

Image Composition in Computer Rendering

by

Li Ji

B.Eng., Shanghai Jiaotong University, 2008

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Computer Science

© Li Ji, 2016

University of Victoria

All rights reserved. This dissertation may not be reproduced in whole or in part, by photocopying or other means, without the permission of the author.

Image Composition in Computer Rendering

by

Li Ji

B.Eng., Shanghai Jiaotong University, 2008

Supervisory Committee

Dr. Brian Wyvill, Supervisor
(Department of Computer Science)

Dr. Amy Gooch, Supervisor
(Department of Computer Science)

Lynda Gammon, Outside Member
(Department of Visual Arts)

ABSTRACT

In this research, we study image composition in the context of computer rendering, investigate why composition is difficult with conventional rendering methods, and propose our solutions. Image composition is a process in which an artist improves a visual image to achieve certain aesthetic goals, and it is a central topic in studies of visual arts. Approaching the compositional quality of hand-made art work with computer rendering is a challenging task; but there is scarcely any in-depth research on this task from an interdisciplinary viewpoint between computer graphics and visual arts. Although recent developments of computer rendering have enabled the synthesis of high quality photographic images, most rendering methods only simulate a photographic process and do not permit straightforward compositional editing in the image space. In order to improve the visual quality of the digitally synthesized images, the knowledge of visual composition needs to be incorporated. This objective not only asks for novel algorithmic inventions, but also involves research in visual perception, painting, photography and other disciplines of visual arts.

With examples from historical painting and contemporary photography, we inquire why and how a well-composed image elicits an aesthetic visual response from its viewer. Our analysis based on visual perception shows that the composition of an image serves as a guideline for the viewing process of that image; the composition of an image conveys an artist's intention of how the depicted scene should be viewed, and directs a viewer's eyes. A key observation is that for a composition to take effect, a viewer must be allowed to attentively look at the image for a period of time. From this analysis, we outline a few rules for composing light and shade in computer rendering, which serve as guidelines for designing rendering methods that create imagery beyond photorealistic depictions. Our original analysis elucidates the mechanism and function of image composition in the context of rendering, and offers clearly defined directions for algorithmic design. Theories about composition mostly remain in the literature of art critique and art history, while there are hardly any investigations on this topic in a technical context. Our novel analysis is an instructive contribution for enhancing the aesthetic quality of digitally synthesized images.

We present two research projects that develop our analysis into rendering programs. We first show an interpolative material model, in which the surface shading is interpolated from input textures with a brightness value. The resultant rendering depicts surface brightness instead of light energy in the depicted scene. We also

show a painting interface with this material model, with which an artist can directly compose surface brightness with a digital pen. In the second project, we ask an artist to provide a sketch of lighting design with coarse paint strokes on top of a rendering, while details of the light and shade in the depicted scene are automatically filled in by our program. This project is staged in the context of creating the visual effects of foliage shadows under sunshine. Our software tool also includes a novel method for generating coherent animations that resemble the movements of tree foliage in a gentle breeze. These programming projects validate the rendering methodology proposed by our theoretical analysis, and demonstrate the feasibility of incorporating compositional techniques in computer rendering.

In addition to programming projects, this interdisciplinary research also consists of practices in visual arts. We present two art projects of digital photography and projection installation, which we built based on our theoretical analysis of composition and our software tools from the programming projects. Through these art projects, we evaluate our methodology by both making art ourselves and critiquing the resultant pieces with peer artists. From our point of view, it is important to be involved in art practices for rendering researchers, especially those who deal with aesthetic issues. The valuable first-hand experiences and the communications with artists in a visual arts context are rarely reported in the rendering literature. These experiences serve as effective guides for the future development of our research on computer rendering.

The long term goal of our research is find a balance between artistic expression and realistic believability, based on the interdisciplinary knowledge of composition and perception, and implemented as either automated or user-assisted rendering tools. This goal may be termed as to achieve a *staged realism*, to synthesize images that are recognizable as depictions of realistic scenes, and at the same time enabling the freedom of composing the rendering results in an artistic manner.

Contents

Supervisory Committee	ii
Abstract	iii
Table of Contents	v
List of Figures	vi
Acknowledgements	viii
1 Introduction	1
2 Variations in Shading	22
2.1 Realistic Perception and Monet’s Haystacks	23
2.2 The Artist’s Shade	31
2.3 An Interpolative Material Model	39
3 Variations in Lighting	46
3.1 Extra-scene Light Transport	47
3.2 Light and Shadow Effects of Tree Foliage	52
3.3 Creating Light Patterns from an Artist’s Sketch	55
3.4 Animation with Harmonic Motion	67
4 Art Practice and Future Work	74
4.1 Brightness Composition in Photography	76
4.2 Light Art Projection Installation	81
4.3 User Study of the Light Pattern Rendering Method	86
5 Conclusion	93
References	96

List of Figures

Figure 1.1 Johannes Vermeer, <i>The Art of Painting</i>	1
Figure 1.2 Tom Blackwell, <i>Triumph Trumpet</i>	2
Figure 1.3 An example of physically based photorealistic rendering.	3
Figure 1.4 Examples of Non-photorealistic rendering.	4
Figure 1.5 Example of computer graphics systems that take image space input.	7
Figure 1.6 Depth-first order of computer rendering.	8
Figure 1.7 Breadth-first order of traditional art media.	9
Figure 1.8 Drawing objects in a linear perspective.	10
Figure 1.9 Drawing with one, two and three point perspectives.	11
Figure 1.10 Three flower paintings of Jan Davidsz de Heem.	13
Figure 1.11 Example based NPR rendering.	16
Figure 1.12 User-assisted lighting composing for photographs.	17
Figure 1.13 Algorithms aimed at non-local image space visual features.	18
Figure 2.1 Tennis game on Atari 2600.	22
Figure 2.2 Histogram analysis of Monet’s Haystack Painting.	24
Figure 2.3 Anatomy of an eye.	26
Figure 2.4 Johannes Vermeer, <i>Woman in Blue Reading a Letter</i>	32
Figure 2.5 Jeff Wall, <i>A View from an Apartment</i>	34
Figure 2.6 Johannes Vermeer, <i>The Music Lesson</i> , and its reconstructions.	35
Figure 2.7 Details of <i>The Music Lesson</i>	36
Figure 2.8 Harmen Steenwijck, <i>An Allegory of the Vanities of Human Life</i>	37
Figure 2.9 The interpolative material model.	40
Figure 2.10 User defined shading curves.	42
Figure 2.11 Painting interface for brightness value editing	43
Figure 3.1 Evaluating the perception of inconsistent lighting.	48
Figure 3.2 Scott McFarland, <i>The Granite Bowl in the Berlin Lustgarten</i> (after Johann Erdmann Hummel).	50

Figure 3.3 Scott McFarland, <i>Torn Quilt with the Effects of Sunlight</i> , 2003.	51
Figure 3.4 Film crew altering the natural light with a cucoloris.	52
Figure 3.5 Real life foliage shadows.	53
Figure 3.6 An example of tree animation methods.	54
Figure 3.7 Sketching the guide image for leaf shadows	56
Figure 3.8 Example scene setup for rendering the foliage shadow effect. . .	57
Figure 3.9 Transforms of a shape instance.	58
Figure 3.10The process of stochastic optimization.	59
Figure 3.11Optimization heuristics.	61
Figure 3.12Foliage shadow effect rendering for the Lucy scene.	63
Figure 3.13Soft shadow effect under sunshine.	64
Figure 3.14Rendering the soft shadow effect.	65
Figure 3.15Field study of foliage shadows.	66
Figure 3.16Simulation of stochastic harmonic motion.	68
Figure 3.17Movement of a shape group.	70
Figure 3.18A Buddha scene rendered with the foliage shadow effects. . . .	72
Figure 3.19More rendering results of the Buddha scene.	73
Figure 4.1 Final image of our photography art project.	75
Figure 4.2 Source photographs of the phtography art project.	77
Figure 4.3 Painted Brightness Mask.	79
Figure 4.4 Charles Sandison, <i>Chamber</i>	81
Figure 4.5 Light art projection installation.	82
Figure 4.6 Implementation plan of the installation.	83
Figure 4.7 Guide image and example video frame.	84
Figure 4.8 Photographs of our installation show.	85
Figure 4.9 User Study Questionnaire.	88
Figure 5.1 The iterative process of image composition.	93

ACKNOWLEDGEMENTS

First of all, I wish to thank the supervisory committee for the comments, suggestions, and guidance they have given me throughout this winding path and finally to this work. In addition, I wish to thank Professor Paul Walde and Professor Vikky Alexander for the instruction and help in exploring the artistic facets of my research. I wish to also acknowledge my colleagues in the graphics labs. In particular, I would like to thank Dr. Jeremy Long for the discussion and help in my research projects, and to thank Eduard Wisernig and Mauricio Andres Rovira Galvez for helping me proofreading this manuscript. I wish to express my sincere gratitude to the department secretaries and staff, who helped me stay on track throughout the administrative steps, and most importantly to my family and friends for all their kind support and help during these years of study.

CHAPTER 1

INTRODUCTION

Image composition is a key topic in various forms of visual arts throughout their historical developments (Figure 1.1); yet computer graphics research on this topic is scarce and limited. In this work, we examine the idea of image composition in detail, and investigate how to facilitate composition with computer rendering. With traditional art media, composition may be thought as a process in which an artist ‘plays’ with an image, changing the colour, light and shade on the image, until the result appears satisfying. For example, an artist making a drawing or painting may take an initial sketch, then adjust the studio setting and the posture of the sitter, return to his canvas and try to reproduce the scene again. With the advancement of digital media, artists also spend a lot of time using software tools to touch up images or videos on the screen. For both traditional and digital art media, each individual artist may prefer a different visual style, and every piece of artwork may have a different goal for visual effects. Despite these differences, there is a common process of making image compositions in both traditional and digital media; and successfully composed images elicit visual response from their viewers with common mechanisms. Reflecting these common properties in computer rendering is the central theme of our research.



Figure 1.1: Johannes Vermeer, *The Art of Painting*, Kunsthistorisches Museum, Austria. This well-known image is an allegory of painting composition, demonstrating the relationship between the artist, the studio and the sitter.



Figure 1.2: Tom Blackwell, *Triumph Trumpet*, 1977. Oil on Canvas, 180 x 180 cm.

The mainstream methodology of computer rendering is to simulate a photographic process, in which the light passes through the lens of a camera, and imprints an image on the negative. Since the optical process of taking a photograph is well understood, and the field was pioneered by scientists from physics and mathematics backgrounds, physically based photorealistic rendering naturally became a major research direction. Furthermore, this preference for a photorealistic rendering style is related to the broader social and historical context. At the time when computer graphics was invented, photographic images dominated the mass media such as newspaper and TV, delivering stories from foreign affairs to community events. The affordable film cameras at that time enabled everybody to take snapshots and fill albums with photos from daily lives. In this photographic era, we rely heavily on photographic images to perceive the world around us. As observed by art historian Linda Chase:

The photograph, for all its ubiquity, offered the Photorealist a realm that had never really been fully explored by art. ... When Degas employed photographic perspective and distortion, it was considered a bizarre way of seeing and depicting the world, an aberration. Today these photographic aberrations are so commonplace that we can look at a painting that employs extreme distortion and out-of-focus areas and comment on how real it looks. In the nineteenth century,



Figure 1.3: An example of physically based photorealistic rendering. Image synthesized by the Intel Embree ray tracing framework (Wald et al. 2014).

arguments raged over what was real — the photographic depiction, artistic convention, or the way the eye perceives — and the photograph’s truthfulness was often questioned. Nevertheless, the visual language of the photographic process soon gained an aura of validity that took precedence over all other ways of seeing. “We accept the photograph as real,” observed Richard Estes in a 1972 interview, “Media has to affect the way you see things.” And Tom Blackwell took the idea even further: “Photographic images, movies, TV, newspapers are as important as actual phenomena. They affect our perception of actual phenomena.” (Chase 2002)

The applications of computer graphics in the consumer market produce imagery to be shown on media that are already dominated by photographic images. By aligning itself with the established photographic style of visual representation, digital rendering acquired broad attention from the visual effects and video game industries, which facilitated its rapid development in the last few decades. The computer graphics literature borrowed the term ‘photorealistic’ from the art world; in its original context it means a genre of painting that “appears like a real (untouched) photograph” (Meisel 2002; Letze 2013). One well-known painting of this genre is shown in Figure 1.2. In the computer graphics context, we tend to use this term to mean ‘as real as a photograph’ (Figure 1.3). This notion assumes that photographs faithfully represent reality, and reflects the previous discussion about the ubiquity and validity of photography in the contemporary world. For some applications the rendering result must be photographic, such as injecting virtual objects into live-action film footages, where the synthesized images have to be consistent with the captured images from a camera.



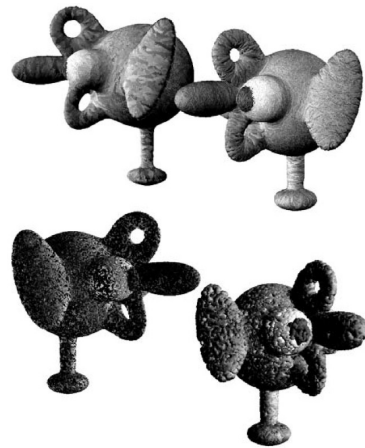
(a) Paint-stroke effect rendered with geograftals (Kaplan et al. 2000).



(b) The Olaf model with Cel-shading (Lake et al. 2000).



(c) A barn rendered with the pencil drawing effect (Meraj et al. 2008).



(d) The implicit painting method using a particle system to place strokes onto the surface of an implicit model (Akleman 1998).

Figure 1.4: Examples of non-photorealistic rendering methods for three dimensional models.

Rendering researchers have also acknowledged that photorealism is only one style of visual representation, and that imagery different from photographs can be quite effective in communicating information and conveying aesthetic qualities. Curious researchers have taken examples and inspirations from various art media, and created rendering methods that imitate them. Since these rendering methods synthesize imagery different from photographs, they are categorized into a field called “Non-photorealistic Rendering” (NPR) (Gooch and Gooch 2001; Strothotte and Stefan 2002) (Figure 1.4). This rather descriptive title reflects the overwhelming dominance of photorealistic rendering in computer graphics (Salesin 2002). NPR research can be roughly categorized into two major topics. The first category deals with how to simulate a particular visual art *gesture*. For example, researchers have thoroughly examined how to simulate various kinds of paint-strokes (Baxter et al. 2004; Lu et al. 2013; Ning et al. 2011; Chu and Tai 2005), and to render a given three-dimensional scene with paint-brushes (Meier 1996; Northam et al. 2012; Kalnins et al. 2002; Klein et al. 2000; Akleman 1998). Research on line drawing and hatching has led to methods that automatically generate line drawings (Meraj et al. 2008; DeCarlo et al. 2003; Lu et al. 2012; Winkenbach and Salesin 1994). The second category of NPR research aims at *stylized shading*, and it is initialized by the two-level Cel-shading methods (Lake et al. 2000). Developments in this category have introduced methods that are capable of simulating a wide range of cartoon-like, illustrative effects, with the ability of rendering real-time, coherent animations (Barla et al. 2006; Anjyo and Hiramitsu 2003; Todo et al. 2007). Research in both categories has also led to various image space rendering methods, which are capable of transferring a given image into a specific painting, drawing or illustrative style (DeCarlo and Santella 2002; Mould and Grant 2008; Winnemöller 2011).

Although NPR methods seem to be more ‘artistic’ than physically based photorealistic rendering, the fundamental photographic metaphor is consistent among these two categories. To illustrate this idea, let us consider film photography as an analogy. A photographer working with a film camera first sets up the studio and adjusts the camera, then makes an exposure on the negative. Before the negative is developed, the photographer cannot know for sure how the photograph will appear. If the photograph does not meet the photographer’s aesthetic criterion, (s)he may need to go back to re-arrange the studio setting and re-take the shot, if that is possible¹. Similarly, in

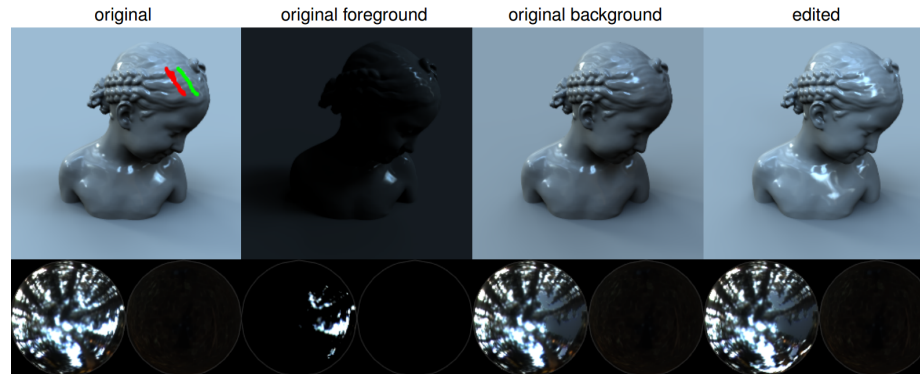
¹Our discussion here is about staged studio photography. Photojournalism and photography documentation have different methodologies, which generally do not permit the photographer to

order to change the light and shade on a rendering result, a digital artist needs to go back to the *modelling interface*, to change the object space geometry, material and lighting configuration. Until the artist runs the rendering pipeline again, either photorealistic or NPR, it is typically difficult to predict exactly what the synthesized image will look like. This photographic metaphor implies that image space visual features exist as a *causal consequence* of the object space information. Within this metaphor, it is meaningless to talk about altering a consequence without changing its cause. The direct interactions between the artist and the *image* is excluded from this approach of computer rendering, and the rendering program serves as more or less a black box that projects the object space onto the image plane.

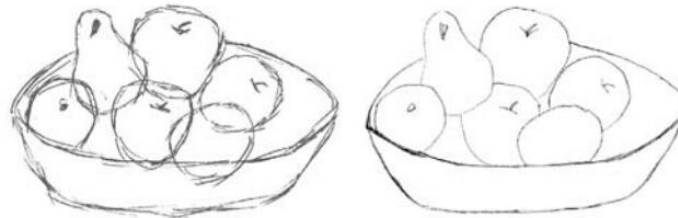
In our opinion, this assumed causality between object space information and image space visual features is at the core of the difficulty of image composition with computer rendering. This difficulty of approaching the visual quality of hand-made art works with digital rendering has been discussed in several essays on expressive and artistic rendering (Durand 2002; Hertzmann 2010). To this end, researchers have noted that an artist tends not to make an exact copy of the observed reality but to add subjective touches on top of the objective observation. Artists who draw or paint by hand frequently invent physically impossible light and shade on objects' surfaces. They also routinely modify objects' positions and sizes on the canvas after the initial sketch is done, often at a point when adjustments of the original real scene have become impossible. Each kind of these artistic inventions and modifications is well understood as a *phenomenon*, and corresponding rendering tools are available. For example, adjustments of light and shade on a canvas can be related to the 'reverse lighting' and material designing methods (Pellacini et al. 2007; Pellacini 2010; Ritschel et al. 2010; Ritschel et al. 2009) (Figure 1.5a). Adjustments of object geometry and position can be implemented with sketch-based modelling and example-based layout methods (Olsen et al. 2009; Pusch et al. 2007; Fisher et al. 2012) (Figure 1.5b). On the other hand, the *reason* for these artistic adjustments remains mostly speculative. Theories about why, how, and to what extent an artist would alter the observed reality remain in the literature of art critique and art history. There are scarcely any interpretations on this issue in the context of computer graphics to serve as a guideline for algorithmic design. We propose our own interpretation in the following.

The end results for both computer rendering and traditional art media are planar

modify the subject matter or retake shots due to aesthetic judgements. For a brief exposition on these topics of photography, see Edwards (2006)



(a) An example reverse-lighting system, which let its user paint the specular light location, then adjusts the environment map to achieve the desired lighting effect (Pellacini 2010).



(b) Sketch-based modelling methods are capable to transfer an artist's drawing into clean curves and build three dimensional models (Pusch et al. 2007).

Figure 1.5: Examples of computer graphics systems that take image space input.

images, either on a piece of paper, a sheet of canvas or a computer screen; but the order in which such an image is constructed is quite different. Rendering methods in computer graphics define each pixel's colour value independently from all other pixels on the same image. The rendering system collects all relevant parameters for a specific pixel, and determines the pixel colour just from this set of information. Because pixel values are supposed to be causal consequences of object space information, the rendering method only consults a subset of the object space data without considering relationships of pixel colours in the image space (Figure 1.6). In recent years, parallel computing has become one of the fundamental features of high performance rendering. This trend further asks for algorithms that locally compute each pixel without interfering with each other. The overall appearance of the output image, either photorealistic or non-photorealistic, emerges as a synergy of these isolated calculations. The following statement from the "Advanced Renderman" book from Pixar (Apodaca and Larry 1999) illustrates this approach:

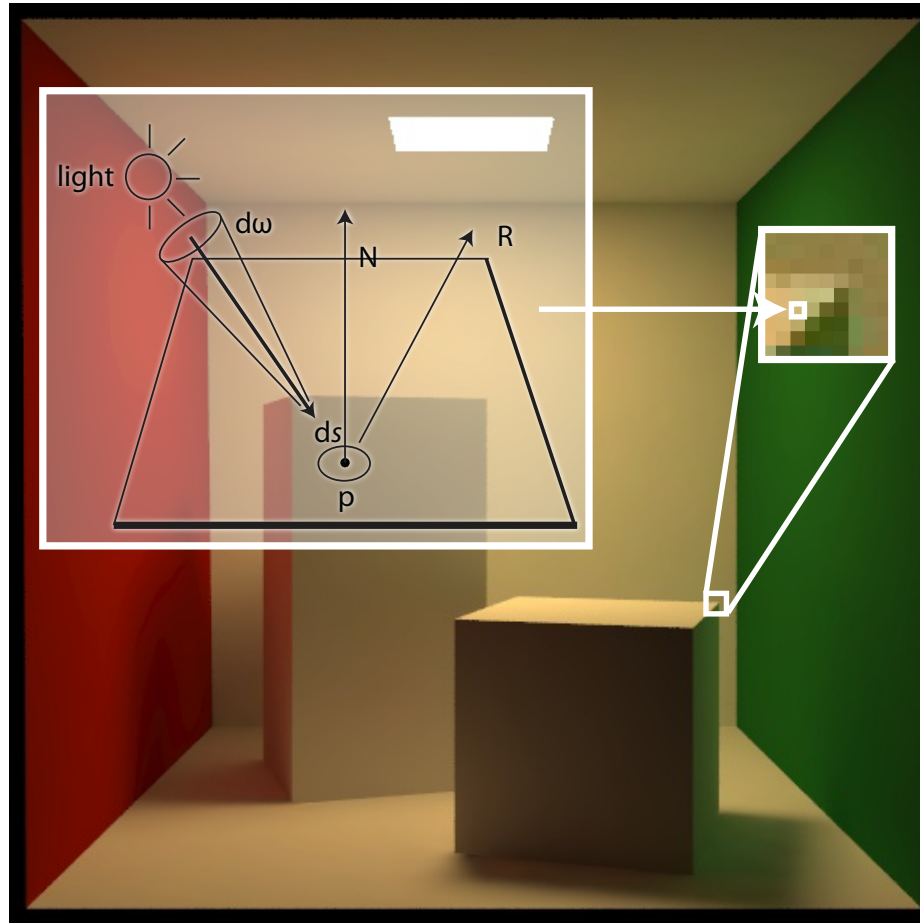


Figure 1.6: Computer rendering is carried out in a depth-first order, with each pixel calculated independently from the object space information of geometry, lighting and material. Image courtesy of the Cornell Program of Computer Graphics (cor 2016).

All shaders answer the question “What is going on at this spot?” The execution model of the shader is that you (the programmer) are only concerned with a single point on the surface and are supplying information about that point. ... The shader starts out with a variety of data about the point being shaded but cannot find out about any other points. (page 171)

For instance, in raster-based rendering pipelines, the vertex shader prepares the geometrical and texture information, and the rasterizer interpolates the output of the vertex shader for each pixel. Then, all information related to one specific pixel, including geometry, lighting and material, will be sent to a fragment shader instance. Henceforth, this shader instance determines the final pixel colour independently of all other instances. For ray-tracing, each ray determines its own colour indepen-

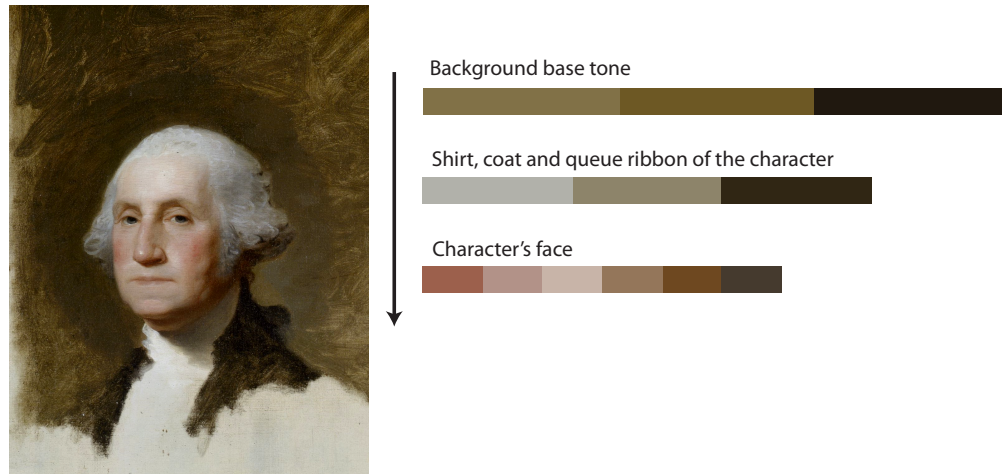


Figure 1.7: Traditional art media operate in a breadth-first order. Left: Gilbert Stuart, *George Washington*, 1796, oil on canvas, National Portrait Gallery, Washington. This unfinished painting reveals the artist’s painting process, in which the overall toner composition is determined before the geometrical structure is defined. In the unfinished parts, large, monochromic blocks are visible. (Barratt and Miles 2014).

dently of every other ray, even with most global illumination methods (Dutre et al. 2006). In multipass rendering, information is propagated between pixels through texture sampling, but each rendering pass still calculates each pixel separately (Saito and Takahashi 1990; Lauritzen 2010). One exception to this claim is the radiosity method (Angel and Shreiner 2011), which solves a finite element problem related to all diffuse fragments in the entire scene. Radiosity operates in the object space and serves to ‘flatten’ the given lighting onto the object’s surfaces without generating a specific image, and thus is not related to our discussion about image space composition. We name this approach of computer rendering as ‘depth-first’.

