviate from the ground truth reference, as the render is accessing parts of the diffusion profile that were not represented in the source image. Nevertheless, the resulting profile is still plausible. Extreme scenarios in which the source image does not contain enough translucency information (no noticeable shading gradients, planar surfaces with no remarkable features or strong back lighting) obviously translate into ill-conditioned linear systems that lead to erroneous profile estimations (which show as different gradients or even color shifts). Figure 9.11, left, shows a small object with little gradients. Conceptually, it only provides information about the leftmost part of the diffusion profile. On the other hand, Figure 9.11, right, presents an object illuminated from behind, which only provides info about the rightmost part of the profile. Both cases translate into numerical instability of the linear system and therefore lead to wrong captures.

Furthermore, by using the diffusion approximation, our work assumes that objects are homogeneous and optically thick, which is not the case for very small objects, or areas that present sharp edges and high curvature surfaces. Violating these assumptions may lead again to wrong profiles, or even make the QMR iterative method fail to converge.

Our method can also be potentially used in an image-editing context, by transferring the captured profile in an image object to another. By applying the same depth estimation technique both to the source and target objects, a new depiction of the latter can be created (see Figure 9.12). The main

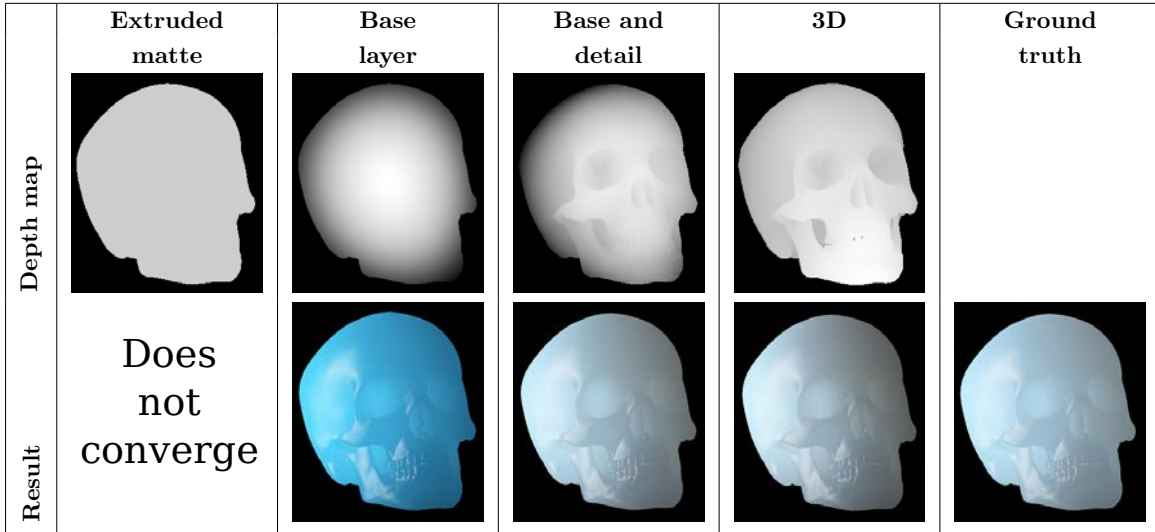


Figure 9.7: Top: results of different depth estimation techniques of increasing complexity, including ground-truth 3D data. Bottom: results of our method for the estimated profiles of the whole milk material, using the different depth maps (source shown at the right-most image). It can be seen how the simplest method does not converge, whereas using only a base layer may lead to unsatisfactory results. A good balance between visual accuracy and simplicity is better achieved with the combination of base and detail layers, yielding results very similar to using the true depth.

drawback of the technique is the double depth estimation process, which tends to accumulate larger errors in the final result.

9.6 Conclusions

The approach presented in this work allows us to approximate a representation of multiple subsurface scattering in optically thick, homogeneous materials from a single image. In the absence of any prior knowledge (geometry and lighting), we face an extremely ill-posed scenario, where a physically accurate solution is simply impossible to obtain. We have shown how to overcome such scenario and still obtain good results, offering an attractive balance between visual accuracy and ease of use. Our acquired data can be directly used for rendering, while also offering a potentially interesting application as an image-editing tool. Our results have given raise to a paper published in the journal *Computer Graphics Forum* (MELM⁺11), which is indexed the 22nd out of 93 of the subject category Computer Science, Software Engineering of the JCR list.

