TONE MAPPING FOR HIGH DYNAMIC RANGE IMAGES

THÈSE N^O 3588 (2006)

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Laurence MEYLAN

Ingénieure informaticienne diplômée EPF

de nationalité suisse et originaire de Le Chenit (VD)

EPFL

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Lausanne, EPFL

2006

Contents

Abstract

Tone mapping is an essential step for the reproduction of "nice looking" images. It provides the mapping between the luminances of the original scene to the output device's display values. When the dynamic range of the captured scene is smaller or larger than that of the display device, tone mapping expands or compresses the luminance ratios.

We address the problem of tone mapping high dynamic range (HDR) images to standard displays (CRT, LCD) and to HDR displays. With standard displays, the dynamic range of the captured HDR scene must be compressed significantly, which can induce a loss of contrast resulting in a loss of detail visibility. Local tone mapping operators can be used in addition to the global compression to increase the local contrast and thus improve detail visibility, but this tends to create artifacts. We developed a local tone mapping method that solves the problems generally encountered by local tone mapping algorithms. Namely, it does not create halo artifacts, nor graying-out of low contrast areas, and provides good color rendition. We then investigated specifically the rendition of color and confirmed that local tone mapping algorithms must be applied to the luminance channel only. We showed that the correlation between luminance and chrominance plays a role in the appearance of the final image but a perfect decorrelation is not necessary.

Recently developed HDR monitors enable the display of HDR images with hardly any compression of their dynamic range. The arrival of these displays on the market create the need for new tone mapping algorithms. In particular, legacy images that were mapped to SDR displays must be re-rendered to HDR displays, taking best advantage of the increase in dynamic range. This operation can be seen as the reverse of the tone mapping to SDR. We propose a piecewise linear tone scale function that enhances the brightness of specular highlights so that the sensation of naturalness is improved. Our tone scale algorithm is based on the segmentation of the image into its diffuse and specular components as well as on the range of display luminance that is allocated to the specular component and the diffuse component, respectively. We performed a psychovisual experiment to validate the benefit of our tone scale. The results showed that, with HDR displays, allocating more luminance range to the specular component than what was allocated in the image rendered to SDR displays provides more natural looking images.

Keywords: high dynamic range images, tone mapping, Retinex, HDR displays

Résumé

La reproduction des tons est une étape essentielle pour l'affichage d'images qui plaisent à l'utilisateur. Cette étape fait la correspondance entre les luminances de la scène originale et les valeurs affichées à l'écran. Par exemple, si le rang dynamique de la scène est supérieur ou inférieur à la capacité de l'écran, un algorithme de reproduction des tons devra compresser ou étendre le rang dynamique de manière à rendre l'image compatible avec l'écran.

Dans cette thèse, nous adressons spécifiquement le problème de reproduction des scènes à grand rang dynamique (appelées "scènes HDR") pour l'affichage sur écrans standards (LCD, CRT) ainsi que sur de nouveaux écrans à très grand rang dynamique appelés "écrans HDR".

Lors de l'affichage sur un écran standard, le rang dynamique d'une scène HDR doit être compressé de manière significative, ce qui peut provoquer une perte de contraste et un manque de visibilité dans les détails. Des opérateurs locaux peuvent être utilisés en plus de la compression globale pour améliorer le contraste local ainsi que la visibilité des détails, mais les méthodes existantes génèrent souvent des artefacts visuels. Notre algorithme de reproduction des tons agit locallement et propose des solutions aux problèmes de halos, de zones grisâtres ou du mauvais rendu de la couleur. Suite à cela, nous avons étudié spécifiquement le traitement de la couleur. Notre recherche confirme que les opérateurs locaux doivent être appliqués seulement sur la luminance. La corrélation entre la luminance et la chrominance joue un rôle dans le rendu final de la couleur mais une décorrélation parfaite n'est pas nécessaire.

Le récent développement d'écrans HDR rend possible l'affichage d'images contenant de très grands contrastes sans presque aucune compression du rang dynamique. L'arrivée de tels écrans sur le marché crée un besoin pour de nouveaux algorithmes de reproduction des tons. En particulier, les images traitées dans le but d'être affichées sur des écrans standards doivent être retraitées pour être affichées sur un écran HDR en tirant le plus grand bénéfice de l'augmentation du rang dynamique de l'écran. Nous proposons l'utilisation d'une fonction linéaire par morceaux dont la forme permet d'augmenter la clarté des spécularités de l'image, améliorant ainsi la sensation de naturel que provoque sa visualisation. Notre algorithme est basé sur une segmentation de l'image en ses composants diffus et spéculaires ainsi que sur le pourcentage de rang dynamique de l'écran réservé à la composante spéculaire. Le bénéfice de notre fonction linéaire par morceaux est validé par une expérience psycho-visuelle. Les résultats de cette expérience montrent qu'avec un écran HDR, allouer une plus grande partie du rang dynamique à la composante spéculaire que celle allouée lors de l'affichage sur un écran standard conduit à des images plus agréables à regarder et plus naturelles.

Mots-clés: Images à grands rang dynamique, reproduction des tons, écrans HDR

Acknowledgments

During the course of my PhD, I had the chance to work with many different persons who all contributed to the outcome of this thesis. First, I would like to thank Wulfram Gerstner who took me as a PhD student and then gave me the freedom to work in collaboration with the IVRG group under the supervision of Sabine Süsstrunk. Then, I would like to thank Sabine Süsstrunk, for welcoming me in her lab, for her continuous help and support, and for her efforts in reading through the thesis many times at the end. I am also grateful to her to have given me the opportunity to work at Sharp Laboratories of America, which was an excellent professional experience.

I would like to thank Scott Daly who supervised my work at Sharp and partly continued his supervision when I was back at EPFL. Many thanks for accepting to be on my thesis committee with such a flexible schedule. Thanks also to all the people at Sharp for some good time in the North-West and for the fruitful discussions on HDR imaging.

Of course, these four years would not have been so enjoyable without all my colleagues and friends at EPFL. I would like to thank them all for the numerous coffee and lunch breaks, for the excellent work atmosphere, and for their help in all domains. Special thanks to the IVRG group, Patrick, Joanna, Roberto, Daniel and RK for all the inspiring interactions.

Last but not least, I would like to thank my family and friends who always encouraged me in pursuing my studies.

Frequently Used Terms, Abbreviations, and Notation

Terms and abbreviations

Brightness: Attribute of a visual sensation according to which an area appears to emit more or less light (CIE 45-25-210).

Contrast ratio: See "dynamic range".

CRT: Cathode Ray Tube.

Chrominance: The two components of an image that contain the color information.

Display black point: Neutral color with the lowest luminance that can be produced by the display (*ISO 22028-1 2004*).

Display white point: Neutral color with the highest luminance that can be produced by the display (*ISO 22028-1 2004*).

Dynamic range: The ratio between the brightest and the darkest object in a scene.

Floating point map: Image built with a multiple exposure technique (also called radiance map).

HDR: High Dynamic Range.

HDR image: An image that represents an HDR scene.

HDR scene: A scene whose dynamic range exceeds the dynamic range of the output medium.

High key image: Image containing mainly bright elements.

HVS: Human Visual System.

Image black point: Pixel value corresponding to the perceived black in a scene.

Image white point: Pixel value corresponding to the perceived white in a scene.

Key: Average log luminance of a scene.

LCD: Liquid Crystal Display.

LED: Light Emitting Diodes.

LDR: Low Dynamic Range.

LDR image: An image that represents an LDR scene.

LDR scene: A scene whose dynamic range is lower than the dynamic range of the output medium.

Low key image: Image containing mainly dark elements.

Luminance: Physical measure of scene radiances, given in candela per meter square (cd/m^2).

Luminance (in a luminance/chrominance representation): Achromatic information in an image.

Lightness: Visual impression of brightness.

Middle key image: Image composed mainly of midtones.

MSR: Multi-Scale Retinex.

MSRCR: Multi-Scale Retinex with Color Restoration.

PCA: Principal Component Analysis.

Radiance map: Image constructed using a multiple exposure technique.

Re-rendering: Mapping of an image tone-mapped to one output medium (e.g. an SDR display) to a different output medium (*ISO 22028-1 2004*).

Scene: Real environment that becomes the topic of an image (Reinhard *et al.* 2005).

SDR: Standard Dynamic Range.

SDR image: An image that represents an SDR scene.

SDR scene: A scene whose dynamic range is approximately the same as the dynamic range of the output medium.

Specular highlight: Reflection of a light source on a non-lambertian surface, such as metal or glossy paint.

Notation and variables

	Matrices, images and binary maps;
$\mathcal{B}$	Binary map;
C_1	Chrominance channel 1;
C_2	Chrominance channel 2;
E	Surround suppressed edge image;
I	Linear image encoded with RGB primaries;
I'	Non-linear image encoded with RGB primaries;
I_{new}	Image encoded with RGB primaries, after processing;
I_{HDR}	Radiance map constructed with the ghost removal algorithm;
$\mathcal{R}$	Image computed with a Retinex operator;
U	Magnitude image;
Z	Image compensated to a reference exposure time;
Δ	Difference image for motion detection;
λ	LED image;
Λ	Luminance image, linear with scene radiances;
Λ'	Luminance image, non-linear;
Λ_{new}	Luminance image, after processing;
Υ	Inhibition term;
$\{,,\}$	Representation of a color image under the form of a vector of three color channels;
$\{R, G, B\}$	RGB image, alternative notation for I;
$\{\Lambda, C_1, C_2\}$	Luminance/chrominance representation of an image.
	Fixed constants;
J	Number of images in a set;
K	Number of paths in an image;
N	Number of pixels in an image;
Q	Number of observations in the psychovisual experiment;
T	Number of tone scale functions tested by the psychovisual experiment;
	Color transforms and filters;
$\mathcal{F}_m$	Two-dimensional averaging filter of size m;

$\mathcal{G}_\sigma$	Two-dimensional Gaussian filter of spatial constant σ;
$\mathcal{H}$	Filter for morphological operators;
M	Color transform;
	Other variables, functions, symbols and operators;
$\overline{\bullet}$	Average operator;
$*$	Convolution operator, matrix multiplication;
$\bullet_{\downarrow n}$	Downsampling by n;
$\bullet_{\uparrow n}$	Upsampling by n;
$\bullet_{lp}$	Low-passed version of an image;
$O(\bullet)$	Order of complexity;
a_1, a_2	Variables for cross-talk correction;
b	Black point of an image;
c	Color channel of an RGB image;
d	Physical distance between two LEDs;
e	Black level error for monitors;
f	Tone mapping function;
h	Histogram of an image;
H	Cumulative histogram of an image;
i, j, k, m, n, o	Indexes, variable numbers;
L	Measured luminance produced at the screen, given in cd/m^2;
$mask(x, y)$	Weighted average of pixels surrounding pixel (x,y);
(x, y)	Coordinate of a pixel;
p	Pixel in I;
P	Set of pixels in an image;
q	Score in a pair comparison experiment;
r	Radius;
s	Slope;
$\mathcal{S}$	Set of images;
$\mathcal{T}$	Threshold operator;
t_1, t_2	Threshold for detecting specular highlights;
u	Coefficient of agreement;
v	Vector;
V	Input voltage controlling a monitor;
w	White point of an image;
α	Factor for inhibition term;

β	Sigmoidal weighting factor;
δ	Shift value in Retinex;
ϵ	Small number;
η	Measure of consistency;
γ	Exponent for gamma correction;
Γ	Path in an image;
ι	Intermediary variable to count circular triads;
κ	Factor for saturation enhancement;
μ	Measure of correlation;
ν	Number of circular triads;
Ψ_{max}	Maximum display luminance given in cd/m^2;
ρ	Percentage of display luminance allocated to maximum diffuse white;
σ	Spatial constant of a Gaussian function;
τ	Exposure time;
θ	Angle;
ω	Maximum diffuse white;
Ω	Retinex reset operator;

Chapter 1

Introduction

1.1 Tone Mapping

The fundamental goal of image reproduction is to display images that correspond to the visual impression an observer had when watching the original scene. Tone mapping is a major component of image reproduction. It provides the mapping between the light emitted by the original scene and display values.

For good image reproduction, it is necessary to take into account the way the human visual system (HVS) processes light information. Scene radiances are captured by the rods and the cones in the retina and passed further to the visual pathway. These signals are processed non-linearly by several layers of neurons. They form an image that we call *percept*, which does not correspond to the physical radiances of the scene.

Natural scenes can have a wide range of illumination conditions, ranging from night scenes to outdoor scenes. Examples of scenes and corresponding maximum luminances are given in Figure 1.1 (Johnson 2005). For us to "see" all these different scenes and form a percept where details are visible, our visual system adapts to the illumination conditions in different ways (Webster 1996, Pattanaik *et al.* 1998). A first adaptation to global illumination takes place at the pupil, which changes its diameter depending on the amount of light entering the eye. Second, the photoreceptors (cones) adapt their sensitivities to the mean luminance in the field of view, given by total retinal illumination. Third, local adaptation modulates local contrasts as our gaze visually scans the scene.

Capturing devices produce an image that is linear with respect to scene radiances (after flare correction, if necessary). Thus, one role of tone mapping is to process the image captured by the camera to simulate the processing of the HVS and make its representation more perceptually meaningful (Figure 1.2).

Figure 1.1: Maximum luminance values for various scenes. This Figure was taken from (Johnson 2005).
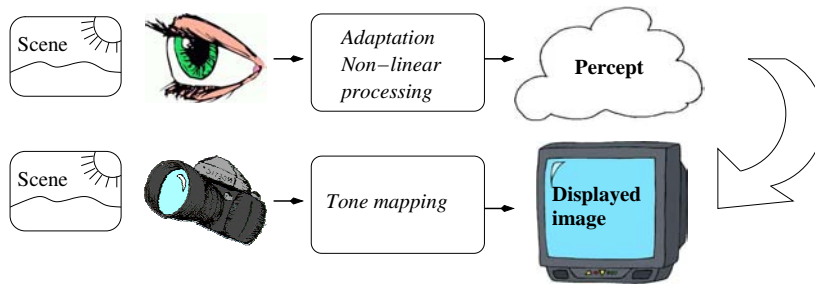
Figure 1.2: The HVS processes the scene radiances in a non-linear manner through different adaptation processes. It forms a percept where all details are visible. Electronic devices capture the scene radiances linearly. A tone mapping operator is necessary to non-linearly encode the image as well as to map it to the display characteristics so that the displayed image corresponds to our memory of the original scene.

The top image of Figure 1.3 shows the image captured by the camera (after demosaicing) and the bottom image shows the corresponding image "seen" by an observer. The image captured by the camera looks too dark. The image perceived by our visual system has more contrast in the shadows.

1.1.1   Dynamic Range of Scenes and Displays

A second role of tone mapping is to match the dynamic range of the scene to that of the display device. The *dynamic range* (or *contrast ratio*) is the luminance ratio between the brightest and the darkest object in a scene. Luminances are the physical measure of scene radiances, given in candela per meter square (cd/m^2). When the dynamic range of the captured scene is smaller or larger than that of the display device, tone mapping expands or compresses the luminance ratios.

Figure 1.3: Top: Image that is captured by the camera (after demosaicing). **Bottom:** Image that is perceived by our visual system.

Scenes

A scene is said to be low dynamic range (LDR) when its dynamic range is lower than that of the output medium. In this case, the dynamic range of the input image has to be expanded to fit the output medium dynamic range. A scene whose dynamic range corresponds approximatively to that of the output medium is called standard dynamic range (SDR). A high dynamic range (HDR) image is the representation of an HDR scene whose dynamic range exceeds by far that of the output medium. With HDR scenes, it is likely that some parts of their representation are clipped by the capturing process[1]. Typical examples of HDR scenes are sunny outdoor scenes or a view of an indoor room with objects visible outside the window.

The dynamic ranges of several natural scenes were measured and 1:160 was

---

[1]The generation of images representing HDR scenes is addressed in Appendix A.

found to be an average contrast ratio (Hunt 1995). Outdoors scenes usually have a larger dynamic range, which can reach a contrast ratio of three orders of magnitude (1:1000) or more. Scenes with fog tend to have a small contrast ratio.

Displays

Currently used display technologies are Cathode Ray Tube (CRT) or Liquid Crystal Displays (LCD). We will call these *standard displays* and define their dynamic range as being standard as opposed to HDR displays which have a much larger dynamic range. The contrast ratio of standard displays is generally 1:100 but new LCD monitors can reach 1:400. Prints have a contrast ratio of 1:50-1:500, depending on the printing technology. Newly developed HDR displays have a contrast ratio reaching 1:25'000, depending on the viewing conditions.

1.2 Global and Local Tone Mapping

Tone mapping methods can either be global (also called spatially invariant) or combined with a local processing (also called spatially variant), modeling either only the global adaptation, or the global and local adaptation of the HVS. Global tone mapping algorithms apply the same function to all pixels of the image, i.e. one input value results in one and only one output value. They can be a power function, a logarithm, a sigmoid, or a function that is image-dependent (see Chapter 2). Local tone mapping algorithms apply different functions for different spatial pixel positions. In this case, one input value can result in more than one output value depending on the pixel position and on surrounding pixel values. A third class of tone mapping algorithms, not treated here, are time-dependent (Pattanaik *et al.* 2000). They consider adaptation over time for the mapping of video sequences.

In general, global tone mapping algorithms are fast and local tone mapping methods are computationally more expensive. With global methods, look-up tables can process the images even faster, which makes them suitable for in-camera and/or video processing.

Global tone mapping methods are suitable for scenes whose dynamic range corresponds approximately to that of the display device, or is lower. When the dynamic range of a scene exceeds by far that of the display (HDR scene), global tone mapping methods compress the tonal range too much, which results in a perceived loss of contrast and detail visibility. A local processing is thus necessary in addition to global compression for the reproduction to be visually

appealing. Local processing allows to increase local contrast, which improves the detail visibility of some parts of the image while the global compression scales the image's dynamic range to the output device's dynamic range.

Figure 1.4 shows an example of an HDR scene that requires local processing when displayed on an SDR monitor. The left image was rendered using just a global tone mapping and the right image was rendered using our tone mapping method described in Chapter 3 that combines global compression and local processing. We observe that our method retrieves details in the shadowed part of the scene, which remains too dark with just a global tone mapping.
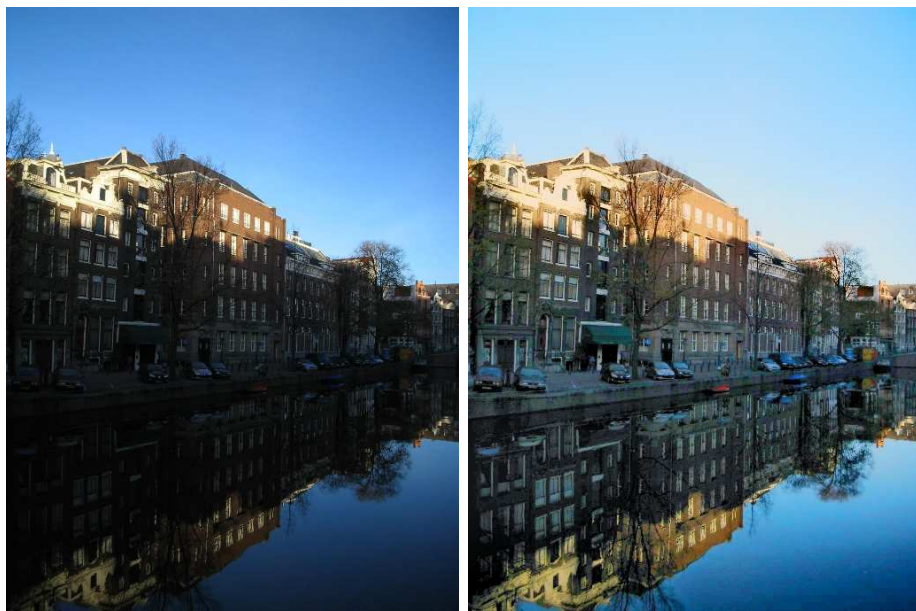
Figure 1.4: Example of an HDR scene that requires local processing. **Left:** Image rendered using a global tone mapping. **Right:** Image rendered with the Retinex-based adaptive filter method described in Chapter 3 that combines global compression and local processing.

HDR monitors whose dynamic range is close to that encountered in the real world have just started to come on the market. These new monitors make it possible to display HDR images without compressing the dynamic range much. Thus, new tone mapping methods must be developed for the rendering of LDR, SDR and HDR scenes to these displays. In addition, since a significant number of images are already tone-mapped to SDR displays, it is necessary to develop tone mapping algorithms that re-render theses images to HDR displays. This conversion can be seen as the inverse operation of tone mapping.

1.3   Tone Mapping and Color Images

The color in a displayed image can be represented by three numbers, usually controlling colors red (R), green (G), and blue (B). A color image is thus referred to as an RGB image, where each R, G, and B component is called a color channel. Tone mapping can be applied to the three color channels independently by performing the same operation three times. This is commonly used for global tone mapping and provides good color rendition.

However, with local tone mapping, treating the R, G, B color channels independently may lead to bad color rendition (see Chapter 4). Another way of applying a tone mapping algorithm to a color image is to transform the RGB image into a *luminance/chrominance* representation where the achromatic and chromatic information are separated, and to process the achromatic channel only. The luminance/chrominance image is composed of one luminance channel containing the achromatic information and two chrominance channels containing the color information. The chrominance channels are often encoded with opponent colors (i.e. red-green and yellow-blue). In this context, the word luminance is different from the physical quantity that we introduced earlier. It refers to the achromatic content of the image and can represent the perceived values if a non-linearity has been applied. Because it is often called "luminance" in the literature, we will keep the same term for both definitions while asking the reader to remain attentive to that.

The color transform from an RGB image $\{R, G, B\}$ to a luminance/chrominance representation $\{\Lambda, C_1, C_2\}$ can be performed by a linear transform as follows:

$$\{\Lambda, C_1, C_2\} = M * \{R, G, B\}, \tag{1.1}$$

where M is a 3×3 matrix, Λ is the luminance channel, and C_1, C_2 are the chrominance channels.

In general, these transforms are applied to non-linear RGB. In this thesis, we use color transforms to decorrelate the color channels and process the luminance only. We apply them on linear RGB.

Several color transforms have been standardized for different purposes. For example, YUV, YIQ, YCrCb (Poynton 2003) are used in the television broadcasting system and compression. CIE Lab, CIE Luv are used in colorimetry to define perceptually uniform color spaces for color difference evaluation (*CIE 15:2004 2004).*

1.4 Contributions and Overview

In this thesis, we focus on the reproduction of HDR scenes, i.e. scenes with very large luminance ratios. Such scenes are problematic when shown on SDR monitors since their dynamic range exceeds by far that of the display. Local tone mapping can be used to increase the local contrast and thus reduce the loss of detail visibility caused by the global compression. However, these algorithms often introduce artifacts such as halos or graying-out of low contrast black and white areas. Also, the rendition of color appears to be an issue to most existing methods.

We propose a local tone mapping method based on the Retinex model of color vision (Section 2.5) that renders HDR images to SDR displays while solving the problems mentioned above. As halos are generally created along high contrast edges, we use an adaptive filter that prevents areas separated by a high contrast edge from influencing each other. Thus, the increase in local contrast is performed only within an area of similar illumination. The benefit of our adaptive filter is illustrated by an example where it helps removing halos. The high contrast edges are detected using a Canny algorithm with fixed thresholds. As an alternative, we propose the use of a boundary detector that takes into account surround information. The local treatment is then weighted by a sigmoidal function to conserve black and white low contrast areas. This processing is applied only to the luminance channel to avoid color artifacts. The application of our algorithm to HDR images shows that it efficiently increases local contrast while preventing artifacts and providing good color rendition.

We then study the issue of color reproduction with local tone mapping algorithms. We apply different local methods with different color transforms, which convert an RGB image into a luminance/chrominance representation. Our research shows that the correlation between luminance and chrominance channels plays a role in the color rendition of the final image. However, a perfect decorrelation is not always necessary.

While working on the rendering of HDR images, we create our own image database. These images are either RAW images captured with a Canon camera or radiance maps created with a multiple exposure technique, which enables the capture of very high dynamic range scenes (Appendix A). The drawback of this technique is that it is limited to still scenes. We propose an extension of the multiple exposure technique to allow the capture of scenes with moving objects (Appendix B).

With the development of HDR monitors, a global compression combined or not with local contrast enhancement is no longer necessary. Nevertheless, it raises new questions about how to take advantage of this increase in dynamic

range when re-rendering images that were tone-mapped for standard displays to HDR monitors. We address this issue for images representing HDR scenes, which are often clipped or compressed by the capturing and rendering process. We develop a tone scale function,[2] whose goal is to recreate the brightness of specular highlights (i.e. the reflections of light sources on metal or glossy surfaces). This enhances the sense of naturalness of the re-rendered images when displayed on an HDR monitor. We validate the benefit of our tone scale with a psychovisual experiment. The results of the experiment show that with HDR displays, allocating more display range to the specular parts of an image than what was allocated when rendered to SDR is preferred over applying a linear scaling.

1.4.1 Overview

Chapter 2 reviews global and local tone mapping methods. We start by describing basic global algorithms and continue with more sophisticated ones that take inspiration from traditional photography techniques. Then, we present the Retinex theory of color vision, which inspired several local tone mapping techniques. We review local algorithms that are derived from Retinex followed by other significant tone mapping methods.

Chapter 3 presents the Retinex-based adaptive filter method that we developed to render HDR images to SDR displays. We first present the treatment of the luminance channel that includes a global pre-processing followed by a local processing based on Retinex. The local processing uses an edge map computed using a Canny edge detector. We propose an improved edge detector that uses surround suppression to differentiate texture edges from object boundaries. We integrate it in our local processing instead of the Canny edge detector and compare the results. We then discuss computational complexity and describe a way to reduce it by applying the processing on a downsampled image. Finally, we explain the processing of the chrominance. We show the results of our algorithm applied to HDR images and compare them with images rendered by state of the art methods.

In **Chapter 4**, we study the influence of the choice of color space for local tone mapping methods. We apply two local tone mapping algorithms to the luminance channel given by four different color transforms. We introduce a measure to evaluate how well the color transforms decorrelate luminance and chrominance information and relate it to visual results.

Chapter 5 proposes a tone scale function to convert images that were rendered to standard displays into images rendered to HDR displays, with

---

[2]The term *tone scale* is used as a synonym of tone mapping.

the goal of recreating the brightness of specular highlights. We describe the psychovisual experiment that was conducted to validate the use of such a tone scale. We provide a statistical analysis of the collected data and discuss the results.

Chapter 6 presents an automatic way of computing the tone scale previously used in the psychovisual experiment. We propose a masking technique to apply it locally and show the results on images.

Appendix A describes the generation of floating point images called radiance maps, using a multiple exposure technique. **Appendix B** proposes a ghost removal algorithm that extends this technique for the capture of HDR scenes containing a moving object. **Appendix C** explains the processing applied to an image so that what is seen at the screen corresponds to the input image.

Chapter 2

Global and Local Tone Mapping

2.1 Introduction

Global tone mapping algorithms range from basic functions (logarithm, power function, sigmoid) to more sophisticated ones that are image dependent. Their goals can be to approximate the HVS non-linearity, to compensate for the display characteristics, or to render visually more appealing images.

In Section 2.2, we describe "simple" tone mapping functions. Image dependent algorithms whose aim is to obtain visually pleasing images are presented in Section 2.3. Then, Section 2.4 provides a method to scale the image black and white points to the black and white points of the display. This scaling is included in all global tone mapping algorithms and can be performed in many ways. We present one technique that is subsequently used in Chapter 3.

Local tone mapping operators can be used in addition to global tone mapping to improve the local contrast. Many local tone mapping algorithms take inspiration from a theory of color vision called Retinex. The goal of Retinex is to recover the perceived colors from the captured scene radiances. In Section 2.5, we present the Retinex theory and the experiments that lead to it. Section 2.6 describes tone mapping operators derived from Retinex followed by other local tone mapping operators not based on it (Section 2.7). Finally, Section 2.8 reviews psychovisual experiments conducted for the evaluation of existing tone mapping algorithms.