In contrast, traditional art media operates in a ‘breadth-first’ manner of constructing visual images. For artists who paint with either a paint-brush or a digital tablet, they do not determine the colour of each point on the canvas one after another. Instead, the entire *image* is always immediately visible to the artist at any intermediate stages of working. Because an artist keeps evaluating the whole planar image to determine what to do next, properties related to the visual appearance of the image take priority. For example, to paint a portrait, the tonality of the foreground, background and the character’s face are determined at the beginning with large strokes and even colours. Afterwards, an artist could further work on defining structural details, using much thinner paint strokes and more vibrant colours to paint over the initial colour

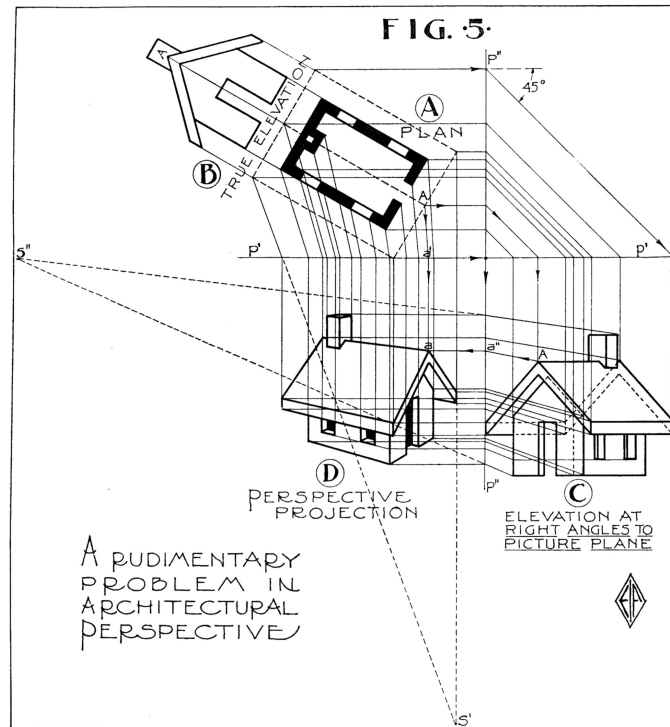
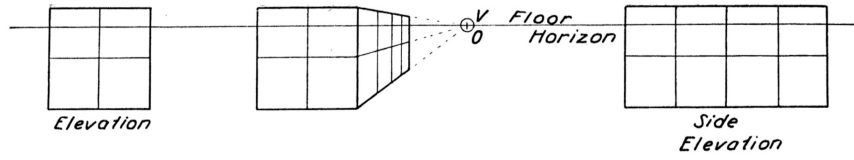


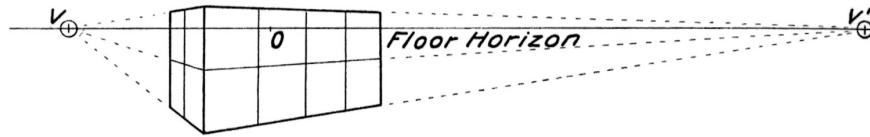
Figure 1.8: Drawing objects in a accurate linear perspective is a complicated task, even when the subject matter is just a toy house (Freese 1930).

blocks. In this approach, the image is the primary subject matter of working and the object space only serves as a reference. A partially completed painting will show the basic colour composition, but leave out much geometric detail in the unfinished parts (Figure 1.7). As a comparison, if a rendering process was interrupted and a partial rendering result is shown, then for raster graphics some objects will be completely missing without affecting their background or their surroundings. And an incomplete ray tracing result will have some pixels undefined while leave others possibly perfectly rendered.

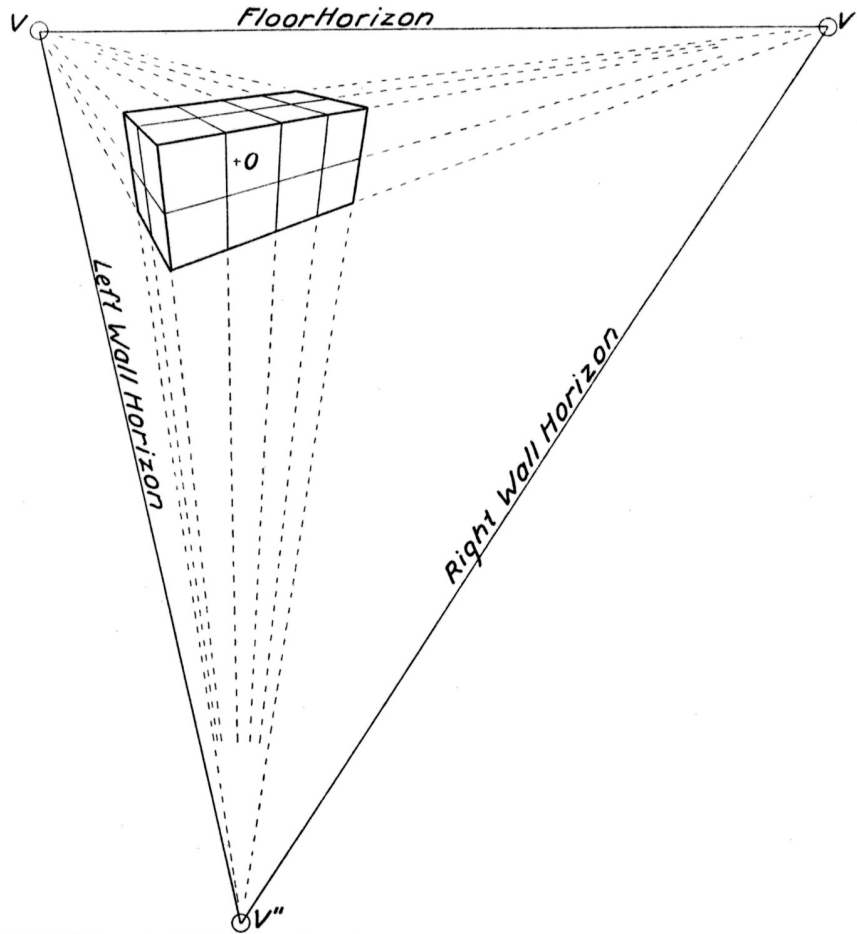
These two distinct approaches lead to different levels of difficulty for accomplishing certain tasks in image construction. For an artist who draws or paints by hand, adjusting the layout of colour and shade is straightforward, but accurately depicting complicated object space structures is difficult. We will examine examples of artistic adjustments in light and shade and relate them to computer rendering in the next chapter. In contrast, object space information are always immediate to the computer rendering programs, while non-local image space visual features are mostly invisible. It is easy to render accurate and consistent geometric structures and light transports,



(a) One point perspective.



(b) Two point perspective.



(c) Three point perspective.

Figure 1.9: Drawing with one, two and three point perspectives (Morehead 1952).



(a) Jan Davidsz de Heem,
Flowers in Glass and Fruits,
Oil on canvas, Gemäldegalerie, Dresden.



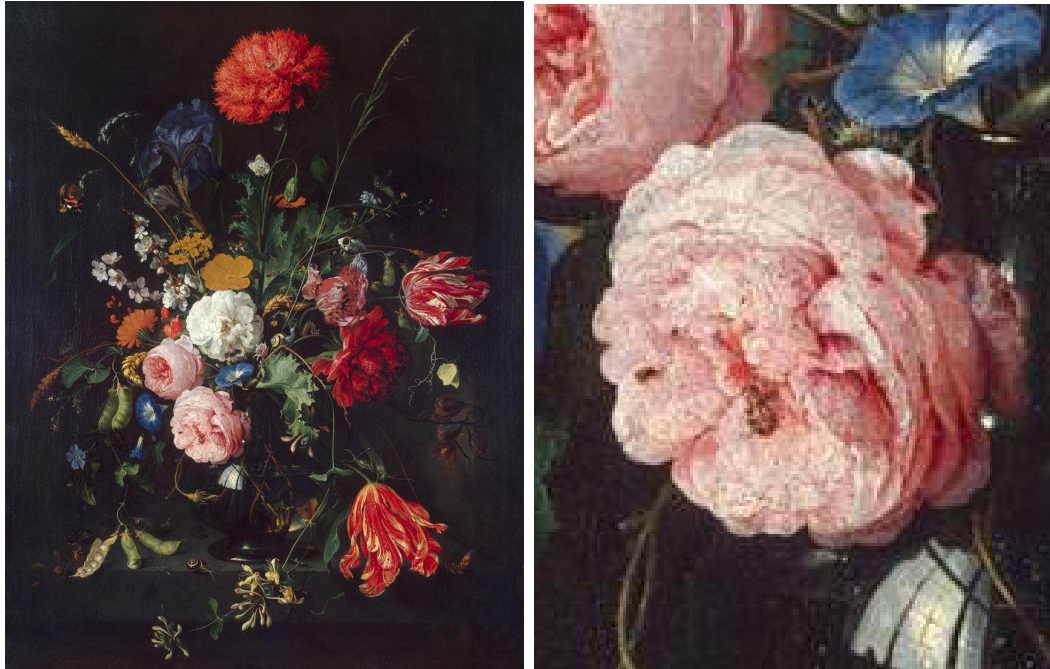
(b) Enlarged part of the rose on the
middle-left of the painting.



(c) Jan Davidsz de Heem,
Vase with Flowers, c. 1670,
Oil on canvas, Mauritshuis, The Hague.



(d) Enlarged part of the rose on the lower-
right of the painting.



(e) Jan Davidsz de Heem,
Vase with Flowers, Oil on canvas,
The Hermitage, St. Petersburg.

(f) Enlarged part of the rose on the lower-
left of the painting.

Figure 1.10: Three flower paintings of Jan Davidsz de Heem. The artist combined flowers sketched from different seasons, and reused motifs such as a left-facing rose with an insect. This particular flower always had a bright lit centre and a dimmed rim.

while it is difficult to adjust the rendering result for a visual property that involves the relationship of colour and shade on the image space.

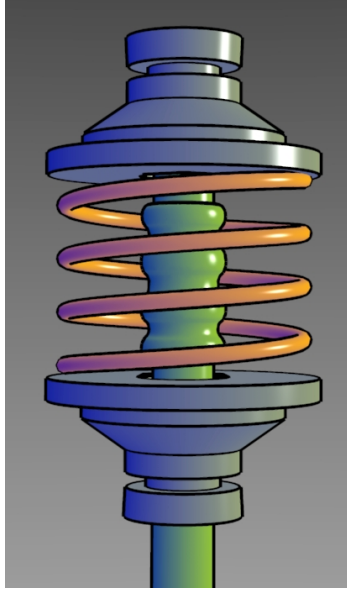
As a notable example, accurate linear perspective can be rendered using merely sixteen numbers in a four-by-four projection matrix (Shirley and Marschner 2009). This problem of rendering linear perspective was first solved by artists during the Renaissance (Pirenne 1970). Their solution, still taught in today's art school, involves drawing a system of complicated plots (Figure 1.8), and the solution works differently for one, two and three points perspectives (Figure 1.9). The considerable effort spent in constructing accurate perspective *on a planar image* demonstrates the inherent difficulty with traditional art media, in which object space information must be conveyed but is not immediately present on the drawing surface. It takes years of training to correctly draw objects in perspective with rich structural details, even if there are real-life models in front of one's eyes. To depict a fictional scene requires

more artistry, and the celebrated painters copy their still-life sketches across paintings instead of inventing new motifs free-handly by imagination (Figure 1.10). Indeed, one important application of computer graphics is computer-aided design (CAD), in which the tedious task of accurately transferring a three-dimensional design to a planar plot is accomplished by a software tool.

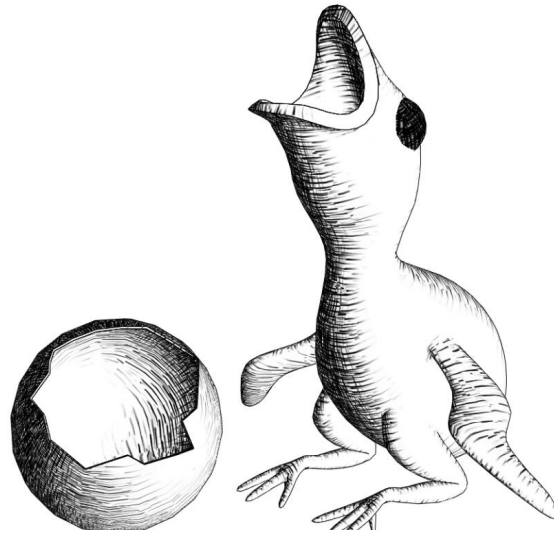
* * *

We state our research goal as enhancing the aesthetic quality of digitally synthesized images by combining the strengths of photographic projection and traditional drawing and painting. We seek to create rendering methods that are both accurate in depicting complex object space information and versatile in image space composition. Here, we shall briefly review the related researches in the computer graphics literature. Image space visual features are easier to understand compared to parameters in the three-dimensional object space, and researchers have designed various rendering methods that modify the object space information according to image space inputs. In order to effectively communicate constraints in the image space, a *painting interface* can be used on top of a numerical interface. With a painting interface, an artist looks at an image, then uses a digital pen to draw the intended modification on top of the image. The paint-strokes can be attached with various semantic meanings, for example to lighten or darken specific parts of the scene. Then, the software tool serves as an interpreter to set the object space parameters that best match the user's intention with some evaluation metrics (Schoeneman et al. 1993; Pellacini et al. 2007; Grimm and Kowalski 2007; Schmid et al. 2011; Pellacini 2010). In our research, we also make use of painting interfaces, and propose novel and intuitive interfaces for painting light and shade.

If a rendering method generates pixel values merely from object space information, then it cannot place any constraint on those pixels' image space expressions. On the contrary, the possibilities of styles and expressions in artworks made by hand are restricted by the medium. An artist drawing with a charcoal pencil cannot produce any colour beyond black and white. If the artist wants to express varying shade across an object's surface, s/he will have to use either an eraser to create grey gradients, or use hatching. With large, dry brushes, a painter will leave visible strokes on the canvas regardless of the subject matter being depicted. Observing this, NPR researchers introduced various *example based* rendering methods. This category of methods take the geometric information from the object space, then combine it with examples from a given database to determine the surface shading. The example database may consist of a collection of hatching textures (Salisbury et al. 1997; Webb et al. 2002; Praun et al. 2001), or a few paint stroke primitives which can be placed onto the objects' surfaces (Kaplan et al. 2000; Lu et al. 2013). Also, research of stylized shading has



(a) The blue-orange shading scheme for technical illustrations.



(b) A result from the real time hatching rendering method.

Figure 1.11: Non-photorealistic methods that construct surface shading from pre-defined rules or example databases. In Figure 1.11a, the shading method always adds orange tint to the bright parts on the object’s surface, and adds blue to the shadows (Gooch et al. 1998). Figure 1.11b shows the interactive hatching rendering method, in which surface brightness is expressed by pre-given hatching textures with different line density (Praun et al. 2001).

demonstrated using a pre-given colour palette for surface shading (Gooch et al. 1998; Barla et al. 2006). In our research, we take this idea further and create rendering methods whose outputs are restricted and predictable, but do not include any specific gestures from a certain traditional art medium.

As discussed previously, three-dimensional rendering algorithms synthesize pixel values individually without interfering with each other, and generally do not evaluate the resultant image for non-local visual features. On the other hand, image space re-lighting or re-colouring methods that aim at image space non-local visual properties usually involve optimization or solve large linear systems, which may be computationally expensive (Fattal et al. 2007; Mertens et al. 2009; Gooch et al. 2005; Bhat et al. 2010). Image space pattern fitting algorithms operate in a similar manner, but in this context, the smallest visual primitive will be an elemental shape or a texture patch, instead of pixels (Song et al. 2008; Hurtut et al. 2009). Recently, researchers

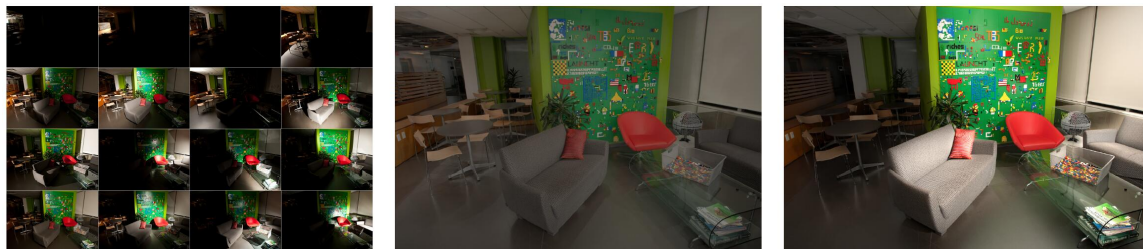


Figure 1.12: User-assisted lighting composing for photographs. The program uses optimization to turn a large stack of photos with different lighting (left) into several reference images for lighting composing (centre, the ambient lighting image). An user then compose the final image by layering the reference images together with a slider-bar interface (right) (Boyadzhiev et al. 2013).

have demonstrated how to combine a large stack of photographs and automatically generate images with ‘good lighting’ as sources for lighting composition (Boyadzhiev et al. 2013) (Figure 1.12). In our research, optimization is also an important technique for generating non-physical lighting according to user inputs.

To approach our research goal of facilitating image composition in computer graphics, we built two rendering projects. Each of these projects concentrates on a specific facet of the general research goal, and develops our initial idea of a painting-based, composition-motivated rendering approach into algorithmic implementations with increasing proficiency. We also introduce our art projects developed with the same methodology of digital image composition. A major part of our research has been published in two peer-reviewed papers (Ji et al. 2016; Ji et al. 2015). Here, the presentation of these projects is organized in the following order:

1. In chapter 2, we analyse image composition in the context of computer graphics rendering, visual perception and visual arts. We then construct a novel interpolative material model with a painting interface. This material model demonstrates a possible approach of creating predictable surface light and shade; and the painting interface allows direct composing of the brightness on objects’ surfaces.
2. In chapter 3, we continue our discussion of image composition, and show a method that can automatically create customized lighting from an input sketch. In this project, we demonstrate an approach in which an artist only needs to approximately paint the overall lighting design. Details in light and shade are automatically added and animated by our program. This research is staged in a context of rendering animated foliage shadows.



(a) The salience-preserving colour removal algorithm generates grey scale images in which the major visual features of a given image are kept.



(b) The ‘Arty Shape’ algorithm automatically fit geometric shapes guided by a given image.

Figure 1.13: Algorithms aimed at non-local image space visual features. If we use a conventional filter to transfer Monet’s sunrise painting (Figure 1.13a, left) into grey scale image, the sun and its reflection is lost (Figure 1.13a, middle), because this painting features the equiluminant composition technique. To preserve these equiluminant visual features, the ‘color2grey’ method uses optimization to adjust values of the result grey level image (Gooch et al. 2005). Optimization methods can also be used to adjust shapes larger than one pixel to express given visual features in an input image (Figure 1.13b) (Song et al. 2008).

3. In chapter 4, we introduce our art projects and our user study for evaluating the rendering methods. These projects validate our algorithmic design and illustrate future directions of research. We then summarize our research in the last chapter.

Through the presentation of these research projects, we explore the problem of image composition in computer rendering from its aesthetic and technical dimensions. The contribution of our research consists of two major aspects:

1. We examine image composition in the context of computer graphics, and show the necessity of incorporating knowledge of composition in rendering. Composition is a central topic in visual arts, but it is rarely discussed in computer rendering

research, even though both disciplines seek to create pictorial depictions of imaginary scenes. We show that the fundamental aspects of composition can be understood in the term of visual perception and computer rendering. This understanding has three facets:

- (a) First, we explain why the composition of celebrated artworks can elicit aesthetic responses from their viewers. We interpret art theories based on visual perception, and show how an artist may regulate surface light and shade beyond a photographic depiction to direct the viewers' eyes.
- (b) Secondly, we compare the methodology of traditional art media with computer rendering, and relate the compositional techniques from visual arts to digital image synthesis and processing. We show that with computer rendering we construct an image in a fundamentally different order from traditional art media, and treat image space visual features as causal consequences of object space information. On the other hand, with traditional art media we consider image space visual features as the primary subject matter; while the object space information is only considered as a reference.
- (c) Lastly, we discuss why conventional three-dimensional rendering is insufficient for the purpose of planar image composition. With computer rendering, an artist must go back to the modelling interface to change parameters in the object space, in order to indirectly adjust visual features in the image space. This twisted work-flow leads to difficulties in achieving compositional effects with computer rendering.

This original analysis constitutes the introduction chapter and the beginning part of chapter 2 and 3. To our knowledge, this is the first in-depth theoretical analysis on the topic of image composition in the context of digital image synthesis.

2. Based on this analysis, we create novel, composition based rendering programs, and show their effectiveness through the rendering results and user evaluation. We also support our rendering methodology and evaluation with a few art projects, through which we validate our approach in art making. These projects are presented in the previously listed order, and the contribution of them can be summarized in the following three aspects:

- (a) First, we implement the idea of regulating surface brightness as rendering methods. Unlike existing rendering methods that simulate light transport and record lighting intensity, our methods render surface brightness with example based approaches, and let artists design the light and shade with painting interfaces. Our methods are not strictly physically based, and they do not simulate specific artistic styles or gestures like conventional non-photorealistic rendering. We show that our methods are capable of creating rendering results that are believably realistic and visually attractive.
- (b) Secondly, our rendering methods show possibilities of directly adjusting rendering results using image space interfaces for three-dimensional computer rendering, without going back to the modelling interface. With our data structures, such as the interpolative material model or the projective light mask, the rendering work flow is intuitive and straightforward. The adjustments of light and shade placed by an artist using the painting interfaces are guaranteed to have predicable, localized effects on the rendering results.
- (c) Lastly, we report our experience of making artworks with the proposed methodology. We validate the proposed methodology by using it in a visual arts context and discussing the resultant pieces with peer artists. We also conduct user studies with artists from a visual effects background. The opportunities and challenges we discovered thorough our art practice cannot be easily acquired by conventional research within a computer lab.

The presentation of these projects constitutes the later parts of chapter 2 and 3, and chapter 4. Our novel programming and art projects validate our theoretical analysis of composition, and demonstrate the feasibility of incorporating knowledge and compositional techniques in computer rendering. These projects also illustrate important future research directions for enhancing the visual quality of digitally synthesized images.

We seek to use computer graphics as an art medium, and to direct the rendering process with information from both the image plane and the object space. The theme of our research is to treat the planar image as the primary subject matter of rendering, instead of a consequence of a projection from the object space. We also aim at creating better interfaces between artists and rendering programs. The long term goal of our

research is to achieve a balance between artistic expression and realistic believability. This goal may be termed as to achieve a *staged realism*, to synthesize images that are recognizable as depictions of realistic scenes, while at the same time enabling the freedom of composing the resultant image in an artistic manner.

CHAPTER 2

VARIATIONS IN SHADING

The first programs that synthesize images with three dimensional impressions simply draw pixels following the principle of linear perspective (Figure 2.1). The beginning of computer graphics in its contemporary sense is marked by the separation of the *object space*, which contains the virtual world, and the *image space*, onto which the virtual world is projected. Henceforth, mainstream research of computer rendering takes the form of physically based photorealistic rendering; while research on non-photorealistic rendering has also successfully simulated many traditional art media on the computer screen. Both research approaches follow the photographic metaphor, which implies that the image space visual features are causal consequences of projections of object space information, as discussed in the previous chapter. In addition, the photographic metaphor assumes a synthesized image is a record of the appearance of the object space at *an instant*, and from our point of view this assumed instant capturing process is also related to the difficulty of image composition with rendering. This chapter begins with an analysis of this assumption.



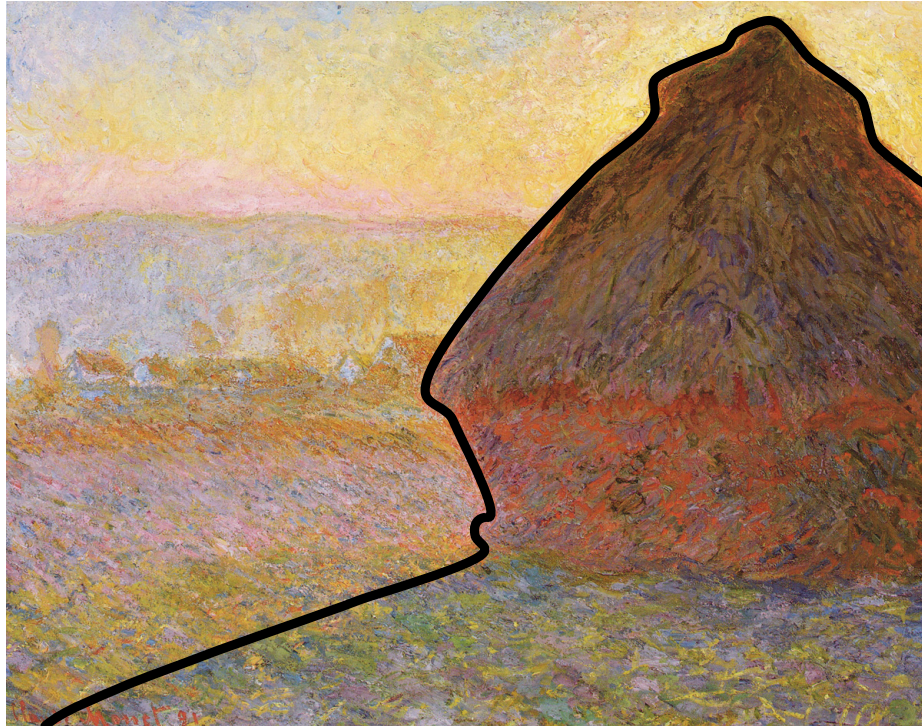
Figure 2.1: Tennis, published by Activision in 1981, is a video game on the Atari 2600. With very limited hardware capability, this video game draws pixels with linear perspective, and uses two black pixels to represent the shadow of the tennis ball, giving its player a sense of the height of the flying ball.

Interactive computer graphics applications typically render at thirty or more frames per second. Computer rendered feature animations are shown with a similar frame rate, although their high-quality, off-line rendering process can take a lot of time per frame. It seems appropriate for a rendering program to exclude time parameters, and to leave the time related calculations to the animation methods. On the other hand, image composition is a way of eliciting visual response from its viewers. To make any response at all, a viewer needs at least some time to look at an image, and the composition of the image unfolds itself during that time. Before investigating this in detail, we shall briefly review the way a camera works, and the role of the instant exposure in a photographic metaphor.

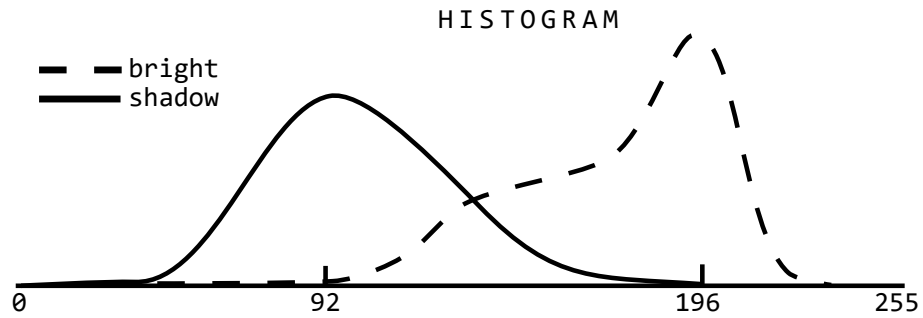
A camera records a slice of light transport on its negative during a short, continuous period of time. In physically based photorealistic rendering, the light transport is simplified to be of infinite speed and the conceptual negative to be of infinite sensitivity. The synthesized image is therefore a representation of a mathematically zero thickness cross-section in the light transport volume of the virtual scene. This simplification holds well for most bright scenes, except for some low-light cases with fast movements in which the motion blur due to a long exposure needs to be considered (Haeberli and Akeley 1990; Hou et al. 2010). NPR methods typically do not simulate detailed light transport, but the rendering results also resemble the given scene in an isolated instance, with artistic gestures such as paint strokes or line drawings. For either photorealistic rendering and NPR, the important implication here is that *the rendering result is a depiction of the whole appearance of the entire scene in one single moment*. In this chapter, we shall see that a human observer cannot do this with eyes; and the fact that we must take time to perceive a large scene is closely related to image composition. We begin this analysis with an example of Monet’s haystack painting.