Future research lines include the extension of our technique to heterogeneous materials or more complex BSSRDF models. Our method will also benefit from more advanced light detection and depth extraction algorithms. Our modular image processing design allows for improvements on these aspects, allowing us to extend our results to more complex objects in a wider range of scenarios. Specifically

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

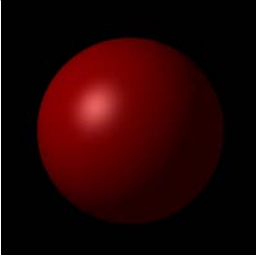

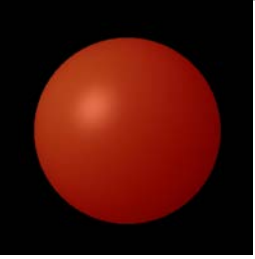
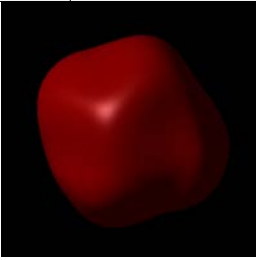
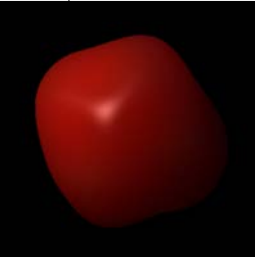
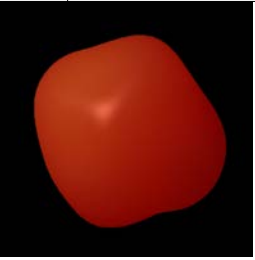


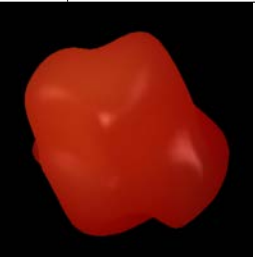
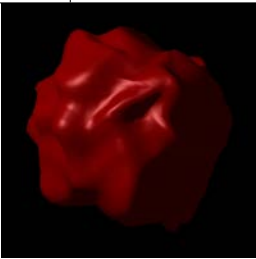

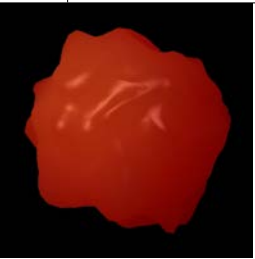
	Opaque		Translucent		Very translucent	
Geometry 1						
	θ	ϕ	θ	ϕ	θ	ϕ
	145.0°	52.1°	142.0°	51.0°	142.7°	50.5°
Error	0.0°	7.1°	-3.0°	6.0°	-2.3°	5.5°
Geometry 2						
	θ	ϕ	θ	ϕ	θ	ϕ
	152.5°	60.5°	149.9°	60.1°	156.2°	52.5°
Error	7.5°	15.5°	4.9°	15.1°	11.2°	7.5°
Geometry 3						
	θ	ϕ	θ	ϕ	θ	ϕ
	155.3°	61.3°	149.9°	60.1°	154.3°	59.8°
Error	10.3°	16.3°	4.9°	15.1°	9.3°	14.8°
Geometry 4						
	θ	ϕ	θ	ϕ	θ	ϕ
	159.2°	50.3°	149.0°	51.8°	153.4°	56.9°
Error	14.2°	5.3°	4.0°	6.8°	8.4°	11.9°

Figure 9.8: Performance of the light detection method for objects with varying degrees of translucency and geometric complexity. The images have been rendered with a directional light source at $(\theta, \phi) = (145^\circ, 45^\circ)$. The results of the light detection algorithm are shown under each image, along with the relative error. The error is always $\epsilon < 20^\circ$, which is below perceptual threshold.

we plan to integrate our parametric shape from shading method (Chapter 4, Section 4.2.2) and *Light source-fitting* estimation method (Chapter 3, Section 3.6.2). In any case, we believe that the range of materials shown demonstrate the current practicality of the method, and hope that the contributions of this work inspire new research in this and other related areas.