2.2 Simple Global Tone Mapping

2.2.1 Logarithm

A logarithm function is often used to approximate the non-linear encoding of the HVS. Thus, in the log-encoded image, equal steps in log-luminances correspond to equal visual sensations. This enables a perceptually uniform quantization where the perceived difference between two digital code values remains constant over the digital code value range. Such a logarithm function is used in the Retinex model of color vision that is described in Section 2.5 of this chapter.

2.2.2 Power of $\frac{1}{3}$

A logarithm function can be approximated by a power law with exponent $\frac{1}{3}$ in a given interval $[1, 100]$. These two functions are illustrated in Figure 2.1. The power of $\frac{1}{3}$ function is used in perceptually uniform color spaces such as CIE Luv and CIE Lab (*CIE 15:2004 2004*) to model the HVS non-uniformity.

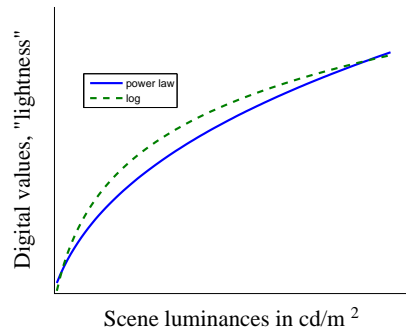**Figure 2.1:** Approximation of the luminances non-linear encoding performed by the HVS.

2.2.3 Gamma

Display devices have a non-linear relationship between input voltage and display luminance. This non-linearity is described by a power law and is commonly called *gamma*, referring to the numerical value of the exponent. The output of a monitor can be modeled as follows:

$$L = V^{\gamma}, \tag{2.1}$$

where V is the input voltage, γ is the gamma value of the display and L is the luminance produced at the screen.

This non-linearity has to be inverted in order to display luminances that correspond to those of the captured scene. To do so, each color channel of an input image I is processed as follows:

$$I'_c = I_c^{\frac{1}{\gamma}}, \qquad (2.2)$$

where c denotes one of the R, G, B color channel of the input image I, and I' is the gamma corrected image. The γ value depends on the monitor; a common average value is 2.2.

In addition to compensating for the display non-linearity, an advantage of the gamma encoding is that it approaches the functions described above that model the HVS non-linearity. Thus, a gamma-encoded image is also approximately perceptually uniform.

Gamma Correction for Real Monitors

In practice, a pure power function is not sufficient to model real monitor characteristics. Indeed, it forces zero voltage to map to zero luminance and can not accommodate errors around zero voltage called *black level errors*. Black level errors offset the curve upward or downward as illustrated in Figure 2.2.
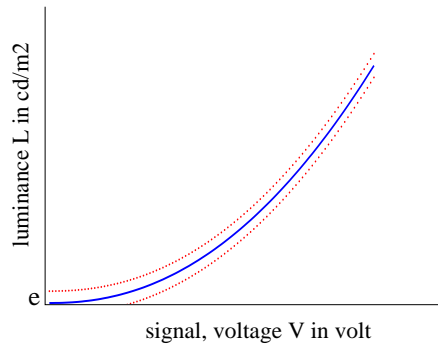

Figure 2.2: Example of black level error (dotted red curves) versus theoretical model (solid blue curve).

A better model is obtained by using an offset to accommodate for black level errors (Poynton 2003):

$$L = (V + e)^{\gamma}, \qquad (2.3)$$

where e models the black level error.

Gamma Correction with Adaptation to the Image Key

In addition to compensating for the output medium non-linearity, we may also want to improve the reproduction of an image depending on its content. An image can be characterized by its dominant tones, which is called the *key*. The key of an image indicates whether it is subjectively light, normal or dark. Tumblin and Rushmeier (1993), Reinhard *et al.* (2002), and Kuang *et al.* (2004) approximated it by the log average luminance $\overline{\Lambda}$:

$$\overline{\Lambda} = \exp(\frac{1}{N} \sum_{p \in I} \log(\epsilon + \Lambda(p))), \qquad (2.4)$$

where p is a pixel in the image I whose luminance channel is given by Λ, N is the number of pixels, and ϵ is a small value to avoid singularities caused by the presence of black pixels.

A good indication of an image key is given by the histogram of the luminance channel. Figure 2.3 shows the histograms for a low key, normal key and high key image (from top to bottom) as well as the corresponding images. For the low key image, the histogram peaks toward the left end, representing the dim tones. The histogram peak moves to the right as the key becomes higher, i.e. the image becomes globally brighter.

The gamma exponent value may be adapted to the key of the image to render more pleasing images. When rendering a low key image, it is desirable to carry out gamma correction with a greater gamma value to improve detail visibility in dim areas. Figure 2.4 shows an example of a photograph with $\gamma = 1$ (top left), $\gamma = 2.2$ (top right) and $\gamma = 3$ (bottom). We observe that with $\gamma = 1$ and $\gamma = 2.2$, the center of the image appears too dark. $\gamma = 3$ provides better detail visibility.

In Chapter 3, we apply a power function with an adaptive exponent value in addition to a logarithm for the pre-processing of very low key images. Other methods (Holm 1996, Reinhard *et al.* 2002, Ward *et al.* 1997) also adapt the shape of the tone mapping function to the key of the image. They are reviewed in the next section.

2.3 Sophisticated Global Tone Mapping

Many global tone mapping methods take inspiration from the traditional photography technique called *zone system* (Adams 1971*a*, Adams 1971*b*, Adams 1972, White *et al.* 1977). The principle of the zone system is to define zones

in a scene using physical measures and match them to output display zones. Traditionally, the zones are indicated with roman letters with I indicating a black zone containing detail and IX indicating a white zone also still containing detail. A crucial part of the zone system is to predict how the mapping from scene to display zones will happen. Traditionally, a middle-gray surface is mapped to zone V, the darkest object in the scene is mapped to zone I, and the brightest object to zone IX.

Holm (1996) defines an s-shaped (sigmoidal) tone reproduction function. The parameters are defined according to the zone characteristics of the input scene (maximum luminance, mean luminance and minimum luminance) and the output medium's dynamic range. This method works well for SDR images but is not always sufficient for HDR images as it may compress the local contrast too much.

Braun and Fairchild (1999) also use a sigmoidal function, but apply it for gamut mapping tasks. The sigmoid parameters are tuned according to the input image statistics, such as mean and standard deviation.

The method of Reinhard *et al.* (2002) first maps the log average luminance (2.4) to a display luminance value defined by the image key. If the image is normal key, it is matched to the middle-gray of the displayed image. If the scene is low key or high key, the log average is mapped to a lower or higher value, respectively. As many scenes contain more information in the midtones than in low and high luminance areas, s-shaped curves compress these areas. Instead of an s-shaped curves, Reinhard *et al.* use a function that only compresses the highlights. They also introduce a clipping since it is not always desirable to bring all luminances within display range. A local processing based on automatic dodging and burning photographic technique is added when an increase in local contrast is necessary.

The histogram adjustment method of Ward *et al.* (1997) is not directly inspired by the zone system but is based on similar principles. It is an extension of histogram equalization, which redistributes pixel values so that the treated image histogram has a uniform distribution. In the histogram adjustment method, the histogram equalization is performed on the log luminance image that was first low-passed, so that each pixel corresponds to an area of 1° of visual field. The goal of histogram adjustment is to conserve the perceived contrast while preventing details that were not visible to the human eye in the original scene to become visible in the image after processing. In short, just noticeable difference in the real world is mapped to just noticeable difference on the display device. A model of glare is also used in the post-processing stage to further improve the displayed image realism.

(a)

Low key

dim   normal   bright

(b)

Normal key

dim   normal   bright

(c)

High key

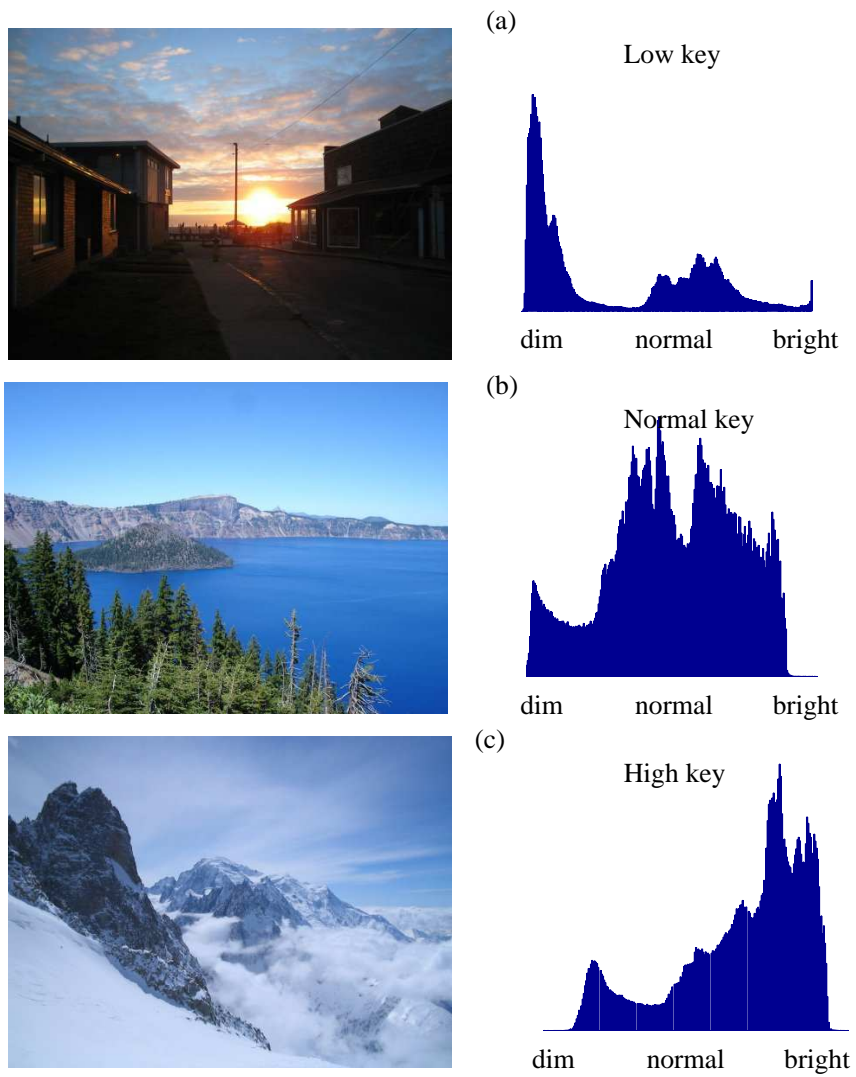dim   normal   bright

Figure 2.3: Histograms for (a) low key, (b) normal key, (c) high key image (from top to bottom).
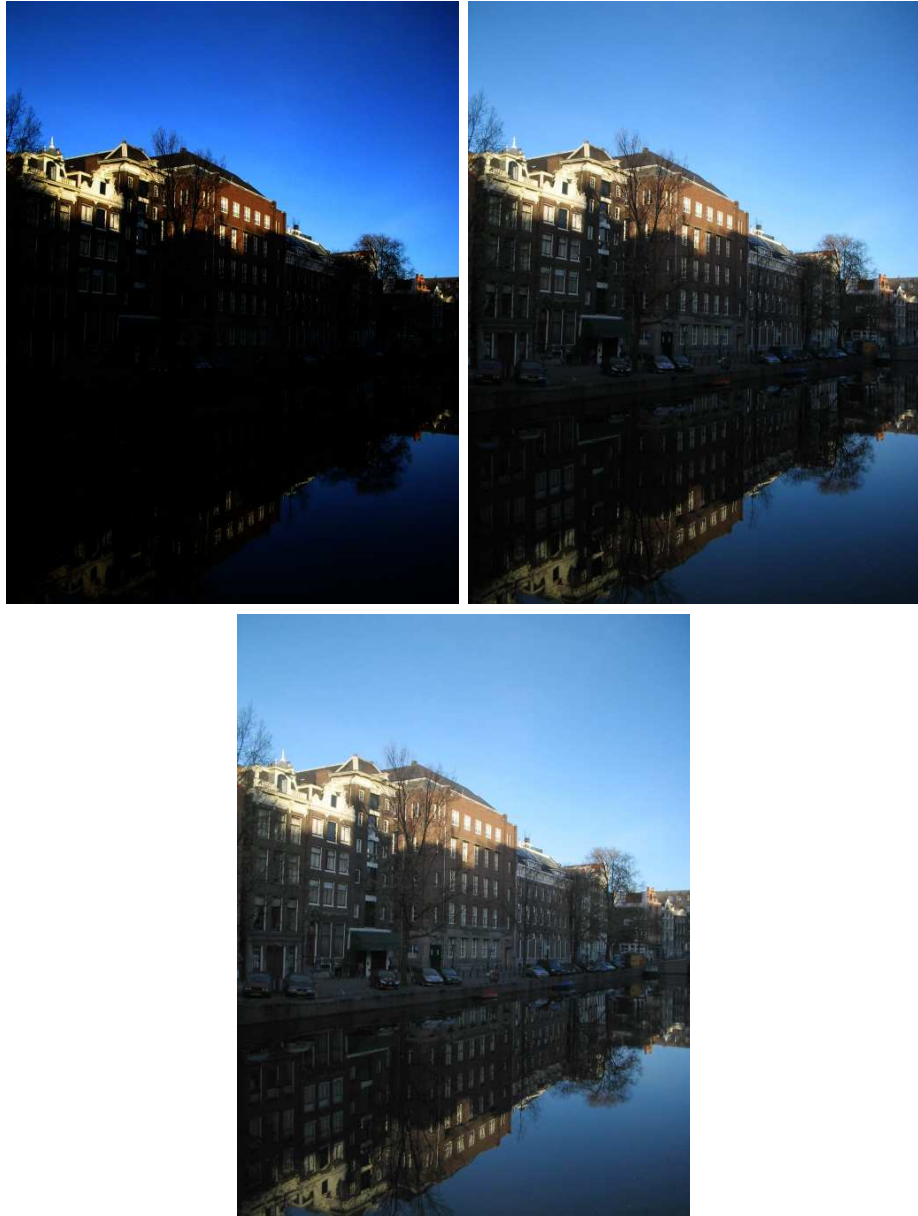
Figure 2.4: Example of gamma correction with different $\frac{1}{\gamma}$ exponent values. **Top left:** $\gamma = 1$. **Top right:** $\gamma = 2.2$. **Bottom:** $\gamma = 3$

2.4 Black and White Point Correction

Tone mapping always includes an operation that matches the perceived black (image black point) to the darkest display luminance and the perceived white (image white point) to the brightest display luminance. This operation is called black and white point correction. It is critical for a pleasing reproduction of images.

Black and white point correction can be automatically included in the tone mapping algorithm such as with histogram equalization method, or it can be performed as a pre- or post-processing in addition to the tone mapping operator. Here, we present the black and white point correction method that we implemented as a post-processing of our tone mapping operator based on a histogram scaling.

The black and white points of the input image can not simply be determined by the darkest and brightest pixel. Indeed, one pixel can be an outlier due to noise and not represent significantly the image content. Group of pixels of low and high digital values must be used. One solution to define the black and white points that significantly represent the image content is to use a histogram-based method on the luminance image. The luminance image is obtained by applying a color transform on the RGB image as described in Section 1.3.

A histogram is a mapping that counts the number of pixels whose value falls in an given interval of the image encoding range. Each interval is called a *bin*.

Let N be the number of pixels in the image and n the number of bins. The histogram $h(i)$ corresponds to the number of pixels in bin i and meets the following condition:

$$N = \sum_{i=1}^{n} h(i), \tag{2.5}$$

where i is the index over the bins.

A cumulative histogram H is a mapping that counts the cumulative number of pixels in all the bins up to to the specified bin j.

$$H(j) = \sum_{j'=1}^{j} h(j') \tag{2.6}$$

H is a monotonically increasing function. The black and white points b and w are defined by the pixel value associated to the bin number in which 1% and 99% of the image data points fall, respectively (illustrated in Figure 2.5).

The input image I is then linearly scaled by matching b and w to normalized code values (2.7). I_{new} is encoded in the range [0,1]. The pixel values below

b and above w are clipped to 0 or 1. The scaling is applied similarly to each color channel c of I.

$$I_{new,c}(p) = min(1, \frac{\max(0, I_c(p) - b)}{w - b}) \tag{2.7}$$

Figure 2.5 shows an example of black and white point correction. The image on the left is the original badly scaled image. We can see in the corresponding histogram (bottom left) that only a few pixels have very low or high digital values. The black and white correction defines new minimum and maximum values and redistributes the pixel values over the image range. The resulting image and histogram are shown in the right images of Figure 2.5.
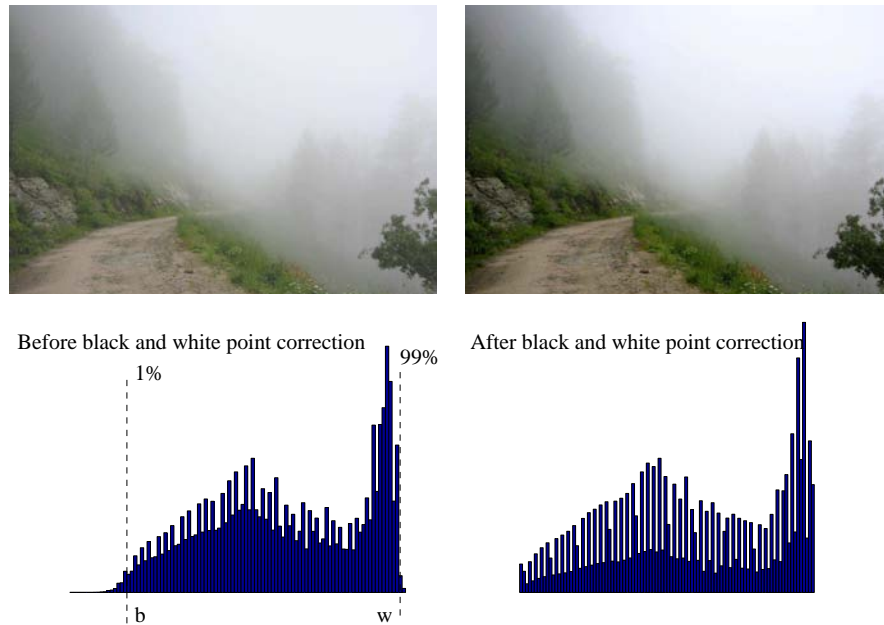
Figure 2.5: Example of black and white point correction using a histogram-based method. **Top left:** Original badly scaled image. **Bottom left:** Badly scaled histogram. The dark and bright extremities of the histogram form long tails. **Top right:** Image after black and white point correction. **Bottom right:** Corrected histogram.

2.5 The Retinex Theory of Color Vision

The Retinex theory, developed by Edwin Land, intends to explain how the visual system extracts reliable information from the scene despite changes of illumination (Land 1964, Land and McCann 1971, Land 1977). It is based

on a series of experiments carried out with one or three projectors and a flat surface made of color patches called a *Mondrian*[1]. This theory inspired the development of several tone mapping algorithms presented later in this chapter.

2.5.1 Experiment with Black and White Mondrian

In a first experiment, a black and white Mondrian was illuminated uniformly with a projector. The illumination falling on the gray patches was measured by a light meter. Then, a filter was inserted in front of the projector so that the illumination varied smoothly and monotonically over the vertical direction of the Mondrian. The measured energy coming from the surface was changed dramatically. Despite this change of illumination, the observer reported that the perceived gray levels remained the same. Land varied the illumination in different ways. In some cases, the black could have the same energy than the white in a previous scene but the perceived gray level did not change. An example of black and white Mondrian illuminated non-uniformly is given in Figure 2.6.
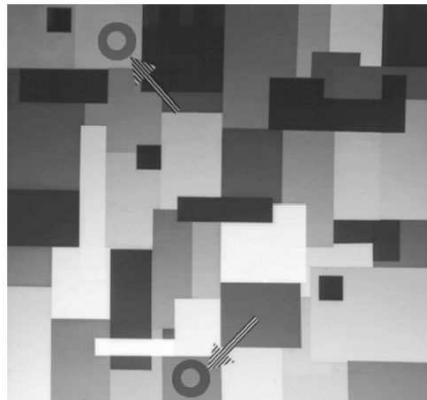
Figure 2.6: Black and white Mondrian illuminated non-uniformly. The bottom is brighter than the top but appears similar to an observer. This figure was taken from (Land and McCann 1971).

2.5.2 Experiment with Color Mondrian

In a next experiment, the gray Mondrian was replaced with surfaces made of colored papers, which were illuminated by three independent light sources of short, medium, and long wavelength radiations.

---

[1]The stimuli were called Mondrian because they resembled the paintings of Pieter Mondrian *(Pieter Cornellis Mondrian, Dutch painter, 1872-1944)*.

Land used two identical color Mondrians. One was illuminated by three monochromatic lights that could be varied and the other was illuminated by a constant light. He selected one color patch from the first one and adjusted the power of the three monochromatic light sources (R,G,B) so that the color signal emitted by the selected color patch equates the color signal of the white color patch in the second Mondrian. Even though the measured values were the same, the observers did not perceive the same color when looking at both Mondrians together. The white patch remained white while the selected color patch retained the color given by its reflectances.

Land showed with these experiments that there was little correlation between the amount of radiation falling on the retina and the apparent lightness of an object. He concluded that the perceived color of a unit area was determined by the relationship between this unit area and the rest of unit areas in the image, independently in each wave-band, and did not depend on the absolute value of light. In other words, *ratios* between color surfaces tend to remain constant instead of absolute values. This was also studied by Foster and Nascimento (1994) who showed by computational simulation that spatial ratios of cone excitations were almost invariant under changes of illumination. Land supposed the formation of lightness to occur partly in the retina and partly in the cortex. He thus constructed the term Retinex for his theory of color vision.

2.6 Retinex Computational Models

The Retinex theory of color vision is a simplified model of the HVS, thus making it possible to apply it on images with a not too high computational complexity. A large number of local tone mapping methods take inspiration from it.

A common principle of these methods is to assign a new value to each pixel of an image based on spatial comparisons of the input pixel values. Their differences are the order in which the pixels are addressed, as well as the distance weighting functions.

2.6.1 Path Version

The primary goal of Retinex is to decompose the image into the reflectance image and the illuminant image to remove the illumination effect and retain the lightness of the objects. Land implemented an algorithm called *path-based* Retinex (Land 1964). This algorithm computes subsequent additions of pixel differences along a set of one-dimensional random paths contained in the image (2.8). A threshold operation $\mathcal{T}$ is added to remove the effect of smooth illumi-

nant (2.9). The new value of each pixel, which represents the lightness image (i.e. the perceived reflectance of objects in their surround, independent of the illuminant) is determined by the average over all paths (2.10). These operations are performed independently in each color channel. The log-encoded image is used such that ratios become differences and products become additions.

The image computed by Retinex (called Retinex image) along a path Γ, at pixel p is noted $\mathcal{R}_c^\Gamma(p)$ and is given by:

$$\mathcal{R}_c^\Gamma(p) = \sum_{k \in \Gamma, k < p} \mathcal{T}[\log(I_c(k+1)) - \log(I_c(k))], \qquad (2.8)$$

where $I_c(k)$ is the pixel value at pixel k for color channel c and $\mathcal{T}$ is a threshold operator. Here the operation $k < p$ is true if pixel k is located before p along the path Γ.

$$\mathcal{T}[x] = \begin{cases} 0 & x < \text{threshold} \\ 1 & \text{otherwise} \end{cases} \qquad (2.9)$$

Finally, the Retinex image is averaged across different paths:

$$\mathcal{R}_c(p) = \frac{1}{K} \sum_{\Gamma=1}^{K} \mathcal{R}_c^\Gamma(p), \qquad (2.10)$$

where $\mathcal{R}_c(p)$ is the reflectance estimate for pixel p in color channel c and K is the number of paths.

The generation of random paths is a non-trivial problem. It can be performed in various ways and leads to different results. Rizzi *et al.* (2003) have developed a Retinex path-based version using Brownian motion for path generation. This idea was first proposed in 1993 (Marini and Rizzi 1993) and is inspired by the results of neurophysiological research on human cortical vision areas, where the distribution of receptive fields' centroids can be approximated by Brownian motion. This way of generating paths has also been applied by Finlayson *et al.* (2002) to remove shadows in an image. An extension of this work uses a Hamiltonian path that visits each pixel once (Fredembach and Finlayson 2005).

A major drawback of the Retinex path version is that it does not converge to a stable solution. In particular, the final value of a pixel depends on the choice of paths. This instability raised the interest of researchers that started to formalize the algorithm and study its convergence.

Horn (1974) reformulated Land's Retinex and showed that the illuminant can be estimated using a two-dimensional Laplacian. Brainard and Wandell (1986) attempted to formally describe the Retinex model focusing on path gen-

eration. They demonstrated that the process is stochastic (due to the random order in which pixels are compared) and is well described by a Markov model. They studied the convergence properties of Land's Retinex and showed that, as the number of paths and their lengths increase, the result converges to a simple normalization by the maximum pixel value. Hurlbert (1986) formalized the Retinex theory mathematically and showed that it is equivalent to solving a Poisson equation. She reviewed several algorithms based on Retinex and put them into a compact and single mathematical form. She postulated that the full solution is given by Green's theorem[2] and demonstrated that these algorithms can be seen as an approximation to the real solution.

In addition to its lack of formalism, the practical problems with the Retinex path version are high computational complexity and free parameters, such as the number of paths, their trajectories, and their lengths.

An interesting approach to Retinex is provided by Kimmel et al. (Kimmel *et al.* 2003, Elad *et al.* 2003). They showed that Retinex can be formulated as a quadratic programming optimization problem. As decomposing the image into reflectance image and illumination image is mathematically ill-posed (West and Brill 1982), they redefined the problem using physically motivated considerations such as illumination smoothness and limited dynamic range of the reflectances. Adding these constraints allows their algorithm to converge to a unique solution, the optimal illumination. However, the computational complexity of quadratic programing optimization is high since each pixel is an unknown to the minimization formula. In a later article, Elad *et al.* (2003) propose several methods for reducing the complexity of the above approach by restricting the solution to have a pre-defined structure using either a look-up table, linear or non-linear filters, or a truncated set of basis functions. These simplifications involve less free parameters and yield reasonable yet sub-optimal results.

2.6.2 Iterative Version

The *iterative* version of Retinex is a two-dimensional extension of the path version. It computes a new value for each pixel by iteratively comparing pixels in the image (Frankle and McCann 1983, McCann 1999, Funt *et al.* 2004). The algorithm starts by computing long-distance shifts between pixels and then progressively moves to shorter distances. Like the path version, it is based on four basic operations, ratio, product, reset, average. The reset operation is different than the threshold as it acts as a clipping to the maximum value. In

---

[2]Green's theorem expresses the relationship between surface and line integrals of a scalar function.

(2.12), the maximum value is 1. New pixel values are computed as follows for each color channel c:

$$\mathcal{R}_c^j(p) = \frac{\Omega[\log(I_c(p)) - \log(I_c(p+\delta)) + \mathcal{R}_c^{j-1}(p+\delta)] + \mathcal{R}_c^{j-1}(p)}{2}, \quad (2.11)$$

where

$$\Omega[x] = \begin{cases} 1 & x > 1 \\ x & \text{otherwise,} \end{cases} \quad (2.12)$$

δ is the pixel shift for comparison, and j is the index for recursive iteration. One shift is defined by the distance between the treated pixels and a direction. Equation (2.11) is iterated an arbitrary number of times for each shift. It is repeated for four directions corresponding to horizontal and vertical shifts and for each shift distance. The shift distance starts from $\delta = \frac{1}{2} \times$ the image size in one dimension and is then divided by two until $\delta = 1$ is reached.