2.1 REALISTIC PERCEPTION AND MONET’S HAYSTACKS

Around 1890, artist Claude Monet started a series of paintings depicting haystacks under different lighting conditions, at various times of a day and at various seasons across years. This series became a systemic study of the colour and shade in outdoor scenes revealed by transient natural light (C. Seitz 1960; Spate 1992). During this project, the painter took many canvases with him to the field, working on each version only when a particular lighting effect appeared. One painting from this series, *Haystack*



(a) Claude Monet, *Haystack at Sunset near Giverny*, 1891, Oil on Canvas, Museum of Fine Arts, Boston. We added a black curve to separate the painting into two parts. The relatively brighter part of the scene is on top of the curve, and the shadows of the haystack is on the lower side.



(b) Histogram analysis of Figure 2.2a. The bright part on top of the curve is represented by the dashed plot, which peaks at 196 on a 0-255 pixel value range. The solid plot shows the shadowed part at pixel value 92.

Figure 2.2: Histogram analysis of Monet’s Haystack Painting.

at Sunset near Giverny, is shown in Figure 2.2a.

From a photometric point of view, a clear daytime sky exhibits a luminance around the magnitude of 10^6cd/m^2 ; and the shadows behind an outdoor object are several hundred times darker. In comparison, paintings are usually shown with indoor

gallery lighting with a luminance around 10^3cd/m^2 , three magnitudes lower than the daytime sky. The darkest oil pigment reflects about $1/20$ light compared to the whitest pigment. To depict a bright landscape on the canvas, an artist must contend with these huge differences in both the absolute luminance levels and the relative contrast ranges. Considering the Weber-Fechner law, which states that human perception is proportional to the contrast of a stimuli (Wandell 1995), we may assume that the absolute luminance level does not matter much, as long as the contrast ratio between visual elements are maintained. Still, the contrast alone provided in the oil painting pigments seems to be quite small compared to the depicted scene. We may therefore expect a painter to perform an artistic tone mapping, reducing the vast contrast range to his available palette, and using every possible colour, from the whitest to the blackest, in order to minimize the loss of information. Under this circumstance, restricting one's palette to a few middle-luminance colours and abandoning most brighter or darker tones seem to be an unreasonable choice; yet this is what we have observed in Monet's haystack paintings.

We divide the haystack image into two parts and perform histogram analysis on them. In Figure 2.2a, the part of the scene under bright lighting is on top of the black curve: the sky and the far-away fields bathed in sunshine. This part is represented by the dashed plot in the histogram. The part below the curve is under shadow, which contains the back-lit haystack and nearby grassy field, and is denoted by the solid plot. On the histogram, we can see that most colours used by the painter are fairly close to the centre, with two peaks sitting at about one-third and two-thirds of the total possible contrast range. Both ends of the histogram contain very little colour. Especially on the dark end, there are scarcely any shades darker than one fifth of the maximum luminance. The histogram indicates that the painter used little of his brightest or darkest pigments. Nevertheless, when we look at the painting, the golden sunset appears stunningly bright behind the heap of hay, almost forcing us to move our gaze away from the contour of the haystack. (The black curve we added in Figure 2.2a undermines Monet's lighting effect considerably. In the original painting the sunshine appears much brighter.) The deep blue and red shades on the haystack unmistakably depict shadow, showing the vivid details of the individual stalks of hay and the way they were piled together. With colours in a small luminance range, the painter constructed an image that depicts a scene with a vast range of contrast. One may spend time contemplating such a painting and discover the sublime in a landscape with humble haystacks.

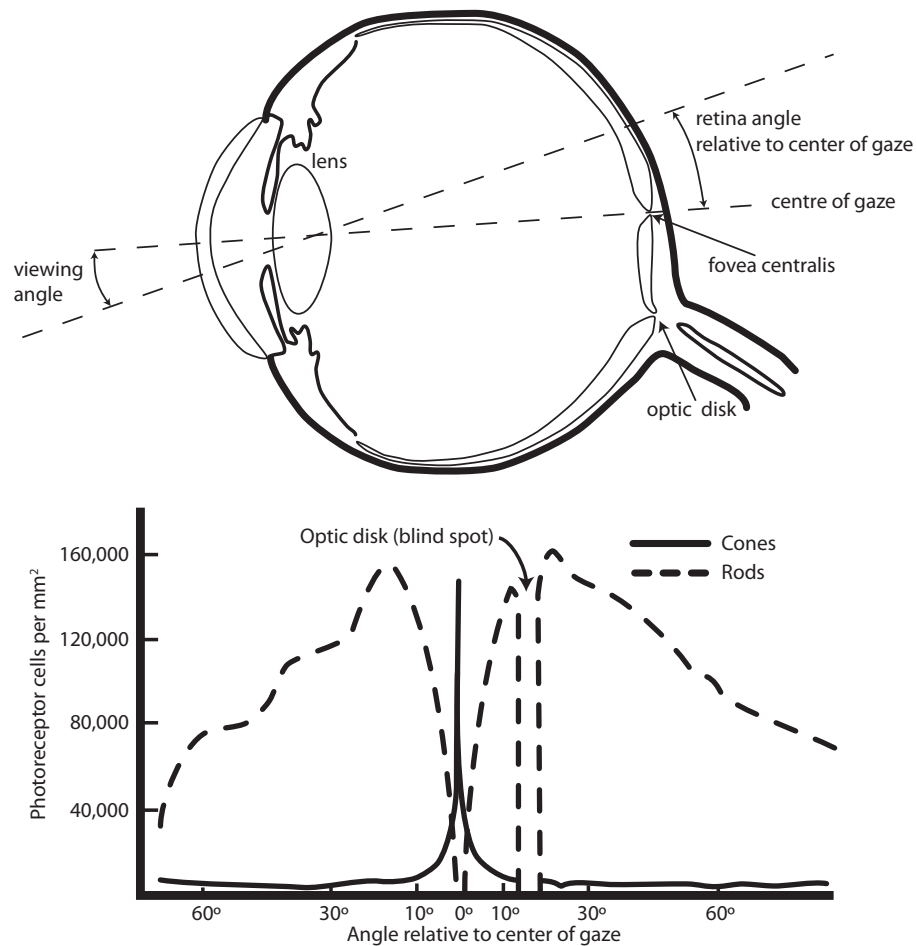


Figure 2.3: The fovea centralis is a small dent on the retina, where the colour photoreceptor cells (cones) are most densely packed. It corresponds to a small viewing angle of approximately 6 degrees, in which we can see the highest spatial and colour resolution. Moving away from the fovea centralis, the density of cones drops rapidly, and the low light photoreceptor (rods) starts to appear (Boynton 1979; Cornsweet 1970).

Perception theories about human visual systems (HVS) may account for part of the painting's visual effects. In the rest of this section, we shall relate our discussion on composition to the luminance adaptation of the HVS. Assuming we stand in front of a landscape with bright sunshine like the haystack scene, we will be able to see details all around us, including clouds in the bright sky and the shadowed wall of a house. We achieve this by staring at each part of the scene for a short time before shifting our lines of sight onto the next region, and after a while we have seen the entire landscape. Our sights need to be constantly rotated, because the viewing angle within which our eyes can see the maximum spatial and colour resolution is quite small. This viewing angle is approximately 6 degrees, which is about the same size as the thumbnail as it is seen when the arm is fully stretched to the front. The narrow viewing angle corresponds to the small fovea centralis on the retina, where the colour-sensitive photoreceptor cells (cones) are packed with the highest possible density (Figure 2.3). Moving away from the fovea centralis, the density of the colour-sensitive photoreceptor decreases rapidly; and the colour-insensitive photoreceptor used for night vision (rods) begins to appear (Wandell 1995). To see a large scene, we look at one part of the scene at a time, and construct the appearance of the entire scene afterwards, by stitching a sequence of visual memories together in our minds.

During each short gaze, the HVS adapts itself to the image projected on the fovea centralis to best distinguish details. The adaptation behaviours can be categorized into two kinds: those we can feel and those we cannot. For example, we clearly feel it if we change the focal point of our eyes to look at objects with a different distance to us. This action involves muscle movement to morph the shape of the lens, and it rapidly changes the retina image. The contraction and dilation of the pupil and strong bleaching after-images can also be clearly felt when the surrounding lighting changes abruptly. Since these adaptation behaviours generate a direct sensation, we can clearly tell their absence when we look at planar, low contrast visual media, such as photographs, paintings or computer screens, even if a similar visual effect is being simulated. On the other hand, more subtle visual adaptations that generate weak or no direct sensations have granted artists opportunities for adding less noticeable touches to their work. Since a viewer cannot reliably tell if an adaptation behaviour is present or absent, artists can simulate an adaptation effect to suggest a specific viewing context. Research of perception based tone mapping has explored various methods for simulating visual adaptation in compressing HDR images (Krawczyk et al. 2005; Pattanaik et al. 2000; Ashikhmin and Goyal 2006), but they generally do not

distinguish between those adaptation behaviours that can be sensed and those cannot, nor do they relate these behaviours to image composition. As an example of visual adaptation, we shall sketch the bleaching-regeneration process here to prepare our discussion of painting composition. Interested readers are referred to the textbooks and research on vision, perception and tone mapping for the details (Boynton 1979; Cornsweet 1970; E. Jacobs et al. 2015; Ritschel and Eisemann 2012).

In our photoreceptor cells, photosensitive pigment molecules change their chemical configurations when they absorb photons within their sensitive frequency range. This change results in a series of biochemical reactions, and in the end triggers neural signals on the subsequent visual pathway. A pigment molecule is said to be *bleached* after it absorbs a photon, and will be no longer capable of absorbing photons. This terminology came from the fact that the photosensitive pigment extracted from animal eyes has a colour. When a solution of the pigment is placed in a test tube and exposed to a strong light for a while, the solution becomes transparent, indicating that all pigment molecules have changed their configuration, and no more photons will be absorbed. Obviously, our photoreceptor cells have a way of constantly replenishing the photosensitive pigment molecules after they are bleached, otherwise we would not be able to see anything after an initial exposure to light.

The mostly accepted theory about how the photoreceptor cells replace the bleached pigment molecules with photosensitive ones is the *regeneration* theory (Boynton 1979). It proposes that the photoreceptor cells rarely make new pigment molecules; instead, they keep reverting the bleached molecules back to their photosensitive status, such that there is always a constant number of total pigment molecules in a photoreceptor cell. In a healthy eye, photoreceptor cells maintain stable biochemical environments for pigment regeneration; and the regeneration speed is proportional to the concentration of bleached pigment molecules in a particular cell. Assuming the concentration (proportion) of photosensitive molecules is p , $p \in [0, 1]$, the bleaching-regeneration kinetics can be modelled as:

$$\frac{dp}{dt} = \frac{1-p}{T_0} - \frac{Ip}{T_0 I_0}.$$

In this equation, T_0 and P_0 are constants for each kind of photoreceptor cells, and are different for rods and various kinds of cones. The term t stands for time. The first term on the right hand side represents the regeneration process, and the second term represents the bleaching process with a given light intensity I . The strength of the

neural signal emitted from a photoreceptor cell depends on how many photosensitive pigment molecules are being bleached, which is approximately proportional to the second term. When luminance I increases from a steady state, this term increases proportionally, meaning we see a brighter light. The increase in the second term will give dp/dt a negative value, which immediately causes p to decrease. In this way, the strength of the emitted neural signal is decreased, meaning now we have a diminished sensation of the incoming light; and the bleaching-regeneration process moves to a new steady state with a smaller p . More importantly, this process does not happen in an instant, and is not carried out at a constant speed. At the moment of a change in light intensity I , dp/dt reaches the largest absolute value and the luminance adaptation is the fastest. After a while, when the bleach-regeneration process is close to its next steady state, dp/dt becomes smaller and it takes longer to fully adapt to the new luminance condition. For the red and green cones used for daylight vision, it only takes a few seconds to perform the major amount of the bleaching-regeneration adaptation in common situations. In contrast, it may take up to half an hour to completely stabilize p with a given luminance in a vision lab. When luminance environments gently change, the bleaching-regeneration kinetics moves between steady states through a mild biochemical process, and is mostly unnoticeable.

The bleaching-regeneration process, combined with other adaptation mechanisms in the photoreceptor and the subsequent visual pathway, creates a constantly shifting visual impression of the physical retina image. Every time we move our gaze, our eyes always attempt to adapt to the luminance condition in a local region to best distinguish details. How well our vision can adapt depends on how long we look at that region and what has been viewed just before. For artists who need to carefully observe a scene to depict it, they usually stare at each part of the scene for a long time and have their eyes accurately adapted during each gaze. In the example of Monet's Haystack painting (Figure 2.2a), the painter needed to attentively observe the shade on the shadowed side of the haystack before he could paint it. This observation gave his eyes ample time for luminance adaptation, which in turn placed the visual neural signal around a moderate strength, and generated a visual impression of a fairly lit side of the haystack. Similarly, a prolonged gaze adapted the painter's eyes to the high light intensity from the sky, and the dazzling vision of the sunset receded to mildly bright. Although the painting appears rather non-photorealistic, the painter faithfully reproduced his visual impression of every part of the scene. The important point to reiterate here, as Monet complained in his letter, is that such a visual impression can

be only formed *over time*:

...for in October (1890) he (Monet) wrote to Geffroy:

“I’m working terribly hard, I’m struggling stubbornly with a series of different effects (stacks), but at this time of year the sun sinks so fast that *I can’t keep up with it. I’m beginning to work so slowly that I despair*, but the longer I go on, the more I see that it is necessary to work a great deal in order to succeed in rendering what I seek - *‘instantaneity’*, above all the envelope, the same light spreading everywhere - and *more than ever I’m disgusted with things that come easily in one go*. I am more and more obsessed by the need to render what I experience...” (Spate 1992). (italics marked by the author)

As a result of such careful observations, the painting not only demonstrates an effective perception-based tone mapping method, but also suggests a specific viewing process that achieves such a perception. The painting does not conform to a visual perception of a quick glance into a sunset landscape. Rather, a conventional photograph with over exposed sky and dark shadows better conforms to a hasty glance; since the output of our photoreceptor can easily be saturated by the bright sunset, given insufficient time for adaptation. By confining his palette to the medium luminance range, the painter instructs his viewers that *in order to see the scene like this*, one must have taken time and carefully observed it in a meditating manner. This calm, motionless viewing experience is further strengthened by the blurry depiction of the horse-drawn carts at a distance, which hints a motion-blur effect from a static, long exposure camera.

The invention of photography has greatly changed the way images are produced. It obviated the need for representational painters, while pushed artists like Monet to produce images clearly dissimilar from photographs. This led to the impressionism movement in the 19th century (Spate 1992; Fried 2008). To differentiate themselves from what is supposed to be done by the camera machine, the impressionists use large paint-strokes and vibrant colours, and intentionally leave out small details. One distinct feature of the impressionism genre from previous representational paintings is that the impressionists use continuous shade variation to define spatial geometry, instead of discrete gestures such as line drawings or contours. Ironically, this is also the essential property of photographs. Before the invention and popularization of cameras and camera-like devices such as camera-obscuras, this visual effect cannot be observed from previously existing artworks. We could say that the technology of photography

both undermined the established practice of representational art, and at the same time facilitated the creation of new forms of visual images. Likewise, computer rendering is approaching its consummate ability to synthesize photographic images. Pursuing better aesthetic quality and deviating from an exact photographic depiction will be a natural development, a next step similar to the one that the impressionist artists took after photographs dominated the market of representational images. At this turning moment, we believe an in-depth study of image composition is instructive and helpful. We shall further explore the topic of composition and perception with examples of a few more paintings and photographs in the following section.

2.2 THE ARTIST'S SHADE

Among the many forms of traditional visual arts, painting and photography are two forms that receive a lot of attention from computer graphics and image processing communities. NPR researchers often refer to painting and drawing for inspirations (Gooch and Gooch 2001; Strothotte and Stefan 2002; Klein et al. 2000); and the tone mapping research takes examples from photography in addition to knowledge of perception and signal processing (Reinhard et al. 2002; Yuan and Sun 2012). These rendering methods seek to best communicate the information contained in the input three-dimensional scenes or high dynamic range imagery (Durand and Dorsey 2002; Mantiuk et al. 2004; Shan et al. 2012). A few tone mapping methods also include interactive interfaces to let their users annotate the image and help in the process (Lischinski et al. 2006; Kang et al. 2010). Research on these topics, in general, does not seek to add information on top of the input model or image. In contrast, the discipline of creative art despises the idea of straightforwardly mimicking the subject of depiction. From one of the earliest art theories of painting (van Hoogstraten 1678) to a recent summary on photography (Edwards 2006), it has been insisted that the artistry in creating images lays not in how accurately the subject matter appears, but in how to construct and deliver a novel idea through the depiction. In other words, the aesthetic value of visual imagery is correlated to inserting and communicating additional information beyond objective depiction. Particularly for representational art, an important function of image composition is to facilitate to this task of conveying aesthetic effects without breaking the realistic visual appearance of the image (Goldstein 1989).



Figure 2.4: Johannes Vermeer, *Woman in Blue Reading a Letter*, c. 1662-64, oil on canvas, Rijksmuseum, Amsterdam. 46.6x39.1 cm. The background wall is painted as the three blocks marked as (a), (b) and (c) (Wheelock 1995).

In this section, we examine two of Vermeer’s paintings, “Woman in Blue Reading a Letter” and “The Music Lesson” to further explore the relationship between visual perception and composition. We choose Vermeer’s work as examples because his paintings of domestic scenes are praised as accurately photorealistic yet visually attractive. Unlike many other painters, such as the impressionists, who intentionally differentiate their paintings from photographs, Vermeer only deviates his representational paintings from an exact photographic depiction in a subtle but powerful manner. A discussion of his work is closely relevant to three-dimensional computer rendering, whose major function is to synthesize representational images depicting a virtual scene.

In “Woman in Blue Reading a Letter” (Figure 2.4), part (b) of the background wall is tinted towards blue. This blue shade reminds us of the global illumination effects (Shirley 2003), as the blue coat on the foreground figure will reflect blue light onto the wall. In the painting, Vermeer’s execution of this effect is different from an exact physically based solution. The blue light fills up part (b) in a uniform and exaggerated way, and cuts off sharply on the boundary between part (a) and part (b) of the background wall. In contrast, in part (c) to the right of the figure, the wall is tinted to brown-yellow, instead of blue. From a physically based point of view, one would expect the blue shade on the wall to fall off gradually around the figure in both directions. In the painting, however, the three blocks of the background wall are constructed as three monochromatic blocks with little colour gradients (Wheelock 1995; Liedtke 2012).

The boundaries of the blue shade in part (b) are delineated by their visual contexts in the image space, such as the chair, the map and the main figure. Since blue is an ostensible colour in oil painting, the artist suppressed its expression on the background wall to avoid distracting the viewers from the main character in the front. In contrast, the blue dress on the character is painted with bright and intense (high saturation) blue colours. The silky reflections on the dress, rendered with sophisticated colour variations and indiscernibly fine paint strokes, show the artist’s exceptional ability to construct accurate and realistic surface shading. Similarly, the yellow-brown shade on part (c) of the wall can be related to its lower right position in the painting. This earthy tint serves to stabilize the visual structure of the painting and echoes the brown map on the top. The manipulations of detail and colour do not break the impression of a realistic scene, but instead draw our sight unmistakably onto the foreground character. Equipped with eye-tracking devices, contemporary perception researchers have shown that variations in colour hue and value do attract a viewer’s



Figure 2.5: Jeff Wall, *A View from an Apartment*, 2004-05, transparency on light-box, 1670 x 2440 mm

gaze; while monochromic shade blocks do not catch as much attention (Livingstone 2015). Modern X-Ray examination of this painting has also revealed that Vermeer initially painted the map on the background wall at a higher position. Later, the artist over-painted the map to a lower position to align a dark lake with the bright forehead of the character to create a stronger contrast for the character's face (Wheelock 1995).

An artist can also encourage a viewer to carefully examine every part of an image even if there appear to be some central characters, by intentionally enhancing details in the seemingly peripheral parts. A well-known example in contemporary photography is Jeff Wall's "A View from an Apartment" (Figure 2.5). The artist, who has a background of traditional painting, used computer to edit and combine many photographs together and created a hyper-realistic image that has sharp details and intense colours everywhere. The reflections on the foreground TV, the two characters in the middle ground and the port outside the window are all in perfect focus; and every object in the scene seems to be equally bright. The artist presented this image on a transparent layer on top of a large (2.4 by 1.6 meters) light box. The sheer size of the work prevents a viewer from looking at the whole image in one gaze (Fried 2008). The sharp, equally bright details encourage a viewer to spend time and closely examine the photograph part by part. In comparison, we remember that the size of the "Women in Blue Reading a Letter" painting is only 50 by 40 centimetres.

"The Music Lesson" (sometimes titled "*A Woman at a Virginal with a Gentleman*")



Figure 2.6: (a). Johannes Vermeer, *The Music Lesson*, 1662-65, oil on canvas. 74 x 64.5 cm. The Royal Collection. (b) A photographic reconstruction of the painting. (c) A computer graphics reconstruction of the painting. The shadowed Turkish carpet is marked by the white square. In the original painting it exhibits vivid blue fabrics. In both the photography and rendering reconstruction, where the main characters are properly captured, the carpet in shadow is completely black.

is another important work in Vermeer's oeuvre. Because of its apparent photorealistic quality, researchers have experimented with reconstructing the painting using both film photography and computer graphics rendering (Steadman 2001; Hilliard 2002). When we place the original painting side by side with its photographic and rendering reconstructions, the differences between them are obvious (Figure 2.6). Apart from the different overall tonality and perspective, the difference on the shadowed part of the Turkish carpet (white square in Figure 2.6. a) is evidently clear. In the original painting, this part of the carpet contains vibrant details of the fabric in a shimmering blue colour (Figure 2.7), while in both the photographic and the computer graphics reconstructions this part is completely black. This blacked-out region is not due to technical mistakes of the reconstruction experiments. In Vermeer's studio setting, bright sunshine penetrates the studio window and is reflected directly towards the viewpoint by the white wall and the brilliant, silky clothes on the characters. Without the latest HDR photography technologies, one has to stop down (shrink the aperture) or to shorten the exposure to properly portray the main characters at the end of the room, to an extent that details on the shadowed carpet become invisible. The computer graphics reconstruction with physically based rendering resulted in a similar image.



Figure 2.7: Details of the foreground Turkish carpet in Vermeer's *The Music Lesson*.

If we were to look at Vermeer's studio in reality and to fix our gaze on the shadowed carpet, we would clearly see the exotic fabrics on it. Then, we may lift our eyesight to look at the warm sunshine on the characters farther into the studio. After a short moment, our eyes adapt to the bright light, and the characters reveal themselves to us with full detail. The painter's depiction mimics this experience, and therefore suggests this process to its viewers. On a canvas approximately twice as big as "Woman in Blue Reading a Letter", we cannot see the entire "Music Lesson" painting in one gaze but must shift our sight. The luxurious Turkish carpet, which was a symbol of wealth during Vermeer's time, is the nearest, largest object directly facing the viewer. The bright blue shade on the shadowed carpet trap a viewer's gaze on this foreground object, before the viewer moves his/her attention to the smaller figures farther into the room. In this way, the composition creates an articulated distance between the viewer and the characters in the painting, and establishes an intimate atmosphere around the characters that is undisturbed by the crowd of viewers in front of the frame (Wheelock 1995; Steadman 2001). In contrast, the blacked-out carpets in the



Figure 2.8: Harmen Steenwijck, *An Allegory of the Vanities of Human Life*, ca. 1640, oil on canvas, 39.2 x 50.7 cm, National Gallery, London.

photographic and computer graphics reconstructions do not contain any interesting visual details. With these images, a viewer will immediately focus on the characters in the centre, and the sense of a distance between the depicted characters and the viewer is lost.

If an artist does not want viewers to spend too much time on a piece of front-facing fabric in the foreground, it may indeed be painted black, as the example in Figure 2.8. Here, the fabric dripping in the foreground is painted only with a hint of dark blue, and the major part of it is lost in the deep black. In this way, the painter directs our attention onto more important objects on top of the table, such as the skull, the book and the flute.

The compositional techniques we examined above include suppressing and articulating details on different parts of the image, and stitching together visual impressions from different adaptation contexts. These techniques can be considered as a reverse-engineering of the visual adaptation process, such that the part that an artist wants to attract attention is rendered with more details, and vice versa. Composition is an additional layer of information constructed by the artist on top of the depicted objects in the image. In addition to what is in the image, the artist instructs a viewer

how the image should be seen. It is important to note here that these adjustments cannot be deduced merely from the object space. For example, if we move the viewing point and angle of “The Music Lesson” slightly to the left, the area occupied by the Turkish carpet on the image will be reduced, and it may no longer be an effective compositional device to catch the viewer’s attention in the foreground. Similarly, the decision to over-paint the background map in “Woman in Blue Reading a Letter” can be made only after the main character has been fixed at a position *on the image plane*. These visual features that artists seek to compose reside in the planar image space, instead of a consequence of a projection from the three dimensional object space. In addition, an artist needs to consider the presentation conditions such as image size, exhibition lighting, viewing distance and framing to make an effective composition (Gooch and Gooch 2005).

Painting is an art medium particularly suitable for sophisticated visual composition, because its common presentation conditions suggest viewers to spend a period of time to look at it. In galleries and museums where paintings are shown, the white gallery wall, the elaborated frames and the professional exhibition lighting all encourage a viewer to pay close attention and take time to contemplate. It is only during this time and through the viewer’s attention that the composition gradually reveals itself. Extending painting techniques such as paint-strokes and compositional rules onto animated and interactive rendering should be carefully considered for their possible aesthetic effects.

In summary, our visual impressions of reality are closely related to the context of seeing. The order in which we look at objects, the amount of time we fix our gaze and the surrounding environments all contribute to the formation of the visual impression of a real scene. Because these visual contexts are highly subjective, and are dependent on particular locations and moments, to define or evaluate an objective ‘realism’ with human viewers is difficult (Ashikhmin and Goyal 2006). At the very least, we can be confident that visual realism cannot be deduced merely from the physical properties in the object space, and it cannot be treated as an optical image recorded in a single instant. We show that artists seek to better represent realistic visual impression by creating compositions based on the manner in which the subject matter should be viewed. As suggested by Aaron Hertzmann (2010), realistic visual impression results from an interactive process between the physical world and its observers. An effective composition serves as a summary and guideline of such a process.