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

	Opaque		Translucent		Very translucent	
Light (0, 0)						
	θ	ϕ	θ	ϕ	θ	ϕ
	11.85°	8.63°	-9.75°	2.79°	-10.71°	2.61°
Error	11.85°	8.63°	9.75°	2.79°	10.71°	2.61°
Light (0, 35)						
	θ	ϕ	θ	ϕ	θ	ϕ
	-11.84°	33.17°	-11.84°	36.62°	-11.84°	43.02°
Error	11.84°	1.83°	11.84°	1.62°	11.84°	8.02°
Light (0, 70)						
	θ	ϕ	θ	ϕ	θ	ϕ
	-13.97°	68.33°	-17.30°	72.77°	-12.84°	81.53°
Error	13.97°	1.67°	17.30°	2.77°	12.84°	11.53°
Light (120, 0)						
	θ	ϕ	θ	ϕ	θ	ϕ
	122.62°	22.57°	120.96°	20.60°	120.11°	12.84°
Error	2.62°	22.57°	0.96°	20.60°	0.11°	12.84°
Light (120, 35)						
	θ	ϕ	θ	ϕ	θ	ϕ
	120.96°	35.59°	123.42°	33.78°	124.99°	34.73°
Error	0.96°	0.59°	3.42°	1.22°	4.99°	0.27°

	Opaque		Translucent		Very translucent	
Light (120, 70)						
	θ	ϕ	θ	ϕ	θ	ϕ
	123.42°	59.55°	120.96°	63.81°	93.43°	71.81°
Error	3.42°	10.45°	0.96°	6.19°	26.57°	1.81°
Light (60, 0)						
	θ	ϕ	θ	ϕ	θ	ϕ
	56.58°	21.38°	57.38°	10.47°	61.63°	8.92°
Error	3.42°	21.38°	2.62°	10.47°	1.63°	8.92°
Light (60, 35)						
	θ	ϕ	θ	ϕ	θ	ϕ
	55.78°	23.59°	56.58°	23.30°	60.75°	24.45°
Error	4.22°	11.41°	3.42°	11.70°	0.75°	10.55°
Light (60, 70)						
	θ	ϕ	θ	ϕ	θ	ϕ
	54.25°	57.51°	55.01°	60.81°	59.63°	65.71°
Error	5.75°	12.49°	4.99°	9.19°	0.37°	4.29°

Figure 9.9: Performance of the light detection method for an object with varying degrees of translucency and different light directions. The images have been rendered with a directional light source at the specified (θ, ϕ) directions. The results of the light detection algorithm are shown under each image, along with the error. For translucent materials, the error is always $\epsilon < 20^\circ$, which is below perceptual threshold. An exception occurs in the case of $(\theta = 120^\circ, \phi = 0^\circ)$, for which the error in ϕ is 20.60° .