The drawback of this implementation is that the number of iterations, which controls the degree of spatial interaction is not defined and has a critical effect on the final result. One iteration is generally not enough and an infinite number of iterations would converge to an output image that corresponds to the input image scaled by its maximum. Although attempts to automatically define the number of iterations using an early stopping technique have been carried out (Ciurea and Funt 2004), it remains an important issue.

Sobol (2004) contributes to the improvement of the Retinex iterative version by introducing a ratio modification operator in the form of a look-up table. This additional operation allows a better compression in high contrast areas while increasing the visibility in low contrast areas and thus improve the rendering of HDR images. However, the problems of non-convergence and non-defined parameters remain.

2.6.3 Surround-Based Version

A non-iterative version of Retinex called *center/surround* or *surround-based* was proposed by Land (1986). With surround-based Retinex, each pixel is selected sequentially and treated only once. New pixel values are given by the ratio between each treated pixel and a weighted average of its surround.

Based on the early work of Land, Jobson and Rahman (Jobson *et al.* 1997, Rahman *et al.* 2004) developed a method called Multi-Scale Retinex with Color restoration (MSRCR). A *single-scale Retinex* image $\mathcal{R}_{c,\sigma}$ is computed for three different surround scales (2.13). The weighted average of each scale is given

by a two-dimensional Gaussian filter $\mathcal{G}_\sigma$ of spatial constant σ. σ is usually defined by a ratio of the image size. Logarithmically, the ratio is expressed as a difference operator:

$$\mathcal{R}_{c,\sigma}(p) = log(I_c(p)) - log(I_c(p) * \mathcal{G}_\sigma), \tag{2.13}$$

where c indicates one color channel and the two-dimensional Gaussian filter is given by (2.14) for p, a pixel of coordinate (x, y).

$$\mathcal{G}_\sigma(x, y) = \frac{1}{2\pi\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.14}$$

The multi-scale version aims to improve the results obtained with one scale. The final image is simply the average of the three single-scale Retinex.

$$\mathcal{R}_c(p) = \frac{1}{3}(\mathcal{R}_{c,\sigma_1}(p) + \mathcal{R}_{c,\sigma_2}(p) + \mathcal{R}_{c,\sigma_3}(p)). \tag{2.15}$$

In addition, a color restoration factor is introduced to compensate for the loss of color saturation inherently present in their method. This color correction greatly enhances the saturation but does not ensure a correct rendition of colors (Barnard and Funt 1998).

In a previous work, we proposed a method based on MSRCR (Meylan and Süsstrunk 2004a). Instead of using three single-scale Retinex, all spatial constants are included into a single filter. The algorithm is applied to the luminance channel and no color restoration is added. Figure 2.7 gives an example of our filter and the three MSRCR filters.
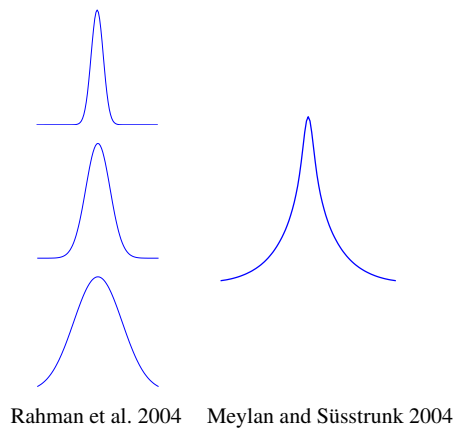

Rahman et al. 2004   Meylan and Süsstrunk 2004

Figure 2.7: Example of filters used for MSRCR of Rahman et al. 2004 (left) and the filter used in Meylan and Süsstrunk 2004a (right).

iCAM (Fairchild and Johnson 2004) is another rendering algorithm based on spatial properties of vision (local adaptation and spatial filtering). The first stage of iCAM accounts for chromatic adaptation. The adapted image is transformed into an opponent representation. Then, a local treatment using Gaussian kernels is applied to the luminance only. Unlike other surround-based methods, it is a complete model of image appearance and quality. It was specifically developed to render HDR images.

Common drawbacks of surround-based local tone mapping methods are the possible appearance of halo artifacts around light sources and graying-out of low contrast areas (see Figure 2.8). Some of the Retinex-based methods mentioned above accounted for theses problems: Kimmel et al. (Kimmel *et al.* 2003, Elad *et al.* 2003) added a halo reduction term in their quadratic programming optimization formulation. The methods of Sobol (2004) also contributes to reducing halo artifacts by introducing a ratio modification operator.

2.7 Solving the Halo Artifact Problem

Halo artifacts and local graying-out are well-known issues when rendering HDR images. Using a local operator involves a trade-off between the increase in local contrast and the rendition of the image: a strong increase in local contrast leads to artifacts while a weak increase in local contrast does not provide the expected improvement of detail visibility. Halo artifacts are due to the local filtering of two areas of very different illumination. For example, if a dim area is close to a bright window, the bright pixels influence the processing of the dim pixels, which causes a black halo around the bright area. Moreover, local filtering tends to make pure black and pure white low contrast areas turn gray. These phenomena are illustrated in Figure 2.8. The top image was obtained with single-scale Retinex using a small spatial constant. Detail visibility is improved but undesirable shadows (halos) appear along high contrast edges and the black t-shirt looks washed out. The shadow on the face of the person is a halo artifact due to the bright window. The bottom image was obtained with the same method using a larger filter. No artifacts are created but the increase in local contrast is not satisfying.

These issues were addressed by a number of local tone mapping methods not based on Retinex. Tumblin and Turk (1999) propose a method called Low Curvature Image Simplifiers (LCIS) to increase the local contrast while avoiding halo artifacts. LCIS uses a form of anisotropic diffusion to enhance boundaries while smoothing non-significant intensity variations. This method does not take into account whether details are visibly significant. Consequently, the resulting images tend to look unnatural due to excessive detail visibility.

Fattal et al. compress the dynamic range using a gradient attenuation function defined by a multiresolution edge detection scheme (Fattal *et al.* 2002). A new image of lower dynamic range is obtained by solving a Poisson equation on the modified gradient field. This approach provides good results but requires parameter tuning. Reinhard *et al.* (2002) developed a local method based on the photographic dodging and burning technique. They use a circular filter, whose size is adapted for each pixel by computing a measure of local contrast. A related method was proposed by Ashikhmin (2002), which computes a measure of the surround luminance for each pixel. This measure is then used for the definition of the tone mapping operator. Both methods provide an efficient way of compressing the dynamic range while reducing halo artifacts. However, the restriction to a circular surround limits their performance. DiCarlo and Wandell (2000) investigated tone mapping algorithms and the creation of halo artifacts. They suggest the use of robust operators to avoid these. A robust Gaussian includes a second weight that depends on the intensity difference between the current pixel and its spatial neighbors. This technique preserves the sharpness of large transitions. A recent approach based on LCIS and robust operators was proposed by Durand and Dorsey (2002). Their method renders HDR images using bilinear filtering, an alternative for anisotropic diffusion. It does not attempt to imitate human vision.

2.8 Evaluation of Tone Mapping Algorithms

Considering the numerous existing tone mapping algorithms, being able to assess their quality has become an important issue. Subjective methods of comparisons have been performed under the form of psychophysical experiments. However, since each of the tone mapping algorithms has different strengths and weaknesses and different reproduction goals, it is difficult to draw general conclusions. Tone mapping algorithms have to be tested against perceptual attributes and with a large variety of scenes. Here, we review existing evaluation procedures although our algorithm has not yet been tested by any of them.

Drago *et al.* (2003) asked observers to judge similarity and dissimilarity of pairs of tone-mapped images. They obtained psychophysical scale values for three perceptual attributes (apparent image contrast, apparent level of details, apparent naturalness). They analyzed the results for the perceptual attributes in a "stimulus space" derived from the analysis of dissimilarity rating.

Kuang *et al.* (2004) developed a paired comparison using eight tone mapping algorithms and ten HDR images. Observers were presented two tone-mapped images and selected the one they prefer according to different perceptual attributes. General rendering performance, tone compression, natural appear-

ance, color saturation, image contrast and image sharpness were evaluated.

Yoshida *et al.* (2005) performed a comparison with real scenes. Observers stood in front of a real scene that was captured prior to the experiment. It was then rendered using different tone mapping methods and displayed on a standard monitor. The subjects could browse through the tone-mapped images and had to rank them with respect to perceptual attributes, using the real scene for comparison. Tested attributes were image naturalness, overall contrast, overall brightness, and detail reproduction in the dark and bright regions.

Ledda *et al.* (2005) proposed a psychophysical experiment to validate six tone mapping operators against a reference image displayed on an HDR monitor. Using an HDR display rather than direct comparison with real scene produces less uncontrolled variables. Observers were presented two tone-mapped images on an SDR display along with the reference scene projected on the HDR display. Their task was to choose the image that matches the best the reference scene. Overall similarity to reference and detail reproduction were tested. The data were studied with a multiple comparisons score test. For the first task of overall similarity, the iCAM algorithm (Fairchild and Johnson 2004) and the photographic operator of Reinhard *et al.* (2002) performed the best. For the second task of detail reproduction, the local eye adaptation (Ledda *et al.* 2004) which is based on Pattanaik's model (Pattanaik *et al.* 1998), had the best score. Histogram adjustment (Ward *et al.* 1997) had a fair overall performance. It is interesting to note that algorithms such as the bilateral filtering (Durand and Dorsey 2002) that were well ranked in other comparisons did not obtain a good score in this experiment due to the presence of a reference image.

2.9   Conclusion

Global tone mapping methods are essential to display visually appealing images. Not only they perform the mapping between the scene's to the display's dynamic range and compensate for the display non-linearities, but they can also treat the image differently according to scene statistics. In addition to global tone mapping, a local tone mapping may be applied to improve local features of the image.

The Retinex theory of color vision has often served as a basis for local tone mapping algorithms. Most of these methods give satisfying results for SDR images. However, in the case of HDR images, halo artifacts start to appear as well as graying-out of black and white low contrast areas. In the next chapter, we propose a method to solve the problem of halo artifact using a variant of the surround-based Retinex.

Figure 2.8: Example of the trade-off between increase in local contrast and image rendering. **Top:** Image treated with single-scale Retinex using a small scale. The increase in local contrast is significant but halo artifacts are visible along high contrast edges and the black t-shirt looks washed-out. The shadow on the face of the person is a halo artifact. **Bottom:** Image treated with single-scale Retinex using a large scale. There is no halo artifact nor graying-out but the increase in local contrast is not sufficient.

Chapter 3

The Retinex-Based Adaptive Filter Tone Mapping Method

3.1 Introduction

This chapter presents our local tone mapping algorithm that renders HDR images to SDR displays. It belongs to the class of surround-based Retinex algorithms (Land 1986, Rahman *et al.* 2004, Meylan and Süsstrunk 2004*a*) but differs in many ways. First, an adaptive filter is used instead of the traditional circular surround, thus avoiding halo artifacts. Then, the local processing is weighted by a sigmoidal function to conserve black and white low contrast areas. Finally, instead of treating the R, G, and B color channels independently, we treat the luminance channel only to ensure good color rendition.

Figure 3.1 shows the global framework of our method that processes the luminance and the chrominance in parallel. The left part describes the global processing that is applied to the RGB input image I, which is linear with respect to scene radiances and encoded with RGB primaries, and the right part shows the local processing applied to the luminance channel only. The luminance channel Λ is obtained by performing a principal component analysis (PCA) on the input image. A PCA is a statistical procedure that transforms a set of correlated variables into a set of uncorrelated variables called principal components. The transform is given by the eigen vectors of the covariance matrix computed for the centralized data (i.e. the mean of the data is subtracted). The first principal component corresponds to a vector that provides the largest variance of projected data. Applied to an RGB image, it provides a representation where the first principal component corresponds the luminance channel Λ, and the second and third components correspond to the chrominance chan-

nels C_1, C_2 (Buchsbaum and Gottschalk 1983). With PCA, the luminance is decorrelated from the chrominance, which allows the color to remain relatively unchanged despite the processing of the luminance.

This chapter is structured as follows: Section 3.2 presents the processing applied to the luminance channel illustrated in the right part of Figure 3.1. It describes the global compression followed by the local processing. The local processing includes the construction of a mask using an edge map, and the application of a sigmoidal function that weighs the effect of the local filtering. Section 3.3 presents an alternative to improve the edge detection and shows the results obtained by applying our method using the new edge map. Then, Section 3.4 describes the color image processing illustrated in the left part of Figure 3.1. The PCA transform and the saturation enhancement are described. In Section 3.5, we analyze the computational complexity of our method. We propose a way to reduce it by computing the mask on a downsampled version of the image. Finally, Section 3.6 provides comparisons with other methods that are state of the art; we discuss the different properties of these methods and show our results. A conclusion is given in Section 3.7.
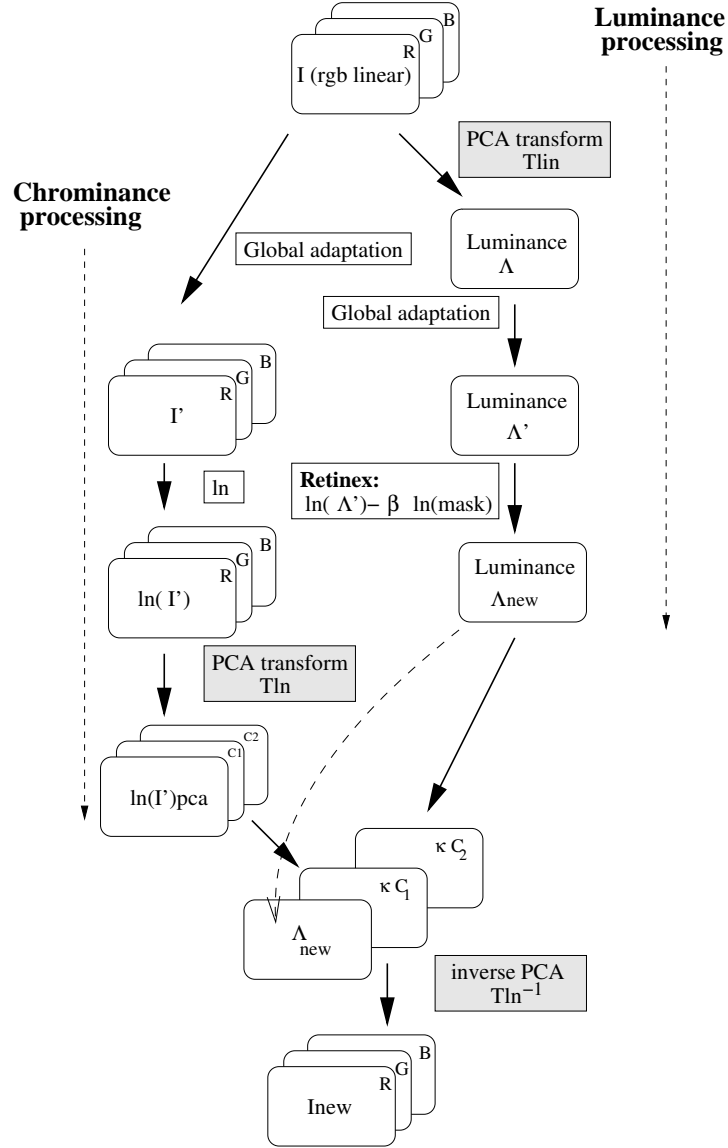
Figure 3.1: Simplified global framework. Our method uses parallel processing; one for luminance and one for chrominance. A first global compression is applied to both the luminance image Λ and the linear RGB image I. Then, the Retinex-based adaptive filter method is applied to the logarithm of the globally corrected luminance Λ' (see 3.2.2) while a logarithm is also applied to the globally corrected RGB image I'. I' is then transformed into a luminance/chrominance encoding through PCA. Its first component is replaced by the treated luminance Λ_{new} (3.17). The image thus obtained is transformed back to RGB.

3.2   The Retinex-Based Adaptive Filter Method: Luminance Processing

3.2.1   Step 1: Global Tone Mapping

Our algorithm consists of two parts: a preliminary global tone mapping followed by the Retinex-based local processing. The global tone mapping that is applied to the linear luminance image Λ performs a first compression of the dynamic range. It can be compared to the early stage of the human visual system where a global adaptation takes place (Alleysson and Süsstrunk 2002, Poynton 2003). We design our global tone mapping function to be similar to the adaptation of photoreceptors, which can be approximated by a power function. The curvature of the function that determines the adaptation state depends on the mean luminance of the total field of view (Alleysson and Süsstrunk 2002).

The average image luminance Λ_{av} is computed by taking the average of the log-encoded pixels:

$$\Lambda_{av} = \frac{\sum_{p \in I} \ln(\Lambda(p))}{N}, \tag{3.1}$$

where N is the number of pixels and p is a pixel in I. Λ is the luminance given by the first principal component of a PCA applied to the RGB image I. It is encoded linearly and normalized to 1 but in order to perform the logarithm operation, it is temporarily scaled to a maximum value of 100.

The non-linear luminance Λ' is given by

$$\Lambda'(p) = \Lambda^{\frac{1}{\gamma}}(p), \tag{3.2}$$

where the value of $\frac{1}{\gamma}$ is an affine function of the average luminance Λ_{av} (3.3). The resulting luminance image Λ' is still normalized to 1.

$$\frac{1}{\gamma} = \min\left(1, \frac{1}{6}\Lambda_{av} + \frac{2}{3}\right). \tag{3.3}$$

The coefficients of the affine function were defined experimentally as follows: a high or normal key image is not globally compressed and is therefore assigned $\frac{1}{\gamma} = 1$. As the average luminance decreases, the exponent $\frac{1}{\gamma}$ decreases, increasing the sensitivity for dark areas.

3.2.2   Step 2: Local Adaptation

After global processing, local adaptation is performed using a surround-based Retinex method. Traditionally, surround-based Retinex methods (Land 1986, Rahman *et al.* 2004, Meylan and Süsstrunk 2004a) compute a new value for

each pixel by taking the difference between the log-encoded treated pixel and the log-encoded value of a mask (3.4). Here, we use a natural logarithm (ln) but a logarithm in base 10 (log) could be used as well. The mask represents a weighted average of the treated pixel's surrounding area.

$$\Lambda''(p) = \ln(\Lambda'(p)) - \ln(mask(p)), \tag{3.4}$$

where Λ' is the non-linear luminance image computed in (3.2) and $mask$ is computed by convolving the luminance Λ' with a surround function (i.e. a two-dimensional filter).

The Sigmoidal Weighting Function

A drawback of surround-based methods is that small filters tend to make pure black or pure white low contrast areas turn gray, due to the local normalization. We overcome this problem by introducing a weighting factor $\beta(p)$ that ensures the conservation of white and black areas.

With our method, each pixel value of the treated luminance image Λ'' is computed as follows:

$$\Lambda''(p) = \ln(\Lambda'(p)) - \beta(p) \cdot \ln(mask(p)), \tag{3.5}$$

Both Λ' and $mask$ are encoded in floating points in the range [0,1]. To compute the logarithm, we temporarily scale them to a larger range and clip them to a minimum value of 0.1. The logarithm is thus performed in the range [0.1,100]. The log-encoded Λ' and $mask$ that we call Λ'_{ln} and $mask_{ln}$, respectively are then normalized back to 1. The operation performed by (3.6) and (3.7) lets Λ'_{ln} and $mask_{ln}$ have negative values. However, since $mask_{ln}$ represents a low-pass version of the image, most of the very small values have disappeared. In practice, $mask_{ln}$ is unlikely to have negative values.

$$\Lambda'_{ln}(p) = \frac{1}{\ln(100)} \cdot \ln(\max(0.1, \Lambda'(p) * 100)) \tag{3.6}$$

$$mask_{ln}(p) = \frac{1}{\ln(100)} \cdot \ln(\max(0.1, mask(p) * 100)) \tag{3.7}$$

We thus reformulate (3.5):

$$\Lambda''(p) = \Lambda'_{ln}(p) - \beta(p) \cdot mask_{ln}(p), \tag{3.8}$$

The log operation performed on Λ' and $mask$ in (3.6) and (3.7) can be approximated by a power function of exponent $\frac{1}{6}$ (illustrated in Figure 3.2). Compared to a logarithm in the range [1,100], which can be approximated by

$y = x^{\frac{1}{3}}$ (generally used to model the HVS non-linearity), our log operation expands the shadows more (Figure 3.2). This helps retrieve details in dark areas but also has the drawback of compressing the highlights extensively, as shown later in Figure 3.6.
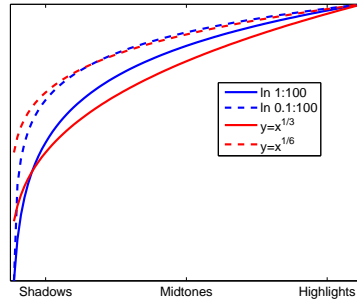

Figure 3.2: Comparison of a logarithm function applied in the range [1,100] and [0.1,100]. These functions can be approximated by a power law of exponent $\frac{1}{3}$ and $\frac{1}{6}$, respectively.

As for β, it weighs the mask depending on the pixel values of Λ'_{ln}:

$$\beta(p) = 1 - \frac{1}{1 + e^{-10 \cdot (\max(\Lambda'_{ln}(p), 0) - 0.5)}}. \tag{3.9}$$

The β factor is based on a sigmoid function (Figure 3.3). It maps the white to white and the black to black, which is necessary to obtain visually pleasing images. For a pixel of high intensity, the mask is weighted by a value close to 0. Since the mask is subtracted from the log-encoded luminance, it effectively keeps the pixel bright. Similarly, a pixel of low intensity is weighted by a value close to 1, which has the effect of maintaining black. This function lets the middle gray values change without constraint while restricting the black to remain black and the white to remain white.

Λ'' resulting from (3.5) may have negative values. A final processing is needed to remove outliers and to scale the luminance in the range [0,1] before it is integrated back in the color image. This is done using histogram scaling and clipping (Section 2.4). Λ'' is scaled to [0,1] with 1% of its pixels clipped at both extremities of the histogram.

The scaled luminance image Λ_{new} is computed as follows:

$$\Lambda_{new}(p) = min(1, \frac{\max(0, \Lambda''(p) - b)}{w - b}), \tag{3.10}$$

where b and w are given by the pixel values associated to the histogram bin
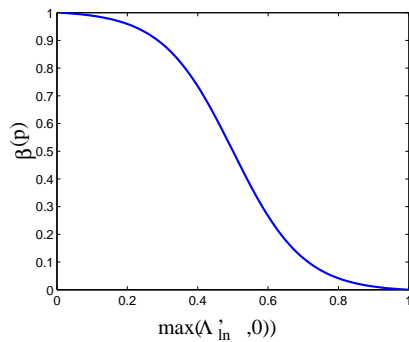
Figure 3.3: Illustration of the $\beta(p)$ function that weighs the mask according to the log luminance value.

in which 1% and 99% of the image data points fall, as explained in Section 2.4.

The Adaptive Filter

Another important drawback of surround-based Retinex methods is that there is a trade-off between the increase in local contrast and a good rendition of the image (Section 2.7). A small surround allows a significant increase in local contrast but induces halo artifacts along high contrast edges. Using a larger surround reduces the artifacts, but provides less increase in local contrast. This trade-off has already been mentioned by Jobson *et al.* (1997) and Barnard and Funt (1998), remarking that MSRCR caused halo artifacts along high contrast edges.

Our adaptive filter method prevents halo artifacts by adapting the shape of the filter to the high contrast edges in the image. The position of the edges are given by a binary map (Figure 3.5). Since the filter follows image contours, a bright area has less influence on the treatment of a neighboring dim area. This modification does not change the principle of Retinex surround-based methods, that is, to compute the difference between each pixel value and a weighted average of its surround. What changes with the adaptive filter method is the way the mask is computed. Since the filter is different for each pixel, it is not possible to use a convolution anymore. The mask is thus computed specifically for each pixel (3.11). The weighted sum of pixels is normalized by the sum of weights so that each pixel has an equal contribution to the mask even if it is surrounded by edges.

$$mask(x,y) = \frac{\sum_{\theta=0}^{360} \sum_{r=0}^{r_{max}} \Lambda'(x + r \cdot \cos(\theta), y + r \cdot \sin(\theta)) \cdot e^{-\frac{r^2}{\sigma_{\theta,r}^2}}}{\sum_{\theta=0}^{360} \sum_{r=0}^{r_{max}} e^{-\frac{r^2}{\sigma_{\theta,r}^2}}}, \quad (3.11)$$

where (x, y) is the coordinate of pixel p. θ is the angle of the radial direction, r is the distance to the central pixel and $\sigma_{\theta,r}$ is defined as follows:

$$\sigma_{\theta,r} = \begin{cases} \sigma_0 & \text{no high contrast edge was crossed along } r \text{ with } \theta \text{ const.} \\ \sigma_1 & \text{a high contrast edge was crossed along } r \text{ with } \theta \text{ const.} \end{cases}$$

Equation (3.11) can be interpreted as follows. The value of the mask at coordinate (x, y) is given by a weighted average of pixels surrounding the position (x, y). The weights of surrounding pixels are given by a Gaussian function, whose spatial constant varies according to the image high contrast edges. It is done by selecting one pixel after the other in a radial manner. The first pixel to be selected is the central pixel. Then, all pixels along a radial direction are added, weighted by a Gaussian function with spatial constant σ_0. If an edge is crossed along the radial direction, $\sigma_{\theta,r}$ is assigned a smaller value σ_1 and keeps the same value until $r = r_{max}$ with $r_{max} = 3 \cdot \sigma_0$. For each new radial direction, $\sigma_{\theta,r}$ is reset to its initial value σ_0. The weighted sum of pixels continues for each direction until the surround is completed.

Some limitations of precision arise from the fact that a pixel has only 8 neighbors. For simplification, we described the *mask* computation with an iterative process. Practically, this process is performed recursively starting from the center, as illustrated in Figure 3.4. It is thus possible to explore more than eight directions since each time a pixel is visited, three directions are added in the process. Each pixel is used only once in the weighted sum.
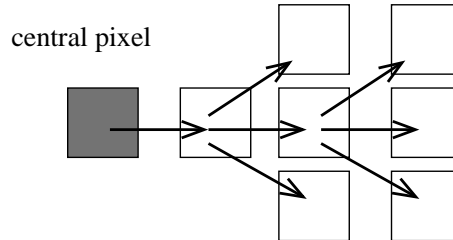

Figure 3.4: Illustration of the recursive process that visits each pixel in the surround.

The numerical values for σ_0 and σ_1 are chosen to be fractions of the image size ($\frac{1}{16}$ for σ_0). Experimentally, we have found that σ_1 needs to be at least $\frac{1}{2}\sigma_0$ to avoid halos in most of our images. We did not use $\sigma_1 = 0$ to ensure that no artifacts will be introduced by the hard threshold even when the edge is very close to the treated pixel.

Construction of the Edge Map

We use a Canny edge detector to detect high contrast edges (Canny 1986). The Canny method finds edges by looking for local maxima of the image gradient. It detects strong and weak edges. Weak edges appear in the output only if they are connected to strong edges. The thresholds for strong and weak edges are fixed values chosen experimentally and kept the same for all images. Fixed thresholds are desirable since we only want to detect high contrast edges. It is thus possible to obtain no edge for an image that has no high contrast edge and where circular, non-adaptive surrounds are sufficient.

The construction of the filter according to a segmented image is illustrated in Figure 3.5. The top left and top right images represent the original image and its corresponding edge map, respectively. The filter was computed for the pixel indicated by the cross using (3.11). The bottom image shows the corresponding adaptive filter.