2.3 AN INTERPOLATIVE MATERIAL MODEL

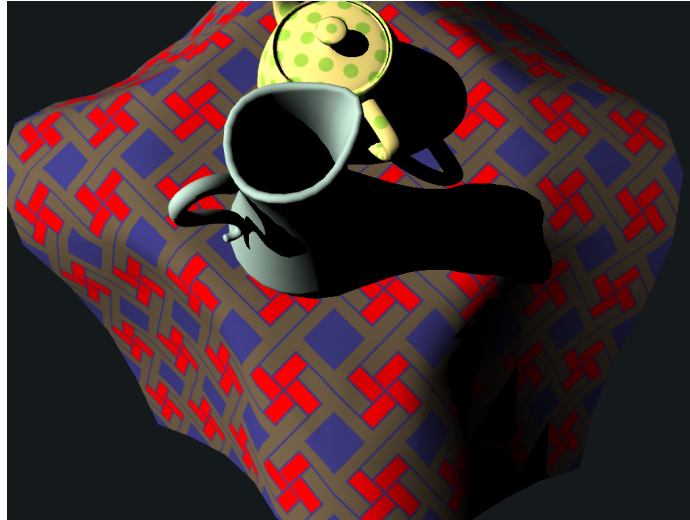
In this section, we show a novel interpolative material model with a painting interface (Ji et al. 2016). This experiment is motivated by the analysis of the Turkish carpet in “The Music Lesson”. One example of our rendering result is shown in Figure 2.9. To create this example, we first make multiple texture images for the table-cloth mesh in an external software tool. When rendering this mesh, we define a ‘brightness value’ for each point on the mesh from the Lambertian diffuse term $\mathbf{n} \cdot \mathbf{l}$ and a few given light sources, taking into account the shadows from other objects (Shirley and Marschner 2009). The ‘brightness value’ is stored on the object surface in a barycentric coordinate per-face texture, similar to the ‘Mesh Colour’ method proposed by Yuksel et al. (2010). Then, we *interpolate* the final pixel colour by sampling all texture images and taking the ‘brightness value’ as the interpolation weight. The interpolation does not need to be linear with the $\mathbf{n} \cdot \mathbf{l}$ diffuse term, and a customized mapping can be used to transfer the diffuse term into brightness value. In the shown example with three texture images (Figure 2.9), the material model can be written as:

$$s_{\mathbf{p}} = f_s(\mathbf{n} \cdot \mathbf{l}), s_{\mathbf{p}} \in [0, 1] \text{ and}$$

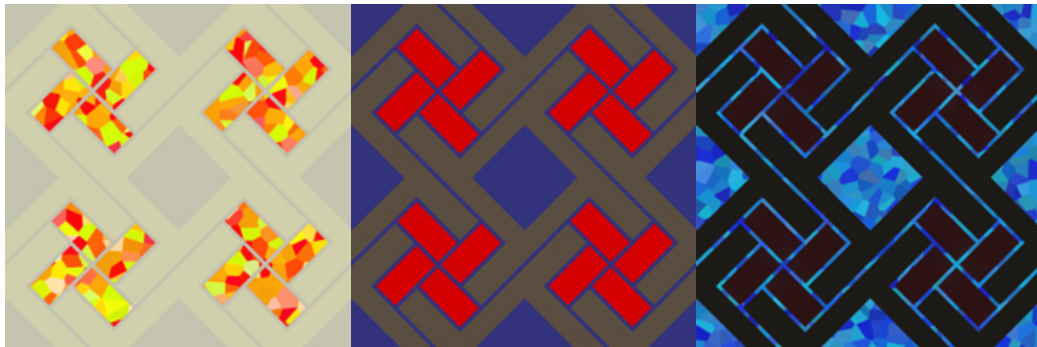
$$c_{\mathbf{p}} = \begin{cases} 2(c_d(0.5 - s_{\mathbf{p}}) + c_a s_{\mathbf{p}}), & 0 \leq s_{\mathbf{p}} < 0.5 \\ 2(c_a(1 - s_{\mathbf{p}}) + c_h(s_{\mathbf{p}} - 0.5)), & 0.5 \leq s_{\mathbf{p}} \leq 1. \end{cases}$$

In the above equation, $s_{\mathbf{p}}$ stands for the brightness value, which is mapped from the diffuse term $\mathbf{n} \cdot \mathbf{l}$ by a customized function f_s . On the user interface, the function f_s is represented as a curve defined in the range of $[0, 1]$, and is called the *shading curve*. The terms c_h , c_a , and c_d stand for the colour sample on point \mathbf{p} from the texture images representing the appearance of the object in highlight, moderate and dim luminance conditions. The final shading colour for surface point \mathbf{p} will be $c_{\mathbf{p}}$.

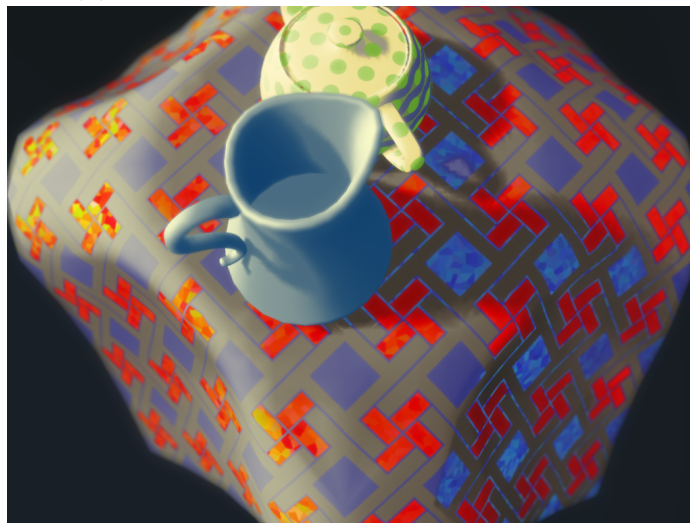
A user can create different shading curves, which will lead to different shading styles in the resultant images, as shown in Figure 2.10. To adjust a shading curve, a user can place and move a series of control points, and the curve will be constructed using monotonic spline interpolation (Carlson and Fritsch 1985). Through our experiments with several artists, we agreed that the curve shown in Figure 2.10a best approaches “The Music Lesson” and several other paintings we discussed. The classical Cel-shading



(a) A table top scene with diffuse shading.

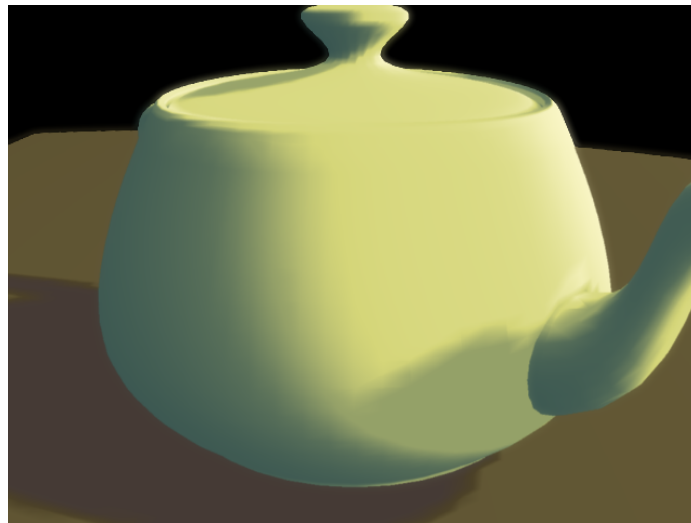
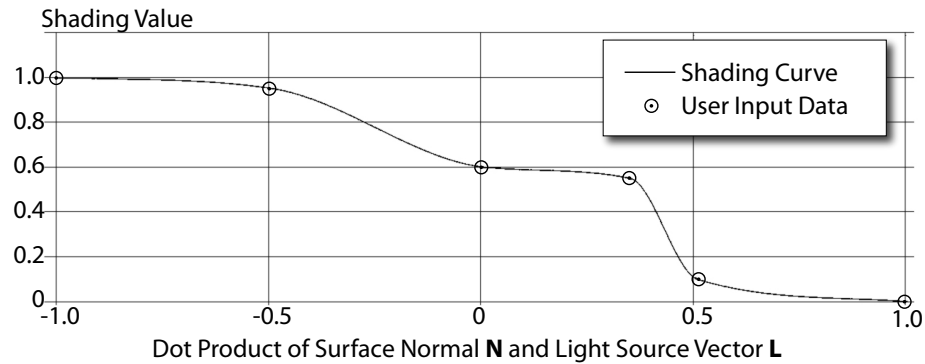


(b) Three textures for the table-cloth mesh.



(c) Rendering with the interpolative material model.

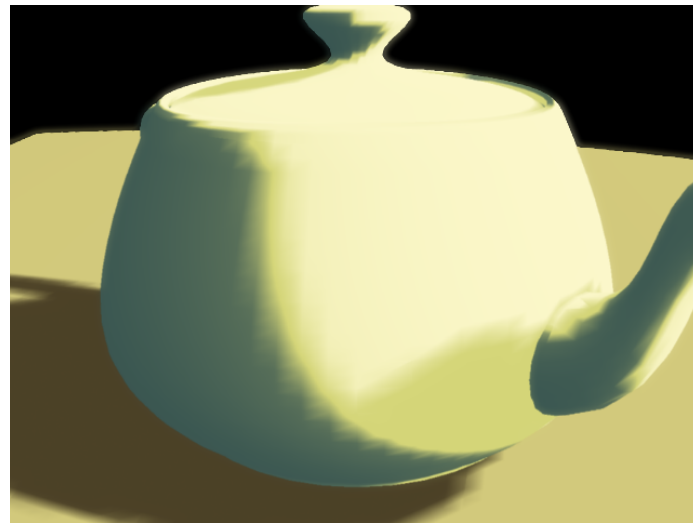
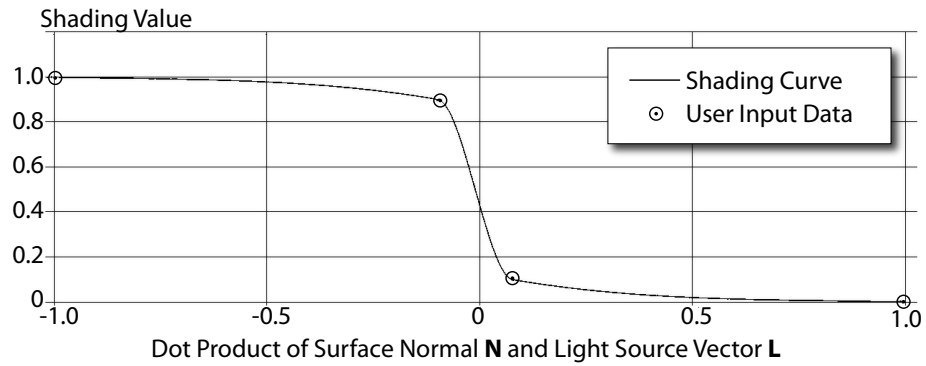
Figure 2.9: The interpolative material model. Figure 2.9a is rendered with conventional diffuse shading. In comparison, Figure 2.9c is rendered with our material model taking the textures in Figure 2.9b. The image clearly defines surface brightness distribution, at the same time exhibits vivid details in the shadows.



(a) A customized shading curve creates a surface shading that smoothly transits between light and shadow.

can be approximated with the curve in Figure 2.10b. With this curve, the rendering result exhibits an illustrative, non-photorealistic style.

With a given shading curve f_s , our interpolative material model can be incorporated into a fully automatic rendering framework. In addition, the brightness value $s_{\mathbf{p}}$ can be edited through a painting interface after it is initialized using the shading curve. The barycentric texture data-structure used for storing $s_{\mathbf{p}}$ is suitable for a painting interface, since it guarantees a bounded sampling rate over the objects' surfaces. Our painting interface is shown in Figure 2.11. An artist can freely move the camera around and choose a brush function from either 'increase brightness', 'decrease brightness' or 'restore the default brightness'. The 'increase brightness' function will increase the brightness value $s_{\mathbf{p}}$ by a given amount set on the interface wherever the brush touches the object's surface, and 'decrease brightness' works in a similar manner. In addition, the 'restore' brush will revert $s_{\mathbf{p}}$ to the result given by f_s . The brush can be set to



(b) A shading curve that mimics the Cel-shading style.

Figure 2.10: User defined shading curves and the corresponding rendering results.

various sizes and alpha blending values. If the alpha is set to a value in between 0 and 1, the result brightness value will be interpolated according to the newly placed stroke and the value stored before. During the editing process, the artist can display a rendering of either the edited brightness value, the default brightness value from the shading curve, or a regular Lambertian shading with the moderate brightness texture image. Our painting software tool allows the edited brightness to be saved using a binary file that contains all the brightness values in the scene, and the edited shading can be loaded later with the corresponding mesh file. The rendering result can also be saved as an image file. An artist can also change one or more of the three texture images without affecting the edited brightness values.

Since the brightness value s_p denotes the intended brightness of local areas on the object's surface, we can adjust the pixels' colour intensity (value) in the resultant

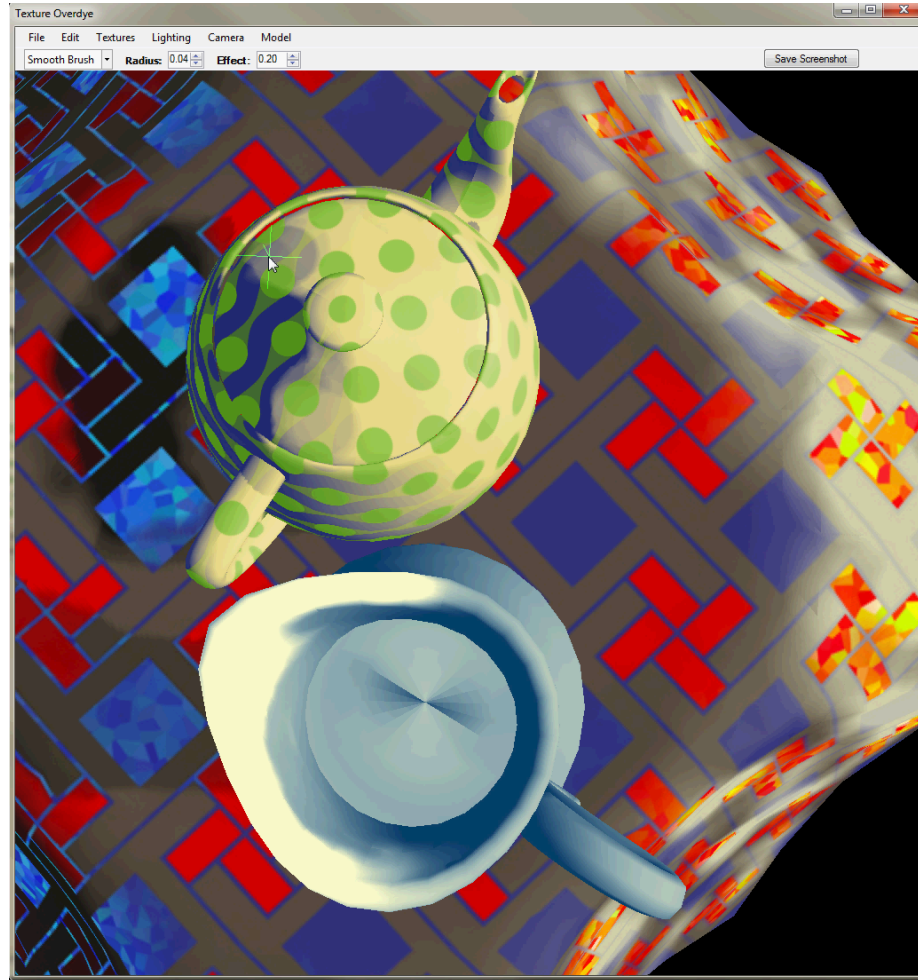


Figure 2.11: The painting interface for brightness editing.

image if they are very different from their corresponding s_p . This is an optional step, and is carried out after the interpolation. If the artist chooses this function, our program will transfer each pixel into the HSV colour space and compare the value channel of the pixel to s_p (Reinhard et al. 2008). If the difference exceeds a given threshold, the program reduces the difference by a percentage and using the modified pixel for the output image.

Our discussions with artists who used our software tool show that our material model is easy to understand and helpful in creating artistic shading. The artists commented that working with the interpolative material model is more similar to painting by hand than conventional shading methods, since the lighting calculation is excluded from the framework; and the artists are guaranteed to get what they put into the textures as the rendering result. The artists also commented that the software

is not advanced enough to include detailed lighting effects such as soft shadows and global illumination effects in the brightness value initialization. The painting interface is useful for constructing large brightness blocks, but it is tedious to work out lighting details by hand. We will address these problems in detail in the next chapter.

* * *

Technically speaking, our method can be related to non-photorealistic rendering methods that mix palettes of colours or textures to create stylized shadings (Barla et al. 2006; Todo et al. 2007; Praun et al. 2001). Rendering tools such as Autodesk Maya sometime come with a ‘ramp shader’ that allow the user to blend multiple input textures into one material. The goal and implementation of our material model is different from these techniques. The most important technical difference here is that we removed the mixing of the colours of the light and the surface material. With our material model, the interpolation result is directly used as the final surface shading (with an optional adjustment of pixel intensities). Existing methods, such as the ramp shader, can generate one procedural texture from multiple images, but the generated texture still needs to go through the standard lighting calculation to create the final surface shading. Our idea can be related to the “Barycentric Shader” method, in which the final shading is also interpolated from a stack of control images (Akleman et al. 2016).

The core idea of our project is to depict brightness instead of light intensity. Light intensity is the physical lighting strength over a unit area or solid angle (Shirley 2003). Most computer graphics shading and material models calculate physically based light intensity or some form of approximation. Brightness is a measurement of visual perception about how much an object appears to radiate or reflect light (Fairchild 2013). Through the discussion of this chapter, we have demonstrated that the visually perceived brightness correlates to physical light intensity in a complicated manner; and the correlation involves the objects in the scene, the viewing context and the subjective intention of the viewer. We have also shown that artists can adjust surface brightness to direct viewers’ eyes and achieve compositional effects. In our research, we seek to prioritize image space visual properties over object space information; and a facet this goal is to depict surface brightness instead of light intensity in the synthesized image.

In our method, the light sources do not emit colour information, but serve as

a reference to initialize the ‘brightness value’ s_p over the scene in a plausible and consistent manner. Beside the shading curve, the surface brightness can also be edited using the painting interface. On the other hand, the entire possible set of appearances of a surface is confined within the range of its textures, since the final pixel colour is *interpolated* without lighting calculations. The interpolative material model provides two important features: first, the rendering result is predicable, and the artist is guaranteed to see exactly what has been put into the textures. Secondly, our material model separates the image composition into several clearly defined steps. The brightness value is localized on the object’s surfaces, and it can be edited without adjusting the object space light sources. In this way, the brightness on the objects’ surfaces is separated with the light sources, and the expression of brightness is separated for the bright, middle and dark tones. If an artist is satisfied with the brightness distribution, but needs to adjust the appearances of brighter or darker parts, s/he can just edit one of the three texture images and render with the updated texture image. Editing of the brightness value or one of the textures is guaranteed to have localized effects, and will not affect either the brightness composition on other parts of the scene, nor the object’s appearance in other brightness ranges. In this way, we alleviated the problem that an artist must go back to the modelling interface to adjust parameters in the object space, in order to modify the light and shade in the synthesized image.

Our method resides in the middle between purely manual painting and fully physically based rendering. When painting by hand, a representational artist has the freedom to directly define the colour of every point on an image, but the artist is also responsible for delivering a three-dimensional illusion in the resultant image. When a physically based rendering framework is used, we input the light sources, geometric models and material properties, and let the program calculate both the perspective of geometry and the light and shade on the rendering result. In the interpolative material model, the perspective projection of geometric models are calculated by the program, while the brightness and surface shading are constructed much like hand based painting. As discuss previously, accurate perspective drawing is difficult for human artists but trivial to calculate in a program. The layout and design of light and shade in an image, however, are easier to improve for an artist, but difficult to evaluate based on current rendering technologies. Our method inherits the freedom of hand based traditional media while leveraging the advantages of digital rendering.

CHAPTER 3

VARIATIONS IN LIGHTING

In the previous chapter, we presented an interpolative material model with a painting interface. Through our experiments and discussions with participating artists, we found that our interface is not effective enough for composing complicated surface shading in detail. Artists like the customizable shading curve f_s and the painting interface to approximately sketch the brightness design onto the objects' surfaces. Using a large brush to brighten and darken a part of a mesh is quite intuitive; and an artist can quickly reach a satisfying overall brightness design. At the same time, they felt that it is difficult and cumbersome to manually paint all the details of light and shade in the depicted scene.

This difficulty consists of three major aspects: first, it is difficult for an artist to draw detailed brightness variation by hand on an object's surface. Introducing automatic algorithms to assist artists in defining details is a necessary task. It conforms with user studies of lighting design (Kerr and Pellacini 2009; Jarabo et al. 2014), which show that a human user cannot reliably draw or paint consistent and accurate light transport effects. Secondly, artists commented that the free-moving viewpoint in the painting interface is confusing. In our brightness painting program, the viewpoint can be freely moved and rotated like a conventional modelling software tool. In contrast, our discussion in the previous chapter shows that image composition depends on a specific layout of objects in the image plane. After a satisfying composition is manually done with a certain viewpoint, changing the view can undermine the existing compositional effects. Finally, to produce any animation with such a manually-painting approach is difficult. Artists will have to paint all key frames of the brightness value mask; and every sample point of brightness value will have to be key-framed in order to be interpolated through time. Because we use barycentric coordinate texture to store brightness values, the number of sample points is proportional to the total surface area of the models in the object space. A large amount of memory corresponding to all surfaces in the scene are needed for the animation.

There are several known methods that could be used to alleviate these difficulties. The first problem of painting and drawing lighting details may be addressed with the aforementioned reverse-lighting methods, in which a program will attempt to match a

set of light sources or environment maps with input paint-strokes. Most methods in this category are limited to physically based lighting, and will reject inputs if they cannot be explained by physically meaningful parameters. Recently, researchers have also demonstrated creating customized lighting effects by extending physically based lighting in an artistic manner (Kerr et al. 2010; Nowrouzezahrai et al. 2011). In our research, we are interested in creating localized surface brightness that does not necessarily have a physical cause. For the second problem, we can fix the view point or let the artist paint on a stack of pre-rendered images. Several published methods of image space relighting can be used for this task (Mertens et al. 2009; Boyadzhiev et al. 2013), but these approaches completely abandon the object space information and prevent any change of the viewpoint. The core problem here is to find a stable *painting plane* for brightness design that is undisturbed by the freely moving viewpoint. Finally, in order to create animation, we need to find a solution that can morph the surface brightness in a natural and coherent way, while still maintaining the overall chiaroscuro throughout the animation.

In this chapter, we present our second research project through which we seek to address these difficulties. This project further develops the idea of a composition-driven brightness design tool with a painting interface. In the following sections, we first examine a few more examples from contemporary photography to illustrate the rationale of our research, then present the technical implementation and application of the novel rendering method.

3.1 EXTRA-SCENE LIGHT TRANSPORT

When composing an image, a representational artist needs to make sure that the colour and shape appear coherently on the planar image surface. At the same time, the image needs to create an illusion of a believable three dimensional scene. Reconciling these two requirements in a satisfying manner may require personal artistry and case-based investigation. Strictly following physically based methods often leads to complicated lighting configurations in order to reach a desired composition, as the light intensity usually does not directly correspond to the perceptual brightness that an artist wants to express. On the other hand, letting a user freely paint surface brightness introduces difficulties in keeping a realistic appearance of the subject matter. Here, we propose two rules that can guide the design of non-physically based light



Figure 3.1: An example of evaluating the perception of inconsistent lighting. The top image has physically consistent lighting, while the bottom image has the Santa Klaus doll re-lit by a different light source (Lopez-Moreno et al. 2010).

and shade:

1. If a lighting effect seems to be originated from a place or object that is not depicted in the resultant image, then it can be customized without breaking the realistic impression. If a lighting effect is resulted from objects visible in the image, then it must be physically consistent.
2. The behaviours of the light transport can change across boundaries of semantically separated objects without breaking the realistic impression. Light and shade on the surface of a single object must be physically consistent.

Research on inconsistent lighting perception with computer graphics has explored related ideas (Lopez-Moreno et al. 2010; Tan et al. 2015) (Figure 3.1); but there is rarely any discussion in the context of image composition. For an example of painting, let us revisit Harmen’s work of vanitas still-life (Figure 2.8). In this painting, a beam

of strong light appears on the background wall. The bright specular reflections on the water jug, on the skull forehead and on several other glossy objects suggest that the beam of light is coming from the upper-left direction in the foreground space. In contrast, the dark edge-side of the table and the deep-blue fabric dripping on the edge seem to indicate that the light volume is cut off on that edge. This sharp halt of lighting on the table's edge creates a clearly delineated major pictorial space, and focuses the viewer's attention onto the things on top of the table. The background light beam is pointing to the forehead of the skull, thus enhancing a sacred, religious feeling in the context of vanitas still life (Sander 2008). One would wonder about the studio environment and be tempted to turn around the view point to look at the source of such lighting. Such a light source does not and needs not exist, given the fact that the painting depicts a frozen moment in time, in which the viewpoint cannot be changed. We call this kind of light transport effects as *extra-scene light transport* effects. The carefully composed extra-scene light transport in this vanitas painting will be intricate to emulate with physically based light sources.

We should also note that the *intra-scene light transport* in the painting is realistic. The laboriously rendered soft shadows on the sea shell, on the pocket watch and of the string on the water jug demonstrate the painter's understanding of realistic light and shadow. The carefully constructed self occlusion shadows on the skull, with the positioning of the corresponding specular reflections consistently suggests a single lighting direction from the upper-left. It is only when depicting a lighting effect originated from an invisible source, that the artist will have the freedom to deviate from physically consistent lighting. The believability of the painting will be undermined if a viewer can see an object in the painting but cannot find the corresponding light transport effects around it.

The powerful and accessible digital cameras and computers have enabled contemporary photographers to create imagery beyond one instant exposure. Many contemporary visual artists spend considerable amount of effort in front of computers to edit their photographs. As discussed in the previous photography example from Jeff Wall (Figure 2.5), compositional techniques of painting are frequently used by photographers in their recent work. The traditional idea that photographs should be an instant, physically consistent depiction of reality is challenged in contemporary times. Here, we will examine an exhibited photograph by artist Scott McFarland to further illustrate our idea. In *The Granite Bowl in the Berlin Lustgarten (after Johann Erdmann Hummel)* (Figure 3.2), the directions of shadows behind characters



Figure 3.2: Scott McFarland, *The Granite Bowl in the Berlin Lustgarten (after Johann Erdmann Hummel)*, 2006. In this composite photograph, every character's shadow points to the central granite bowl sculpture in the Berlin Lustgarten.

are obviously inconsistent. The artist took many photographs of the same scene at various times of a day, and stitched these photographs together into the final image. In this composition, all characters on the right have their shadows pointing to the left, and vice versa. The angle of the shadow is proportional to the distance between the character and the centre of the image, such that it seems the lighting direction for each character is always approximately pointing to the granite bowl. Since this image is also presented on a large format print, this inconsistency in lighting direction does not immediately break the realistic perception, because a viewer needs to move gaze to see these life-sized characters one after another. The disparate directions of shadows is a compositional device that always leads the viewer's gaze to the central granite bowl wherever the viewer started to apprehend the image. In this example, the artist constructed a non-physical extra-scene light source, which casts light onto characters according to their position in the image space. Also, in this image the intra-scene light transport is accurate. For each character, the shape of its shadow matches the character's body shape and consistently points to one single direction that conforms with the direction of lighting perceived on the front of the character. The change of lighting direction only happens in between different characters. Another well-known photograph from the same artist with a similar compositional method is shown in



Figure 3.3: Scott McFarland, *Torn Quilt the Effects of Sunlight*, 2003. In this photograph, each patch of the quilt is a different exposure at a different time of a day. The artist stitched these exposures together skilfully as an analogy of the stitched quilt itself, and created a single image that exhibits an effect of a time-lapse animation. The photograph is exhibited on a large (44 inch by 60 inch) panel.

Figure 3.3.

The concept of extra-scene light transport implies fixed, or at least restricted viewpoints. It suggests a viewer cannot turn the viewpoint around and see how the light was cast on to the scene. In the next section, we further develop this idea and discuss how an artist can use a painting interface to design and customize this kind of extra-scene lighting effects.