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES

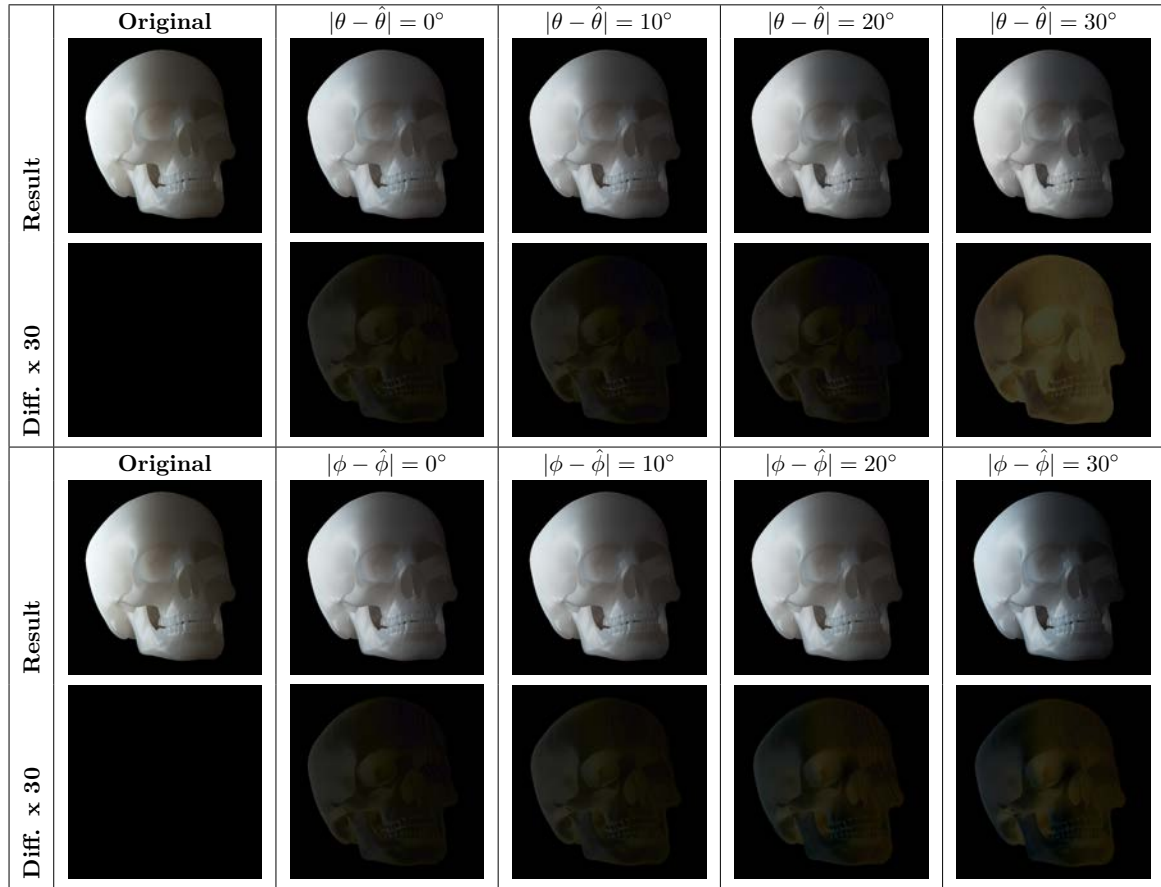


Figure 9.10: Behavior of the BSSRDF estimation algorithm according to the error on the light estimation (both on azimuth and zenith). The resulting renderings are visually accurate up to an error of 20° .

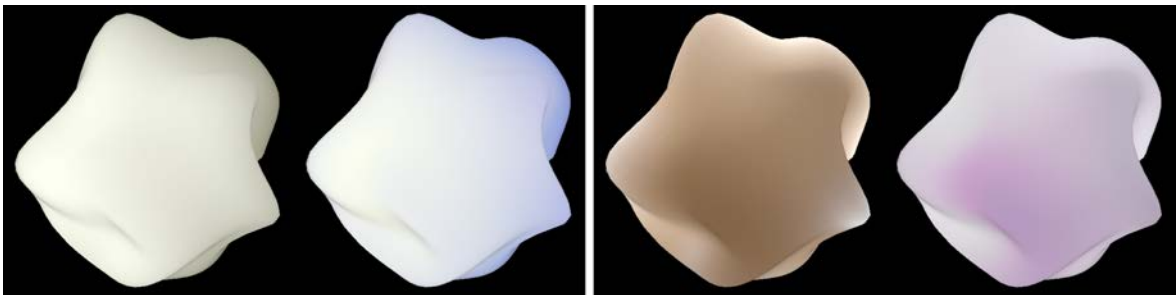


Figure 9.11: Examples of failure cases. Left: source apple rendering with poor subsurface scattering information and its captured material. Right: source marble rendering with strong backlight and its estimated material. The lack of information on the image or breaking our initial assumptions may lead to wrong profiles, even in controlled setups.



Figure 9.12: Example of our technique as an image-editing tool. From left to right: original photograph, transfer of the wax material from the candle to the owl, and transfer from the purple wax in Figure 9.1 to the owl.



Figure 9.13: The estimated BSSRDF for the grape material in Figure 9.1, used to render different geometries under different lighting conditions.