Some Limitations of the Method

Our method works best for low key images containing important details in dark areas. It efficiently increases the local contrast in the shadow while preventing halo artifacts. However, it is not adequate for high key images as the logarithm operation applied in the range [0.1,100] (3.6) tends to compress the highlights too much. This effect is illustrated in Figure 3.6. With this image, our algorithm was able to retrieve details in the room but the outdoor details seen through the window are compressed extensively.
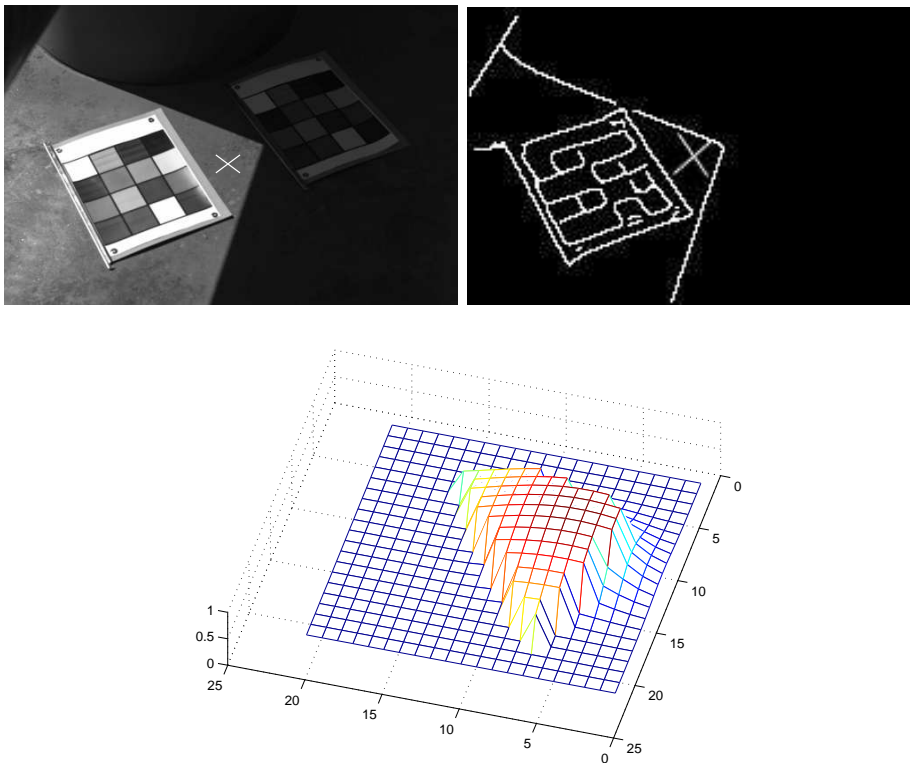
Figure 3.5: Construction of the adaptive filter for the pixel indicated by the cross. **Top left:** Luminance image Λ'. **Top right:** Luminance image segmented with a Canny edge detector. **Bottom:** 3D representation of the filter corresponding to the pixel indicated by the cross.

Figure 3.6: Limitation of the method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. Our algorithm was able to retrieve details in the room but the outdoor details seen through the window are compressed too much.

3.3   An Alternative to Canny Edge Detection using Surround Suppression

The drawback of Canny operator (Canny 1986) is that it only detects local luminance changes and thus can not make a distinction between texture edges and object boundaries. Despite the high threshold value that we use, texture edges may appear in the edge map controlling the adaptive filter. For our Retinex-based adaptive filter algorithm, it would be interesting to make this distinction and only retain edges corresponding to region boundaries illuminated by two different illuminants.

We implemented a boundary detector based on the work of Grigorescu et al. (Grigorescu *et al.* 2004, Grigorescu *et al.* 2003)[1]. Their method takes inspiration from the surround suppression mechanisms of the HVS and allows to distinguish between texture edges and object boundaries.

In this section, we study the advantage of this boundary detector compared to a Canny edge detector. We then include the boundary detection algorithm in our Retinex-based adaptive filter method and compare the resulting images.

3.3.1   Surround Suppression in the Human Visual System

Surround suppression or equivalently called non-classical receptive field inhibition takes place in the early stages of the HVS visual information processing (Petkov and Westenberg 2003). It changes the perception of a stimulus depending on the presence of other such stimuli in the surrounding neighborhood. In other words, the response of a neuron to a stimulus is reduced by the addition of stimuli in the surround. There exist two types of surround suppression: the anisotropic surround suppression that takes into account the relative angle of the stimuli, and the isotropic surround suppression that is independent from the stimuli angle. In the anisotropic behavior, the inhibitory effect is small when the orientations of the surround stimuli are orthogonal to the central stimulus. In the isotropic behavior, the inhibitory effect is present irrespectively of the relative orientation of the surround stimuli.

3.3.2   A Boundary Detector

Grigorescu et al.'s method (Grigorescu *et al.* 2004, Grigorescu *et al.* 2003) integrated a surround suppression step in a Canny-like edge detector so that the edges in an image that are part of a texture are suppressed while edges that belong to a an object contour or region boundary are retained. First, the

---

[1]This work was done in collaboration with Sebastian Pittet (semester project, June 2005).

magnitude of the input image gradient is computed to detect the high contrast local luminance changes. The image resulting from this first step is called the *magnitude image* and is noted U:

$$U(x,y) = \sqrt{(\nabla_x \Lambda'(x,y))^2 + (\nabla_y \Lambda'(x,y))^2}, \qquad (3.12)$$

where Λ' is the non-linear luminance image resulting from (3.2) and (x,y) is the coordinate of a pixel in the image.

The *surround suppressed image E* is obtained by subtracting an inhibition term Υ to the magnitude image U. Thus, the importance of texture edges is reduced.

$$E(x,y) = U(x,y) - \alpha \cdot \Upsilon(x,y), \qquad (3.13)$$

where α is a factor that determines the extent of the inhibition.

The suppression step can be isotropic or anisotropic. Here, we only use isotropic surround suppression as we saw with our natural images that the stimuli relative angle did not influence the results much. For isotropic suppression, the inhibition term Υ is given by the convolution between the magnitude image U and the normalized difference of Gaussian operator (DoG).

$$\Upsilon(x,y) = U(x,y) * g, \qquad (3.14)$$

where

$$g = \frac{DoG(x,y)}{\|DoG(x,y)\|} \qquad (3.15)$$

and

$$DoG(x,y) = \frac{1}{2\pi(4\sigma)^2} e^{-\frac{x^2+y^2}{2(4\sigma)^2}} - \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}. \qquad (3.16)$$

The next two steps, similar to Canny edge detection, include thinning the detected edges by non-maxima suppression and a binarization using hysteresis thresholding. They are applied on the surround suppressed image where edges are characterized by high pixel values (close to 1). A pixel value close to zero indicates the absence of an edge. As the edges of the surround suppressed image can be a few pixel wide, they must be thinned to single pixel. *Thinning by non-maxima suppression* transforms the wide edges into single pixel edges and position them at the location of greatest magnitude. This results in a *thinned image* which still has to be transformed into a *binary map* also called *edge map* (Figure 3.7 and Figure 3.8). This binarization operation must remove false edges while preserving true edges connectivity. This is done by using two

different threshold values. The first threshold detects strong edges, and the second threshold detects weak edges. The binary edge map includes strong edges, and weak edges under the condition that they are linked to strong edges. This method is called *hysteresis thresholding*. Images (a) to (d) of Figure 3.7 and 3.8 illustrate each of these steps and (e) is the resulting edge map. Image (f) shows the edge map obtained by a simple Canny edge detection when using the same parameters than for (e). We see that the surround suppression step brings an improvement over a simple Canny edge detection.

3.3.3   Using Surround Suppression with the Adaptive Filter Method

Figure 3.9 and 3.10 compare the images obtained by our Retinex-based adaptive filter method using either an edge map computed by a simple Canny edge detector (top left) or by the boundary detector using surround suppression (top right). The top images show the edge maps while the middle and bottom images show the corresponding final images rendered by our Retinex-based adaptive filter algorithm. We see that the spatial processing is performed differently, which induces visual differences in the final image. However, it is difficult to judge which one of the two edge detection methods provides the most visually pleasing images.

This study leads to the conclusion that the addition of a biologically motivated step to the traditional Canny algorithm could potentially improve the result of the Retinex-based adaptive filter. Nevertheless, further investigation and testing is needed to fully validate this statement.

Original image

Magnitude image

Surround suppressed magnitude image

Figure 3.7: Illustration of the different steps of the boundary detection using surround suppression method of Grigorescu et al. and comparison with a simple Canny edge detection. (a) Original image. (b) Magnitude image. (c) Magnitude image after surround suppression.

Thinned image

Edge map after hysteresis thresholding

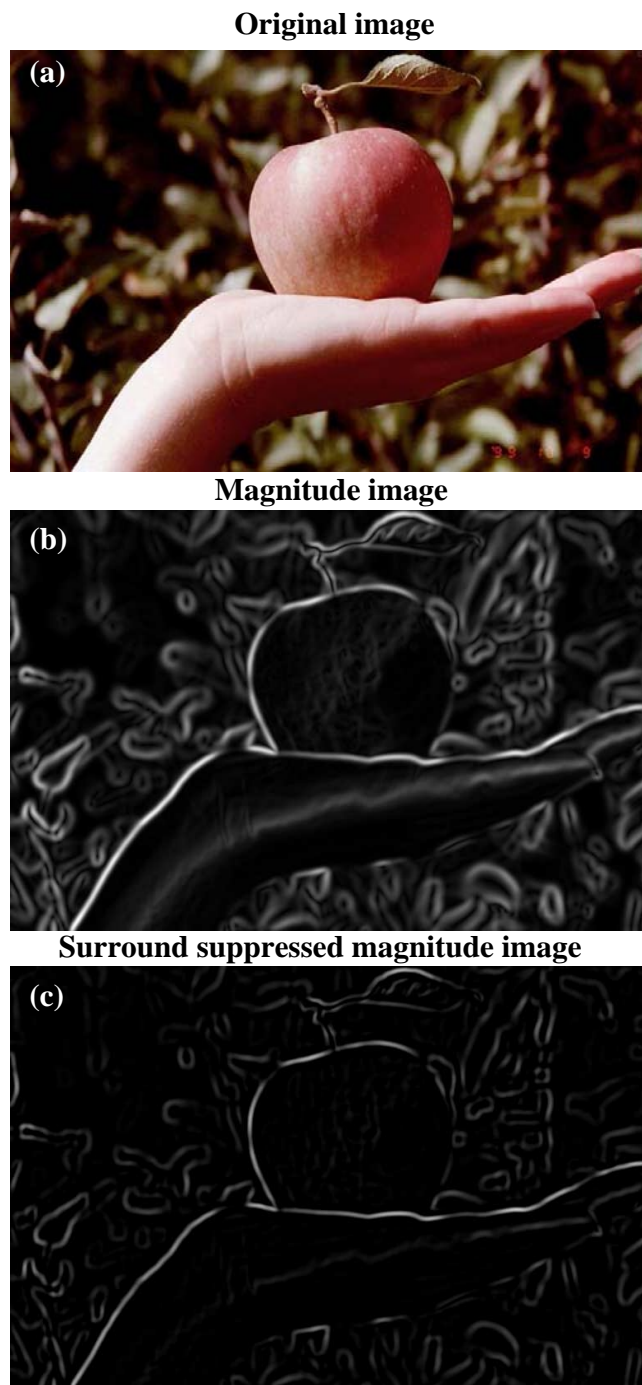**Canny edge detection**
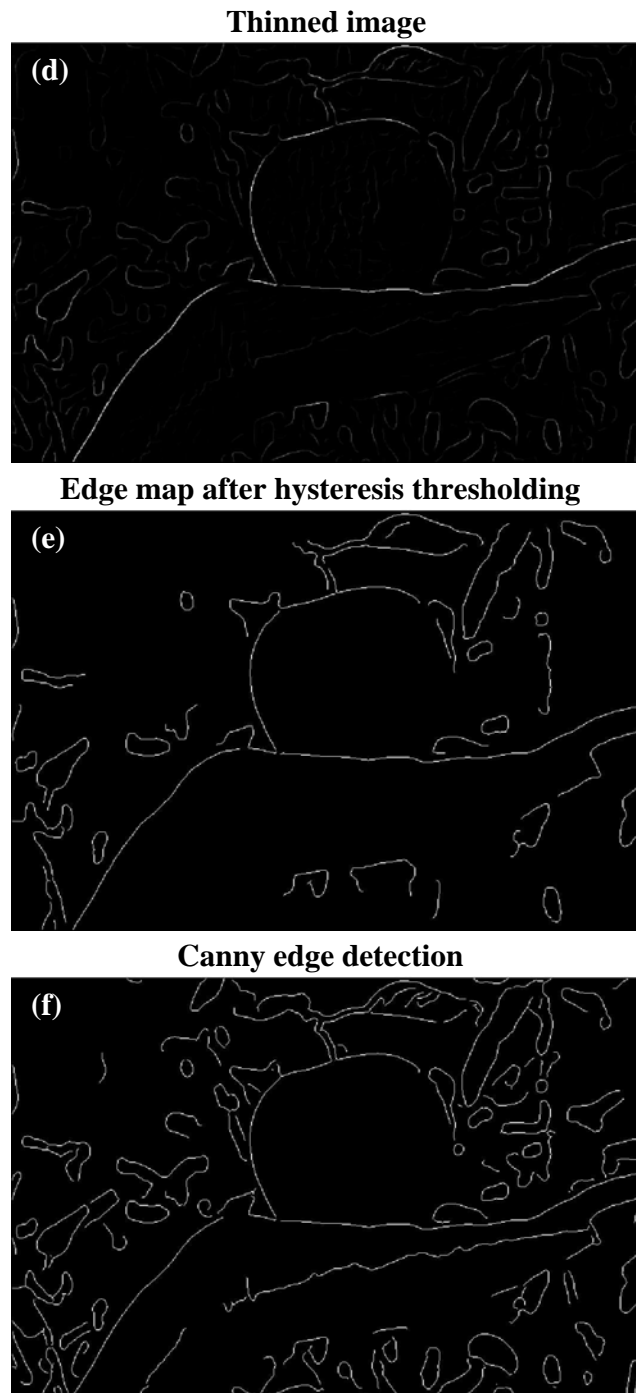
Figure 3.8: Illustration of the different steps of the boundary detection using surround suppression method of Grigorescu et al. and comparison with a simple Canny edge detection. (d) Image after thinning by non-maxima suppression. (e) Edge map obtained after applying a hysteresis thresholding on (d). (f) Edge map obtained with a simple Canny edge detection for comparison.

Original edge map **Edge map with surround suppression**

Original

With surround suppression
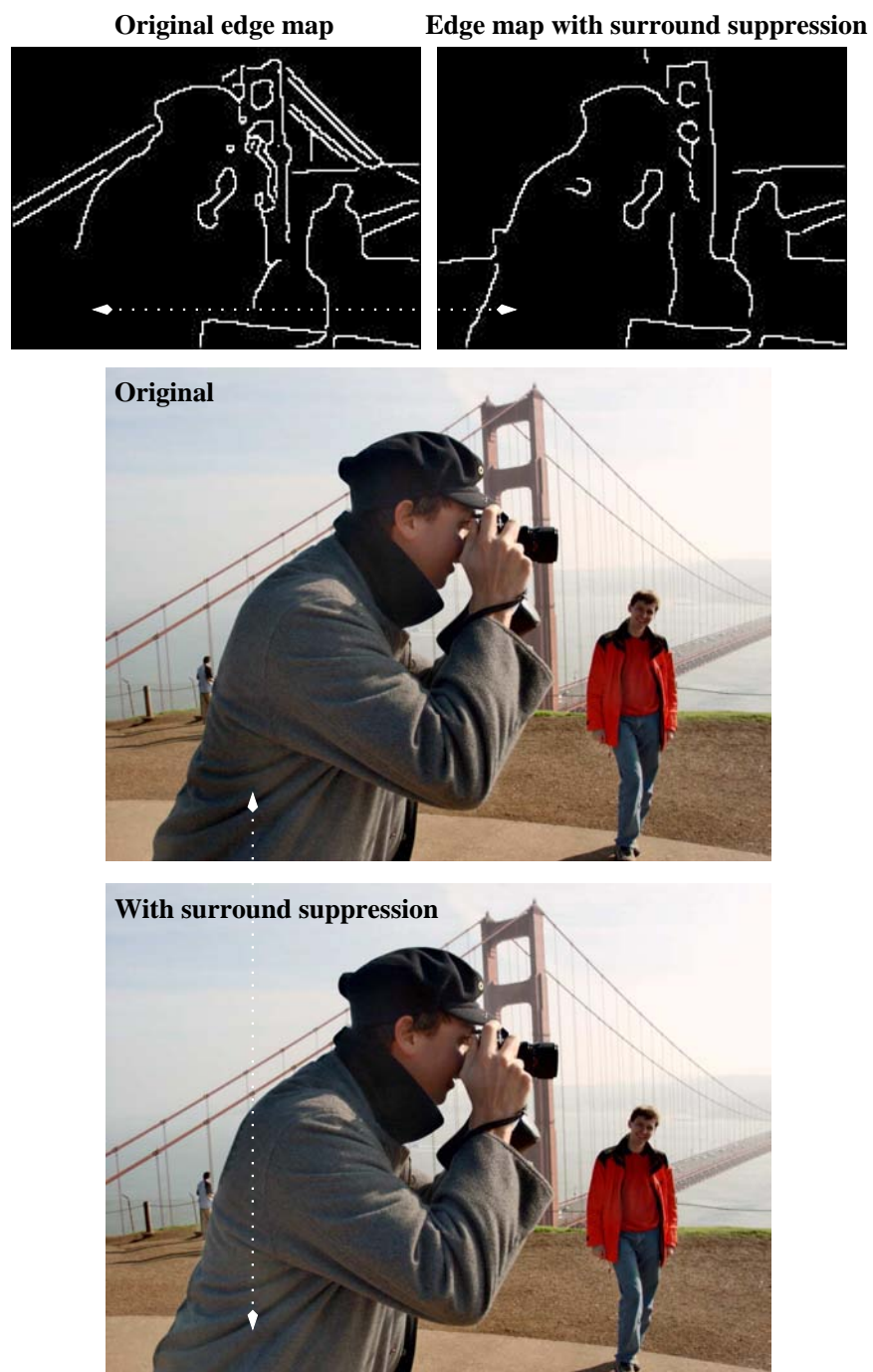
Figure 3.9: Comparison of Canny edge detection and boundary detection using surround suppression, when included in our adaptive filter algorithm. **Top left:** Edge map obtained with a simple Canny edge detection. **Top right:** Edge map obtained with the boundary detector described in this section. **Middle:** Final image obtained with a simple Canny edge detection. **Bottom:** Final image obtained with the boundary detector described in this section.

Original edge map   **Edge map with surround suppression**

Figure 3.10: Comparison of Canny edge detection and boundary detection using surround suppression, when included in our adaptive filter algorithm. **Top left:** Edge map obtained with a simple Canny edge detection. **Top right:** Edge map obtained with the boundary detector described in this section. **Middle:** Final image obtained with a simple Canny edge detection. **Bottom:** Final image obtained with the boundary detector described in this section.
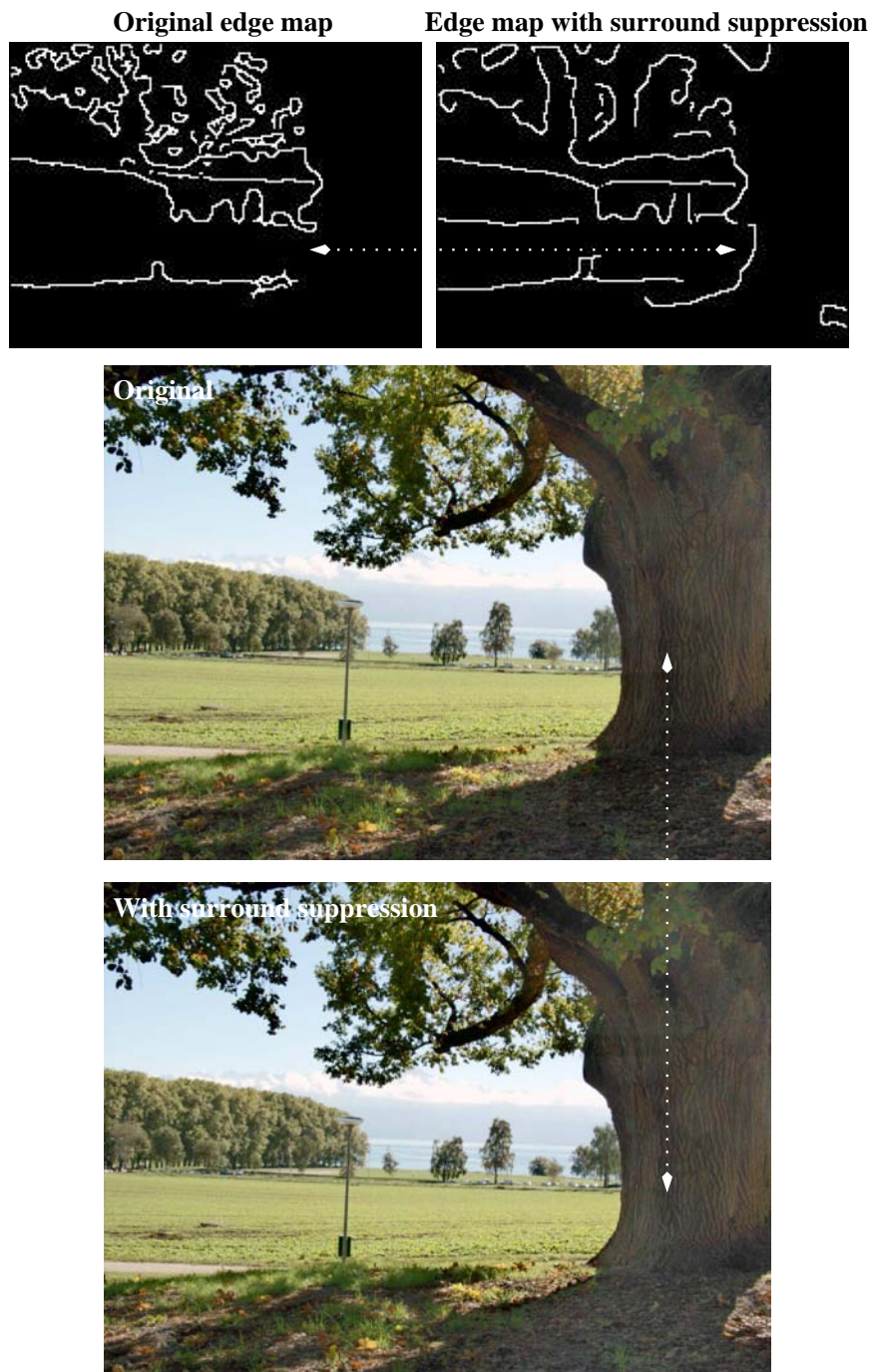
3.4 The Retinex-Based Adaptive Filter Method: Color Processing

The color processing of our method takes inspiration from the HVS, which treats chromatic and achromatic data independently (Figure 3.1). In the retinal pathways, the long, medium, and short wavelengths (LMS) signals captured by the cones are decorrelated. After being processed by subsequent neural stages, they form two major parallel circuits. One is achromatic and non-opponent. The other is chromatic and opponent (Wandell 1995, Kaiser 1996).

Based on this knowledge, we use a PCA to decorrelate the RGB representation of the input image into three principal components. Our motivation is that PCA has properties that intrinsically lead to an opponent representation of colors. Buchsbaum and Gottschalk (1983) described the relation between PCA and the HVS. They showed that optimum transformations in terms of information processing, such as PCA, result in one and only one component that is all positive and has the largest share of signal energy. It is the achromatic channel, carrying luminance information. The second and the third components are opponent. They represent the chrominance channels: Red-Green and Yellow-Blue.

In Section 3.2, we described the treatment applied to the luminance channel only. The result of the luminance processing is inserted back into the parallel color image processing as illustrated in Figure 3.1.

For the processing of the RGB image I, we follow the same steps that were applied to the luminance Λ with the exception of the local Retinex-based processing. The global compression defined in (3.2) is applied to each R, G, and B color channel to obtain I'. Then, a logarithm is applied. The log-encoded image $\ln(I')$ is transformed via PCA into a decorrelated space to obtain $\ln(I')_{pca}$. The first principal component is replaced by the treated luminance Λ_{new} and recomposed with the chrominance channels. The chrominance channels are weighted by a factor κ, in order to compensate for the loss of saturation induced by the increase in luminance. The increase in luminance is partly due to the logarithm operation applied to Λ'. Since this operation is similar to all images, we use a constant factor κ. We found experimentally that $\kappa = 1.6$ is a suitable value.

Converting an RGB image into a luminance/chrominance representation could also be done using an existing color transform. However, with most existing transforms, some luminance information remain in the chrominance and vice versa, due to the non-orthogonality of the color space basis vectors. Unlike these transforms, PCA provides an orthogonal representation of the luminance and chrominance components, which ensures good color rendition

despite the processing of the luminance. We could observe that another color transform such as YUV (Poynton 2003) provided good color rendition as well[2]. However, there are visible color differences in the resulting images. Figure 3.11 compares an image treated by our algorithm using a PCA transform with the same image treated using a YUV transform. The image computed using YUV looks slightly green but it is hard to justify which one is more visually appealing.

A limitation of using PCA is that it works only for natural scenes which contain a reasonable diversity of colors. Indeed, particular cases such as a singular color image lead to an ill-conditioned transformation matrix and thus to the failure of the PCA algorithm. This does not usually happen when treating natural images even in the presence of a color cast, but is more likely to happen with synthetic images. In this case, it is necessary to use an existing color transform such as YUV.

3.5 Computational Complexity

The use of an adaptive filter instead of a fixed surround shape involves a significant increase in computational complexity. Indeed, when the filter is the same for each pixel, the mask can efficiently be computed by a convolution or equivalently, a multiplication in the Fourier domain. The introduction of the adaptive filter prevents the use of a convolution and therefore makes the method computationally very expensive. The adaptive filter method such as computed in (3.17) has an order of complexity of $O(N + N^2)$, where N is the number of pixels in the image. The first term N is due to the difference operation and the N^2 term is due to the mask computation.

We propose two solutions to reduce the computational time. The first solution consists in limiting the size of the surround by taking the decay of the Gaussian weighting function into account. Our default value for the radius surround size is $r_{max} = 3\sigma_0$. This reduces the computational complexity to $O(N + N \cdot (2r_{max})^2)$.

The second solution is to use a downsampled version of the image to compute the mask. The mask is then upsampled before being subtracted from the high resolution image (3.17). The upsampling and downsampling operations are performed using bilinear interpolation. A similar method to reduce the computational complexity was introduced by Moroney (Moroney 2004).

Let us use the symbol $\bullet_{\downarrow n}$ for downsampling by n and and $\bullet_{\uparrow n}$ for upsampling by n. Equations (3.8) and (3.11) for computing the treated luminance become

---

[2]The influence of the luminance definition on the rendered image is discussed in Chapter 4.

$$\Lambda''(p) = \Lambda_{ln}(p) - \beta(p) \cdot mask_{ln,\uparrow n}(p), \tag{3.17}$$

$$mask(x,y) = \frac{\sum_{\theta=0}^{360} \sum_{r=0}^{r_{max}} \Lambda'_{\downarrow n}(x + r \cdot \cos(\theta), y + r \cdot \sin(\theta)) \cdot e^{-\frac{r^2}{\sigma_{\theta,r}^2}}}{\sum_{\theta=0}^{360} \sum_{r=0}^{r_{max}} e^{-\frac{r^2}{\sigma_{\theta,r}^2}}}, \tag{3.18}$$

where (x,y) is the coordinate of a pixel p and n is chosen such that the larger dimension of the downsampled image $\Lambda_{\downarrow n}$ equals a constant, whose default value is 200.