3.2 LIGHT AND SHADOW EFFECTS OF TREE FOLIAGE

Directly painting detailed realistic lighting onto objects' surfaces is difficult, and freely moving and rotating the viewpoint while designing surface brightness is confusing. To build a painting interface for designing extra-scene light transport, we turn to traditional art media to find a stable painting surface suitable for this task. For a long time, artists working on photography, theatrical performance and film making have been using a light regulating instrument for a similar purpose. This instrument has many different names, such as gobos, cuculoris or cookies, but their basic forms are always a thin plate of material with different shapes of holes cut out on the surface



Figure 3.4: Film crew altering the natural light into one of the buildings for a scene of National Treasure (2004). The sunshine is modulated by this metal cuculoris and produces a special lighting pattern (Devine 2003).

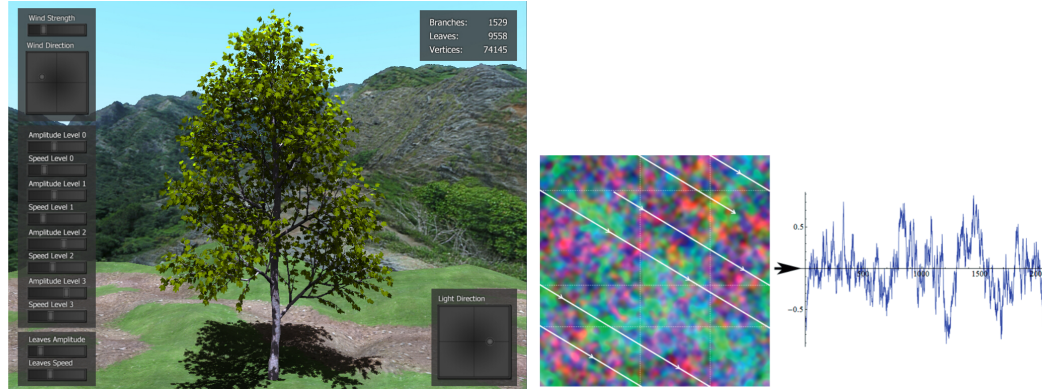


Figure 3.5: Real life foliage shadows exhibit attractive light and shade patterns on the ground. At a certain distance, all light spots appear circular regardless of the shape of the slits inside the foliage.

(Figure 3.4). From an abstract point of view, these instruments can be considered as a plane with varying transparency defined on its surface. When a light source shines from behind, the emitted light will be filtered by the transparency patterns on the plane, and corresponding light and shadow patterns will be projected onto a scene. In computer graphics, similar features may also have different names in various rendering systems (Birn 2006). In our research, we call this feature *projective light mask*. It can be implemented with a plane mesh with a texture image that contains a transparency channel.

The projective light mask provides a stable painting plane for an artist to design light and shade in a scene, which is unaffected by a free viewpoint. A planar projective light mask also implies that the incoming light is approximately orthogonal to the mask. Incoming light around a direction parallel to a planar mask creates largely distorted light and shadows. Keeping these properties in mind, we stage our next research project in the context of rendering animated foliage shadows under sunshine.

In a forest during a clear, sunny day, the vivid light spots on the ground may easily catch one's eyes (Figure 3.5). When the sunshine penetrates tree cupolas in a forest, the light is broken into shattered patterns by the dense tree foliage. When a breeze sweeps through the forest, the light patterns will move around with the swinging branches. If a visitor raises the eyesight to look at a tree cupola, it is nearly impossible to tell which leaf caused a particular light spot on the ground, due to



(a) The rendered tree.

(b) The motion texture for generating stochastic movements.

Figure 3.6: An example of tree animation methods (Habel et al. 2009). The motion texture (b) is used to construct randomized movements for animation.

the highly complicated structure of the tree foliage. The modelling and animation of trees are difficult topics, given the biological complexity of various trees and the non-predictable natural winds. A realistic tree model usually consists of millions of triangles, and to render and animate such a tree with its shadows is computationally expensive. Because every single leaf has a distinct distance to the ground, precisely rendering the soft shadow effects from the tree cupola under sunshine usually requires costly ray-tracing methods.

On the other hand, if the light and shadow from the tree foliage is visible in a view, but the tree cupola is not depicted, then this visual effect can be approached as the aforementioned extra-scene light transport effects. Because the sunshine is close to a directional light, such an effect can be expressed on a planar projective light mask that is orthogonal to the direction of the sunshine. This mask may in turn serve as a painting surface for lighting design. Furthermore, because typical tree cupolae are so complicated that a viewer cannot make direct visual correspondence between individual leaves and their shadows, the modelling and animation of such a visual effect can be based on statistical models and random numbers, instead of physically based animation methods.

Before presenting the implementation of our foliage shadow rendering method, let us briefly review the related research. For plant and tree modelling, most existing methods are inspired by biological rules (Palubicki et al. 2009; Pirk et al. 2012) or mathematical models (Prusinkiewicz et al. 1990; Livny et al. 2011). Recently, researchers also demonstrated creating plant structures based on photo or video inputs (Tan et al.

2007; Bradley et al. 2013; Li et al. 2011). Methods for animating plants in wind combines randomized excitation force with harmonic systems (Van Haevre et al. 2006). In our implementation, we use the $\frac{1}{f^\beta}$ statistical model to initialize our excitation force (Peitgen and Saupe 1988; Ota et al. 2003). This is a statistical model commonly observed in natural phenomena such as the strength of wind or ripples on lakes. Researchers have suggested that a human observer can only judge a highly complex dynamical system by its overall movement frequency and amplitudes (Chuang et al. 2005; Habel et al. 2009) (Figure 3.6). Such complexity is typical of the moving light patterns that can be seen on the ground under trees. The above perception research supports our idea of generating foliage shadow effects with procedural methods on the projective light mask, without actually creating models for trees.

Using a static real-life photo as a digital ‘gobo’ is a well-known practice (Birn 2006), yet little research has further explored the possibility of regulating directional lighting with algorithms, and procedurally creating animated projective light masks. The only related research to the extent of our knowledge is Pellacini et al. (2002), in which a light ‘cookie’ can be painted by the user. Our program creates projective light masks with patterns of light and shadows from an input sketch. To this end, this method is also related to the user-guided pattern generation methods (Hurtut et al. 2009; Ma et al. 2011).

3.3 CREATING LIGHT PATTERNS FROM AN ARTIST’S SKETCH

Our approach to render the foliage shadow effects consists of two steps (Ji et al. 2015). In the first step, we take a sketch that resembles an artist’s lighting design and generate a projective light mask. The light mask produces light and shadow patterns that appear like patterns cast from a foliated tree under sunshine. In the second step, the light mask is animated in a way that resembles foliage movements in a gentle breeze. In this section, we present our method for the first step. Details for creating animation are discussed in the following section.

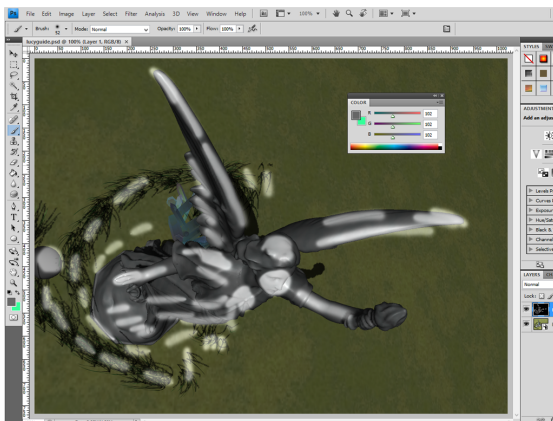
As a starting point of the lighting design, we first render the scene with orthogonal projection from the direction of the assumed sunshine, with a pre-defined view port size that encapsulates the key objects in the virtual scene (Figure 3.7b). Next, an artist is asked to sketch the lighting design on top of the the rendering (Figure 3.7c).



(a) A scene with the Lucy statue.



(b) View from the direction of lighting.



(c) An artist sketches the lighting design.



(d) The extracted sketch.

Figure 3.7: An artist sketches the lighting composition on top of a scene with the Lucy statue. With a view of the scene from the lighting direction as the background (b), the artist sketches a lighting design (c), denoting a bright statue and shadowy surroundings. The sketch is then extracted (d) as the input guide image for synthesizing the projective light mask.

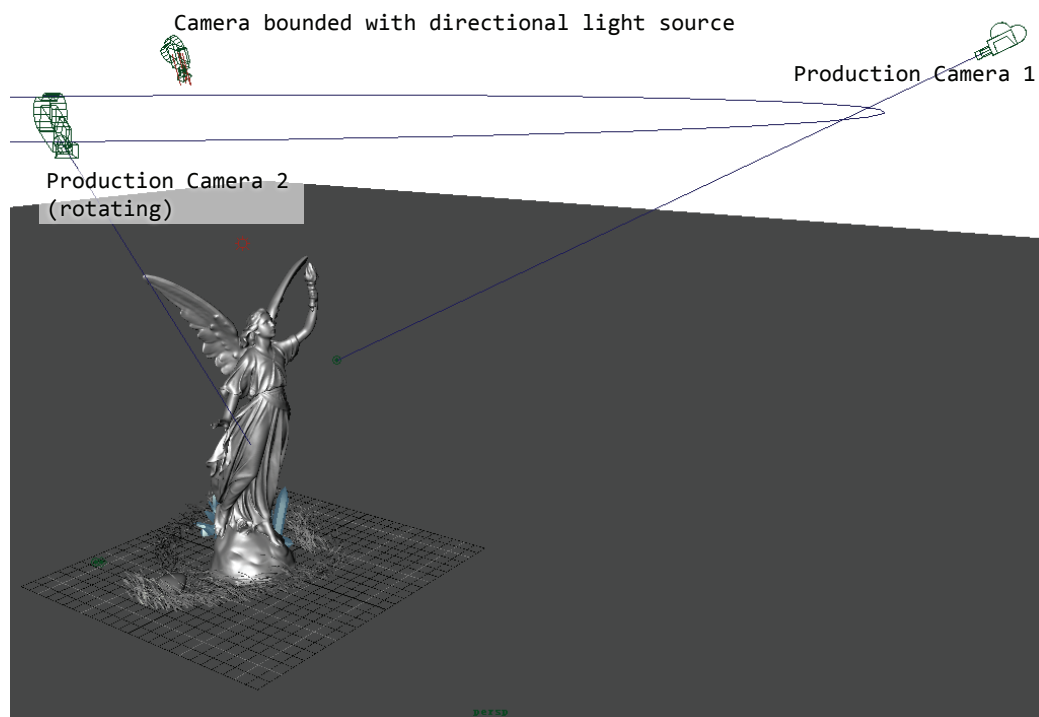


Figure 3.8: An example of scene setup for rendering the foliage shadow effect. A camera is bounded with the directional light source to produce the background for designing the guide image (Figure 3.7b). After the projective mask is created, the scene can be rendered from production cameras at different positions.

The artist can use white strokes to denote light and black for shadow. The painted layer can be initialized as either all-black or all-white for the artist, and it can be painted in any external painting software tools. Afterwards, the transparent layer is extracted as an grey-scale image and is called the *guide image* for the input of our program (Figure 3.7d). This painting process illustrates the idea of a stable painting plane with a directional light source, such as the sunshine. We illustrate the setup of the Lucy scene in Figure 3.8. When an artist designs the light and shade on the projective light mask, which is stationary if the light source is fixed, (s)he does not need to worry about the viewpoint for the final rendering.

An artist only needs to use large paint strokes to sketch the design of light and shade in the guide image. In addition to the guide image, our program also requires the input of a few three-dimensional models as the *elemental shapes*. In the context of rendering foliage shadows, the elemental shapes will be models of tree leaves. Then, the program transfers the coarsely sketched guide image into a projective light mask

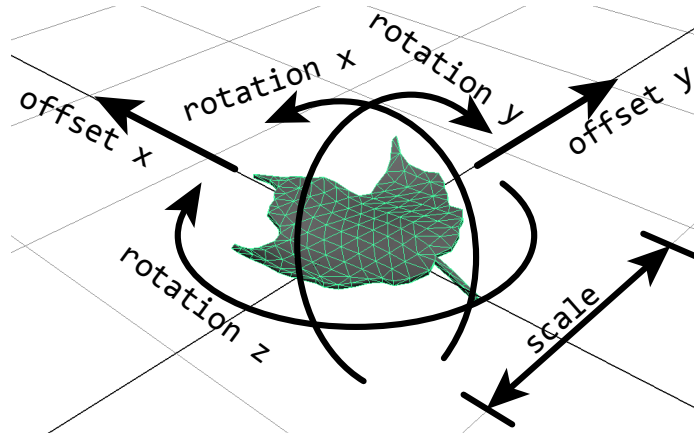


Figure 3.9: Transforms of a shape instance. Shape instances can have different scale, rotation and planar translation, but their geometric centres are fixed on a plane.

with detailed shape patterns. Stochastic optimization is used in this process: we start with a random shape pattern as an initial candidate, then adjust the candidate pattern and evaluate whether it better resembles the guide image, and repeat this process until a satisfying pattern is reached.

The candidate shape pattern is created by duplicating elemental shapes, then rendering with orthogonal projection. The duplicated shape instances can have different scale and rotation transforms, but their geometry centres are constrained in the plane of the projective light mask, and therefore can only have planar offsets (Figure 3.9). The candidate shape patterns are rendered with vertex colours set to black against a white background (Figure 3.10). We compare the rendering of the candidate shape pattern against the guide image, and set the goal function of the optimization z to be the accumulated pixel value difference between them:

$$z(\mathbf{I}') = \left(\sum |V(\mathbf{I}_{x,y}) - V(\mathbf{I}'_{x,y})| \right) / N.$$

In the above equation, \mathbf{I} stands for the guide image, and \mathbf{I}' represents the rendering of a candidate shape pattern in the optimization. V denotes the value of a pixel. N represents the total number of pixels in the rendered image, so the goal function is normalized in $[0, 1]$. Here, we assume that the guide image has exactly the same pixel resolution as the rendering resolution of the shape patterns. If \mathbf{I} and \mathbf{I}' have different number of pixels, our program will scale the guide image to the size of the shape pattern rendering by tri-linear interpolation.

In each iteration of the optimization, we start with a candidate shape pattern as the

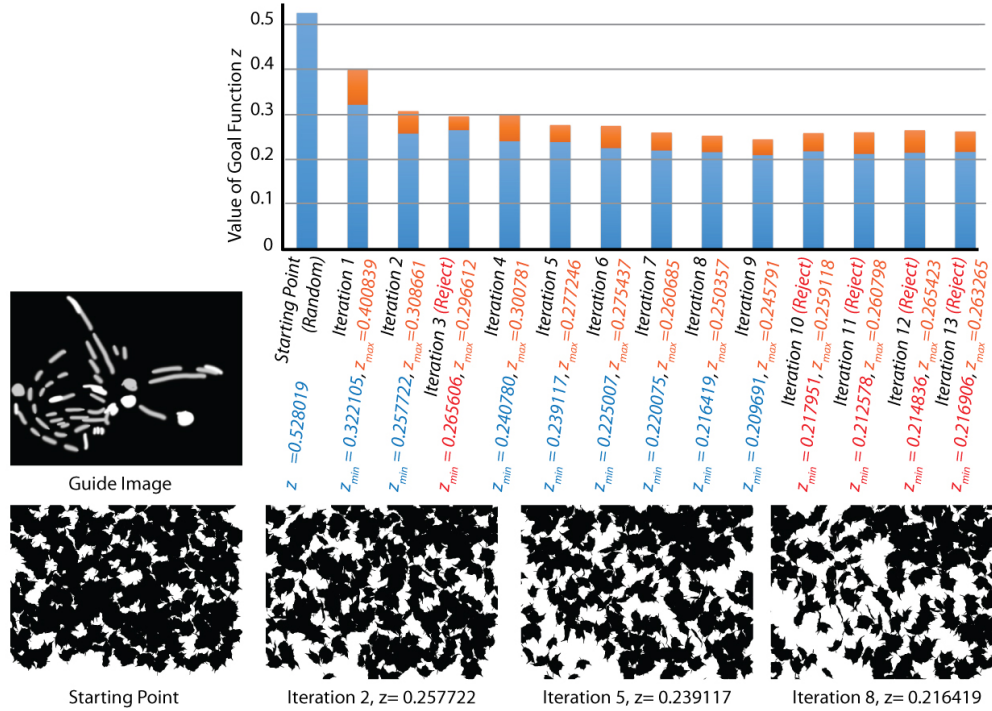


Figure 3.10: The stochastic optimization starts with a random shape pattern, and converges to a shape pattern that resembles the guide image. In this example, 12 new candidates are generated in each iteration; the lowest (blue) and highest (orange) goal function values of each iteration are illustrated in the top plot. Reference solutions (best shape pattern candidates) from some of the iterations are shown below.

reference solution. From the reference solution, we generate multiple new candidates by adding or deleting shape instances and adjusting the transforms of existing shape instances. We then evaluate the goal function z on the newly created candidate, and compare it with the z value of the reference solution. If any new candidate results in a better (smaller) z value, we pick the best candidate shape pattern in the current iteration, and set it as the reference solution for the next iteration (Figure 3.10). Otherwise, we keep the reference solution from the previous iteration, and discard all newly generated candidate shape patterns (these iterations are marked with a red z_{min} value and a ‘Reject’ in the top plot of Figure 3.10). Our program visualizes the optimization process in real time by rendering the current reference solution onto the screen. The optimization keeps iterating until one of the following criteria is met:

1. The goal function z of a candidate shape pattern evaluates to a value below a given threshold z_0 ;

2. The maximum time set by the artist for the optimization has expired;
3. The artist decides to terminate the optimization because the current reference solution appears satisfying.

To make optimization converges efficiently, we calculate the difference between a candidate shape pattern and the guide image *part by part* for optimization heuristics. We assume that if we can make a shape pattern look similar to the guide image in every part, then their overall appearance will also be similar. We illustrate this optimization heuristics with a simplified example in Figure 3.11. We first calculate a *difference image* \mathbf{R} by subtracting the candidate shape pattern from the guide image:

$$\mathbf{R}_{x,y} = V(\mathbf{I}_{x,y}) - V(\mathbf{I}'_{x,y}).$$

The difference image \mathbf{R} will have both positive and negative pixels, and is handled with a special implementation. We then create a set of *evaluation cells* \mathbf{C} on \mathbf{R} . Each evaluation cell \mathbf{C}_i contains a square clip of the difference image. There are two kinds of evaluation cells:

1. A planar tiling of evaluation cells of the same size covers the entire image. They form the *evaluation lattice* on \mathbf{R} . The evaluation lattice is shown with dark brown lines in Figure 3.11c. Cells in the evaluation lattice are created when the optimization starts.
2. *Instance Cells* are evaluation cells with their centres aligned with the geometric centres of the shape instances, which are represented by the yellow squares in Figure 3.11c. When we create a shape instance by duplicating an elemental shape and placing the instance onto the shape pattern, we create an instance cell with it. When a shape instance is deleted, its corresponding instance cell is also removed.

Our program then calculates an *accumulated difference index* g_i within the region of each evaluation cell \mathbf{C}_i :

$$g_i(\mathbf{C}_i) = \frac{\sum \mathbf{R}_{x,y}}{N_i} = \frac{\sum (\mathbf{I}_{x,y} - \mathbf{I}'_{x,y})}{N_i}, \quad (x, y) \in \mathbf{C}_i.$$

In this equation, g_i represents the difference between the shape pattern and the guide image in the local region of \mathbf{C}_i . N_i denotes the total number of pixels in \mathbf{C}_i , so

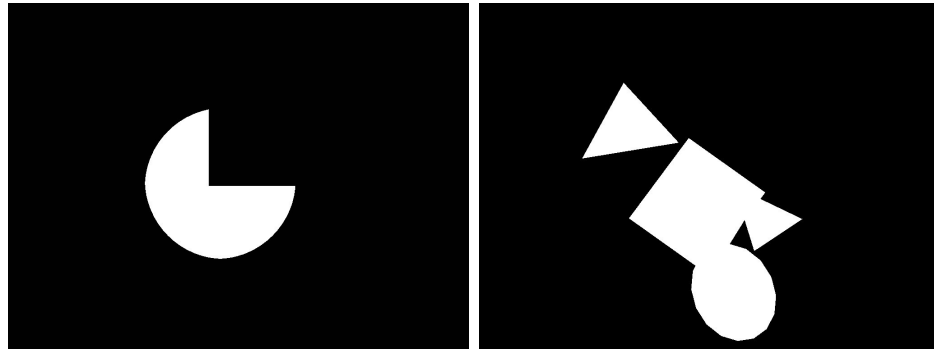
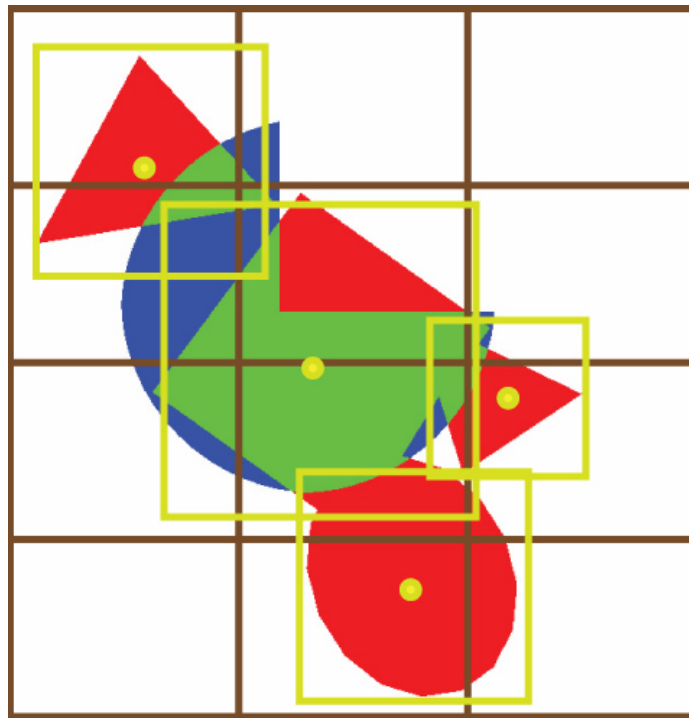
(a) Example guide image \mathbf{I} .(b) Candidate shape pattern \mathbf{I}' .(c) Difference image \mathbf{R} and the evaluation cells (enlarged centre part of \mathbf{R}).

Figure 3.11: Optimization heuristics with a simplified example. Blue in the difference image (c) denotes pixels covered only by the guide image but not the candidate shape pattern, while red stands for pixels covered only by the candidate shape pattern but not the guide image. The dark brown squares represent the evaluation lattice. The yellow squares represent the instance cells, which have their centres aligned with the geometric centres of the shape instances.

g_i is normalized. If g_i is positive, it means that the shape pattern coverage needs to be increased in \mathbf{C}_i ; and if g_i is negative, it means that the shape pattern covers too much area.

To guide the optimization with g_i , we set up a series of threshold values, and determine the appropriate adjustment on the shape pattern by comparing g_i of a specific evaluation cell \mathbf{C}_i to these threshold values. In the following list, λ_{cre} and λ_{del} are threshold values for creating and deleting shape instances. λ_{inc} and λ_{rec} denote threshold values for increasing or reducing a shape instance's rendering coverage. λ_{del} and λ_{rec} should have negative values, and the inequality $-1 < \lambda_{\text{del}} \leq \lambda_{\text{rec}} \leq 0 \leq \lambda_{\text{inc}} \leq \lambda_{\text{cre}} < 1$ should hold.

Given an evaluation cell \mathbf{C}_i and its corresponding difference index g_i :

- if the evaluation cell \mathbf{C}_i belongs to the evaluation lattice,
 - if $g_i > \lambda_{\text{cre}}$, we calculate a ‘create’ probability $p_{\text{cre}} = \frac{g_i - \lambda_{\text{cre}}}{1 - \lambda_{\text{cre}}}$. When generating a new candidate shape pattern, our program creates a new shape instance with the probability of p_{cre} in \mathbf{C}_i . The initial geometric centre of the newly created shape instance is a random point inside \mathbf{C}_i . Other rendering transforms of the newly created shape instance, such as the rotation and scaling, are randomly initialized.
- if \mathbf{C}_i is an instance cell attached to a shape instance i ,
 - if $g_i < \lambda_{\text{del}}$, we calculate a ‘delete’ probability $p_{\text{del}} = \frac{g_i - \lambda_{\text{del}}}{-1 - \lambda_{\text{del}}}$. When generating a new candidate shape pattern, our program deletes shape instance i with the probability p_{del} .
 - if $g_i < \lambda_{\text{rec}}$, our program decreases the coverage of shape instance i on the shape pattern by reducing its scale parameter with a small random amount.
 - if $g_i > \lambda_{\text{inc}}$, our program increases the coverage of shape instance i by increasing its scale parameter with a small random amount.