9. APPLICATION 4: BSSRDF ESTIMATION FROM SINGLE IMAGES



Figure 9.14: Results of our algorithm. The small insets show the original images where the material properties are acquired from (please refer to the supplementary material for the complete data). In reading order, blue soap, whole milk, purple soap, ketchup, orange juice, whitish soap, liquid detergent, skin and greenish soap.

References

- [BBC⁺94] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, R. Pozo, V. Eijkhout, H. Van der Vorst, and C. Romine, *Templates for the solution of linear systems: Building blocks for iterative methods, 2nd edition*, SIAM, 1994. 182, 185
- [BG01] S. Boivin and A. Gagalowicz, *Image-based rendering of diffuse, specular and glossy surfaces from a single image*, ACM Transactions on Graphics (SIGGRAPH) (2001), 107–116. 181
- [dLE07] E. d’Eon, D. Luebke, and E. Enderton, *Efficient rendering of human skin*, Eurographics Symposium on Rendering (Grenoble, France) (J. Kautz and S. Pattanaik, eds.), Eurographics Association, 2007, pp. 147–157. 185
- [DRS07] Julie Dorsey, Holly Rushmeier, and François Sillion, *Digital modeling of material appearance*, Morgan Kaufmann/Elsevier, 2007. 181
- [DS03] Carsten Dachsbacher and Marc Stamminger, *Translucent shadow maps*, , EGRW ’03, Eurographics Association, 2003, pp. 197–201. 183
- [DWd⁺08] Craig Donner, Tim Weyrich, Eugene d’Eon, Ravi Ramamoorthi, and Szymon Rusinkiewicz, *A layered, heterogeneous reflectance model for acquiring and rendering human skin*, ACM Transactions on Graphics (SIGGRAPH Asia) **27** (2008), no. 5, 1–12. 181
- [FB05] R. W. Fleming and H. H. Bühlhoff, *Low-level image cues in the perception of translucent materials*, ACM Transactions on Applied Perception (TAP) **2** (2005), no. 3, 346–382. 188
- [GHP⁺08] Abhijeet Ghosh, Tim Hawkins, Pieter Peers, Sune Frederiksen, and Paul Debevec, *Practical modeling and acquisition of layered facial reflectance*, ACM Transactions on Graphics (SIGGRAPH Asia) **27** (2008), no. 5, 1–10. 181
- [GJJD09] Diego Gutierrez, Henrik Wann Jensen, Wojciech Jarosz, and Craig Donner, *Scattering*, , SIGGRAPH ASIA ’09, ACM, 2009, pp. 15:1–15:620. 179

REFERENCES

- [GLL⁺04] Michael Goesele, Hendrik P. A. Lensch, Jochen Lang, Christian Fuchs, and Hans-Peter Seidel, *Disco: acquisition of translucent objects*, ACM Transaction on Graphics (SIGGRAPH) (2004), 835–844. 181
- [GLMF⁺08] Diego Gutierrez, Jorge Lopez-Moreno, Jorge Fandos, Francisco J. Seron, Maria P. Sanchez, and Erik Reinhard, *Depicting procedural caustics in single images*, ACM Transaction on Graphics (SIGGRAPH Asia) **27** (2008), no. 5, 1–9. 188
- [JB02] H. W. Jensen and J. Buhler, *A rapid hierarchical rendering technique for translucent materials*, ACM Transactions on Graphics (SIGGRAPH) (2002), 576–581. 182
- [JMLH01] H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan, *A practical model for subsurface light transport*, ACM Transactions on Graphics (SIGGRAPH) (2001), 511–518. 179, 181, 182, 183, 185, 186, 187, 191, 192
- [Joh02] Scott F. Johnston, *Lumo: illumination for cel animation*, NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering (New York, NY, USA), ACM, 2002, pp. 45–52. 188
- [KRFB06] E. A. Khan, E. Reinhard, R. W. Fleming, and H. H. Bühlhoff, *Image-based material editing*, ACM Transactions on Graphics (SIGGRAPH) (2006), 654–663. 182, 188
- [LB00] Michael Langer and Heinrich H Bühlhoff, *Depth discrimination from shading under diffuse lighting*, Perception **29** (2000), no. 6, 649–660. 188
- [LKG⁺03] H. P. A. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H-P. Seidel, *Image-based reconstruction of spatial appearance and geometric detail*, ACM Transactions on Graphics (TOG) **22** (2003), no. 2, 234–257. 181
- [LMHRG10] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez, *Compositing images through light source detection*, Computers & Graphics **In press** (2010). 188, 189
- [MELM⁺11] Adolfo Muñoz, Jose I. Echevarria, Jorge Lopez-Moreno, Francisco Serón, Mashhuda Glencross, and Diego Gutierrez, *Bssrdf estimation from single images*, Computer Graphics Forum (Proc. of EUROGRAPHICS) (2011). 179, 187, 193
- [MMTG09] Adolfo Munoz, Belen Masia, Alfonso Tolosa, and Diego Gutierrez, *Single-image appearance acquisition using genetic algorithms*, , CGVCIIP '09, 2009, pp. 24–32. 181, 182
- [MSY09] Y. Mukaigawa, K. Suzuki, and Y. Yagi, *Analysis of subsurface scattering based on dipole approximation*, IPSJ Transactions on Computer Vision and Applications **1** (2009), 128–138. 181, 183, 186
- [Muñ10] Adolfo Muñoz, *Light transport in participating media*, Ph.D. thesis, University of Zaragoza, SPAIN, 2010. 179, 180, 181