Considering this second simplification, the computation time of the mask is fixed and is bounded by $\frac{N}{n^2} \cdot \frac{(2r_{max})^2}{n^2}$. That makes the computational complexity of order:

$$O(N + \frac{N}{n^2} \cdot \frac{(2r_{max})^2}{n^2}). \tag{3.19}$$

Transforming the RGB input image into a luminance/chrominance encoding takes $O(N)$ operations but this is compensated by the fact that processing only the luminance requires less computational time than processing an RGB image. Indeed, treating separately the R,G,B channels would multiply by three the complexity of (3.19), which is significantly more time consuming than adding an $O(N)$ operation. The PCA transform costs more in term of computational time than a fixed transform such as YUV. However, the additional time spent to compute the PCA is neglectable compared to the time spent to compute the mask.

3.6 Discussion, Comparisons and Results

In this section, we justify the need for an adaptive filter by showing an example where it helps to prevent halo artifacts. Then, we compare our method to other local tone mapping methods: the MSRCR of Rahman *et al.* (2004), the gradient attenuation method of Fattal *et al.* (2002) and the fast bilateral filtering method of Durand and Dorsey (2002), currently recognized as one of the best published methods (Kuang *et al.* 2004). Finally, we apply the adaptive filter method to a set of images and show the results.

3.6.1 Importance of the Adaptive Filter

Figure 3.12 illustrates the difference between using an adaptive filter that follows the high contrast edges in the image and a non-adaptive filter, whose

shape is circular for every pixel. The two images were computed with exactly the same method except for the filter's shape. The non-adaptive case was computed with an edge map set to zero everywhere, such that the surround's shape is always circular and does not follow high contrast edges.

The benefit of the adaptive filter is clearly shown in Figure 3.12: the details of the tower and in the forest are more visible using the adaptive filter method.

This is due to the edge-preserving properties of the mask as illustrated in Figure 3.13. The use of the adaptive filter method prevents the areas of different intensity to influence areas beyond high contrast edges.

Figure 3.11: Difference between using a PCA and a YUV transform to compute the luminance. The image computed using YUV looks slightly green. **Top:** Image computed using PCA. **Bottom:** Image computed using YUV.

Figure 3.12: The adaptive filter method allows to preserve detail visibility even along high contrast edges. **Top:** Non-adaptive filter method. **Bottom:** Adaptive filter method.

Figure 3.13: The edge-preserving properties of the mask prevents areas of different intensity to influence areas beyond high contrast edges. **Top:** Input image Λ'. **Middle:** Mask with adaptive filter. **Bottom:** Mask without adaptive filter.

3.6.2 Comparison with Other Methods

Our algorithm finds its basis in the MSRCR algorithm of Rahman *et al.* (2004). It is therefore natural to make a comparison with their results. The MSRCR images were obtained with the free version of the software "PhotoFlair" using the default settings[3], which puts "demo" tags across the image. Figure 3.14 and 3.15 show a comparison between MSRCR and our adaptive filter method. The benefit of the adaptive filter is clearly seen in both images. With MSRCR (top image of Figure 3.14), there is a shadow on the person near the window and on the red dog. Moreover, the black t-shirt tends to become gray. Due to the adaptive filter, our method does not generate halos on the face of the person and on the t-shirt. The β factor (3.5) prevents the t-shirt to turn gray. Similarly, in the bottom image of Figure 3.15, the detail of the tower is more visible on the image treated by the adaptive filter method.

The presence of halo artifacts comes from the fact that MSRCR is based on the assumption that the illuminant is spatially smooth. This results in a mask similar to the bottom panel of Figure 3.13, which leads to halos when subtracted from the log-encoded luminance. Other methods that are based on the same smooth illuminant assumption (Land 1986, Rahman *et al.* 2004, Horn 1974, Funt *et al.* 2004) suffer from the same drawback. Nevertheless, they are good at rendering images of lower dynamic range or in the absence of large intensity ratios.

Fattal *et al.* (2002) treat HDR images with a gradient attenuation method. Figure 3.16 shows their result. Their method is very good at increasing local contrast without creating halo artifacts but the effect tends to be exaggerated. A border effect appears on the left of the image and the colors seem unnatural.

---

[3]PhotoFlair was developed by TruView Imaging Company (http://trueview.com)

Figure 3.14: Top: Image treated with MSRCR. **Bottom:** Image treated with the adaptive filter method.

Figure 3.15: Top: Image treated with MSRCR. **Bottom:** Image treated with the adaptive filter method.

Figure 3.16: **Top:** Image treated with Fattal's gradient attenuation method. **Bottom:** Image treated with the adaptive filter method.

3.6.3   Comparison with Fast Bilateral Filtering

We also compared our method to the fast bilateral filtering method developed by Durand and Dorsey (2002) for two reasons. First, it is recognized as one of the best algorithms tested on HDR images that has been published so far (Kuang *et al.* 2004). Second, although the initial approach is different, the actual treatment of pixels is comparable to that of our method. Fast bilateral filtering is based on an alternative of anisotropic diffusion to enhance boundaries while smoothing non-significant intensity variations. The new pixel values are computed by weighting surrounding pixels as a function of their spatial position as well as their intensity difference. Our method computes the new pixel values by weighting surrounding pixels as a function of their spatial position and their spatial relation to high contrast edges. Our method gives more importance to spatial information. Figure 3.17 illustrates this difference. In the case of fast bilateral filtering, the weight of one surrounding pixel is decreased if its intensity is different from that of the intensity of the treated pixel. However, the weight increases again if the intensity of the next pixel is similar to the currently treated pixel. With our method, as soon as a high contrast edge is detected, the weight of the current pixel is decreased as well as that of pixels located after the edge.

Figure 3.17: In the case of fast bilateral filtering, the weight of one pixel is decreased if its intensity is different from that of the intensity of the treated pixel. However, the weight increases again if the intensity of the next pixel is similar to the currently treated pixel. With our method, as soon as a high contrast edge is detected, the weight of the current pixel is decreased as well as that of pixels located after the edge.
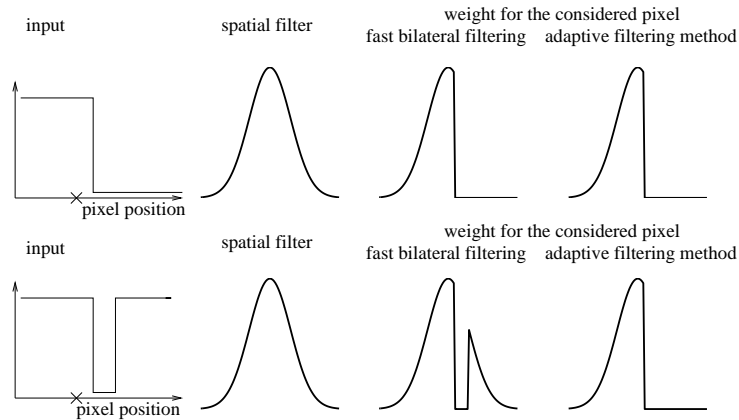
Figure 3.18 compares the images obtained with the two methods. We observe that Durand and Dorsey's method is better at increasing the local contrast

in bright areas while our method provides a better result in dim areas. The head of the person is rendered better by our algorithm. This is due to the adaptive shape of the filter that prevents the sky to influence the color of the face and thus avoids the usual backlight effect. The way the color is rendered also influences the judgment of images. Fast bilateral filtering algorithm renders images that are more saturated than our method. It is suitable for some images but leads to unnatural impression in other cases, such as the reddish skin in Figure 3.18.

3.6.4   Image Acquisition and Results

We applied our algorithm on radiance maps constructed with a multiple exposure technique[4] as well as RAW images. In both cases, the captured images represent HDR scenes. We assume that input images use sRGB primaries (*IEC 61966-2-1 1999*). No color transformation is applied prior to processing. For the images that were generated by the multiple exposure technique (Debevec and Malik 1997), there is no guarantee that the images are in sRGB color space. As many of them are used in the literature, we still use them for comparison. The output of our algorithm are 24 bits/pixel images rendered for standard displays, i.e. the color image encoding is sRGB (*IEC 61966-2-1 1999*).

Figure 3.19 to 3.26 show HDR images treated by our algorithm. They were all obtained with the same parameter values given in this chapter. However, it is possible that an image-dependent tuning of the parameters such as the exponent γ or the adaptive filter spatial constants σ_0 and σ_1 still lead to better visual results.

3.7   Conclusion

The problem of rendering HDR images has been widely studied and a large number of methods exists. These methods enhance the quality of rendered images but still suffer from some problems. Common drawbacks are the apparition of halo artifacts when increasing the local contrast, graying-out of low contrast areas and bad color rendition.

We provide a method to render HDR images taking inspiration from the Retinex model of color vision. Our method is based on center/surround Retinex but uses an adaptive filter whose shape follows the image high contrast edges instead of a circularly-symmetric Gaussian filter. In this way, the influence of applying a local filtering on a bright area located next to a dim area is decreased, thus preventing halo artifacts. We also include a sigmoid function

---

[4]The generation of HDR images was discussed in Appendix A.

that weighs the mask in order to prevent the graying-out of pure white or pure black low contrast areas. The Retinex-based adaptive filter is applied to the luminance channel only, which is defined by the first component of a PCA performed on the input image. Using PCA provides an image-dependent color-space transform that guarantees perfect decorrelation between channels. It minimizes the chromatic changes induced by the processing of luminance.

Figure 3.18: Top: Gamma-encoded image. **Middle:** Image treated with the adaptive filter method. **Bottom:** Image treated with the fast bilateral filtering method.

Figure 3.19: Results of the Retinex-based adaptive filter method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. This image is courtesy of the RIT Munsell Color Science Laboratory.

Figure 3.20: Results of the Retinex-based adaptive filter method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. This image is courtesy of Paul Debevec.

Figure 3.21: Results of the Retinex-based adaptive filter method. **Left:** Gamma-encoded image. **Right:** Image treated with our method. This image is courtesy of Paul Debevec.

Figure 3.22: Results of the Retinex-based adaptive filter method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. This image is courtesy of Clement Fredembach.

Figure 3.23: Results of the Retinex-based adaptive filter method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. This image is from our database.

Figure 3.24: Results of the Retinex-based adaptive filter method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. This image is from our database.

Figure 3.25: Results of the Retinex-based adaptive filter method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. This image is from our database.

Figure 3.26: Results of the Retinex-based adaptive filter method. **Top:** Gamma-encoded image. **Bottom:** Image treated with our method. This image is from our database.

Chapter 4

The Influence of Luminance on Local Tone Mapping

4.1 Introduction

The method presented in Chapter 3 ensures good color rendition by applying the local processing to the luminance channel only. Indeed, many local tone mapping methods do not perform well when applied independently to the three color channels of an RGB image. While treating R, G, and B independently provides good results with global tone mapping methods, local tone mapping algorithms may create artifacts such as local graying-out, hue shifts, or color fringes. This is illustrated in Figure 4.1. The top image was obtained by applying the Multi-Scale Retinex (MSR) algorithm (Jobson *et al.* 1997, Rahman *et al.* 2004) to all three R, G, B channels independently. The middle image was processed similarly with the Retinex-based adaptive filter algorithm discussed in Chapter 3 (Meylan and Süsstrunk 2004*b*, Meylan and Süsstrunk 2006). Processing R, G, and B independently causes a hue shift in both cases.

Treating the luminance independently from the chrominance is a well-accepted solution to avoid these artifacts (Funt *et al.* 1997, Fairchild and Johnson 2004, Durand and Dorsey 2002, Sobol 2004, Meylan and Süsstrunk 2004*b*, Meylan and Süsstrunk 2006). However, none of these publications investigates the influence of the chosen color transform on the appearance of the treated image. The correlation between luminance and chrominance is known to have an influence on the rendered image but the relation between a measure of correlation and the appearance of the image has not yet been found.

Figure 4.1: **Top:** MSR applied to all R, G, B channels (green hue shift). **Middle:** The Retinex-based adaptive filter method applied to all R, G, B channels (pink hue shift). **Bottom:** Original gamma-encoded image.

In this chapter, we investigate the influence of the color transform in the case of luminance-based local tone mapping methods on the rendered image color appearance. In particular, we focus on surround-based Retinex methods that were introduced in Section 2.6.3.

Our aim is to relate a measure of the correlation between luminance and chrominance with the color rendition of images treated by surround-based Retinex methods. We consider four color transforms and define a measure to evaluate how well they decorrelate luminance and chrominance. Then, we test two Retinex-based local tone mapping algorithms with the four different color transforms and relate the results with our measure. We show that there is a relation between the visual representation of the rendered image and the measure of correlation. Color artifacts become visible when luminance and chrominance are significantly correlated.

This chapter is structured as follows: Section 4.2 reviews background work about color rendition in the case of local tone mapping algorithms. Section 4.3 presents our measure and the four color transforms that we consider. Section 4.4 presents the two algorithms used for the test. Then, Section 4.5 comments the images obtained with the two algorithms and the different color transforms. A conclusion is given in Section 4.6.

4.2 Background

The way color is processed by tone mapping methods has already been discussed in the literature. Rahman et al. (Jobson *et al.* 1997, Rahman *et al.* 2004) commented on the graying-out effect of surround-based Retinex algorithms and added a color restoration step to their MSR algorithm. The MSRCR was studied by Funt and Barnard (Funt *et al.* 1997, Barnard and Funt 1998). They argued that MSRCR tends to desaturate the colors, due to the averaging operation on small neighborhoods that have a graying-out effect on the image. Moreover, the color restoration step added to compensate for the loss of saturation can at best approximate the color that was removed and acts in an unpredictable way. Funt and Barnard thus suggested applying MSR to the luminance channel only. The treated luminance is then combined with the chrominance to obtain the final color image. They defined the luminance as the average of the three color channels R, G, B. With this definition of luminance, some chromatic information remains in the luminance and vice-versa, which may lead to artifacts.

In Chapter 3, we presented a Retinex-based method that applies an adaptive filter to the luminance channel only. The luminance is defined by a PCA computed over the RGB input image.

iCAM, developed by Fairchild and Johnson (2004), also applies a local treatment only to the luminance channel to avoid desaturating the image. This method was described earlier in Chapter 2.

Kimmel et al. developed their quadratic programming optimization algorithm for a monochrome image (Kimmel *et al.* 2003, Elad *et al.* 2003). They apply it separately either to the three color channels of an RGB image or to the V channel of an HSV-encoded image. They found that the first approach could lead to exaggerate color shifts or to a loss of saturation. As previously found (Barnard and Funt 1998, Funt *et al.* 1997), applying it only to the V channel yields better results.

Sobol (2004) also applies his Retinex-based algorithm to the luminance channel. Unlike previously mentioned methods that define the luminance as the weighted sum of R, G, B color channels, his luminance definition is given by the maximum between these three channels. The final color image is obtained by adding the new luminance to the log-encoded RGB image.

Thus, many local tone mapping methods first transform the input image into a luminance/chrominance representation and treat the luminance only:

$$\{\Lambda, C_1, C_2\} = M_{cs} * \{R, G, B\}, \tag{4.1}$$

where $\{\Lambda, C_1, C_2\}$ is the luminance/chrominance opponent representation of $\{R, G, B\}$. M_{cs} is defined by the color transform considered.

In most cases, the luminance is defined by a weighted average of R, G, B color channels (Jobson *et al.* 1997, Rahman *et al.* 2004, Meylan and Süsstrunk 2004*b*, Funt *et al.* 1997, Fairchild and Johnson 2004, Durand and Dorsey 2002, Barnard and Funt 1998), with the exception of Sobol' s method (Sobol 2004). Then, the final RGB image is obtained either by converting the luminance/chrominance image back to RGB (4.2) or by using a scaling technique where the ratio of the initial luminance and the treated luminance multiplies the three color channels (4.3). The particular case of Sobol' s method adds the treated luminance to the log-encoded RGB image (4.4).

$$\{R, G, B\}_{new} = M_{cs}^{-1} * \{\Lambda_{new}, C_1, C_2\} \tag{4.2}$$

$$\{R, G, B\}_{new} = \frac{\Lambda_{new}}{\Lambda} \cdot (\{R, G, B\}) \tag{4.3}$$

$$\{R, G, B\}_{new} = \Lambda_{new} + (\{\log(R), \log(G), \log(B)\}) \tag{4.4}$$

Here "·" and "+" are component per component operations, and "∗" is a matrix multiplication.

In this chapter, we study the effect of different luminance definitions on the rendered image for the case of MSR (Jobson *et al.* 1997, Rahman *et al.* 2004) and Retinex-based adaptive filtering of Chapter 3 (Meylan and Süsstrunk 2004*b*, Meylan and Süsstrunk 2006).

4.3 A Measure of Correlation

The four color transforms that we chose for our tests are described in Table 4.1. Each of them transforms the linear RGB input image in a luminance/chrominance encoding (4.1). The first one, "M_{RGB}" transform, simply defines the green channel G as being the luminance and the red and blue channels R, B as being the chrominance. With this transform, the luminance is strongly correlated with the chrominance. The second transform is "M_{YUV}", which is a linear transform widely used for video processing (Poynton 2003). The third one "M_{Lab}" is the CIE definition of opponent color and is not a linear transform (Hunt 1995). The last one "M_{PCA}" is an image-dependent, linear transform based on a PCA applied to the input image. It guarantees perfect decorrelation between components. The luminance is defined by the first principal component.

Table 4.1: The four color transforms tested.

	M_{cs}	Luminance Λ	Chrominance C_1, C_2	Transform
RGB	Linear	G	R,B	M_{RGB}
YUV	Linear	Y	U,V	M_{YUV}
Lab	Non-Linear	L	a,b	M_{Lab}
PCA	Linear	Λ: 1st principal component	C_1: 2nd principal component C_2: 3rd principal component	M_{PCA}: defined by the eigen vectors of the input's covariance matrix

We define a simple correlation measure μ obtained by computing the mean of the correlation coefficients between the luminance and the chrominance channels over a set of representative images $\mathcal{S}$:

$$\mu = \frac{1}{J} \sum_{j \in \mathcal{S}}^{J} \mu_j, \qquad (4.5)$$

where J is the number of representative images in the set $\mathcal{S}$.

The correlation measure for one image j is given by the average correlation between luminance and chrominance channels:
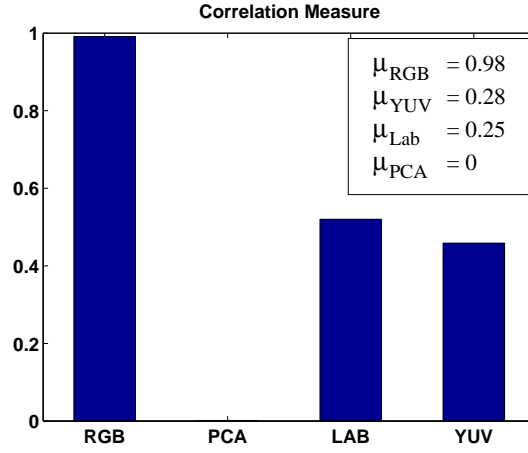
Figure 4.2: μ for M_{RGB}, M_{PCA}, M_{Lab}, M_{YUV}.

$$\mu_j = \frac{corr(\Lambda_j, C_{j,1}) + corr(\Lambda_j, C_{j,2})}{2}, \tag{4.6}$$

where the correlation between luminance and chrominance is defined by the normalized covariance (4.7),(4.8).

$$cov(\Lambda_j, C_{j,c}) = \frac{1}{N} \sum_{p=1}^{N} (\Lambda_j(p) - \overline{\Lambda}) \cdot (C_{j,c}(p) - \overline{C_{j,c}}) \tag{4.7}$$

$$corr(\Lambda_j, C_{j,c}) = \frac{cov(\Lambda_j, C_{j,c})}{\sqrt{cov(\Lambda_j, \Lambda_j) \cdot cov(C_{j,c}, C_{j,c})}} \tag{4.8}$$

In (4.7) and (4.8), $C_{j,c}$ represents one of the two color channel $C_{j,1}$ or $C_{j,2}$. N is the number of pixels in the image j.

Figure 4.2 shows the measure of correlation of the four considered color transforms. Uncorrelated data results in a correlation coefficient of 0; equivalent data sets have a correlation coefficient of 1. It shows how well the color transforms decorrelate the luminance from the chrominance. Our aim is to see how this measure is related to color rendering. In particular, we want to test if M_{PCA}, which is image dependent and guarantees perfect decorrelation between components, leads to the best reproduction.

4.4 Two Local Tone Mapping Methods

Two surround-based local tone mapping are used for our tests. For the first one, we implemented the MSR algorithm of Rahman et al. (Jobson *et al.* 1997, Rahman *et al.* 2004). In these articles, the MSR is applied separately to the R, G, B color channels. A color restoration factor is added. Here, we do not use the color restoration factor but apply MSR only to the luminance channel as suggested by (Funt *et al.* 1997, Barnard and Funt 1998). The second tested tone mapping method is our Retinex-based adaptive filter method (Meylan and Süsstrunk 2004*b*, Meylan and Süsstrunk 2006). They were both described in Chapters 2 and 3.

We apply these two local tone mapping algorithms to a set of test images. Each input image is first converted from RGB to a luminance/chrominance opponent representation using the color transforms of Table 4.1. Then, each of the local tone mapping methods is applied to the luminance channel. Finally, the chrominance channels and the new luminance are transformed back to RGB encoding. In the next section, we visually compare the images obtained with different color transforms and algorithms and comment them.

4.5 Results and Discussion

Figure 4.4 and 4.5 shows the results obtained with MSR and the four color transforms defined in Table 4.1. It shows that different transforms, thus different luminance definitions, result in different output images. The images obtained using M_{RGB} and MSR algorithm clearly show a pink color shift, more apparent in the center of the image. This is due to the fact the G channel is strongly correlated with the R and B channels. Then, there is little visible difference between the images obtained with M_{Lab}, M_{YUV} and M_{PCA}. The graying-out of low contrast areas such as the sky and the clouds is due to the local averaging induced by Retinex surround-based methods. As already mentioned, the Retinex-based adaptive filter method prevents graying-out by introducing a factor that weighs the mask depending on the input image values. This, in addition to the saturation compensation factor, results in visually more appealing images.

Figure 4.6 and 4.7 show the same image treated by our Retinex-based adaptive filter method and the four color transforms. The image computed with M_{RGB} also presents a color shift (green). We can observe as well that M_{YUV} and M_{PCA} lead to images with a better increase in local contrast than M_{Lab}. In other words, the detail of the central part of the image is more visible. Moreover, the sky is slightly more saturated with M_{PCA} and M_{YUV} than with

M_{Lab}[1]. These differences may come from the fact that the M_{Lab} transform is non-linear with respect to the scene radiances. We see no difference between the image computed using M_{YUV} and the image computed using M_{PCA}.

If we compare the results of M_{PCA} and M_{YUV} on another image (Figure 4.8), color differences appear. In Figure 4.8, the top image (M_{YUV}) appears greener than the bottom image (M_{PCA}). This hue shift is confirmed by the a, b chromaticity plane of these two images (Figure 4.3). The image treated using M_{YUV} is plotted in cyan, while the image treated using M_{PCA} is plotted in magenta. The cyan cloud is shifted to the left of the magenta cloud, i.e. the image treated using M_{YUV} tends to appear greener than the image treated using M_{PCA}. This causes the face of the person to look slightly green on the M_{YUV} image.

Figure 4.4 to 4.8 show that the choice of the luminance definition on which a local tone mapping algorithm is applied plays a role for the image appearance. However, a small correlation between luminance and chrominance does not affect significantly the final result. M_{RGB} had the worst decorrelation measure and the images obtained using this transform clearly present color shift artifacts. M_{YUV} and M_{Lab} had good decorrelation measure but not as good as M_{PCA} that ensures perfect decorrelation between components. The fact that the luminance of Lab-encoded image is non-linear induces some differences with the M_{YUV} and M_{PCA} images. However, the images computed using M_{YUV} and M_{PCA} are very similar, which makes it difficult to judge which transforms leads to the best resulting image.
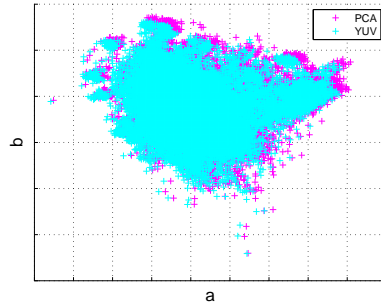

Figure 4.3: Plot of the a,b chromaticities of the two images of Figure 4.8. The image treated using M_{YUV} is plotted in cyan while the image treated using M_{PCA} is plotted in magenta.

---

[1] As printing the images might introduce other color shifts, we suggest the reader looks at the images on a display.

4.6 Conclusion

The goal of our study was to investigate the role of the luminance definition on the final image appearance in the case of luminance-based local tone mapping algorithms. For that purpose, we tested two algorithms and four color transforms (M_{RGB}, M_{YUV}, M_{Lab} and M_{PCA}). A measure of correlation was established for these four transforms. The M_{RGB} transform had highly correlated components. M_{YUV}, M_{Lab} were slightly correlated and M_{PCA} was designed to ensure perfect decorrelation for all images. We observed that there were little visible differences between the images treated using M_{YUV}, M_{Lab}, and M_{PCA}. Local color shifts started to appear when using a transform where luminance and chrominance are highly correlated such as M_{RGB}. That suggests that there is a relationship between the amount of correlation between luminance and chrominance, and the quality of the image appearance. However, a perfect decorrelation is not necessary to obtain visually pleasing images.

Figure 4.4: Results obtained with MSR. **Top:** M_{RGB}. **Bottom:** M_{Lab}.

Figure 4.5: Results obtained with MSR. **Top:** M_{YUV}. **Bottom:** M_{PCA}.

Figure 4.6: Results obtained with the adaptive filter method. **Top:** M_{RGB}. **Bottom:** M_{Lab}.
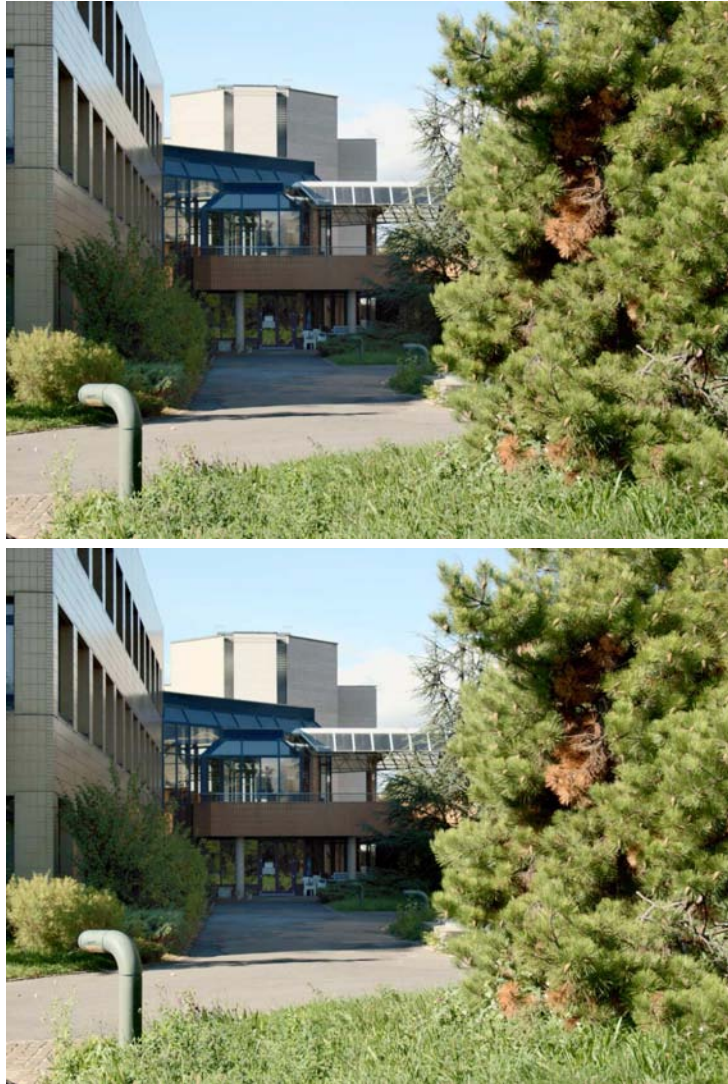
Figure 4.7: Results obtained with the adaptive filter method. **Top:** M_{YUV}. **Bottom:** M_{PCA}.