Typically, an optimization iteration starts with creating the instance evaluation cells for the reference solution, continues with calculating the difference index g_i for each cell, then generates multiple new candidate shape patterns by examining all difference indices and performing corresponding adjustments. Our program also adds

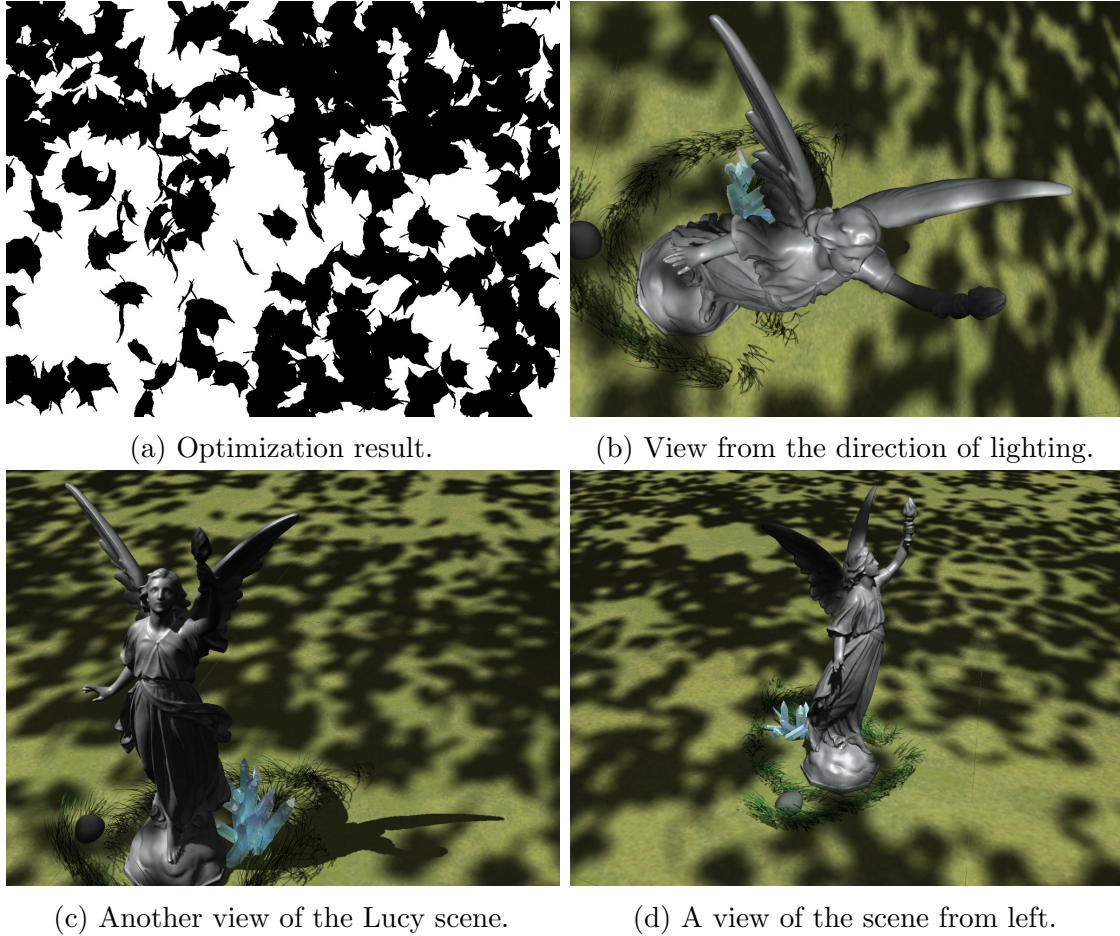


Figure 3.12: Foliage shadow effect rendering for the Lucy scene. From the guide image (Figure 3.7d), our program generates a projective light mask (a) using a leaf model (Figure 3.9) as the elemental shape. In the final rendering (b, c, d) the initial lighting design is preserved and details of the lighting are automatically filled in.

small amount of random perturbations on the rotation and translation transforms of every shape instance when creating a new candidate shape pattern. Although the evaluation cells and difference indices g_i stay the same for the same reference solution, we can generate non-repetitive new candidates as long as we use independent random numbers through the process. Therefore, if our program discards all new candidates in an iteration because they all evaluate to larger z than the reference solution, the program can keep using the evaluation cells and the difference indices calculated in the previous iteration. As long as the program uses independent random numbers, it only needs to calculate the heuristics information once for one reference solution, until a better shape pattern is reached. We show the optimization and rendering

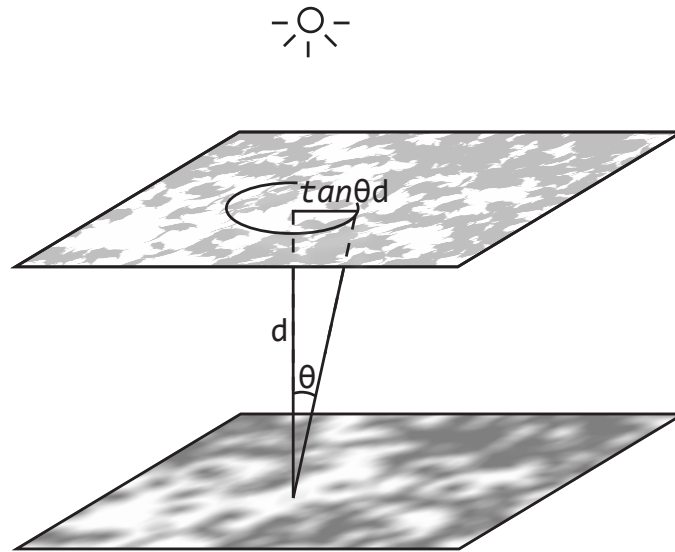
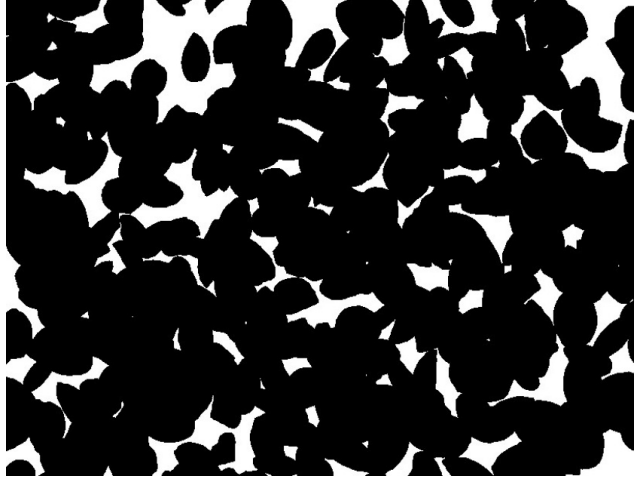


Figure 3.13: Soft shadow effect under sunshine. Observed from the earth, the sun has a constant angular diameter of $\theta = 0.5^\circ$, thus the blurring radius of a leaf with distance d towards the ground is $\tan \theta d$.

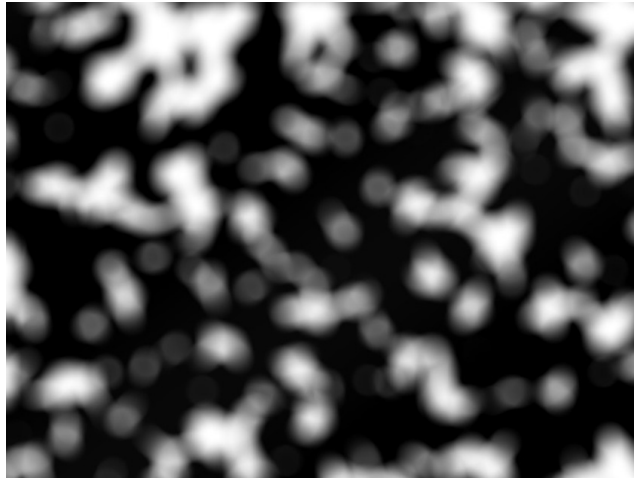
results in Figure 3.12. In these final renderings, the initial lighting design of a bright Lucy statue with shadowy surroundings is preserved. The view from the direction of lighting (Figure 3.12b) fits well with the given sketch of lighting design (Figure 3.7c).

* * *

The above optimization algorithm generates a shape pattern that matches the input guide image with elemental shapes. This shape pattern can be directly used as a projective light mask, but the rendering result is too sharp for shadows under sunshine. Because the sun is not a point light source, shadows from objects with a distance to the ground exhibit a soft shadow effect (Figure 3.13). Viewed from Earth, the sun has a constant angular diameter of approximately $\theta = 0.5$ degree (Seeds and Backman 2010). If the assumed distance from a leaf to the ground is d , then its shadow on the ground should be blurred with a circular kernel of diameter $\tan \theta d$. In our implementation, the blur is treated as a convolution of the sharp rendering result with a filter image (Soler and Sillion 1998). The filter image is initialized as all-black with a circular white spot of diameter $\tan \theta d$ at the centre. The artist can pick another image as the filter, such as a photograph of the sun (Max 1991). The blurred shape pattern closely resembles the real life photograph of foliage shadows (Figure 3.14).



(a) Sharp shadow pattern.



(b) The pattern blurred with a radius of 20 pixels.



(c) Photograph of real foliage shadows.

Figure 3.14: Rendering the soft shadow effect, and comparison with a photograph.



Figure 3.15: Field study of foliage shadows. We photographed a white foam core in a forest. The foam core is placed with an angle approximately orthogonal to the sunshine.

For a tall tree, foliage on different heights can cast shadows with different degrees of blur onto the ground. To render this effect, our program can create multiple layers of shape patterns on the projective light mask, applying different blur radii to each layer and combining them together. In practice two or three layers suffice for a believable rendering.

A notable effect of the foliage shadows is the circular shape of light patterns on the ground regardless of the shapes of the slits inside the foliage (Figure 3.5, 3.14c). This is because slits inside foliage act as pin-hole cameras and project the sun's image onto the ground at a proper distance. This effect naturally emerges when proper distance parameters are set. The photograph of real foliage shadows in Figure 3.14c is taken in a field study. We placed a framed white foam-core board in a forest on an angle approximately orthogonal to the sunshine, and photographed the shadows on the foam-core (Figure 3.15).

The blur convolution is executed in the frequency domain as a multiplication, with corresponding forward and inverse two-dimensional Fourier transforms on the rendered image. As the distance parameters do not change throughout the animation, the frequency domain filter is pre-calculated. In our implementation, the Fourier transforms and texture multiplications are executed on the GPU with Microsoft Direct 11 Compute Shader. With two or three layers of shadow patterns, which are black-and-white and only contain one floating point number per pixel, the blur convolution can be completed in approximately 200 milliseconds on the GTX 260 graphics card. A projective mask with RGB colours will require three passes of convolution, and therefore will take longer to complete.

3.4 ANIMATION WITH HARMONIC MOTION

In addition to rendering static projective light masks resembling foliage shadows, we also show our method for creating animations for them. When a breeze passes a forest, the light and shadow patterns on the ground move around in a visually attractive manner. As discussed previously, tree animation in natural wind can be approximated with harmonic systems (tree branches) driven by signals resembling the force of the wind. Since the shadows are projected from the tree foliage to the ground by the directional sunshine, we can consider this as an orthogonal projection. Therefore, movements of the light and shade on the ground are also harmonic. Because we only render the light and shadows on the projective light mask, we can directly construct harmonic trajectories for the shape instances on the mask, without animating structures of trees. To drive the harmonic movement, we use the $\frac{1}{f^\beta}$ statistical model which resembles the frequency distribution of many natural phenomena (Peitgen and Saupe 1988). A signal conforming to the $\frac{1}{f^\beta}$ model appears random in the time domain, but exhibits a reciprocal curve similar to $\frac{1}{f^\beta}$ in its power spectrum, where f denotes frequency and β is a given constant. Smaller β values result in larger high-frequency turbulence in the signal; while larger values cause smoother oscillation (Figure 3.16).

We propose a method with two stages for creating animation of the shape patterns. In the first stage, we generate a random excitation that conforms to the $\frac{1}{f^\beta}$ statistical model. This is achieved by creating the absolute value part and the phase (argument) part separately in the frequency domain. The second stage runs a harmonic motion simulation with the excitation created in the first stage as the excitation force. Our

method is listed as follows:

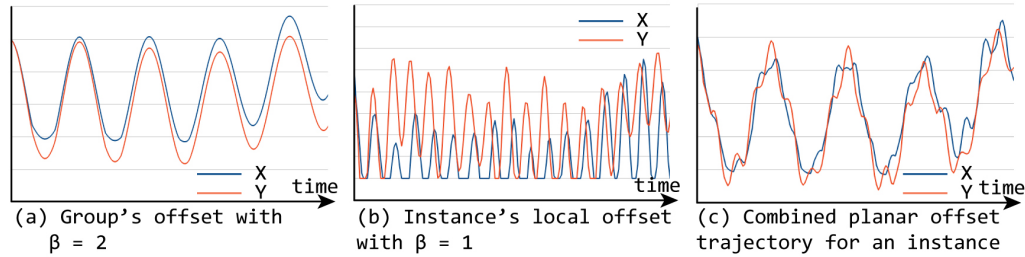


Figure 3.16: Stochastic harmonic motion simulation with the $\frac{1}{f^\beta}$ signal as the excitation. (a) Trajectory for a shape group driven by an excitation with $\beta = 2$. (b) The local movement of a shape instance is driven by an excitation with $\beta = 1$, resulting in larger high frequency oscillation. The movement is also constrained in the instance's local movement range. (c) Combined shape instance trajectory over time. Each plot has individually scaled vertical axis, and in this example the local movement has a smaller range than the group movement. Time (horizontal) axes are of the same scale.

Stage 1. Creating the $\frac{1}{f^\beta}$ excitation force.

Assuming that the animation contains N frames, with k frames per-second:

1. Set the first value of the frequency domain representation for the excitation to 0, since we need shape instances to fluctuate around its original position and to have no static offset.
2. Calculate the frequency step $f_\lambda = \frac{k}{N}$. In the frequency domain representation, the i_{th} value stands for the sinusoidal component of frequency if_λ .
3. Generate $N/2$ random numbers for the absolute value part of the frequency domain representation. The values should approximately conform to the $\frac{1}{f^\beta}$ distribution (with the first value set to 0). For the i_{th} absolute value:
 - (a) Construct a normal distribution $N(\mu, \sigma^2)$ with the mean $\mu = \frac{1}{(if_\lambda)^\beta}$ and variance $\sigma = m\mu$, where m is a given small deviation factor ($\ll 1$).
 - (b) Use a Gaussian random number generator to create a random value that conforms to the above normal distribution. In our implementation, the Box-Muller method is used (Box and Muller 1958).

The absolute values need to be greater or equal than zero, so the result is set to zero if the Gaussian random number generator gives a negative value.

4. Generate $\frac{N}{2}$ uniform random values in the range of $[-\pi, \pi]$ to be used as the phase part of the frequency domain representation.
5. Combine the phase part with the absolute value part to form complete complex numbers, and extend the sequence to size N with complex conjugation (explained below). Then, the time domain excitation is calculated using the inverse Fourier transform.

Since this sequence of excitation needs to be real numbers in the time domain, its frequency domain representation must be conjugation symmetric (Oppenheim et al. 1983). Therefore, our program only needs to generate $N/2$ complex numbers in the frequency domain, and the high frequency half of the frequency domain representation can be deduced with complex conjugation. The values in the excitation sequence are in the range of $[0, 1]$. They are multiplied with a given excitation magnitude before being used to drive the harmonic motion simulation. If an indefinite length of animation is required instead of a fixed length of N frames, our program simply continues to drive the harmonic motion from the start of an excitation sequence when the sequence is exhausted. Repeating the entire excitation sequence in the time domain does not alter its frequency domain $\frac{1}{f^\beta}$ distribution. In our implementation, we set $N = 2000$ and $k = 30$.

Stage 2. Creating the movement trajectory

Instead of letting every shape instance move at an independent random direction, we create coherent movement for them by making them move in groups. The groups of shape instances resemble the foliage clusters on large tree branches; and this setting improves the visual believability of the animation. We set the number of groups according to the number of shape instances, and determine the groups of shape instances with the fuzzy C-Mean clustering algorithm (FCM) (Bezdek 1981). Aesthetically, the shape groups should overlap each other a little; thus a fuzzy clustering algorithm is chosen over a binary clustering algorithm such as K-Means.

We implement the Duhamel's integral method for simulating a single degree of freedom harmonic motion with an arbitrary excitation (Craig and Kurdila 2006). As our program focuses on visual believability instead of physical accuracy, satisfying

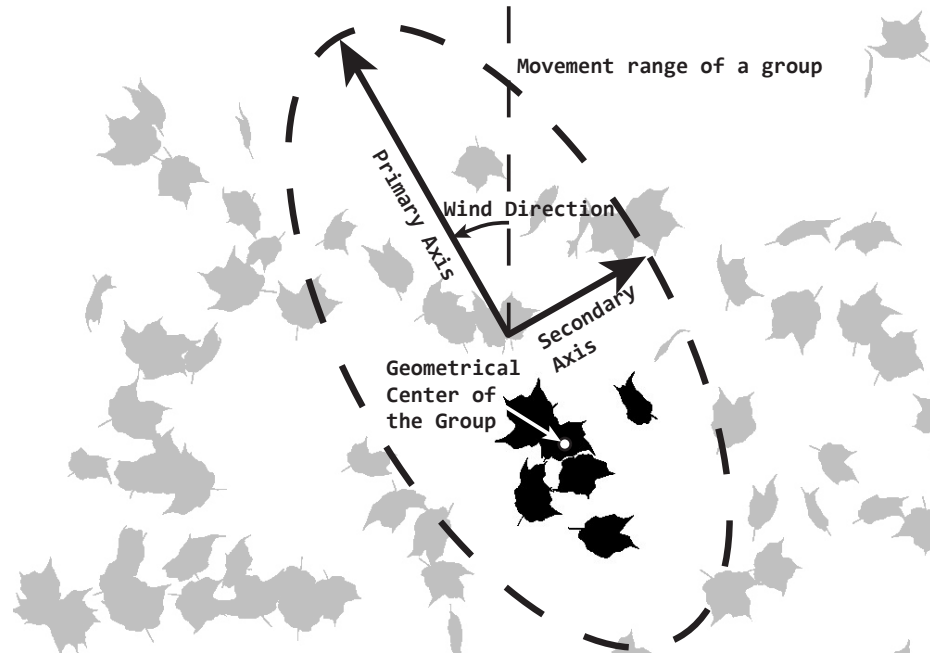


Figure 3.17: Movement of a shape group, which is calculated as the offset of the group centre and follows a given wind direction.

results are achieved using one simulation time step per rendering frame. We list the harmonic movement algorithm as follows:

1. Arrange the shape instances into several groups by applying the FCM algorithm to their geometry centres. The FCM algorithm calculates a probability matrix that represents the probability of each shape instance belonging to a particular group. Our program determines the group of a shape instance by drawing a uniform random number in the range of $[0, 1]$, and compares it to the group entries in the matrix. The first group that has a greater accumulated probability than the random number is chosen as the instance's group.
2. Create the planar trajectories of the groups by simulating two orthogonal, one-dimensional harmonic motions for each group, driven by independent excitation sequences. The movement of the shape groups are also confined in the image plane, like the geometry centres of the shape instances. One of the two excitations is multiplied with a larger magnitude; and the group will travel further in this direction. The group trajectories are also rotated by a given angle to conform with the given wind direction (Figure 3.17).

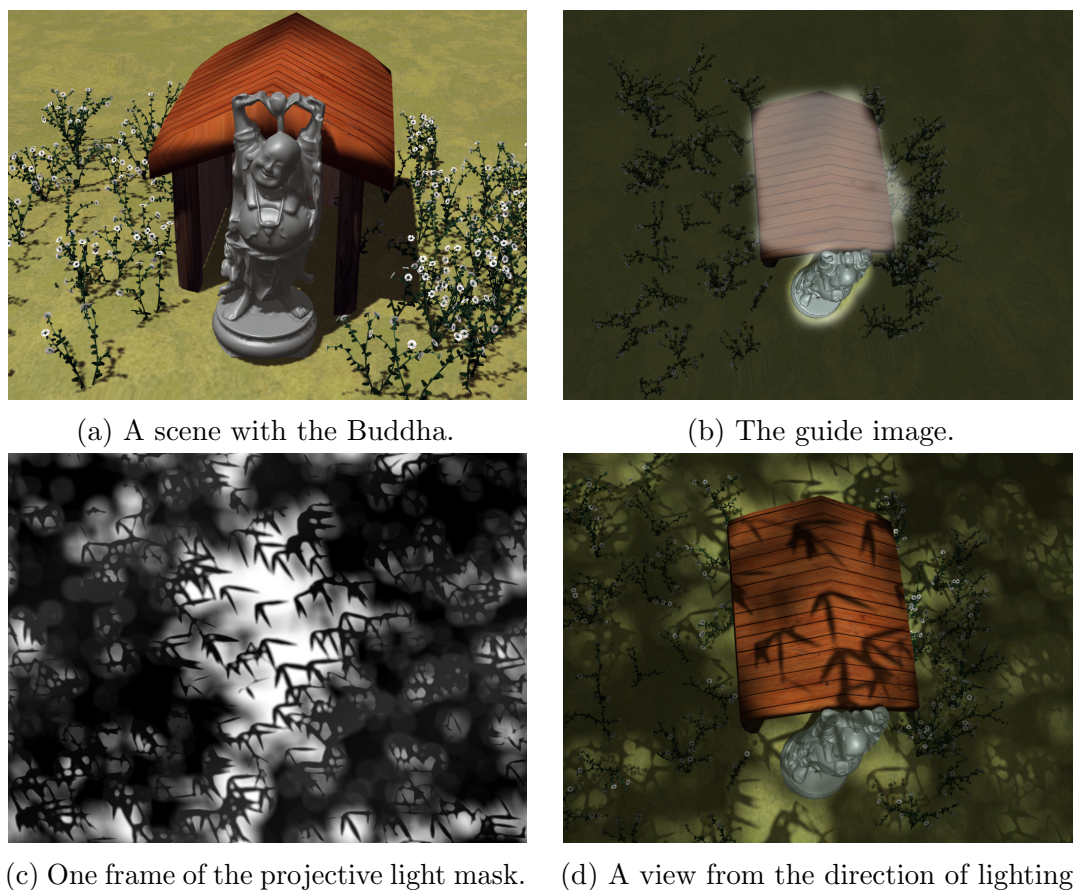
3. Create the trajectory of every shape instance by simulating five one-dimensional harmonic motions with independent random excitation. They are used as the movements on the three rotational axes plus two planar translation axes of a shape instance. (Remember that shape instances have the geometric centres constrained inside a plane, but are free to rotate and move otherwise.) Excitations used in this step are multiplied with small magnitudes (Figure 3.16).

In our implementation, the movements of groups are driven by $\frac{1}{f^\beta}$ excitations with $\beta = 2$. The high β value creates smooth movements that appear similar to sinusoidal trajectories with slow moving centres. Movements of shape instances are driven by excitations with $\beta = 1$, which create large amounts of high frequency oscillations. When rendering an animation frame, the movement of a shape instance is added on top of its group movement to produce the final rendering transform for that shape instance (Figure 3.16). The movements of shape groups and instances are also confined within given ranges. These movement ranges are given as parameters to our program, and they ensure that at any time during the animation, the shape patterns in the animation will not deviate too much from the guide image.

* * *

Our rendering program reads an image file as the guide image, and loads the elemental shapes from an Autodesk .fbx file (Autodesk 2014). The artist using our program can design the guide image and elemental shapes with any preferred software tools. The program stores numerical parameters in an XML file, such as the optimization heuristics thresholds (λ_{cre} , λ_{del} , etc.) and the intrinsic frequency and damping for simulating the harmonic motion. The random number generator we used throughout our program is the Mersenne Twister from Saito and Matsumoto (2008). It is a pseudo-random number generator, which ensures our program to render exactly the same shape pattern and animation as long as the random seed does not change.

On a desktop computer with an Intel i7 processor and a Nvidia GTX 280 graphics card, the optimization process takes around 10 seconds per iteration and one minute in total. The animation algorithm creates approximately 100 frames per-minute and writes them to the hard drive as images or videos. The exact performance depends on the complexity of the elemental shape, the number of shape instances and the rendering resolution. The entire process is visualized on the screen in real time, and the artist can pause or halt the program any time during the optimization or



(a) A scene with the Buddha.

(b) The guide image.



(c) One frame of the projective light mask.



(d) A view from the direction of lighting.

Figure 3.18: A Buddha scene rendered with the foliage shadow effects.

animation stage. Through our project, we worked closely with artists and improved our methods according to their feedback. The field study in which we observed real foliage shadows in nature provided the initial motivation and guided our algorithmic design (Figure 3.15).

One technical limitation of our method is that the generated shape pattern will only approximately resemble the guide image. The method is designed to create shape patterns that match sketchy inputs, and to automatically add details on the light mask using elemental shapes. Therefore, if the guide image contains details smaller than the elemental shape, these details will likely be lost in the generated light mask. When we create animation, the shape patterns will further change their appearances. Although there are range parameters to limit the movement of groups and shape instances, very detailed drawings in the guide image are still likely to be lost during the animation.

A Buddha scene composed using our method is shown in Figure 3.18. The guide image (Figure 3.18b) denotes a bright-lit statue, a partly lit roof and shadowy



(a)

(b)

Figure 3.19: More rendering results of the Buddha scene.

surroundings. In this scene, the elemental shapes are models of bamboo leaves, and there are two layers of shadow patterns with different blur radius (Figure 3.18c). In the final rendering (Figure 3.18d and Figure 3.19), the lighting composition in the guide image is preserved. We will further evaluate this method in depth in the next chapter.

CHAPTER 4

ART PRACTICE AND FUTURE WORK

Besides programming projects, our interdisciplinary research of image composition also consists of our practices in visual arts. In this chapter, we describe two art projects that were developed alongside the programming projects, and a user study with the visual effects artists who tried to use our rendering tool. The art projects and the user study not only validate the analysis and algorithms we presented before, but also serve as effective guides for the future development of our research. From our point of view, the benefits of being involved in art making as a computer rendering researcher consists of at least three aspects:

- First, art making is result oriented, and an artist is not required to generalize the work flow of making a piece to form a theory or an algorithmic design. In this context, we can freely experiment with a wide range compositional techniques by hand, evaluating their effectiveness and complexity. Experiences acquired through this process are helpful for the development of our rendering software tools.
- Secondly, a visual artist needs to carefully consider many factors of visual presentation that are not usually examined in computer graphics research. In a computer lab, we typically only look at rendering results on monitors or with small digital projectors. Through our art projects, we explore how synthesized images appear on large format prints, or when projected to a large area in an outdoor installation. Working with these presentation forms leads to opportunities and challenges that cannot be discovered in a computer lab.
- Lastly, we strive to communicate with peer artists using their conventional language to evaluate our rendering results. The common way to acquire feedback from artists is by conducting a user study, which usually takes the form of asking artists to fill out pre-defined questionnaires made by the software developer. In addition to this kind of user studies, we also held oral critiques with peer artists that allow all research participants to communicate their thoughts more effectively.



Figure 4.1: The final image of our photography art project.

4.1 BRIGHTNESS COMPOSITION IN PHOTOGRAPHY

In chapter 2, we have discussed how artists modify the light and shade in a depicted scene, in order to direct viewers' eyes and to achieve compositional effects. Alongside the interpolative material model project, we experimented the discussed compositional theories with digital photography (Ji et al. 2016). In this art project, we experiment with image composition techniques without involving computer graphics rendering. In contemporary professional rendering systems, there are usually a large number of various rendering algorithms that can affect the quality of the synthesized image. It is difficult to clarify whether a visual feature is due to a particular type of rendering algorithms or due to a compositional technique. We therefore choose to work with digital photography, which is by definition perfectly photorealistic, and to experiment with the proposed compositional techniques using photographs taken in a studio setting.

The final image of our photography project is shown in Figure 4.1. In this art project, we follow the style of historical flower still-life paintings (Taylor 1995) (Figure 1.10), with contemporary motifs such as a computer box. We take multiple shots towards a table-top arrangement under different lighting conditions, including direct sunshine from the side window, and various indoor studio lighting (Figure 4.2). For each shot, we keep the camera's position, orientation, aperture stop and most other parameters fixed, but adjust the exposure time for the different lighting conditions. Afterwards, the photos are combined with painted alpha layer masks (Figure 4.3). The alpha masks specify which exposure will be visible on each pixel position. In the final image, the red lily in the centre, the red daisy on the upper-right and the violet orchid on the lower left come from the photo taken under direct sunshine (Figure 4.2a). A few other central objects such as the smaller yellow daisy in the lower-right are from the photo with bright indoor lighting (Figure 4.2b). Objects in the middle ground are lit by indoor studio lighting (Figure 4.2c). The background fabrics, the part of the computer box behind the bouquet and the blackberry vein come from a dark photo (Figure 4.2d). Intuitively speaking, the goal is to create a picture in which the red lily appears very bright and attractive, while other surrounding flowers gradually come into view. In the actual project approximately 20 exposures of the same scene are used to compose the final image. This technique can be related to the work of Jeff Wall



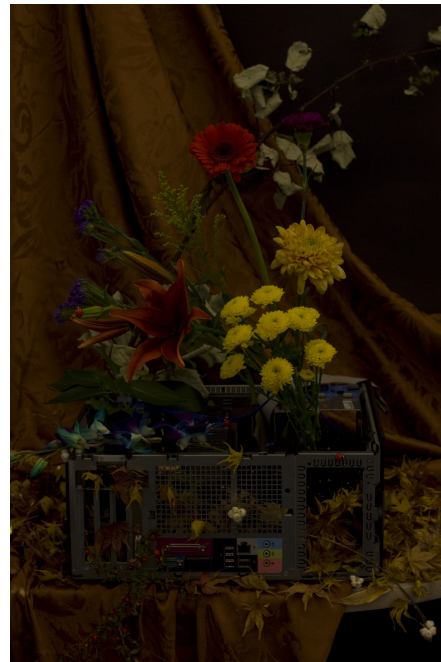
(a) Exposure with direct sunshine.



(b) Exposure with bright indoor studio lighting.



(c) Exposure with standard studio lighting.