-
- [NKGR06] S.K. Nayar, G. Krishnan, M. D. Grossberg, and R. Raskar, *Fast Separation of Direct and Global Components of a Scene using High Frequency Illumination*, ACM Transactions on Graphics (SIGGRAPH) **25** (2006), no. 3, 935–944. 181
- [PvBM⁺06] Pieter Peers, Karl vom Berge, Wojciech Matusik, Ravi Ramamoorthi, Jason Lawrence, Szymon Rusinkiewicz, and Philip Dutré, *A compact factored representation of heterogeneous subsurface scattering*, ACM Transactions on Graphics (SIGGRAPH) (2006), 746–753. 181
- [SMGG01] Peter-Pike J. Sloan, William Martin, Amy Gooch, and Bruce Gooch, *The lit sphere: a model for capturing npr shading from art*, , GRIN’01, Canadian Information Processing Society, 2001, pp. 143–150. 181
- [ST06] L. Shen and H. Takemura, *Spatial reflectance recovery under complex illumination from sparse images*, Computer Vision and Pattern Recognition, IEEE, 2006, pp. 1833–1838. 181
- [STPP09] Y. Song, X. Tong, F. Pellacini, and P. Peers, *Subedit: A representation for editing measured heterogeneous subsurface scattering*, ACM Transactions on Graphics (SIGGRAPH) **28** (2009), no. 3, 1–10. 181, 189
- [TGL⁺06] S. Tariq, A. Gardner, I. Llamas, A. Jones, P. Debevec, and G. Turk, *Efficient estimation of spatially varying subsurface scattering parameters*, Workshop on Vision, Modeling, and Visualization (VMW) (Aachen, Germany), 2006. 181
- [WLL⁺08] Tim Weyrich, Jason Lawrence, Hendrik Lensch, Szymon Rusinkiewicz, and Todd Zickler, *Principles of appearance acquisition and representation*, ACM SIGGRAPH 2008 classes (New York, NY, USA), ACM, 2008, pp. 1–119. 179, 181
- [WMP⁺06] Tim Weyrich, Wojciech Matusik, Hanspeter Pfister, Bernd Bickel, Craig Donner, Chien Tu, Janet McAndless, Jinho Lee, Addy Ngan, Henrik Wann Jensen, and Markus Gross, *Analysis of human faces using a measurement-based skin reflectance model*, ACM Transactions on Graphics (SIGGRAPH) **25** (2006), no. 3, 1013–1024. 181
- [WS02] Y. Wang and D. Samaras, *Estimation of multiple illuminants from a single image of arbitrary known geometry*, European Conference on Computer Vision (ECCV), Springer, 2002, pp. 272–288. 189
- [WT06] T-P Wu and C-K Tang, *Separating subsurface scattering from photometric image*, International Conference on Pattern Recognition (ICPR), IEEE Computer Society, 2006, pp. 207–210. 181
- [WZT⁺08] J. Wang, S. Zhao, X. Tong, S. Lin, Z. Lin, Y. Dong, B. Guo, and H-Y. Shum, *Modeling and rendering of heterogeneous translucent materials using the diffusion equation*, ACM Transactions on Graphics **27** (2008), no. 1, 1–18. 181, 183