Figure 4.8: Results obtained with the Retinex-based adaptive filter method. The top image (M_{YUV}) appears greener than the bottom image (M_{PCA}).

Chapter 5

The Reproduction of Specular Highlights on High Dynamic Range Displays

5.1 Introduction

HDR monitors capable of displaying simultaneously bright highlights and dark shadows have just started to come on the market. As their usage will increase, we will be confronted with the problem of re-rendering images that have been mapped to SDR displays so that they look "good" on HDR monitors. A straightforward solution is to use a linear scaling from the input image range to the HDR display range. Instead of a linear scaling, we propose a tone mapping method that enhances the specular parts of the image. We validate its use with a psychovisual experiment.

This chapter starts with a short description of the HDR display (Section 5.2). The software correction algorithm to display images on an HDR display (Seetzen *et al.* 2004) is described in Appendix C. Then, Section 5.3 presents our proposed tone scale, which is evaluated by a psychovisual experiment. The generation of stimuli, the experimental set up and procedure are described in Section 5.4. Measurement performed on the HDR display in the experimental conditions are presented in Section 5.5. Finally, Section 5.6 explains the statistical analysis applied on the experiment data and discusses the results.

5.2   High Dynamic Range Display Technology

Current flat panels LCD are composed of a uniformly bright backlight and an LCD array. These displays have a contrast ratio of about 1:400, which is not sufficient to display scenes of high dynamic range such as outdoor scenes. With HDR displays (Seetzen *et al.* 2004), the uniform backlight is replaced by an array of white light emitting diodes (LEDs). These LEDs can be as bright as 200,000 cd/m^2 at their maximum capacity and emit no light when turned off. Each LED is controlled individually and lights up a small area of the LCD array through a diffusion screen. Multiplying the modulation of the LCD and LED arrays provides a gain in dynamic range.

Practically, HDR displays can reach a contrast ratio up to 1:25,000, depending on the ambient light. This enables rendering HDR scenes without having to compress the dynamic range much. It results in an impressive and highly realistic representation of HDR scenes. Figure 5.1 shows a picture of the small HDR display prototype. A larger display (37 inches) is now also available.

Figure 5.1: HDR display prototype. This Figure was taken from (Seetzen et *al.* 2004).

5.3   A Tone Mapping Method to Enhance the Representation of Specular Highlights

The input to HDR displays are generally radiance maps encoded in the "hdr" format (Appendix A). These radiance maps represent an accurate representation of the scene radiances. However, most available images are already mapped to SDR displays and have lost part of their dynamic range. In particular, specular highlights are badly reproduced due to strong luminance compression and/or clipping taking place during the image capturing and rendering process

to SDR. As they offer important visual cues about three dimensional shapes and increase the sense of realism (Interrante *et al.* 1997, Blake and Bülthoff 1991), it would be beneficial to use part of the extension of dynamic range of HDR displays to enhance their representation.

Figure 5.2 illustrates luminance profiles of a diffuse (a.) and glossy (b.) half-sphere illuminated by a point light source. The narrow region with high luminance is referred to as specular highlight. A specular highlight is the reflection of a light source on a non-lambertian surface, such as metal, glossy paint, etc. It occurs when the specularly reflective surface is illuminated by a light source that has the same half-angle as the camera, i.e. the surface normal is located exactly halfway between the direction of incident light and the camera. Figure 5.2 (c.) and (d.) show the luminance profile of a glossy half-sphere illuminated under the same point light source after it was captured by a camera. The effects of compression (c.) and clipping (d.) are illustrated.
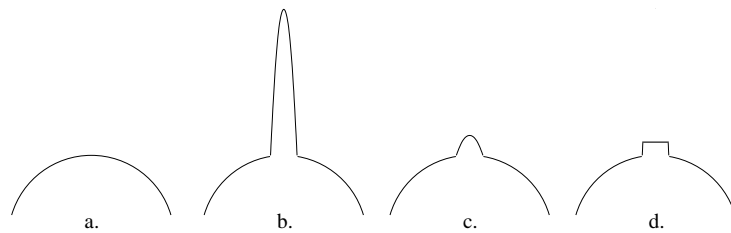

Figure 5.2: Luminance profiles of a half-sphere illuminated by a point light source. **a.** Diffuse surface. **b.** Glossy surface. **c.** Glossy surface captured by a camera, example of compression. **d.** Glossy surface captured by a camera, example of clipping.

The presence of specular highlights in an image suggests that the original scene had a high dynamic range, as specular highlights can be several orders of magnitude brighter than diffuse highlights (Wolff 1994). We address the problem of rendering images mapped to SDR and representing HDR scenes to HDR displays. We propose the use of a tone scale function that expands the luminance range allocated to the specular parts of an image with the goal of recovering the natural look of the original HDR scene.

In a psychovisual experiment, we test different tone scale functions by varying the display luminance range allocated to specular highlights. We prove that allocating some of the additional display range provided by an HDR monitor to specular highlights leads to a more natural displayed image than using a simple linear scaling of code values. In addition, the proposed tone scale prevents the re-rendered image to look too bright, which is likely to happen when applying just a linear scaling (illustrated in Figure 5.3).
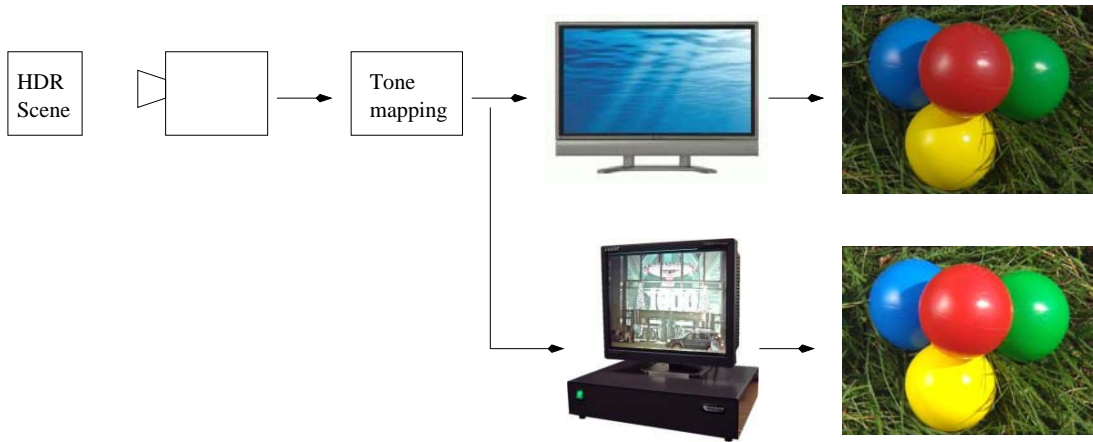
Figure 5.3: Simulation of re-rendering SDR images to HDR displays. If a simple linear scaling is applied, the image can appear too bright.

5.3.1 The Tone Scale Function

Input Images and Pre-processing

The images used for our psychovisual experiment were either standard images rendered to SDR display or radiance maps built with a multiple exposure technique, clipped to a standard dynamic range, and stored in a ".hdr" file format. The set of images is shown in Figure 5.4. A pre-processing step ensures that the input to the algorithm is an image that is a linear representation of the scene radiances.

Most of our images are legacy images, i.e. images whose original capture device is unknown. We assume that these images are encoded in sRGB as most SDR images from consumer cameras currently are. We thus apply the inverse color component transfer function specified by the sRGB standard (*IEC 61966-2-1 1999*). For images built with a multiple exposure technique and clipped to standard dynamic range, we assume that they are already a linear representation of the scene radiances.

An image resulting from this pre-processing step is linear and encoded with sRGB primaries. It is noted I or equivalently $\{R, G, B\}$.

From RGB to Luminance

Our tone scale function is applied only to the luminance channel to avoid color artifacts. We thus transform our RGB encoded image into a luminance/chrominance representation (5.1) and only process the luminance Λ. We chose to use the NTSC color transform M_{ntsc} (Poynton 2003) but any
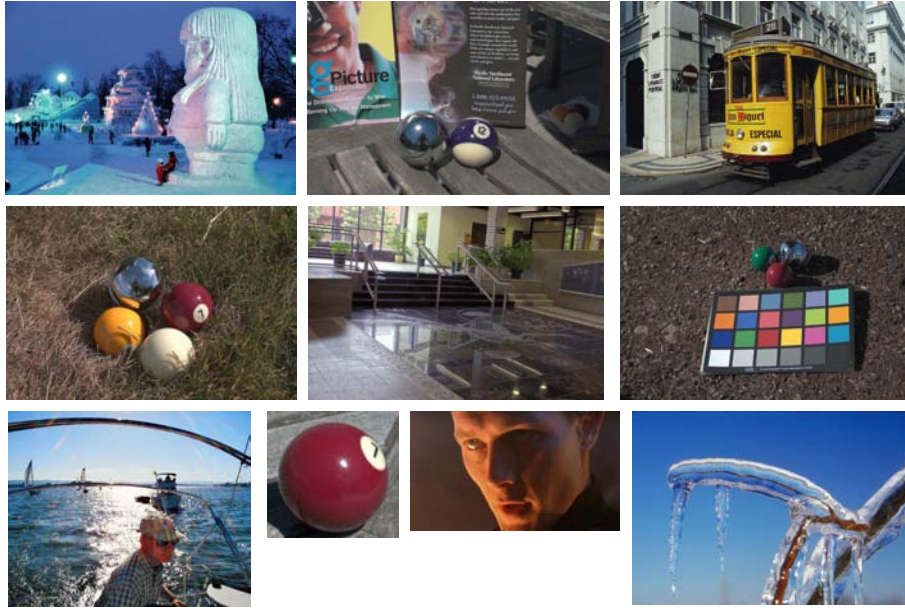
Figure 5.4: Set of images used in the experiment.

transform that converts an image encoded with sRGB primaries into a luminance/chrominance representation would work as well (see Chapter 4).

$$\{\Lambda, C_1, C_2\} = M_{ntsc} * \{R, G, B\}, \tag{5.1}$$

where

$$M_{ntsc} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix}, \tag{5.2}$$

Λ is the luminance channel, C_1, C_2 are the chrominance channels, and $\{R, G, B\}$ is the linear input RGB image.

Construction of the Tone Scale Function

The luminance channel is processed by our tone scale function so that more dynamic range is allocated to the specular image than that allocated in the SDR input. It is a piecewise linear function composed of two slopes (Figure 5.5).

The shape of the tone scale is entirely defined by ω, the normalized code value of the maximum diffuse white in the image, and ρ, the percentage of the maximum display luminance allocated to ω.

ω is determined by segmenting the image into its diffuse and specular com-
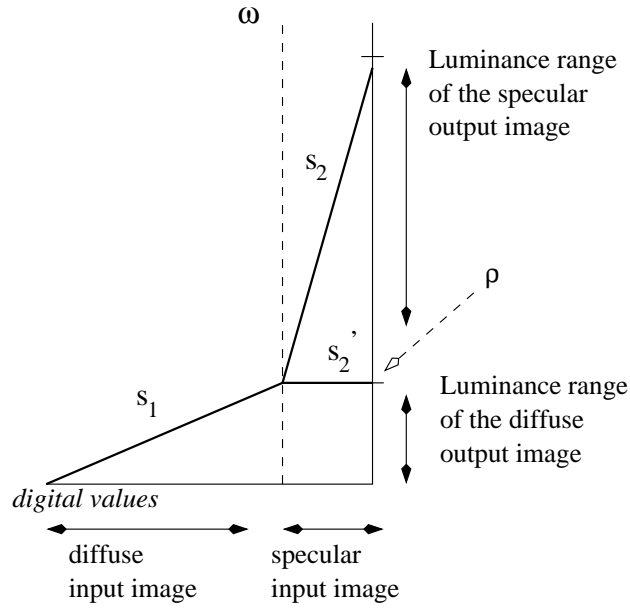
Figure 5.5: Piecewise linear tone scale function.

ponents, which we call "diffuse image" and "specular image," respectively. The specular image is composed of the parts of the image that contain specular highlights. The diffuse image can include glossy and non glossy objects and is composed of the rest of the image that is not part of the specular image. Figure 5.6 gives an example of a segmentation. The minimum digital value of the specular image defines the maximum diffuse white ω. For the psychovisual experiment, the segmentation was done manually for each image. A way to automatically segment the image and compute ω is described in Chapter 6. Since small variations of ω do not change the general shape of the tone scale, the results of the psychovisual experiment will remain valid for the automatic algorithm.

ρ is the parameter tested in the psychovisual experiment. It varies for each tone-scaled image. The tone scale function f is defined as follows:

$$f(\Lambda(p)) = \begin{cases} s_1 \cdot \Lambda(p) & \text{if } \Lambda(p) \leq \omega \\ s_1 \cdot \omega + s_2 \cdot (\Lambda(p) - \omega) & \text{if } \Lambda(p) > \omega \end{cases} , \qquad (5.3)$$

where

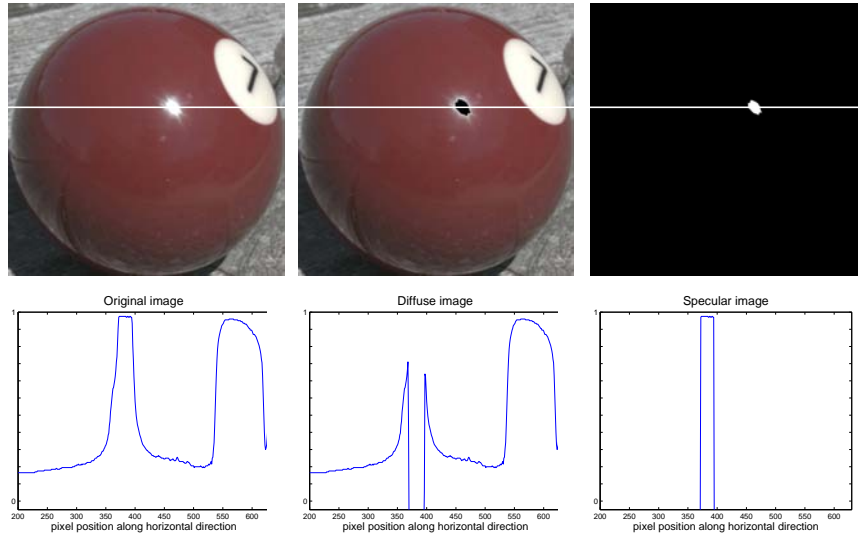$$s_1 = \frac{\rho}{\omega}, \qquad (5.4)$$

Figure 5.6: Example of an image segmentation into its specular and diffuse components. The white line in the top three images represents the position of the traces in the bottom graphs. **Top left:** Original image. **Top center:** Diffuse image. The specular part of the image is filled with black. **Top right:** Specular image. The diffuse part of the image is filled with black. **Bottom left:** Horizontal trace in the original image. **Bottom center:** Horizontal trace in the diffuse image. **Bottom right:** Horizontal trace in the specular image.

$$s_2 = \frac{1 - \rho}{\Lambda_{max} - \omega}. \tag{5.5}$$

Λ is the normalized luminance and p is a pixel in the image. The maximum digital value of the input image is noted Λ_{max}. Λ_{max} is normalized to 1 to make the method independent of the digital code value range.

The shape of the tone scale (Figure 5.5) allows the allocation of more dynamic range to the specular image than that allocated in the SDR input (horizontal axis). All pixels of the input image whose normalized code values are smaller than ω are considered being part of the diffuse image and are scaled by s_1. s_2 has a steeper slope and is used to scale the specular image defined by pixels having a value greater than ω.

We added a clipped version of the tone scale where the specular highlight maximum value is not matched to the maximum display luminance ($s_2\prime$ in Figure 5.5). This enables us to test if participants preferred specular highlights clipped or enhanced given a particular overall image brightness.

From Luminance back to a Color Image

After applying the tone scale to the luminance, the RGB image is recovered by multiplying the original RGB color channels by the ratio of the treated luminance Λ_{new}, where $\Lambda_{new}(p) = f(\Lambda(p))$, and the original luminance Λ (5.6). The result is scaled to the theoretical maximum display luminance called Ψ_{max}. For the display used in the experiment (Brightside 37"), $\Psi_{max} = 2500 \ cd/m^2$. Practically, this value is reached when measuring a large white patch. With smaller areas such as specular highlights, Ψ_{max} tends to have a lower value. However, the effect of our tone scale function remains valid as long as its general shape is conserved, i.e. s_1 remains smaller than s_2. This is the case for all but extremely small specular highlights, as discussed in Section 5.5.

$$I_{new} = \{R, G, B\}_{new} = \frac{\Lambda_{new}}{\Lambda} \cdot \{R, G, B\} \tag{5.6}$$

5.4 Experimental Set Up

5.4.1 Stimuli Preparation

For each tested scene, different tone-scaled images are constructed by varying the luminance allocated to the diffuse white. We tested four different values of ρ varying from 20% to 67% of Ψ_{max} using logarithmic increments, as well as a linear scaling. The tone scale functions used in the experiment are shown in Figure 5.7. Table 5.1 shows the corresponding ρ values. For tone scales 1 to 4, ω is matched to 20, 30, 47, and 67 percent of Ψ_{max}, while the maximum code value of the input image is matched to Ψ_{max}. Tone scale 5 corresponds to linear scaling. For one of these tone scales ($\rho = 0.47$), we constructed a clipped version, where the maximum code value of the input image is matched to $\rho \cdot \Psi_{max}$.

Table 5.1: Tone scales used in the experiment.

	1	2	3	4	5	6
Percentage of max. display luminance allocated to ω	0.2	0.3	0.47	0.67	lin.	0.47 clipped

For the non-clipped tone scales (1 to 5), changing the value of ρ affects both the image global brightness and the reproduction of specular highlights. The more luminance range is allocated to the diffuse image, the brighter the image appears while simultaneously decreasing the range allocated to specular
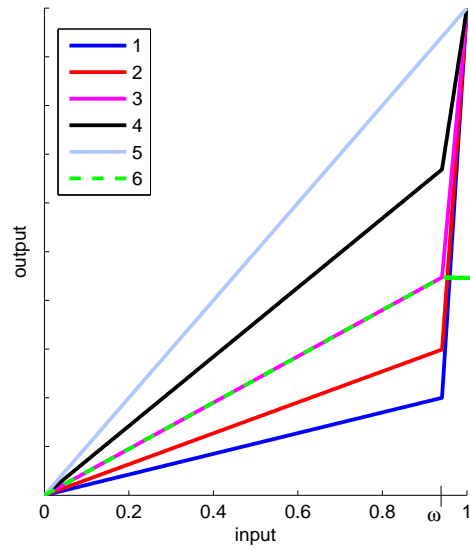
Figure 5.7: Illustration of the 6 tone scale functions used in the psychovisual experiment ($\omega = 0.94$).

highlights. A smaller luminance range allocated to the diffuse image causes the image to look dimmer and the specular highlights to look brighter. Figure 5.8 illustrates these two cases.

Small range allocated to diffuse image

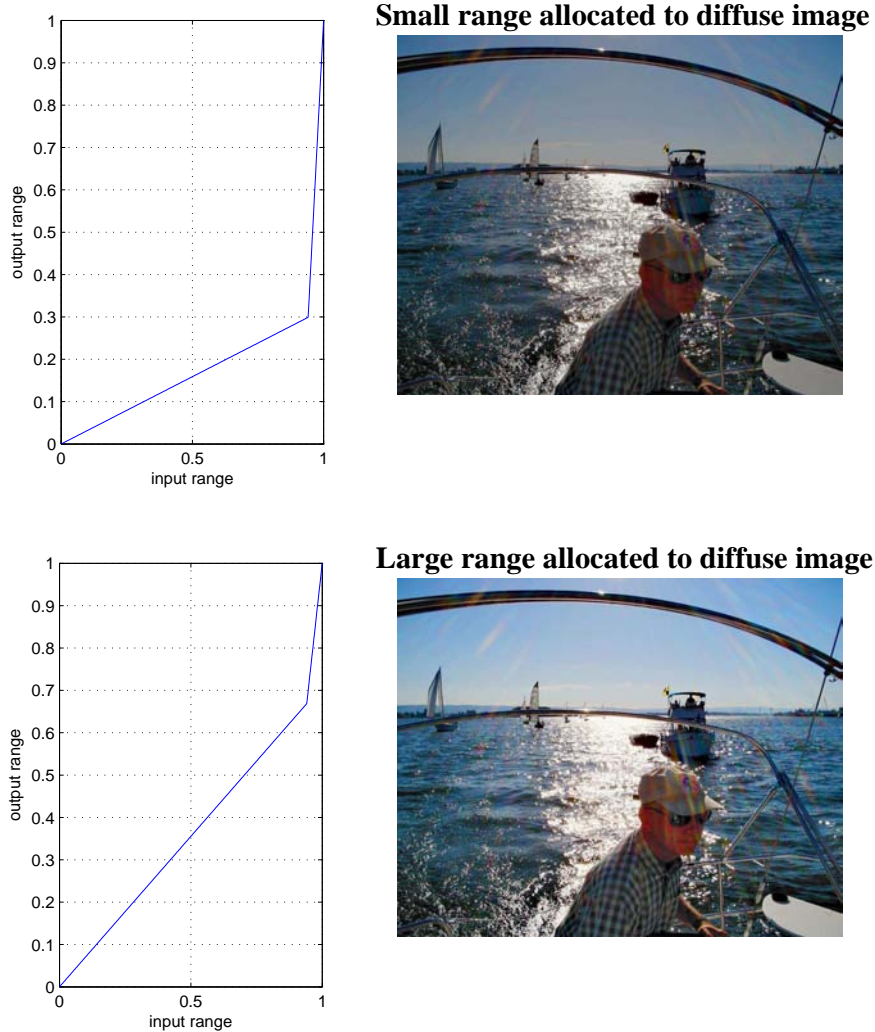
Large range allocated to diffuse image

Figure 5.8: Example of tone scale functions for two different input parameters. The top image corresponds to the case where a small range is allocated to the diffuse image. The bottom image corresponds to a larger range. ω for this image was 0.94.

A Smoothing Technique to Remove Unnatural Contours

The discontinuity in the tone mapping function may produce unnatural contours, which influence the participants' judgment in an undesirable way. We added a smoothing step to our algorithm to overcome this problem. Our solution is to introduce a slight blur around each specular highlight, thus removing unnatural contours.
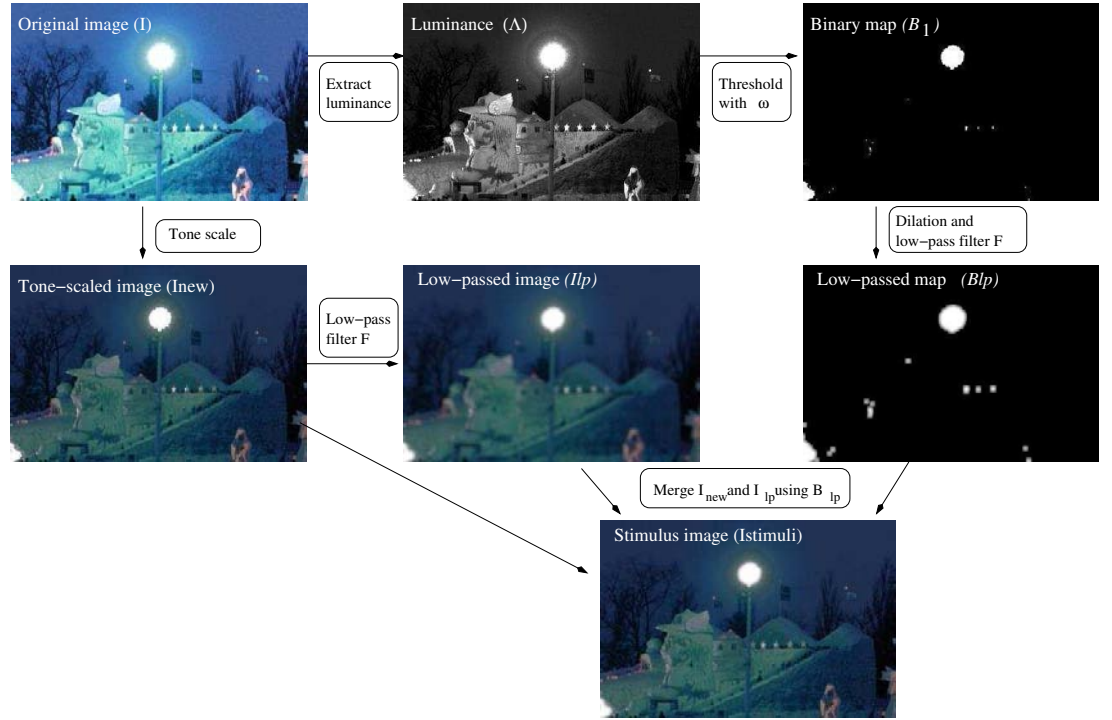

Figure 5.9: Illustration of the smoothing technique to remove unnatural contours (see text for explanation).

The principle of the smoothing step is to smooth the contour around specular highlights by applying a low-pass only at the location of specular highlights (Figure 5.9). First, a thresholding operation is applied to the luminance channel of the original image Λ to find the location of specular highlights:

$$\mathcal{B}_1(p) = \begin{cases} 0 & \text{if } \Lambda(p) \leq \omega \\ 1 & \text{if } \Lambda(p) > \omega. \end{cases} \tag{5.7}$$

$\mathcal{B}_1(p)$ is a binary map, where 1 indicates the presence of a specular highlight. p is a pixel in the image I.

Then, a dilation is applied to $\mathcal{B}_1(p)$. A pixel is turned from 0 to 1 if more than one of its neighbors are 1:

$$\mathcal{B}_2(p) = \begin{cases} 1 & \text{if } \mathcal{B}_1(p) = 1 \text{ or } (\mathcal{B}_1(p) * \mathcal{H} > 1) \\ 0 & \text{otherwise,} \end{cases} \tag{5.8}$$

where

$$\mathcal{H} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}. \tag{5.9}$$

Both the tone-scaled image I_{new} and the binary map $\mathcal{B}_2(p)$ are then low-passed with a moving average filter of size 5 denoted by $\mathcal{F}_5$.

$$\begin{aligned} I_{lp}(p) &= I_{new}(p) * \mathcal{F}_5 \\ B_{lp}(p) &= B_2(p) * \mathcal{F}_5 \end{aligned} \tag{5.10}$$

The final image $I_{stimuli}$ where contours are removed is simply computed by merging the tone-scaled image I_{new} and the low-passed tone-scaled image I_{lp} using the blurred mask B_{lp} as a matrix of weighting coefficients.

$$I_{stimuli}(p) = I_{new}(p) \cdot (1 - B_{lp}(p)) + I_{lp}(p) \cdot B_{lp}(p) \tag{5.11}$$

An example of application of the smoothing technique on a zoomed version of the red pool ball image is given in Figure 5.10. The largest benefit of this method is seen when a low luminance value is allocated to diffuse white (e.g. $\rho = 0.2$). The contour of the specular highlight looks too sharp in the top left image. Its smoothed version at the bottom left looks more natural. The difference between the tone-scaled image and the smoothed tone-scaled image becomes less visible as ρ increases. The pairs of images on the right look more similar.

The Generation of Pairs of Tone-Scaled Images

The images thus processed are presented in pairs to the observers. Each image in the pair is computed by a different tone scale. Prior to the experiment, all possible combinations of pairs of images generated with the tone scale functions are computed.