(d) Exposure with standard studio lighting and reduced shutter time.

Figure 4.2: Four source photographs for composing the final image shown in Figure 4.1. Figure 4.2a and Figure 4.2b provides definition for the central objects, while the other two supplies pixels for the peripheral objects.

and Scott McFarland we discussed previously (Figure 2.5 and 3.2), in which multiple exposures towards the same scene with different lighting conditions are combined together to compose the final photograph.

In computer graphics, combining multiple photographs with different lighting conditions also has been explored in the topic of image space relighting and perception enhancement (Fattal et al. 2007; Mertens et al. 2009). When this method is applied for composition, the goal is not to enhance every detail in the source images but to selectively enhance or *suppress* details according to how a viewer's sight should be directed. From a viewpoint of image perception enhancement, Figure 4.2c might be closest to an optimal image among other source images in Figure 4.2, since the structural details in both background and foreground objects are clearly defined; and every object is illuminated with a moderate strength diffuse lighting without strong specular reflections or hard shadows. Precisely because of this, Figure 4.2c is not successful as an art project since it does not contain clear hints about how it should be seen by a viewer. (Unless the compositional goal is to intentionally overwhelm viewers with sharp details as in Jeff Wall's example (Figure 2.5), in that case the details need to be enhanced much further.)

Our compositional rule of light and shade in this art project is a straightforward correspondence between the importance of objects and their brightnesses in the image. The direct, strong sunshine can create spots of specular reflections and caustic lights, especially around the water drops on the central flowers (Figure 4.2a). These visual features are attractive, and therefore we use this photo to render those objects with which we want to immediately catch the viewer's attention, such as the red lily and the violet orchid. The blue and purple colour of the orchid distinctly stands out in a floral composition, in which most objects are in the red - yellow - green hue range. The direct sunshine also creates sharp and high-contrast shadows, which we excluded in the final image. Objects of the secondary importance are rendered with indoor lighting from several low power incandescence lights. These lights are set up at various angles, so they light up the scene without creating harsh shadows (Figure 4.2c). The background objects are rendered with the dark exposure shown in Figure 4.2d, with a smooth blending into the surrounding foreground objects. By using this very dark image, we can effectively hide details in the background which could be distracting. For example, the veins on the top-right are darkened to be barely visible, with blurry, undistinguishable details on its leaves.

In this art project, we approach the problem of surface brightness composition



Figure 4.3: The painted brightness mask for composing the final image in Figure 4.1. The magenta parts are taken from the exposure with direct sunshine (Figure 4.2a). Light yellow regions are from Figure 4.2b, blue part are from Figure 4.2c and the black parts are from Figure 4.2d.

in a manner similar to the interpolative material model in chapter 2. As discussed previously, light intensity and surface brightness are two distinct concepts, with the former corresponding to the physical light energy over a unit surface or solid angle, while the latter corresponds to the subjective visual perception of how much light is emitted or reflected from a surface. A camera is capable of capturing the light intensity, but image composition deals with regulating surface brightness to suggest a viewing sequence. The interpolative material model and this art project demonstrate similar approaches to transfer light intensity into compositionally based surface brightness. In these approaches, we first construct a sampling space for all possible surface brightness of the depicted scene. In the interpolative material model, this is achieved by the three input texture images, while in our art project it is achieved by taking photographs with different lighting conditions, from strong sunshine to dimmed studio lighting. In the second step, we transfer the intended composition into a mask of interpolation weight, and use this mask to define the final image. In chapter 2, the mask is constructed with the shading curve and the painting interface, and is stored in the barycentric per-face texture. In our art project, the mask is manually painted as shown in Figure 4.3.

To automate a compositional process like this, a first step is to identify the semantic information of every pixel on the image plane. We need to build methods not only capable of recognizing a pixel's corresponding object, but also capable of identifying how much attention this object should receive from a viewer. In the context of image space relighting, such an algorithm may require the incorporation of computer vision methods. If we render from the object space, the compositional information may be attached to the meshes. In either case, the most urgent future work is to develop a language that can describe compositional intentions about the synthesized image, such as the planned viewing sequence and every object's visual importance, and to insert this information into the rendering process (Durand 2002). To our knowledge no popular technical frameworks to solve this problem have been published. We shall continue to explore this topic of designing a machine-human interface for describing compositional rules in the future.

We printed the final image on a large (1 meter by 2 meters) piece of photographic paper and held an oral critique about this piece with instructors and students in the fine arts department. Our peer artists agreed that the post-processing does not break the realistic impression. Although they are aware that there is extensive digital editing behind this piece, the realistic impression holds quite well in a moderate viewing distance of 2 to 3 meters. If one closely examines the print pixel by pixel, some traces



Figure 4.4: Charles Sandison, *Chamber*, 2013, Denver Art Museum, Colorado. This is an example of light art installation created by customized rendering programs and digital data projectors.

of editing can be noticed by the participating artists, who are well trained in digital image editing. Part of the reason that the realistic impression can be retained is because the large format print forces one to step back in order to comfortably see the whole print. Such a ‘tabular’ form of presentation establishes a visual quality similar to painting for the printed digital photograph (Fried 2008). One notable visual effect is that the central red lily appears to come out of the image and into the three dimensional space in front of the print. This is partly because of the relatively high brightness of this flower compared to its surroundings. A visual effect similar to the historical ‘Trompe-l’œil’ technique is achieved with inconsistent lighting (Seckel 2004).

4.2 LIGHT ART PROJECTION INSTALLATION

In chapter 3, we showed our rendering method for automatically creating detailed light and shadow patterns according to an input sketch. The core device of this method is the projective light mask, which can be considered as a transparency texture on a plane orthogonal to the direction of lighting. This technique is inspired by the physical lighting regulating devices such as gobos or cucoloris (Figure 3.4). The concept of projective light mask can also be applied to digital data projectors. In common projectors, a high power light bulb is placed behind a screen whose pixels can change its colour according to digital input, and the output light will be filtered by this screen. Recent developments of powerful LED lights have introduced data



(a) The surrounding environment of our installation (Snider 1976).



(b) Light effects created by the installation in the night.



(c) The installation show attracted viewers to stop by.

Figure 4.5: Our light art installation uses the light pattern rendering method to create animated projections, which rendered the winter nights with cheerful lighting patterns.

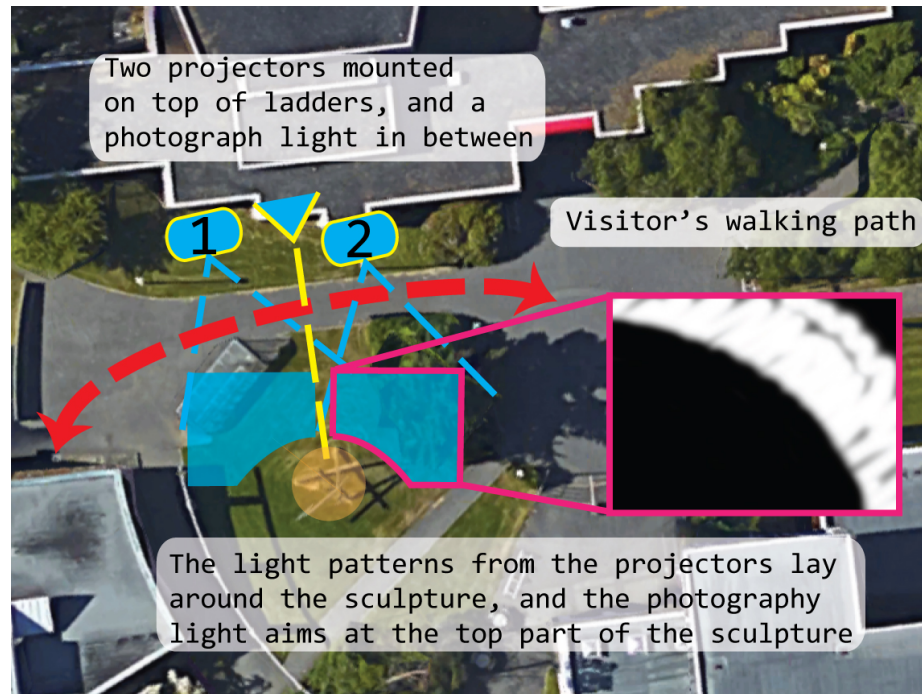
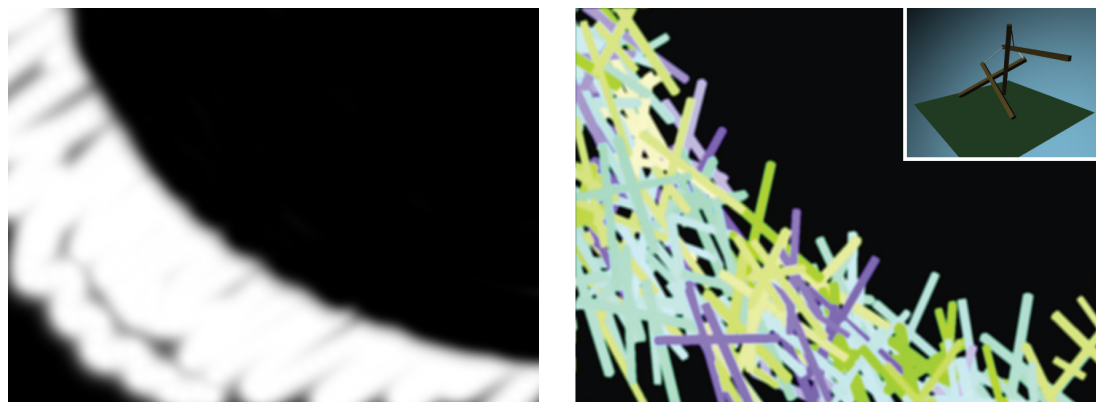


Figure 4.6: The implementation plan of our light art installation.

projectors with high luminosity and contrast ratio. Projectors are important devices in contemporary visual arts, and artists have developed large-scale installations based on these projectors (Figure 4.4). In this section, we introduce a light art installation project with our light pattern rendering method (Figure 4.5), and evaluate the method in a context of large scale, outdoor projection.

Our projection based installation responds to the environmental context of our campus. In northern latitudes on the Canadian west coast, winter brings long nights and much precipitation. The dark and humid climate sometimes creates a depressed emotion among the campus residents, who supported our proposal of a light-hearted projection installation. We planned to project light patterns on the ground surrounding an outdoor sculpture (Snider 1976). The sculpture stands on a sloped grassland on the side of a pedestrian path; and many campus residents routinely take this route to go to work or home (Figure 4.5a). In a misty winter night, we set up our projection and let the colourful, animated light patterns fill up the grassland around the sculpture (Figure 4.5b). During its exhibition, the cheerfully moving lights attracted many viewers to stop by (Figure 4.5c). We took this chance to communicate with the passing viewers and acquired their feedback on both the technical implementation and the aesthetic aspects of this piece.



(a) Guide image used for one of the projectors of the installation.

(b) One example frame in the synthesized video for one of the projectors. The elemental shape is shown on top right.

Figure 4.7: The guide image and an example video frame for one of the projectors.

The implementation plan of our installation is shown in Figure 4.6. The installation contains two data projectors on top of two ladders, and one photographic light on a tripod. The photographic light is gated such that it only illuminates the top half of the sculpture and leaves out the lower part, in order to create a stronger contrast of the projected light pattern on the ground. We use our rendering method in chapter 2 to generate the projected light patterns, and the guide image is shown in Figure 4.7a. This guide image indicates that the light pattern should fill the space around the central sculpture without covering its base. The elemental shape is a model of the sculpture, instead of models of tree leaves (Figure 4.7b). This elemental shape relates the light patterns to its surrounding space. Because we used two data projectors, two light pattern videos are needed with each covering a quarter of the circular space around the sculpture, such that the two projection patches will cover a half circle of ground around the sculpture closer to the pedestrian's path. The light patterns are rendered with a pre-defined palette, with every instance of the element shape taking a random colour from this palette, instead of the black-and-white rendering in chapter 2. We let the program directly output videos of the animated light patterns for the projectors.

During our discussion with the viewers, we validated several features of our rendering method. First, the light patterns follow the design sketch well, moving around without covering the sculpture in the centre of the space of the installation. Secondly, although the animation algorithm we designed was based on statistical



Figure 4.8: More photographs of our light art installation show.

models of natural winds and tree movements, the animation with abstract shapes also appears natural and coherent. We also discussed the aesthetic aspect of this art project with the viewers, and they acknowledged that the colourful, animated light patterns really stand out in the damp winter nights. The vibrant blue and green colours are rarely seen in a natural environment during the dreary winter. The large and bright light patterns in the dark night invite curious passengers to spend time to walk around and watch its movements on the grassland.

This installation project granted us a chance to observe the procedurally generated projective light mask in a large-scale, outdoor setting. The most apparent difference between a real life projection and a synthesized image is that the environment affects the projected light in a much more conspicuous way than in computer rendering. In our computer rendering experiments in chapter 2, the surface shading is mostly determined by the directional light, to which the projective light mask is attached. For this art project, we exhibited the installation at night, in order to create the maximum contrast with the data projectors. At the time of the show, the surrounding road lamps, the interior lights of the buildings and the city light reflected from the distant sky all affect the appearance of the projection (Figure 4.8). In a low light condition such as at night, a viewer will be very sensitive to any environment lighting and their effects on the projection. In addition, the humid and misty weather introduced a thin layer of water on the grass and the sculpture; and this thin layer of water produced strong specular reflections in a random way. Although the misty atmosphere created a preferable volumetric effect for the projected lights, the many spots of specular reflection were distracting. Technically speaking, these effects can be predicted and simulated with environment map and global illumination methods. From a visual arts perspective, a thorough investigation of the installation site is important for planning

and implementing outdoor projects.

Altering the appearance of reality with computer graphics techniques is a long-standing idea. In recent years, augmented reality (AR) technologies with wearable devices have acquired broad attention from the computer graphics research community. Besides wearable devices, projection based technology remains an important category of AR methods due to its non-invasive nature and the ability to create large sized installations. In order to fully exploit the potential of projection based technologies, we need to understand how to represent various lighting effects on a planar projective light mask. Besides approaching the previously mentioned goal of creating detailed lighting effects from a sketch, our research projects also contribute towards better applications of projection based media.

4.3 USER STUDY OF THE LIGHT PATTERN RENDERING METHOD

In the previous sections, we have discussed our experience with artists from the visual arts department. The term ‘artists’ we used was mainly referring to this group of research participants. On the other hand, there is a group of visual effects artists who work for the computer graphics industry; and whose education background and common practice are largely different from artists pursuing fine and visual arts. A visual arts program emphasizes interdisciplinary art practice in a contemporary visual arts context. Students in such a program are educated with a breadth of art techniques including drawing, painting, sculpture, photography and performance art. Critical writing and oral critique of each other’s work are important for students in such a program. A successful artist’s career consists of exhibitions of personal artwork in galleries. In contrast, a visual effects program trains its students to become professional practitioners of digital media production, such as employees in video game or visual effect companies. Students in a visual effects program receive extensive training in applied arts and computer graphics production, especially software tools for three-dimensional modelling, rendering and animation. They may also receive amateur courses in programming, such as Javascript and Adobe Flash based web media development. The goal of a visual effect program is to bring up CG artists who will have a successful career in the digital media industry.

Both communities of artists are important users of computer rendering software,

therefore it is important to evaluate and discuss our rendering research with them. In this section, we present a user study with instructors and undergraduate students in a visual effects program. The user study session consists of an introductory presentation, test-use of the program by the participants, and a questionnaire survey with open-ended discussions. Our communication with the user study participants yields important insights about our accomplished and future research. Below, we first describe the process of the user study, then discuss lessons learned from this experience.

Participant and Method

Participants of our user study came from a 4th year undergraduate animation and rendering class in a visual effects program. Students in this class are well-trained with common modelling and rendering tools, such as Maya, 3ds Max and Z-Brush. The user study session was conducted in their computer lab with 20 Windows 8 workstations that are equipped with high end graphics cards. This computer lab is dedicated to the computer rendering course in the program. There were 18 students who participated in our user study, 14 of whom filled out our questionnaire and joined us in the discussion. A summary of the participants's demographic background information is shown in Table 4.1. The questionnaire we used is shown in Figure 4.9.

Total number of participants : 14	
Gender	Female 78%(11) Male 22%(3)
Age Group	20-25 93%(13) 25-30 7%(1)
Previous knowledge of computer rendering: (Level 1 - 5)	Professional (Level 5): 7%(1) Very Confident (Level 4): 22%(3) Knowledgeable (Level 3): 57%(8) Not Confident (Level 2): 14%(2) Don't know much (Level 1): 0%(0)

Table 4.1: Background and demographic information of our user study participants.

Prior to the user study, we installed the light pattern rendering program in the computer lab. The user study session substituted one lecture of the students' regular rendering class, and took about 2 hours in total. The rendering course instructor audited our session. First, we introduced the rendering method by a presentation with

Participant No. _____

Participant Survey

This survey asks for your back ground information and your comments about our seminar today. As declared in the Participant Consent Form, this survey is anonymous, and your answer will only be identified by a participant number or in aggregated form. Your participation in this research is voluntary, and you can skip any question if you do not feel like to answer. Please tick appropriate options in the questions below.

Background Survey

Gender: Female Male Age Group: Below 20
 Male 20 - 25
 25 - 30
 Above 30

Previous knowledge about computer graphics rendering:
 I am very good at making 3D rendering with computers.
 I feel confident about making 3D rendering with computers.
 I can make some rendering with computers myself.
 I know something about rendering, but easily get confused.
 I know little about rendering and doesn't feel confident about it.

Your most familiar rendering software is:
 Maya 3ds Max Z-Brush Blender
 Other. Please specify:

Feedback and Comments

The following questions will be covered in our discussion. You can make a few notes below, but you don't need to write down a complete answer.

1. Is it straightforward or difficult to incorporate the presented lighting method with a standard rendering framework?
 Very Easy Easy Reasonable Difficult Very Difficult

2. How would you utilize the presented lighting method for visual effect production? For what kind of scenario would you consider it to be useful and why?

3. Taking the animation sequence we just made in the seminar, how do you compare the workflow with and without the presented rendering method?

Figure 4.9: The questionnaire for our user study.

live demo. We talked about the ways to incorporate the new lighting method with Autodesk Maya, and demonstrated how to create the foliage shadow effect in a step-by-step manner. To avoid evaluation bias, we addressed our method without identifying that the presenters are the authors of the introduced software. The presentation took about 20 minutes.

Then, the students were asked to make a simple foliage shadow animation with our program. As a starting point, the students were each provided with a set of pre-made scenes. The students could generate a sample animation by simply click-and-run the provided materials, without changing any parameters. Based on this sample, the students were free to make their own scene and the elemental shapes, to draw guide images, or to change any parameters in the program. As students in a rendering class, the participants have many three-dimensional scene models from their course assignments, which could be readily experimented with our rendering program.

We monitored the user study session, helped the participants for their rendering tasks, and took notes of the problems that the students encountered. We refrained from taking over the students' computers and performing the requested task for them. Instead, we observed how the students performed on the computers with a reasonable amount of verbal explanations. The rendering task took about an hour to complete. The rest of the time (about 40 minutes) was spent discussing the rendering experience with the participants. The classroom was split into two focus groups, with one containing 7 students and the other containing 8; such that everyone would have more chances to speak. The focus groups were run in a semi-structured way: we first went through a list of prepared questions, and then ask the students to freely comment on their experiences. We also helped the participating students fill out the questionnaire during this time. After the classroom session ended, we interviewed the rendering course instructor in a separate conversation.

Results

The first question on the questionnaire (Figure 4.9) asks whether it is easy to incorporate the presented method with a standard rendering tool. On the questionnaire, this question is supplied by a five-level scale choice question. The participants chose one option from “very easy”, “easy”, “reasonable”, “difficult” and “very difficult”. The participants then gave a short comment after the choice. 7%(1) of the participants answered “very easy”, 22%(3) answered “easy”, 57%(8) answered “reasonable” and

14%(2) answered “difficult”. The participants commented that the usability of our rendering method is affected by the fact that it is an external, stand-alone piece of software. The participants suggested that the program will be much more easier to use if it could be programmed as a plug-in of Maya, such that they can use the Maya painting tool to paint the guide image, and viewing the rendering result interactively in the Maya view port.

The second question asks about the possibility of utilizing this rendering tool for visual effects production. Besides the leaf shadows example, the participants are asked to think about another possible scenario in which our method will fit. The participants can choose to answer “no obvious applicable scenario”, if they cannot immediately name a such situation. For this question, the participants gave various answers from indoor lighting effects to underwater caustic lights. Two of the participants suggested that we should concentrate on still image composition rather than producing animation, because the harmonic movements driven by random excitation are difficult to control in a precise manner.

The third question asks the participants to evaluate the rendering work flow with and without our light pattern rendering tool. The participants commented that it is faster with our rendering tool to get an initial rendering result than without it. Meanwhile, fine tuning becomes difficult due to the automatic rendering process and the non-standard user interface of our program.

Discussion

In summary, the major concern raised by visual effects artists is the lack of a standard user interface to control the rendering result on a very detailed level. This concern was not apparent during our interaction with the artists from fine and visual arts. This is because the working environments for the two communities of artists are different in the following two aspects. First, for the purpose of making artworks for gallery exhibitions, an artist is free to explore any hardware or software tools and experiment with innovative ideas. Although artists with a fine arts background are not necessarily well-trained in computer rendering, they usually welcome experimental, non-standard tools, and are willing to spend some extra time to learn how to use them. On the other hand, visual effects artists in the CG industry work in a corporative, deadline-driven schedule. The productivity and reproducibility of the workflow is a primary concern for this group of artists. In this context, a successful application

of a new software tool depends on how quick a large group of working artists can learn to use it, and how well it can be integrated into existing tool-chains. Secondly, in fine and visual arts, an artist or a small group of artists will usually take the responsibility of the entire work flow of making a piece of art. The same group of artists are in charge of initially pitching the idea, making the implementation plan, gathering material and technical resources, and eventually constructing the work. In contrast, visual effects artists work in a highly specialized industry. The artist who is in charge of implementing a visual effect is usually not the same person who designs it. In fact, most visual effects companies work by contract with film companies; and in this situation the visual effects artists who implement the rendering may only communicate with the art director in another corporation with written documents. Under this circumstance, very detailed control of the rendering result is desirable. As long as the design of the user interface is aligned with software tools that the artists are familiar with, the complexity of the interface will not be considered obstructive.

When designing our light pattern rendering method in chapter 2, one of the goals was to take over the tedious detailing work from the artists as much as possible. With this guideline, we designed an algorithm that heavily relies on statistical models and random numbers, and only asks for a sketchy guide image as the input. Although this design is welcomed by artists in fine and visual arts, it is disapproved by visual effects artists working for the CG industry. For them, usually a third party judges the successfulness of a rendering result. Results based on statistical models and random numbers are likely to be unacceptable, unless there is an interface for the artists to manually touch up the rendering results to the very detail. In our context, this may require an interface to adjust the individual elemental shape's size, position, orientation and trajectories. On the other hand, if such a user interface is provided as the major part of the software tool, then the initial research goal of using automatic methods to free artists from tedious manual painting is defied.

* * *

We have evaluated our research through our own art practice and our experiences with artists from both visual arts and visual effects communities. These valuable experiences validate our research, identify limitations of our methods and implementations, and provide a direction for the future research of image composition in computer rendering. We summarize the future work as the following:

1. The most important future development is to build user interfaces to let an artist communicate compositional intentions with the rendering program effectively. The future interfaces may consist of painting interfaces, scripting languages and numerical parameters. A scripting language is suitable for attaching compositional information to models in the object space, while in the context of image space rendering a combination of computer vision methods and painting interfaces may be required. The future rendering methods should also be able to evaluate the synthesized image during the rendering process to see if the specified compositional goals are met, and be able to adjust the rendering process according to the evaluation results.
2. In addition, the future rendering program should be aware of the presentation conditions of the synthesized image. Facts of whether the image will be only displayed on a small sized mobile device, be printed on large format photographic papers or be projected onto a large surface should be considered during the rendering process.
3. Lastly, the future rendering program should have a standard user interface instead of an individually designed one, and can be implemented as a plug-in in mainstream rendering frameworks. The future rendering program should also have real-time, responsive painting interfaces for artists, instead of the optimization based method which may take a minute to generate a rendering result.

CHAPTER 5

CONCLUSION

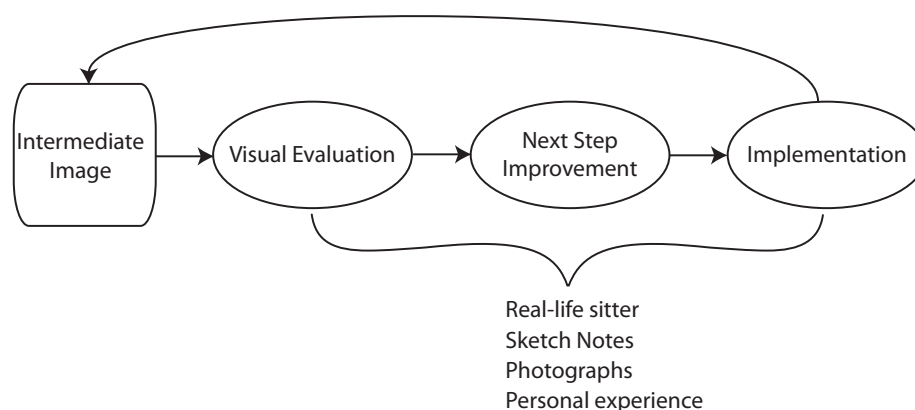


Figure 5.1: The iterative process of image composition.

On an abstract level, image composition can be considered as an iterative process as illustrated in Figure 5.1. Each iteration can be divided into several steps: first, an artist visually evaluates an intermediate, unfinished piece of work. Then, the artist may turn to real-world studio sitters, sketch notes and other resources as references to determine the next step of improvements. Next, the artist implements the improvements and starts over again to evaluate the image. Each of these steps provides ample opportunities for future research in computational composition beyond rendering and image synthesis. For example, the first step in which an artist observes and evaluates an existing image can be explored with computer vision and image processing methods. The next step of creative thinking may be approached with artificial intelligence and data mining technologies. After the desired changes are determined, an artist needs to implement these changes on the canvas with sufficient accuracy to create believable depictions. Our research concentrates on this last step of image composition. We assume that a design of light and shade for the rendering is given, and explore various methods to effectively adjust the rendering result to conform with the design.

The theoretical analysis we presented clarifies key aspects of image composition in the context of computer rendering. Theories and techniques about image composition mostly remain in the literature of art critique and art history, while our original analysis

interprets several important facets of this topic in the terms of visual perception and digital image synthesis. This theoretical analysis serves as explicit and effective guidelines for designing rendering algorithms that combine knowledge from perception and visual arts.

Based on this analysis, our research projects show novel rendering methods that facilitate composition. On the technical dimension of our research, there are two central topics we investigated throughout these projects: painting based user interface and compositional adjustment for a rendering result. We outline our goal, progress and future works on both topics as follows:

- First, a compositional intention must be effectively communicated between an artist and a rendering program before such an intention can be rendered. For user interface design, our primary goal is to let an artist directly edit light and shade in the image space, instead of turning back to the modelling interface and change parameters in the object space. We also want to avoid the ‘reverse rendering’ methodology in which a program interprets the image space input to object space parameters. Our experiments demonstrate the effectiveness of painting based interfaces, and show the possibility of directly constructing non-physical light and shade from image space inputs. Our interface research does not limit us to its conventional sense of creating a program with a graphical user interface. Much of our research deals with creating a novel *data structure* that can be painted in any software tool for the communication between an artist and the rendering program. To build better image space user interfaces to edit light and shade for three-dimensional rendering will be a long-term theme to explore in our research.
- Secondly, after the compositional intentions are communicated from the artist to the rendering program, the program needs to implement this intention by adjusting an existing rendering result. We demonstrated two key strategies for this goal. These strategies shall remain essential techniques for solving composition related problems in the future:
 - Example-based, interpolative shading methods are effective for various artistic shading. With this methodology, surface brightness can be initialized from object space lighting and be edited with a painting interface, while the colour information is taken from pre-defined examples, without calculating the material-light interaction.