REFERENCES

- [XGL⁺07] K. Xu, Y. Gao, Y. Li, T. Ju, and S-M. Hu, *Real-time homogenous translucent material editing*, Computer Graphics Forum **26** (2007), no. 3, 545–552. 183, 186
- [ZY01] Y. Zhang and Y-H. Yang, *Multiple illuminant direction detection with application to image synthesis*, IEEE Transactions on Pattern Analysis and Machine Intelligence **23** (2001), no. 8, 915–920. 189

Chapter 10

Conclusions and Future Work

In Section 1.3 we exposed that the goal of this PhD which is to extend the set of tools available to artists to effect high level edits in single images, without the need to painstakingly paint over all pixels.

Along this dissertation we have proved that this is possible by leveraging the limitations of our perception and extending the edition process to a multidimensional space. To this end, we have presented a single image editing pipeline and proposed several novel algorithms in order to extract information like depth, material properties or illumination from a single image:

- In Chapter 2, we measure quantitatively the accuracy of human vision detecting lighting inconsistencies in images. Our research suggests a perceptual threshold for multiple configurations which have been used in the design of our light source estimation algorithms. We even shown that this threshold seems to be even larger for real-world scenes.
- For depth estimation, we have explored several existing shape from shading techniques, implementing novel variations based either in the perception of depth (used for most of our applications), or in the previous knowledge of the light sources (See Chapter 4).
- Regarding light source estimation, in Chapter 3 we have introduced and validated two novel methods which are, to our knowledge, the first solutions in the literature to multiple light detection from arbitrary shapes in a single image (no depth information required).
- For intrinsic image decomposition, in Chapter 5 we have explored the limits of bilateral filtering, proposing a novel algorithm based in albedo segmentation and optimization.

Furthermore, we have introduced four new applications for single image editing based on our processing pipeline (described in Section 1.2) which turn complex edits, only achievable by skilled artists

10. CONCLUSIONS AND FUTURE WORK

at the expense of considerable time and effort, into semi-automatic processes feasible for unskilled users at interactive rates:

- We have simulated the complex process of light transport in participating media (fog in Chapter 6 and caustics from transparent objects in Chapter 7) by means of two-dimensional analysis and filtering. Our results match perceptually those achievable by ground truth simulation (photon mapping) if 3D information were available. In the case of two-dimensional editing, where physically based simulation is not possible, our methods perform better than professional artists using commercial tools at a fraction of time and effort.
- In Section 3.8.3 we have shown novel relighting and compositing methods based on light detection and depth estimation.
- We have applied our processing pipeline to the design of novel non-photorealistic stylization techniques in Chapter 8, implementing a real-time editing tool as proof of concept.
- Finally, our light and depth estimation methods made possible the capture of complex materials with subsurface scattering properties from a single image, as shown in Chapter 9.

10.1 Future Work

Although this thesis has extended the range of edits available for single image processing, there is still room for improvement and additional research. We have already mentioned some future lines of work at each chapter but let us summarize the most relevant ones:

- Some of our algorithms can be directly applied to video sequences, however, for certain cases (e.g.:our kinetic lines filter in Chapter 8), frame-to-frame coherency is not granted and additional techniques such as optical flow analysis need to be considered.
- Our RBF shape from shading implementation could be extended to incorporate multiple light sources into the system. Likewise, a parallel version is feasible and would allow us to provide a swift interaction in the form of real-time user strokes which add constraints to the solver.
- We would like to explore our algorithms at multiple levels of detail. We think that light detection, intrinsic image decomposition and shape from shading would benefit from a different processing at each frequency level.
- Our perception studies have provide us with useful thresholds for algorithm design, however additional refinement of these limits, taking into account different render styles and visual complexity, would improve the accuracy of future works in the area.