The number of possible pairs N_{pair} generated by T number of tone scale is given by

$$N_{pair} = \frac{T \cdot (T-1)}{2} \tag{5.12}$$

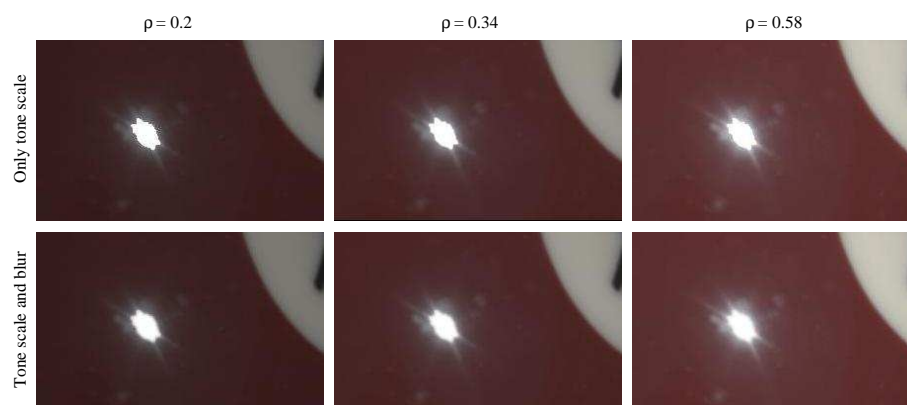In our case, $T = 6$ and $N_{pair} = 15$ for each tested image.

Figure 5.10: An example of application of the smoothing technique on a zoomed version of the red pool ball image.

The two tone-scaled images composing a pair are scaled and stored as another image having the resolution of the HDR display (1980×1280). A black border of 80 pixels (1.3 degree of visual angle) separates them. We experimented with different border sizes and empirically found that 1.3 degree was a good size to prevent the brightness of one image from influencing the color of the other one, which would influence the observer judgment in an uncontrolled way. The left/right position of the tone-scaled images is chosen randomly. Examples of stimuli pairs are shown in Figure 5.11.

5.4.2 Experimental Procedure

A computer program displayed pairs of scaled images in random order. For each image of the test set, 15 pairs were presented. The 15 pairs of the next image were shown until all images from the test set have been used.

Then, the process was repeated one more time with a different image sequence. The pairs for one image were still displayed randomly. The left and right position of the tone-scaled images which was random for the first sequence, was swapped.

Each time a pair was displayed, the observer used the keyboard to select an image according to the following question, which they could read on an information sheet:

Which image looks more natural (i.e. more like a real scene, like real lighting)? Focus on the tone reproduction; try not to be influenced by other factors (contouring, noise, etc).
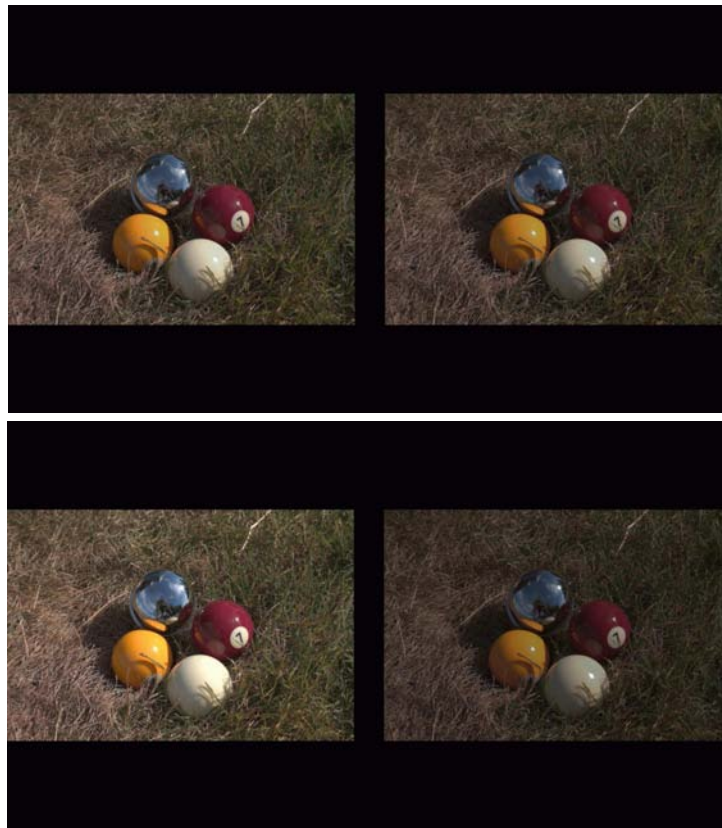
Figure 5.11: Example of stimuli shown in pairs.

Observers

20 observers participated in the experiment. 14 were naive observers, and 6 were experts in judging image quality, including 2 that had some knowledge about the purpose of the experiment. Each of them saw 330 images, which took about 25 minutes.

Viewing Conditions

The experiment was set up in a room with no window. The lights were on, which created an ambient luminance of 22 cd/m^2. The images were displayed on the Brightside's 37" HDR monitor. Observers sat at a viewing distance of three times the display height, which resulted in a total viewing angle of 33 degrees.

5.5   Measurements Performed on the HDR Display

The maximum displayed luminance of our HDR monitor was obtained by displaying and measuring a large white patch. However, for very small bright areas, this measured value can not be reached. This is due to the characteristics of the HDR display and to the software that provides the conversion between the ideal tone-scaled image (input to the HDR monitor) and the image displayed at the screen. The displayed image is the multiplication of the images sent to the LEDs and to the LCD panel, respectively. The algorithm that computes these two images is explained in Appendix C. One important part of this algorithm is the cross-talk correction, which computes the LED driving values. The goal of the cross-talk correction is to compensate for the fact that the luminance measured at one LED physical position is not only due to one LED but also to remaining light emitted by neighboring LEDs. The model used for cross-talk is limited to only six neighboring LEDs, which correspond to direct neighbors. However, the light contribution of other surrounding LEDs, which are not direct neighbors is not null. When measuring a large white patch, these LEDs, not taken into account by the cross-talk correction, tend to increase the measured value. Therefore, a small bright area on a dim background suffers from the fact that there is not enough contribution coming from neighboring LEDs and can not reach the maximum displayed value. Consequently, the measured luminances at the screen differ from what is intended by the tone scale function applied to the image.

To understand this behavior better, we measured white patches of varying sizes (simulating specular highlights) using a spectrophotometer (Pr650). We used patches of 8, 16, 32, and 64 pixels corresponding to 0.14, 0.27, 0.55, and 1.1 degrees of visual angle.

Backgrounds of varying gray levels (black, $0.1 \cdot \Psi_{max}$, $0.5 \cdot \Psi_{max}$) were used to simulate the luminance allocated to the maximum diffuse white. Example of generated images are shown in Figure 5.12.

Figure 5.12: Generated images for measurements. **Left:** Background: 50% of Ψ_{max}, Specular highlight size: 1.1 degree. **Right:** Background: 10% of Ψ_{max}, Specular highlight size: 0.55 degree.

Measurements of these simulated specular highlights are plotted in Figure 5.13. We observe that the smaller the specular highlight is, the lower is the display luminance. Moreover, the luminance of the area surrounding the specular highlight also influences the practical measured value. The darker it is, the lower the specular highlight measured value is.
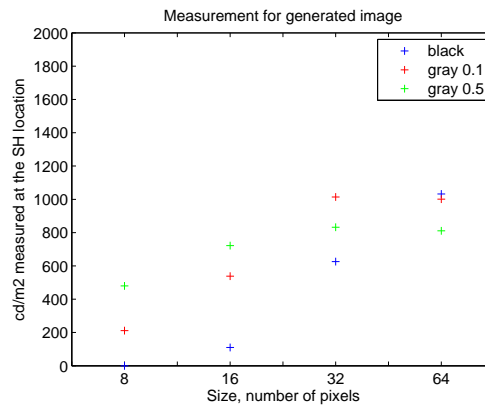

Figure 5.13: Measurements for generated images. The horizontal axis shows four different sizes of specular highlights. Each color corresponds to a different background luminance value.

Consequently, the practical applied tone scale varies locally and depends on the size of the specular highlights as well as on the luminance value allocated to the maximum diffuse white.

Figure 5.14 illustrates a theoretical tone scale and two corresponding practical tone scale functions for different specular highlight sizes. With large specular highlights, the practical tone scale approaches the intended behavior. However, with small specular highlights, it is possible that the measured luminance of a large diffuse white area exceeds that of a specular area, despite the behavior intended by the tone scale function.

This display limitation has an influence on the type of images that need to be chosen for the psychovisual experiment. Images with small specular highlights can not be used to validate our proposed tone scale. Based on our measurements, we consider that the diameter of a specular highlight must be at least 16 pixels for the results to be meaningful.
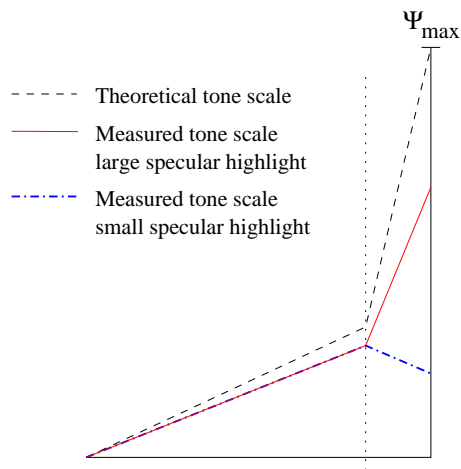
Figure 5.14: Illustration of a theoretical tone scale and two corresponding practical tone scale functions. With large specular highlights, the practical tone scale approaches the intended behavior. With small specular highlights, the measured luminance of a large diffuse white area exceeds that of the specular area.

5.6 Statistical Analysis

5.6.1 Interval Scale of Preferences

Thurstone law of comparative judgment Case V (Engeldrum 2000) was applied to convert the paired comparison observer data into an interval scale of preferences. We used the toolbox provided in (Green 2003), which calculates the z-scores and confidence intervals from such data.

With Thurstone law of comparative judgment, unanimous judgments (i.e. when a stimuli is preferred by all observers or no observer) are problematic as corresponding z-value are undefined. This problem is referred as "zero proportion matrix problem". Is it solved by substituting missing z-values using a linear regression technique.

Results

The interval scale of preferences along with 95 % confidence limits are shown in Figure 5.15. For two tone mapping to be significantly different, their error bars must not overlap. Tone scale number 6 is the clipped version of tone scale number 3. Their diffuse brightness is the same but specular highlights of tone scale 6 are not boosted up.

The six plots represent six different images that we selected from our set to give representative results. Indoor and outdoor scenes are shown. The term "prefer" is used to describe observer choice. However, it is important to

remember that it relates to a sensation of naturalness.

For the two images at the top, participants significantly preferred tone scale 4 over a simple linear scaling (5). At equal brightness (tone scales 3 and 6), they selected the tone scale with bright specular highlights (3) significantly more than the clipped one (6).

For images c, d, and e, our tone scale is slightly preferred than linear scaling but not statistically different. At equal brightness, the bright specular highlight images are statistically judged to be better than the clipped highlights. For these three images it seems that participants expected a very bright scene while images a and b benefit from a low luminance allocated to diffuse white. For image a, which is an indoor scene, observers do not expect a very bright image. For image b, the lower luminance choice can be explained by some capture artifacts becoming more visible as the image becomes brighter.

Image f is an example of a problematic image. We showed with the measurements performed on the HDR monitor that small specular highlights were not scaled as much as predicted due to some display limitations. Consequently, in image f, the increase in luminance of the specular highlights performed by the tone scale function could not be displayed on the screen. This explains why tone scale 3 and 6 are statistically equivalent.
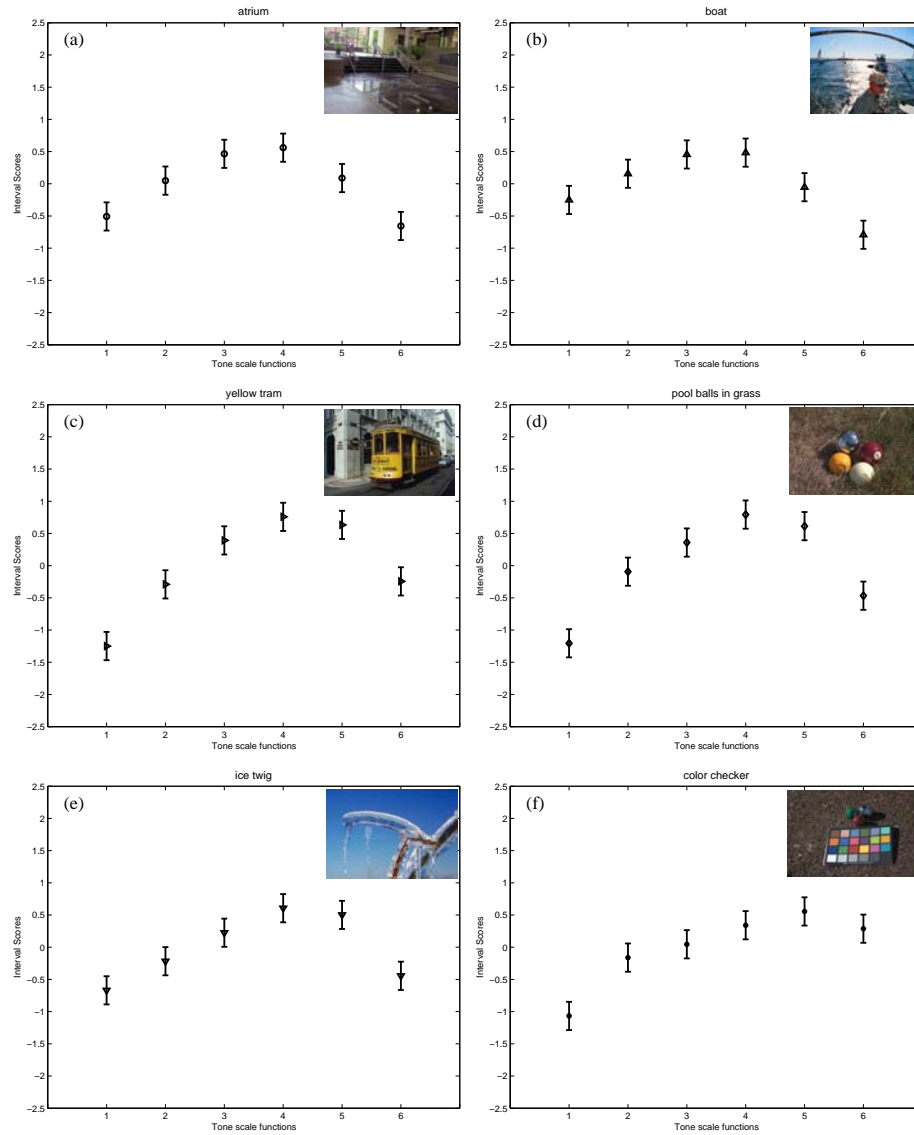
Figure 5.15: For images a and b, participants significantly preferred tone scale 4 over a simple linear scaling (5). At equal brightness (tone scales 3 and 6), they selected the tone scale with bright specular highlights (3) significantly more than the clipped one (6). For images c, d, and e, our tone scale is slightly preferred than linear scaling but not statistically different. At equal brightness, the bright specular highlight images are statistically judged to be better than the clipped highlights. For image f, tone scale 5 (linear) is statistically preferred over 1,2,3. Clipped (6) and non-clipped tone scales (3) are equivalent. These results are due to the small size of specular highlights.

5.6.2 Coefficient of Agreement and Consistency

In addition to the interval scale of preference plots, we tested observer agreement and consistency following the method described in (Kendall and Smith 1940, Ledda *et al.* 2005).

Agreement between observers

Agreement between observers was measured using Kendall coefficient of agreement (Kendall and Smith 1940). Let Q be the number of observations (the number of observers $\times$ the number of repetitions), T be the number of tested tone scale functions, and q_{ij} be the number of times tone scale i was preferred over tone scale j. We first compute

$$\Sigma = \sum_{i=1, j=1, i \neq j}^{T} \binom{q_{i,j}}{2} \tag{5.13}$$

Then, the coefficient of agreement between observers is given by:

$$u = \frac{2\Sigma}{\binom{T}{2} \cdot \binom{Q}{2}} - 1 \tag{5.14}$$

Complete agreement between observers leads to $u = 1$. The value of u if observers were making their selection randomly is given by:

$$u_{random} = \begin{cases} -\frac{1}{Q-1} & \text{if Q is even} \\ -\frac{1}{Q} & \text{if Q is odd} \end{cases} \tag{5.15}$$

In our case, $u_{random} = -0.0256$.

We then performed a test χ^2 to assess the statistical significance of u. Our H0 hypothesis was "There is no agreement among observers". H0 can be rejected when χ_t^2 with $\frac{(T \cdot (T-1))}{2} = 15$ degrees of freedom is larger than the quantile of the χ_{15}^2 distribution (5.16).

$$\chi_t^2 = \frac{T(T-1)(1+u(Q-1))}{2}$$
$$\chi_{15}^2 = 25 \tag{5.16}$$

Coefficient of Consistency

The coefficient of consistency is determined for one image and one observer repetition by the number of circular triads in the data. The number of circular triads in paired comparison data is given by ν (Kendall and Smith 1940).

$$\nu = \frac{T}{24}(T^2 - 1) - \frac{1}{2}\iota, \tag{5.17}$$

where

$$\iota = \sum_{i=1, i \neq j}^{T} (\sum_{j=1}^{T} q_{i,j} - \frac{T-1}{2})^2. \tag{5.18}$$

The coefficient of consistency is then

$$\eta = \begin{cases} 1 - \frac{24\nu}{T^3 - 4T} & \text{if Q is even} \\ 1 - \frac{24\nu}{T^3 - T} & \text{if Q is odd} \end{cases} \tag{5.19}$$

Full consistency leads to $\eta = 1$ while total inconsistency leads to $\eta = 0$.

Results

Table 5.2 shows the agreement and consistency coefficients for images of Figure 5.15. Since our experiment does not lead to a typical ranking of the tone scale functions, it is possible for observers to disagree on some pairs while agreeing on which tone scale provides the most natural overall look. Therefore, global agreement between observers is rather small but still passes the χ^2 test of statistical significance. The H0 hypothesis "There is no agreement among observers" can be rejected for all images of Table 5.2. The average consistency is high for all images.

Table 5.2: Observer agreement and consistency coefficients. For each image, the coefficient of agreement passes the χ^2 test ($\chi^2 > 25$) ensuring a non random agreement between observers.

image name	obs. agr. u	obs. cons. η	χ^2	image name	obs. agr. u	obs. cons. η	χ^2
atrium (a)	0.16	0.82	111	boat (b)	0.15	0.83	105
tramway (c)	0.32	0.81	205	pool balls (d)	0.35	0.89	217
ice twig (e)	0.20	0.80	130	color check.(f)	0.21	0.81	140

In Table 5.2, we computed the consistency coefficient for each image averaged over the observers' responses. We now compute the consistency coefficient for each observer averaged over the set of images. This allows us to detect if one observer was completely inconsistent. Table 5.3 presents the average consistency of all observers. It shows that all observers were consistent in their judgment.

Finally, we test intra-observer agreement and consistency for two observers who did 4 repetitions of the experiment. The results show that the agreement coefficient over the four repetitions was high. The consistency coefficient was good as well.

Table 5.3: Average consistency of observers over all images.

Obs	1	2	3	4	5	6	7	8	9	10
Av. cons.	0.81	0.69	0.91	0.93	0.84	0.88	0.81	0.91	0.79	0.96
Obs	11	12	13	14	15	16	17	18	19	20
Av. cons.	0.83	0.84	0.69	0.79	0.69	0.83	0.81	0.94	0.91	0.73

Table 5.4: Intra-observer agreement with 4 repetitions, for two observers.

	Observer 1	Observer 2
Average agreement	$u = 0.60$	$u = 0.5$
Average consistency	$\eta = 0.86$	$\eta = 0.79$
χ^2	72	68

5.6.3 Discussion

The results of this experiment show that the preferred luminance range allocated to the diffuse image varies with the image content. Different results are obtained for indoor scenes and outdoor scenes. For outdoor scenes, observers tend to select images where only a small part of the dynamic range is allocated to specular highlights. However, with images of equal diffuse brightness, they prefer the image with bright highlights.

For indoor scenes, the participants clearly prefer to allocate more range to the specular highlights instead of a linear scaling, which would result in an unnaturally bright image. When comparing images of equal diffuse brightness, the image with bright specular highlights is also significantly preferred.

At the end of the experiment, each observer had to answer some questions and could write down their comments. An example of the form given to participants is shown in Figure 5.16.

In response to question 1, most observers reported occasional artifacts but almost none of them were disturbed by them. Answers to the second question were "no" for most participants, which confirms that 1.3 degree of visual angle for the black line separating the two images was sufficient. Concerning question 3, some comparisons were reported difficult. They correspond to the images with very small specular highlights where it was really difficult to see any difference in the highlights. For these images, the answers were still consistent if participants agreed on the global diffuse brightness that the rendered image should have. We put one of these images in the results (image f). In general, observers' feed-back was coherent with the statistical results. Finally, more than two third of the participants found that the duration of the experiment was acceptable. Some reported fatigue while looking at the last few images.

Questions and comments

When you are done with the experiment, please write down your
comments and answer the following questions

1) Did you notice artifacts in the specular highlights?

Often   Occasionally   Never

2) Did you feel like a very bright image next to a dark image influenced
your preferences in other way than the tone scale itself?

3) Were some comparisons too difficult?

4) Was it too long?

You comments:

Figure 5.16: Example of form given to the participants.

Nevertheless, our analysis of the observers' consistency did not allow us to
discard any participants from our data analysis.

To conclude, the use of a tone scale that boosts up the specular high-
lights instead of rendering a globally brighter image is validated for indoor
scenes. Most importantly, the results of the comparison between clipped and
non-clipped specular highlights in images of equal diffuse brightness confirmed
that bright specular highlights lead to a more natural impression for all tested
images.

5.7 Conclusion

The recent marketing of HDR displays opens new research opportunities in the
field of HDR imaging as well as related applications. This chapter focuses on
the conversion of SDR images (whose original scenes were HDR) into images
that can be displayed on an HDR monitor. We present a tone scale function
whose goal is to improve the realism of specular highlights. The benefit of such
a tone scale is confirmed by a psychovisual experiment.

This experiment suggests that when using an HDR display, it is preferable
not to use the entire dynamic range for the diffuse component of the input
image despite the reduction in mean brightness. Instead, part of the dynamic
range could be used to provide a better reproduction of specular highlights
and thus increase the realism of the displayed image. More importantly it
shows that at equal diffuse brightness, images with brighter specular highlights
provide a more natural visual sensation.

Chapter 6

The SDR to HDR Algorithm

6.1 Introduction

For the psychovisual experiment described in the previous chapter, the maximum diffuse white of an image, which defines the shape of the tone scale function, was chosen manually. In this chapter, we propose an automatic way to construct the tone scale. The maximum diffuse white ω is determined using an automatic segmentation of the input image into a diffuse and specular image (Figure 5.6). This segmentation is performed by detecting the specular highlights in the image.

First, background work on the detection of specular highlights is reviewed. Then, we introduce our method to find the specular components in an image, compute ω, and construct the tone scale. An alternative way of applying the tone scale function using spatial information is proposed and compared to the global application used in the psychovisual experiment. Finally, we show the results of applying our tone scale using the automatic detection of ω on the images used in the experiment.

6.2 Detection of Specular Highlights

Detecting specular highlights, which often means separating an image into diffuse and specular components, is not an easy task. It has been addressed by many researchers for various purposes. Klinker *et al.* (Klinker *et al.* 1987, Klinker *et al.* 1988) separate diffuse from specular components using the distribution of color clusters in color spaces. They show that diffuse and specular surfaces form a "T" in the dichromatic plane of the RGB color space. The base of the "T" points to the color of the illuminant (Figure 6.1). However,

this does not hold for saturated pixels since these pixels do not generally obey the characteristics of the dichromatic reflection model.
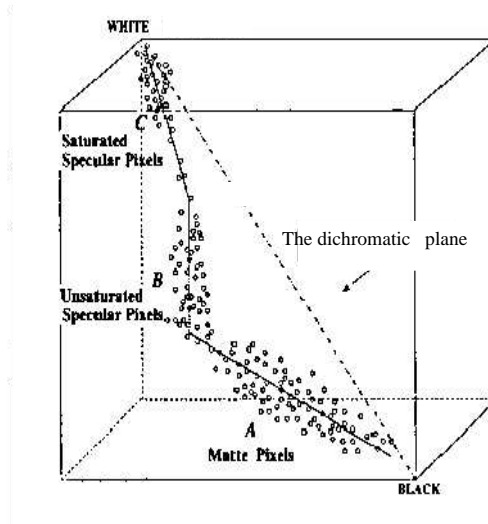

Figure 6.1: Example of "T" shape formed by diffuse and specular pixels (A,B). This figure was taken from (Park and Kak 2003) and also shows some clipped pixels (C).

The specular highlights in an image are often clipped and this creates problems for most detection algorithms based on the dichromatic model. In (Tan *et al.* 2004), the goal is to estimate the illuminant using specular highlights. The first part of their method, based on earlier work (Lee 1986), consists in finding the specular highlights by thresholding on brightness and saturation. They then estimate the illuminant using an inverse chromaticity space in which the correlation between illumination chromaticity and image chromaticity can be obtained. No special care is taken for clipped pixels. A method to discard clipped pixels is proposed in (Park and Kak 2003) so that the distribution obeys the characteristics of the dichromatic model. Osadchy *et al.* (2003) use specularities for recognition of a known 3D object in an image. They detect specular highlight candidates using a threshold operation but also use 3D information of the known object to reject false candidates. They use additional assumptions that are worth mentioning as our technique is also based on them: specular highlights are usually monotonically decreasing and rotationally symmetric. Similar to these methods, the algorithm presented in the next section uses thresholding to discriminate between the diffuse and the specular components of an image. A key difference is that our threshold is determined

by spatial filtering and morphological operators, which makes the detection of specular highlights more robust.

6.3 Automatic Detection of Maximum Diffuse White

Our algorithm detects the specular highlights by applying two low-pass filters to the luminance channel of the input image, combined with morphological operators. The luminance image is obtained by the pre-processing described in Section 5.3.1. The detection of specular highlights results in a binary map where 1 indicates a pixel belonging to a specular highlight. Figure 6.2 illustrates the process.
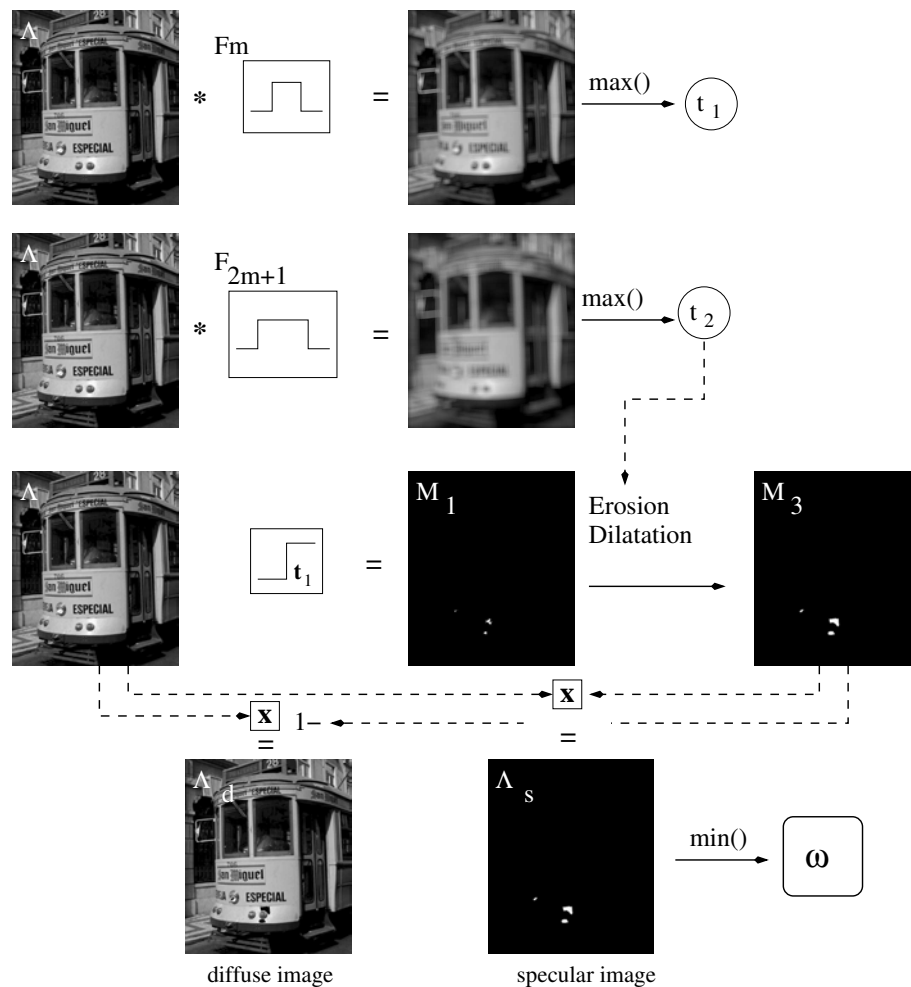

Figure 6.2: Definition of maximum diffuse white ω using two low-pass filters and morphological operators.