- Optimization methods are suitable for solving problems that have approximate solutions, but no clear constructive structures. Image composition falls well into this category, because in many cases an artist can describe the intended visual effect to some extent, with a sketch or natural language, but cannot easily give a step by step solution to achieve it. Our light pattern rendering method shows how to solve such a problem with stochastic optimization.

Computer rendering synthesizes digital images from imaginary scenes, depicting non-existing objects and lighting effects. Like all other visual media, synthesizing digital images requires resources from the maker, such as the time and effort of the artist and the computational resource of the computers; and therefore such an endeavour must come with a purpose. In general, unlike the objective reality whose presence does not depend on the intention of human beings, the production of artificial media always relies on the subjective purpose of our minds: a story must be told, an emotional moment needs to be rendered, or simply a thrilling visual feature should appear at a specific time. On this larger picture, we consider computer rendering as a tool to achieve these purposes, not as a study of the simulation of physical light transport. Therefore, properties related to the end goal, such as the composition and the visual appearance of the resultant image, should be prioritized over the accuracy of the objective depiction of the virtual scenery.

REFERENCES

- AKLEMAN, E., LIU, S., AND HOUSE, D. 2016. Barycentric shaders: Art directed shading using control images. In *Proceedings of the Joint Symposium on Computational Aesthetics and Sketch Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, Expresive '16, 39–49.
- AKLEMAN, E. 1998. Implicit surface painting. In *Proc. of Implicit Surfaces 98*, vol. 2.
- ANGEL, E., AND SHREINER, D. 2011. *Interactive Computer Graphics: A Top-Down Approach with Shader-Based OpenGL, 6th Edition*. Addison-Wesley. Chapter 5 Lighting and Shading.
- ANJYO, K., AND HIRAMITSU, K. 2003. Stylized highlights for cartoon rendering and animation. *Computer Graphics and Applications, IEEE 23*, 4 (July-Aug.), 54 – 61.
- APODACA, A., AND LARRY, G. 1999. *Advanced RenderMan: Creating CGI for Motion Pictures*. Morgan Kaufmann.
- ASHIKHMIN, M., AND GOYAL, J. 2006. A reality check for tone-mapping operators. *ACM Trans. Appl. Percept.* 3, 4 (Oct.), 399–411.
- AUTODESK, 2014. FBX data exchange technology. The official introductory web page can be found at <http://www.autodesk.com/products/fbx/overview>, retrieved September, 2014.
- BARLA, P., THOLLOT, J., AND MARKOSIAN, L. 2006. X-toon: an extended toon shader. In *Proceedings of the 4th international symposium on Non-photorealistic animation and rendering*, ACM, New York, NY, USA, NPAR '06, 127–132.
- BARRATT, C., AND MILES, E. 2014. *Gilbert Stuart*. Metropolitan Museum of Art, New York, NY, USA.
- BAXTER, W., WENDT, J., AND LIN, M. C. 2004. Impasto: A realistic, interactive model for paint. In *Proceedings of the 3rd International Symposium on Non-*

- photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '04, 45–148.
- BEZDEK, J. C. 1981. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA.
- BHAT, P., ZITNICK, C. L., COHEN, M., AND CURLESS, B. 2010. Gradientshop: A gradient-domain optimization framework for image and video filtering. *ACM Trans. Graph.* 29, 2 (Apr.), 10:1–10:14.
- BIRN, J. 2006. *Digital Lighting and Rendering*. New Riders; 2nd edition, 36–37.
- BOX, G. E. P., AND MULLER, M. E. 1958. A note on the generation of random normal deviates. *The Annals of Mathematical Statistics* 29, 2 (06), 610–611.
- BOYADZHIEV, I., PARIS, S., AND BALA, K. 2013. User-assisted image compositing for photographic lighting. *ACM Trans. Graph.* 32, 4 (July), 36:1–36:12.
- BOYNTON, R. M. 1979. *Human Color Vision*. Holt, Rinehart and Winston.
- BRADLEY, D., NOWROUZEZHAI, D., AND BEARDSLEY, P. 2013. Image-based reconstruction and synthesis of dense foliage. *ACM Trans. Graph.* 32, 4 (July), 74:1–74:10.
- C. SEITZ, W. 1960. *Claude Monet*. Harry N.Abrams, Incorporated.
- CARLSON, R. E., AND FRITSCH, F. N. 1985. Monotone piecewise bicubic interpolation. *SIAM Journal on Numerical Analysis* 22, 2, pp. 386–400.
- CHASE, L. 2002. The not-so-innocent eye: Photorealism in context. In *Photorealism at the Millennium*, Harry N.Abrams, Inc, 11–22.
- CHU, N. S.-H., AND TAI, C.-L. 2005. Moxi: Real-time ink dispersion in absorbent paper. *ACM Trans. Graph.* 24, 3 (July), 504–511.
- CHUANG, Y.-Y., GOLDMAN, D. B., ZHENG, K. C., CURLESS, B., SALESIN, D. H., AND SZELISKI, R. 2005. Animating pictures with stochastic motion textures. In *ACM SIGGRAPH 2005 Papers*, ACM, New York, NY, USA, SIGGRAPH '05, 853–860.

2016. Cornell Box, Synthetic Image. Cornell University Program of Computer Graphics, <http://www.graphics.cornell.edu/online/box/data.html>, retrieved 2016.
- CORNSWEET, T. 1970. *Visual Perception*. Academic Press.
- CRAIG, R. R., AND KURDILA, A. J. 2006. *Fundamentals of structural dynamics*, 2nd ed. Wiley.com. Chapter.6 Numerical Evaluation of the Dynamic Response of SDOF Systems.
- DECARLO, D., AND SANTELLA, A. 2002. Stylization and abstraction of photographs. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, ACM, New York, NY, USA, SIGGRAPH '02, 769–776.
- DECARLO, D., FINKELSTEIN, A., RUSINKIEWICZ, S., AND SANTELLA, A. 2003. Suggestive contours for conveying shape. *ACM Trans. Graph.* 22, 3 (July), 848–855.
- DEVINE, S., 2003. Crew members on national treasure using a cookie. <http://en.wikipedia.org/wiki/Cucoloris>, retrieved May 2014.
- DURAND, F., AND DORSEY, J. 2002. Fast bilateral filtering for the display of high-dynamic-range images. *ACM Trans. Graph.* 21, 3 (July), 257–266.
- DURAND, F. 2002. An invitation to discuss computer depiction. In *Proceedings of the 2nd International Symposium on Non-photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '02, 111–124.
- DUTRE, P., BEKAERT, P., AND BALA, K. 2006. *Advanced global illumination, second edition*. CRC Press.
- E. JACOBS, D., GALLO, O., A. COOPER, E., PULLI, K., AND LEVOY, M. 2015. Simulating the visual experience of very bright and very dark scenes. *ACM Trans. Graph.* 34, 3 (May), 25:1–25:15.
- EDWARDS, S. 2006. *Photography: A Very Short Introduction*. Oxford University Press.
- FAIRCHILD, M. 2013. *Color Appearance Models*. The Wiley-IS&T Series in Imaging Science and Technology. Wiley.

- FATTAL, R., AGRAWALA, M., AND RUSINKIEWICZ, S. 2007. Multiscale shape and detail enhancement from multi-light image collections. In *ACM SIGGRAPH 2007 papers*, ACM, New York, NY, USA, SIGGRAPH '07.
- FISHER, M., RITCHIE, D., SAVVA, M., FUNKHOUSER, T., AND HANRAHAN, P. 2012. Example-based synthesis of 3d object arrangements. *ACM Trans. Graph.* 31, 6 (Nov.), 135:1–135:11.
- FREESE, E. I. 1930. *Perspective Projection, a Simple and Exact Method of Making Perspective Drawings*. The Pencil Points Press, Inc., New York.
- FRIED, M. 2008. *Why Photography Matters as Art as Never Before*. Yale University Press.
- GOLDSTEIN, N. 1989. *Design and composition*. Prentice Hall. "... composition refers to the total visual nature of a particular work's dynamic and depictive occurrences and to their resolution into a balanced and unified whole." (page 2).
- GOOCH, B., AND GOOCH, A. 2001. *Non-Photorealistic Rendering*. A. K. Peters, Ltd., Natick, MA, USA. Refer to Chapter 7 for a comprehensive review on feature edge algorithms.
- GOOCH, A. A., AND GOOCH, B. 2005. Enhancing perceived depth in images via artistic matting. In *Proceedings of the First Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, Computational Aesthetics'05, 83–89.
- GOOCH, A., GOOCH, B., SHIRLEY, P., AND COHEN, E. 1998. A non-photorealistic lighting model for automatic technical illustration. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, SIGGRAPH '98, 447–452.
- GOOCH, A. A., OLSEN, S. C., TUMBLIN, J., AND GOOCH, B. 2005. Color2gray: Saliency-preserving color removal. In *ACM SIGGRAPH 2005 Papers*, ACM, New York, NY, USA, SIGGRAPH '05, 634–639.
- GRIMM, C., AND KOWALSKI, M. 2007. Painting lighting and viewing effects. In *GRAPP*, INSTICC, 204–212.

- HABEL, R., KUSTERNIG, A., AND WIMMER, M. 2009. Physically guided animation of trees. In *Computer Graphics Forum*, vol. 28, Wiley Online Library, 523–532.
- HAEBERLI, P., AND AKELEY, K. 1990. The accumulation buffer: Hardware support for high-quality rendering. *SIGGRAPH Comput. Graph.* 24, 4 (Sept.), 309–318.
- HERTZMANN, A. 2010. Non-photorealistic rendering and the science of art. In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '10, 147–157.
- HILLIARD, C. 2002. Vermeer, master of light. Siggraph 2002 CAF, Electronic Theater.
- HOU, Q., QIN, H., LI, W., GUO, B., AND ZHOU, K. 2010. Micropolygon ray tracing with defocus and motion blur. *ACM Trans. Graph.* 29, 4 (July), 64:1–64:10.
- HURTUT, T., LANDES, P.-E., THOLLOT, J., GOUSSEAU, Y., DROUILLHET, R., AND COEURJOLLY, J.-F. 2009. Appearance-guided synthesis of element arrangements by example. In *Proceedings of the 7th International Symposium on Non-Photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '09, 51–60.
- JARABO, A., MASIA, B., BOUSSEAU, A., PELLACINI, F., AND GUTIERREZ, D. 2014. How do people edit light fields? *ACM Trans. Graph.* 33, 4 (July), 146:1–146:10.
- JI, L., GOOCH, A., GAMMON, L., AND WYVILL, B. 2015. Rendering artistic light patterns. In *Proceedings of the Workshop on Computational Aesthetics*, Eurographics Association, CAE '15, 1–13.
- JI, L., WYVILL, B., GAMMON, L., AND GOOCH, A. 2016. Composition and Perception beyond Photorealism. In *Computational Aesthetics*, The Eurographics Association, A. Forbes and L. Bartram, Eds.
- KALNINS, R. D., MARKOSIAN, L., MEIER, B. J., KOWALSKI, M. A., LEE, J. C., DAVIDSON, P. L., WEBB, M., HUGHES, J. F., AND FINKELSTEIN, A. 2002. Wysiwyg npr: drawing strokes directly on 3d models. *ACM Trans. Graph.* 21, 3 (July), 755–762.
- KANG, S. B., KAPOOR, A., AND LISCHINSKI, D. 2010. Personalization of image enhancement. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 1799–1806.

- KAPLAN, M., GOOCH, B., AND COHEN, E. 2000. Interactive artistic rendering. In *Proceedings of the 1st International Symposium on Non-photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '00, 67–74.
- KERR, W. B., AND PELLACINI, F. 2009. Toward evaluating lighting design interface paradigms for novice users. *ACM Trans. Graph.* 28, 3 (July), 26:1–26:9.
- KERR, W. B., PELLACINI, F., AND DENNING, J. D. 2010. Bendylights: Artistic control of direct illumination by curving light rays. In *Proceedings of the 21st Eurographics Conference on Rendering*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, EGSR'10, 1451–1459.
- KLEIN, A. W., LI, W., KAZHDAN, M. M., CORRÊA, W. T., FINKELSTEIN, A., AND FUNKHOUSER, T. A. 2000. Non-photorealistic virtual environments. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '00, 527–534.
- KRAWCZYK, G., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2005. Lightness perception in tone reproduction for high dynamic range images. *Computer Graphics Forum* 24, 3, 635–645.
- LAKE, A., MARSHALL, C., HARRIS, M., AND BLACKSTEIN, M. 2000. Stylized rendering techniques for scalable real-time 3d animation. In *Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, ACM, New York, NY, USA, NPAR '00, 13–20.
- LAURITZEN, A., 2010. Deferred rendering for current and future rendering pipelines. SIGGRAPH Course, *Beyond Programmable Shading*.
- LETZE, O. 2013. *Photorealism: 50 Years of Hyperrealistic Painting*. Hatje Cantz Verlag.
- LI, C., DEUSSEN, O., SONG, Y.-Z., WILLIS, P., AND HALL, P. 2011. Modeling and generating moving trees from video. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, ACM, New York, NY, USA, SA '11, 127:1–127:12.
- LIEDTKE, W. 2012. *Vermeer: The Complete Paintings*. Harry N. Abrams.

- LISCHINSKI, D., FARBMAN, Z., UYTTENDAELE, M., AND SZELISKI, R. 2006. Interactive local adjustment of tonal values. In *ACM SIGGRAPH 2006 Papers*, ACM, New York, NY, USA, SIGGRAPH '06, 646–653.
- LIVINGSTONE, M. S. 2015. *Vision and Art (Updated and Expanded Edition)*. Harry N. Abrams.
- LIVNY, Y., PIRK, S., CHENG, Z., YAN, F., DEUSSEN, O., COHEN-OR, D., AND CHEN, B. 2011. Texture-lobes for tree modelling. In *ACM SIGGRAPH 2011 Papers*, ACM, New York, NY, USA, SIGGRAPH '11, 53:1–53:10.
- LOPEZ-MORENO, J., SUNDSTEDT, V., SANGORRIN, F., AND GUTIERREZ, D. 2010. Measuring the perception of light inconsistencies. In *Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization*, ACM, New York, NY, USA, APGV '10, 25–32.
- LU, C., XU, L., AND JIA, J. 2012. Combining sketch and tone for pencil drawing production. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, NPAR '12, 65–73.
- LU, J., BARNES, C., DIVERDI, S., AND FINKELSTEIN, A. 2013. Realbrush: Painting with examples of physical media. *ACM Trans. Graph.* 32, 4 (July), 117:1–117:12.
- MA, C., WEI, L.-Y., AND TONG, X. 2011. Discrete element textures. *ACM Trans. Graph.* 30, 4 (July), 62:1–62:10.
- MANTIUK, R., KRAWCZYK, G., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2004. Perception-motivated high dynamic range video encoding. *ACM Trans. Graph.* 23, 3 (Aug.), 733–741.
- MAX, N. 1991. Unified sun and sky illumination for shadows under trees. *CVGIP: Graphical Models and Image Processing* 53, 3, 223 – 230.
- MEIER, B. J. 1996. Painterly rendering for animation. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ACM, New York, NY, USA, SIGGRAPH '96, 477–484.
- MEISEL, L. K. 2002. *Photorealism at the Millennium*. Harry N. Abrams, Inc.

- MERAJ, Z., WYVILL, B., ISENBERG, T., GOOCH, A. A., AND GUY, R. 2008. Mimicking hand-drawn pencil lines. In *Computational Aesthetics*, Citeseer, 73–80.
- MERTENS, T., KAUTZ, J., AND VAN REETH, F. 2009. Exposure fusion: A simple and practical alternative to high dynamic range photography. *Computer Graphics Forum* 28, 1, 161–171.
- MOREHEAD, J. C. 1952. *A Handbook of Perspective Drawing*. Elsevier Press, Inc., Houston, Texas.
- MOULD, D., AND GRANT, K. 2008. Stylized black and white images from photographs. In *Proceedings of the 6th International Symposium on Non-photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '08, 49–58.
- NING, X., LAGA, H., SAITO, S., AND NAKAJIMA, M. 2011. Extended papers from npar 2010: Contour-driven sumi-e rendering of real photos. *Comput. Graph.* 35, 1 (Feb.), 122–134.
- NORTHAM, L., ASENTE, P., AND KAPLAN, C. S. 2012. Consistent stylization and painterly rendering of stereoscopic 3d images. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, NPAR '12, 47–56.
- NOWROUZEZHAI, D., JOHNSON, J., SELLE, A., LACEWELL, D., KASCHALK, M., AND JAROSZ, W. 2011. A programmable system for artistic volumetric lighting. *ACM Trans. Graph.* 30, 4 (Aug.), 29:1–29:8.
- OLSEN, L., SAMAVATI, F. F., SOUSA, M. C., AND JORGE, J. A. 2009. Sketch-based modeling: A survey. *Comput. Graph.* 33, 1 (Feb.), 85–103.
- OPPENHEIM, A., WILLSKY, A., AND YOUNG, I. 1983. *Signals and systems*. Prentice-Hall signal processing series. Prentice-Hall, 2nd Edition.
- OTA, S., TAMURA, M., FUJITA, K., FUJIMOTO, T., MURAOKA, K., AND CHIBA, N. 2003. 1/fb noise-based real-time animation of trees swaying in wind fields. In *Computer Graphics International, 2003. Proceedings*, IEEE, 52–59.
- PALUBICKI, W., HOREL, K., LONGAY, S., RUNIONS, A., LANE, B., MĚCH, R., AND PRUSINKIEWICZ, P. 2009. Self-organizing tree models for image synthesis. *ACM Trans. Graph.* 28, 3 (July), 58:1–58:10.

- PATTANAİK, S. N., TUMBLIN, J., YEE, H., AND GREENBERG, D. P. 2000. Time-dependent visual adaptation for fast realistic image display. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '00, 47–54.
- PEITGEN, H.-O., AND SAUPE, D., Eds. 1988. *The Science of Fractal Images*. Springer-Verlag New York, Inc., New York, NY, USA. See Chapter 1.2.3 for an introduction to the $\frac{1}{f^\beta}$ noise.
- PELLACINI, F., TOLE, P., AND GREENBERG, D. P. 2002. A user interface for interactive cinematic shadow design. *ACM Trans. Graph.* 21, 3 (July), 563–566.
- PELLACINI, F., BATTAGLIA, F., MORLEY, R. K., AND FINKELSTEIN, A. 2007. Lighting with paint. *ACM Trans. Graph.* 26, 2 (June).
- PELLACINI, F. 2010. envylight: An interface for editing natural illumination. In *ACM SIGGRAPH 2010 Papers*, ACM, New York, NY, USA, SIGGRAPH '10, 34:1–34:8.
- PIRENNE, M. H. 1970. *Optics, painting & photography*. Cambridge University Press.
- PIRK, S., STAVA, O., KRATT, J., SAID, M. A. M., NEUBERT, B., MĚCH, R., BENES, B., AND DEUSSEN, O. 2012. Plastic trees: Interactive self-adapting botanical tree models. *ACM Trans. Graph.* 31, 4 (July), 50:1–50:10.
- PRAUN, E., HOPPE, H., WEBB, M., AND FINKELSTEIN, A. 2001. Real-time hatching. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, ACM, New York, NY, USA, SIGGRAPH '01, 581–.
- PRUSINKIEWICZ, P., LINDENMAYER, A., HANAN, J. S., FRACCHIA, F. D., FOWLER, D. R., DE BOER, M. J., AND MERCER, L. 1990. *The Algorithmic Beauty of Plants*, vol. 2. Springer-Verlag New York.
- PUSCH, R., SAMAVATI, F., NASRI, A., AND WYVILL, B. 2007. Improving the sketch-based interface. *The Visual Computer* 23, 9-11, 955–962.
- REINHARD, E., STARK, M., SHIRLEY, P., AND FERWERDA, J. 2002. Photographic tone reproduction for digital images. In *Proceedings of the 29th Annual Conference*

- on Computer Graphics and Interactive Techniques*, ACM, New York, NY, USA, SIGGRAPH '02, 267–276.
- REINHARD, E., KHAN, E. A., AKYUZ, A. O., AND JOHNSON, G. 2008. *Color imaging: fundamentals and applications*. A K Peters, Ltd.
- RITSCHEL, T., AND EISEMANN, E. 2012. A computational model of afterimages. *Computer Graphics Forum* 31, 2pt3, 529–534.
- RITSCHEL, T., OKABE, M., THORMÄHLEN, T., AND SEIDEL, H.-P. 2009. Interactive reflection editing. In *ACM SIGGRAPH Asia 2009 papers*, ACM, New York, NY, USA, SIGGRAPH Asia '09, 129:1–129:7.
- RITSCHEL, T., THORMÄHLEN, T., DACHSBACHER, C., KAUTZ, J., AND SEIDEL, H.-P. 2010. Interactive on-surface signal deformation. *ACM Trans. Graph.* 29, 4 (July), 36:1–36:8.
- SAITO, M., AND MATSUMOTO, M. 2008. Simd-oriented fast mersenne twister: a 128-bit pseudorandom number generator. In *Monte Carlo and Quasi-Monte Carlo Methods 2006*. Springer, 607–622.
- SAITO, T., AND TAKAHASHI, T. 1990. Comprehensible rendering of 3-d shapes. *SIGGRAPH Comput. Graph.* 24, 4 (Sept.), 197–206.
- SALESIN, D. 2002. The seven grand challenges for npr. In *Keynote at the 2nd International Symposium on Non-Photorealistic Animation and Rendering, NPAR*, vol. 2002.
- SALISBURY, M. P., WONG, M. T., HUGHES, J. F., AND SALESIN, D. H. 1997. Orientable textures for image-based pen-and-ink illustration. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '97, 401–406.
- SANDER, J. 2008. *The Magic of Things: still-life painting, 1500-1800*. Ostfildern, Germany.
- SCHMID, J., SENN, M. S., GROSS, M., AND SUMNER, R. W. 2011. Overcoat: An implicit canvas for 3d painting. *ACM Trans. Graph.* 30, 4 (July), 28:1–28:10.

- SCHOENEMAN, C., DORSEY, J., SMITS, B., ARVO, J., AND GREENBERG, D. 1993. Painting with light. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, SIGGRAPH '93, 143–146.
- SECKEL, A. 2004. *Escher, Dalí & the Artists of Optical Illusion*. Sterling Publishing Co., Inc., New York.
- SEEDS, M., AND BACKMAN, D. 2010. *Foundations of Astronomy*. Nelson Education.
- SHAN, Q., MEYER, M., DEROSE, T., AND ANDERSON, J. 2012. Tone mapping high dynamic range videos using wavelets. *Pixar Technical Memo*.
- SHIRLEY, P., AND MARSCHNER, S. 2009. *Fundamentals of Computer Graphics, Third Edition*. A. K. Peters, Ltd, Natick, MA.
- SHIRLEY, P. 2003. *Realistic Ray Tracing*. A K Peters; Second Edition, Natick, MA.
- SNIDER, G., 1976. Rough Trade. Douglas fir, steel, 15'x15'x18'.
- SOLER, C., AND SILLION, F. X. 1998. Fast calculation of soft shadow textures using convolution. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, ACM, New York, NY, USA, SIGGRAPH '98, 321–332.
- SONG, Y.-Z., ROSIN, P. L., HALL, P. M., AND COLLOMOSSE, J. 2008. Arty shapes. In *Proceedings of the Fourth Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, Computational Aesthetics'08, 65–72.
- SPATE, V. 1992. *Claude Monet: Life and Work, with over 300 illustrations, 135 in colour*. Thames and Hudson Ltd.
- STEADMAN, P. 2001. *Vermeer's Camera: Uncovering the Truth Behind the Masterpieces*. Oxford University Press.
- STROTHOTTE, T., AND STEFAN, S. 2002. *Non-Photorealistic Computer Graphics: Modeling, Rendering, and Animation*. Morgan Kaufmann.

- TAN, P., ZENG, G., WANG, J., KANG, S. B., AND QUAN, L. 2007. Image-based tree modeling. In *ACM SIGGRAPH 2007 Papers*, ACM, New York, NY, USA, SIGGRAPH '07.
- TAN, M., LALONDE, J.-F., SHARAN, L., RUSHMEIER, H., AND O'SULLIVAN, C. 2015. The perception of lighting inconsistencies in composite outdoor scenes. *ACM Trans. Appl. Percept.* 12, 4 (Sept.), 18:1–18:18.
- TAYLOR, P. 1995. *Dutch Flower Painting, 1600-1720*. Yale University Press.
- TODO, H., ANJYO, K.-I., BAXTER, W., AND IGARASHI, T. 2007. Locally controllable stylized shading. *ACM Trans. Graph.* 26, 3 (July).
- VAN HAEVRE, W., DI FIORE, F., AND VAN REETH, F. 2006. Physically-based driven tree animations. In *Proceedings of the Second Eurographics Conference on Natural Phenomena*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, NPH'06, 75–82.
- VAN HOOGST RATEN, S. 1678. *Inleyding tot de hooge schoole der schilderkonst: anders de zichtbaere werelt (Introduction to the High School of Painting, or the Visible World)*. “Want een volmakte Schildery is als een spiegel van de Natuer, die de dingen, die niet en zijn, doet schijnen te zijn, en op een geoorlofde vermakelijke en prijslijke wijze bedriegt.” (A perfect painting is like a mirror of Nature, in which things that are not there appear to be there, and which deceives in an acceptable, amusing, and praiseworthy fashion).
- WALD, I., WOOP, S., BENTHIN, C., JOHNSON, G. S., AND ERNST, M. 2014. Embree: A kernel framework for efficient cpu ray tracing. *ACM Trans. Graph.* 33, 4 (July), 143:1–143:8.
- WANDELL, B. A. 1995. *Foundations of Vision*. Sinauer Associates Inc.
- WEBB, M., PRAUN, E., FINKELSTEIN, A., AND HOPPE, H. 2002. Fine tone control in hardware hatching. In *Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering*, ACM, New York, NY, USA, NPAR '02, 53–ff.
- WHEELOCK, A. K. 1995. *Vermeer & the Art of Painting*. Yale University Press, New Heaven and London.

- WINKENBACH, G., AND SALESIN, D. H. 1994. Computer-generated pen-and-ink illustration. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, ACM, New York, NY, USA, SIGGRAPH '94, 91–100.
- WINNEMÖLLER, H. 2011. Xdog: Advanced image stylization with extended difference-of-gaussians. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering*, ACM, New York, NY, USA, NPAR '11, 147–156.
- YUAN, L., AND SUN, J. 2012. *Computer Vision – ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part IV*. Springer Berlin Heidelberg, Berlin, Heidelberg, ch. Automatic Exposure Correction of Consumer Photographs, 771–785.
- YUKSEL, C., KEYSER, J., AND HOUSE, D. H. 2010. Mesh colors. *ACM Trans. Graph.* 29 (April), 15:1–15:11.