Assuming that the specular highlights are small and bright in the image, a low-pass filtered image corresponds to the diffuse version of the image. However, it can occur that white diffuse parts of the image are brighter than the region bordering specular highlights. We take that into account by using two low-pass filters of different sizes. The luminance channel Λ of the input image I is low-passed by a first filter $\mathcal{F}_m$. Taking the maximum digital value of the low-passed image results in a first threshold t_1 (6.1) that is used to define the specular highlight candidates under the form of a binary map $\mathcal{M}_1$ (6.2).

$$t_1 = \max_{p \in I}(\Lambda(p) * \mathcal{F}_m), \tag{6.1}$$

where p is a pixel in the image I and $\mathcal{F}_m$ is an $m \times m$ averaging filter. We use m equals $\frac{1}{50}$ of the image size in the vertical direction.

$$\mathcal{M}_1(p) = \begin{cases} 0 & \text{if } \Lambda(p) \leq t_1 \\ 1 & \text{if } \Lambda(p) > t_1 \end{cases} \tag{6.2}$$

Independently of this first step, the input image is low-passed again by a filter that is twice as large $\mathcal{F}_{2m+1}$. The maximum of the second low-pass filtered image defines a second threshold t_2:

$$t_2 = \max_{p \in I}(\Lambda(p) * \mathcal{F}_{2m+1}). \tag{6.3}$$

The specular highlight candidate map $\mathcal{M}_1$ obtained in (6.2) is now processed with morphological operators. At first, an erosion is applied in order to remove single pixels (6.4). After this first step, we consider that each group of white pixels corresponds to the center of a specular highlight. Dilation is performed iteratively on the specular highlight candidate map $\mathcal{M}_2$ constrained by the second threshold t_2. That is, a pixel is turned from non-specular to specular if more than three of its neighbors are specular and if $\Lambda(p)$ is greater than t_2 (specular regions are shown white in Figure 6.2). Iterations continue until a stable state is reached. Usually only a few iterations are necessary. The sequence of morphological operators is given by the following equations:

$$\mathcal{M}_2(p) = \begin{cases} 0 & \text{if } \mathcal{M}_1(p) = 0 \text{ or } \mathcal{M}_1(p) * \mathcal{H} < 1 \\ 1 & \text{otherwise} \end{cases}, \tag{6.4}$$

where

$$\mathcal{H} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}. \tag{6.5}$$

$$\mathcal{M}_3(p) = \begin{cases} 1 & \text{if } \mathcal{M}_2(p) = 1 \text{ or } (\mathcal{M}_2(p) * \mathcal{H} > 3 \text{ and } \Lambda(p) > t_2) \\ 0 & \text{otherwise} \end{cases} \qquad (6.6)$$

The binary map thus obtained can be used as a mask to discriminate between diffuse and specular image. Λ_d and Λ_s are respectively described as:

$$\Lambda_s(p) = \Lambda(p) \cdot \mathcal{M}_3(p), \qquad (6.7)$$

$$\Lambda_d(p) = \Lambda(p) \cdot (1 - \mathcal{M}_3(p)). \qquad (6.8)$$

We can then define a threshold ω given by the minimum digital value of the specular image Λ_s, which in turn determines the maximum diffuse white of the image. The advantage of using the minimum value of the specular image rather than the maximum value of the diffuse image is that it does not depend on the presence of a white surface in the image.

$$\omega = \min_{p \in P}(\Lambda_s(p)), \qquad (6.9)$$

where P is the set of pixels for which $\Lambda_s(p) > 0$.

6.3.1 Application of the Segmentation Method

In Figure 6.3, we show the results of segmentation into diffuse and specular components (Λ_s, Λ_d) for images used in the psychovisual experiment. The detected specular highlights (given by the specular image) are shown in red. Small and middle size specular highlights are detected successfully.

The detection of specular highlights is dependent on the size of the filter $\mathcal{F}_m$. Using two filters brings some flexibility but detecting large bright highlights remains a problem. In the presence of such a bright area, the maximum of both low-pass filtered images equals the maximum of the input image. ω takes the value of the maximum display luminance, which results in a linear tone scale. An extension to our algorithm for bright large light sources should be considered as future work.

6.4 Applying the Tone Scale Function using a Mask

Knowing ω, we can compute the tone scale function as described in Section 5.3.1. For the psychovisual experiment, the tone scale was simply applied globally to each pixel of the image (6.10). Nevertheless, due to the way ω is
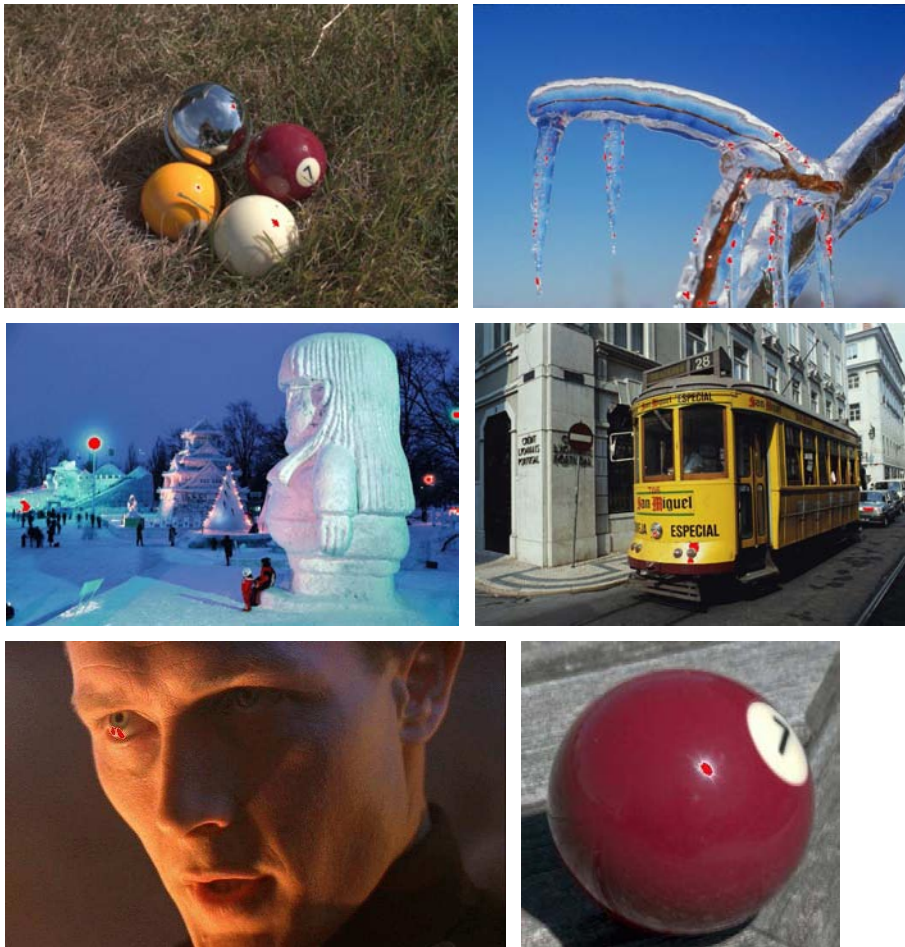
Figure 6.3: Result of the segmentation for images used in the experiment. The detected specular highlights are shown in red. Small and middle size specular highlights are detected successfully.

computed using morphological operators, there could be diffuse parts of the image whose pixel values are greater than ω. In this case, it is advantageous to use the mask $\mathcal{M}_3$ (6.6) to scale the diffuse image and the specular image differently. The diffuse image is scaled by the first linear function even for pixels whose values are greater than ω and the specular image is scaled by the second linear function. This is illustrated in Figure 6.4. We call it local application of the tone scale. Global and local application of the tone scale are given by (6.10) and (6.11), respectively.

$$\Lambda_{new}(p) = f(\Lambda(p)) \tag{6.10}$$

$$\Lambda_{new}(p) = \begin{cases} s_1 \cdot \omega + s_2 \cdot (\Lambda(p) - \omega) & \text{if } \mathcal{M}_3(p) = 1 \\ s_1 \cdot \Lambda(p) & \text{if } \mathcal{M}_3(p) = 0 \end{cases} \qquad (6.11)$$
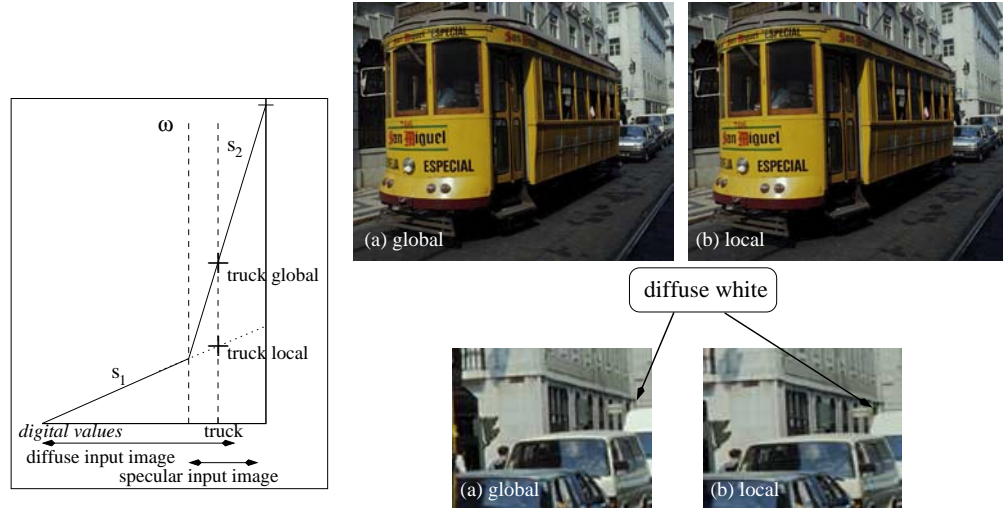

Figure 6.4: Comparison between global and local application of the tone scale. The truck is scaled to a lower value by the local application and appears darker than if it was scaled by the global method since it is not part of the specular image.

Figure 6.4 shows a comparison between global and local application of the tone scale function. For the local application, the diffuse image is scaled with the dotted slope (s_1) while the specular image is scaled by the solid slope (s_2). The difference can be seen in the zoomed images on the roof of the white truck. The truck is part of the diffuse image but has a luminance value greater than ω. It is scaled to a lower value when the tone scale is applied locally than when it is applied globally and appears darker in the image treated with the local method.

Two other examples are given in Figures 6.5 and 6.6. The top left image (a) shows the image scaled by our tone scale applied locally using the specular highlight candidate map (c). The top right image (b) shows the image scaled globally, not taking spatial information ($\mathcal{M}_3$) into account. Diffuse pixels whose value is greater than the computed ω are also boosted up. Differences between (a) and (b) are visible in the area around the number in the pool ball image (Figure 6.5) and in some parts of the big snow statue in Figure 6.6. Another way of illustrating these differences is to compute the binary map $\mathcal{M}_d(p)$ (6.12). In $\mathcal{M}_d(p)$, all pixels whose value is greater than ω is set to 1, as illustrated by the bottom images (d) of Figure 6.6 and 6.5.

$$\mathcal{M}_d(p) = \begin{cases} 0 & \text{if } \Lambda(p) \leq \omega \\ 1 & \text{if } \Lambda(p) > \omega \end{cases} \tag{6.12}$$

Local application   Global application

Figure 6.5: Comparison between local and global application of the tone scale function. A global application enhances the brightness of some diffuse pixels while a local application prevents that.

6.5 Combining the Automatic Detection of ω and the Tone Scale Function

We combined the automatic detection of ω with the local application of the tone scale and present the results in Figure 6.7 to 6.10. The only definitive way to assess the quality of the tone-scaled images is to compare them visually on an HDR display. For this thesis, we simulated an HDR display by increasing the brightness of the displayed images, but the enhancement of specular highlights

Local application   Global application
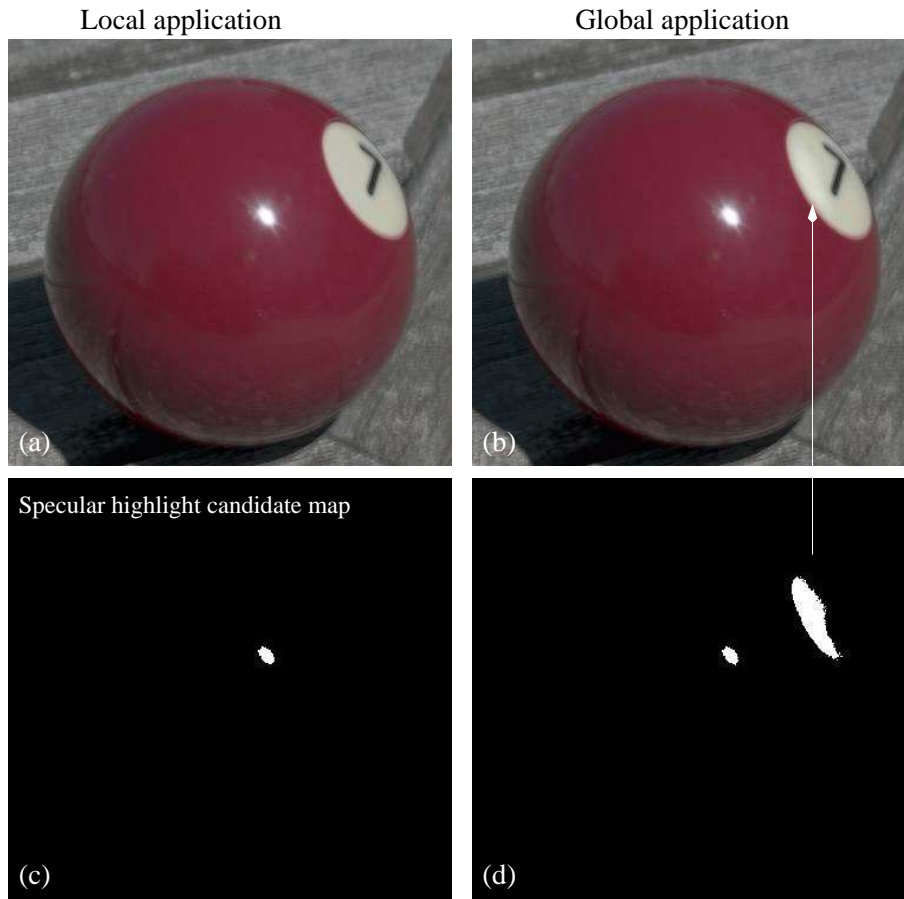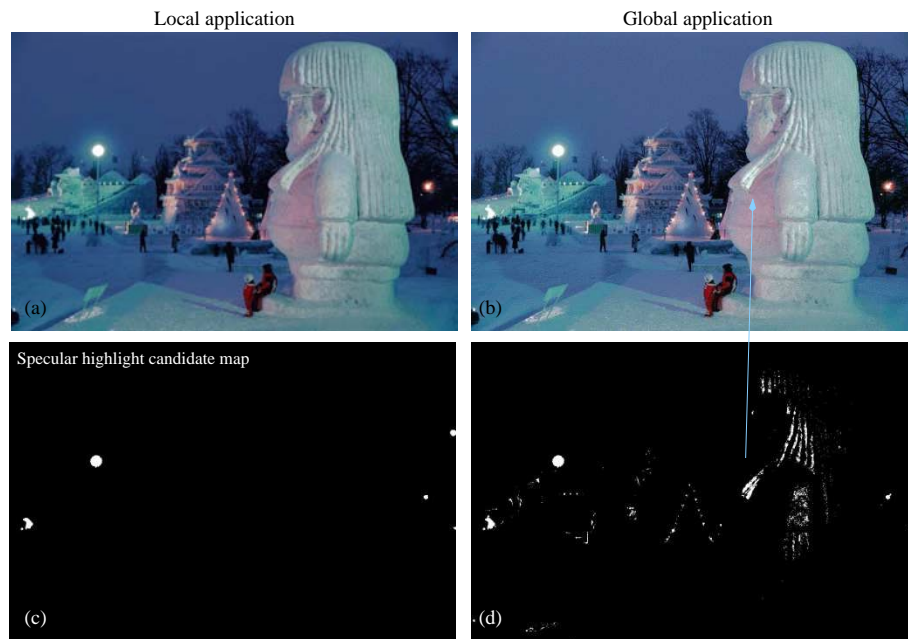

Specular highlight candidate map

Figure 6.6: Comparison between local and global application of the tone scale function. A global application enhances the brightness of some diffuse pixels while a local application prevents that.

can not be reproduced here.

Figure 6.7 to 6.10 compare images originally tone-mapped to SDR display that are scaled linearly (left, a and c), with images treated by our piecewise linear tone scale function (right b and d). According to the results of the psychovisual experiment, we use $\rho = 67\%$. The first row (a,b) shows what happens if the images are shown on an SDR device or a print. We can see that the specular highlights of (b) have been boosted but at the expense of global brightness. The second row (c,d) simulates what would happen if these images were displayed on an HDR monitor. This is done by increasing the brightness of the images (a,b) similarly. Image (d) gives about the same impression of brightness than image (a). The brighter specular highlights of image (d) will only be visible when displayed on an HDR monitor. The simulation of the linearly scaled image displayed on an HDR device (c) shows that the image looks too bright due to the increase in brightness.

6.6 Conclusion

The tone scale presented in Chapter 5 is entirely defined by two parameters, the maximum diffuse white of the input image and the percentage of display

luminance allocated to it. This chapter presents a method to define automatically the maximum diffuse white of an input image. We also propose a more effective way of applying the tone scale, which treats the diffuse and specular images differently. ρ was chosen based on the results of the experiment but remained fixed for all images. Changing ρ automatically according to the image content is suggested for future work.

Figure 6.7: Comparison between linear scaling (left) and our tone scale (right). First row: images displayed on an SDR device or print. Second row: Simulation of images (a,b) when displayed on an HDR device. Only the diffuse brightness of these images is simulated. The bright specular highlights of image (d) can not be reproduced.

Figure 6.8: Comparison between linear scaling (left) and our tone scale (right). First row: images displayed on an SDR device or print. Second row: Simulation of images (a,b) when displayed on an HDR device. Only the diffuse brightness of these images is simulated. The bright specular highlights of image (d) can not be reproduced.
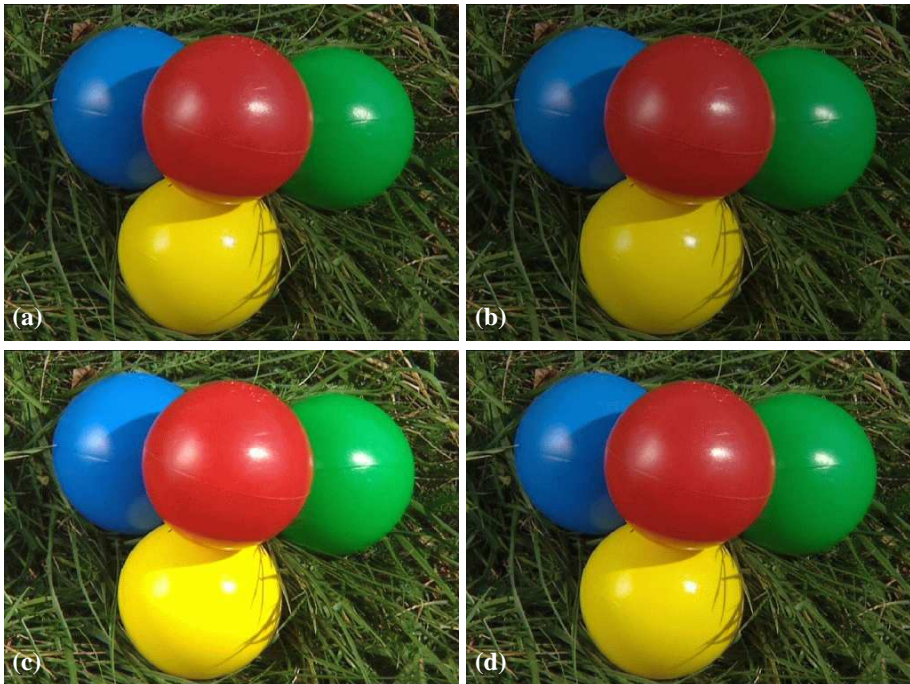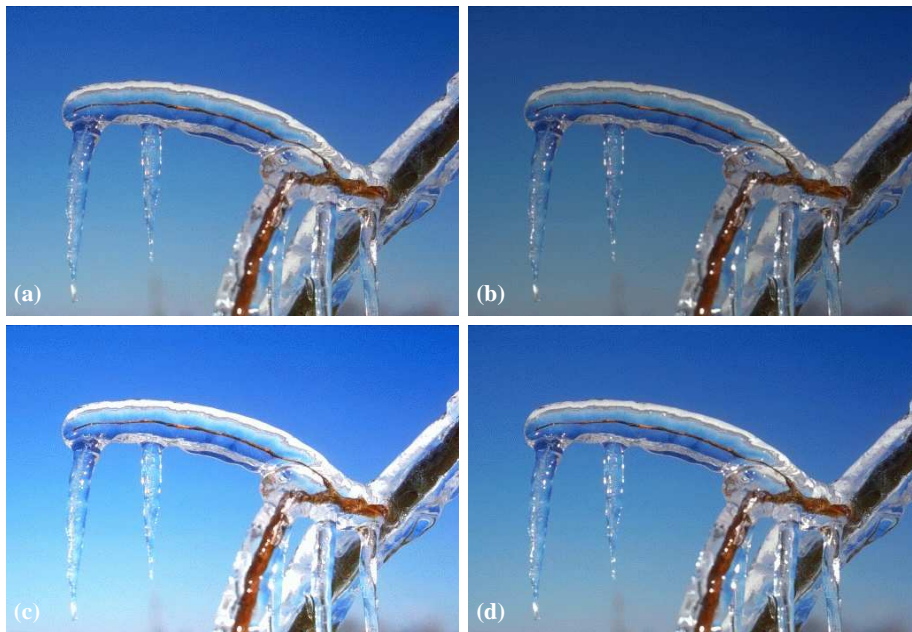
Figure 6.9: Comparison between linear scaling (left) and our tone scale (right). First row: images displayed on an SDR device or print. Second row: Simulation of images (a,b) when displayed on an HDR device. Only the diffuse brightness of these images is simulated. The bright specular highlights of image (d) can not be reproduced.

Figure 6.10: Comparison between linear scaling (left) and our tone scale (right). First row: images displayed on an SDR device or print. Second row: Simulation of images (a,b) when displayed on an HDR device. Only the diffuse brightness of these images is simulated. The bright specular highlights of image (d) can not be reproduced.

Chapter 7

Conclusion

7.1 Thesis Summary

In this thesis, we investigated tone mapping methods for images representing high dynamic range (HDR) scenes and developed new algorithms.

The display of HDR images on standard dynamic range (SDR) monitors is known to be problematic since the dynamic range of the input image has to be compressed. Using only global tone mapping induces too much of a loss of contrast information either in the shadow areas or in the bright areas. This is due to the decrease in local contrast caused by the compression in dynamic range. Local tone mapping methods that increase the local contrast can be combined with a global tone mapping to avoid this loss of contrast and improve detail visibility. However, local tone mapping methods often introduce artifacts such as halos, graying-out of low contrast areas, or bad color rendition. We presented a local tone mapping method that overcomes these problems. It uses an adaptive filter to increase the local contrast while preventing areas of very different illumination to influence each other. Our algorithm also introduces a weighting function of the local mask to prevent the black and white low contrast areas to turn gray and keep a visually pleasing global contrast. This local tone mapping algorithm is combined with a global pre-processing, which applies an additional gamma function depending on the key of the image. The application of our method on HDR images showed that it efficiently increases the local contrast while preventing halo artifacts and graying-out of low contrast areas.

Our method treats only the luminance channel to avoid color artifacts. The color image is recovered by putting the treated luminance back with the chrominance channels. Many existing local tone mapping methods proceed similarly. However, the influence of the choice of the color transform used to extract the

luminance and the chrominance is often not taken into consideration. We investigated two existing local tone mapping methods using four different color transforms to compute the luminance. We found that the correlation between luminance and chrominance plays a role in the final image appearance. However, a perfect decorrelation is not necessary.

The capture of HDR scenes is an issue as standard cameras have a limited dynamic range. A multiple exposure technique can be used to capture the whole dynamic range of still scenes. We extended this technique to the capture of HDR scenes containing a moving object. We tested a preliminary version on a few images and obtained promising results.

With these recently developed HDR monitors, HDR images can be displayed with hardly any compression of dynamic range. This brings the need for new tone mapping operators. After having studied the rendering of HDR images to SDR displays, we focused on the re-rendering of images representing HDR scenes that were mapped to SDR, to display them on HDR monitors. This is a very important topic as legacy images rendered to SDR displays represent a large amount of data. We propose the use of a piecewise linear tone scale function that recovers the brightness of specular highlights, which were clipped or compressed by the capturing and rendering process to SDR. We conducted a psychovisual experiment and showed that our tone scale provides more realistic-looking images than a linear scaling. With this research, we showed that re-rendering images representing HDR scenes to HDR displays benefits from an adequate tone mapping function. We hope that this work will encourage future research on that topic.

7.2   Future Research

During the course of this thesis, new questions were raised, which brought new opportunities for investigations. Here, we present ideas for future research that are a direct continuation of this work.

Tone Mapping Algorithms for SDR Displays

A large amount of work has been performed on the topic of tone mapping HDR images to SDR displays. Different algorithms have different goals, such as our method presented in Chapter 3 whose goal is to render pleasing images while avoiding halo and graying-out artifacts. Subjective testings have been conducted to evaluate tone mapping algorithms but none of the tests has led to satisfying conclusions yet. It is now becoming increasingly important to conduct more psychovisual experiments to evaluate existing algorithms and

assess their performances according to their goals.

Color Rendering. Most tone mapping algorithms focus on the rendering of luminance while the processing of color is barely addressed. Future tone mapping algorithms should study the "correct" rendition of color. Subjective testings focusing on color should also be conducted.

Generation of HDR Content Images and Videos

The extension of the multiple exposure technique using ghost removal has shown promises. It has to be developed further and tested on more images. In particular, a motion compensation algorithm should be integrated to handle global motion of the scene, which may be caused by moving the camera. Then, a natural extension is the generation of HDR videos. This can be done by merging frames of changing exposures. Another direction for improvement would be to combine the multiple exposure technique applied on video frames with a superresolution technique.

Tone Mapping for HDR Displays

The arrival of HDR displays provides a large number of opportunities for future work. In this thesis, we proposed a tone mapping method to re-render images representing HDR scenes. A continuation of this work is to extend it to the re-rendering of LDR scenes and SDR scenes. Direct rendering of RAW images and radiance maps (not tone-mapped to SDR) to HDR displays should be addressed as well.

Extension to Video. The extension of these tone mapping methods to video represent a key step toward their use in practical applications. We believe that the simplicity of our proposed algorithm is an advantage and makes the extension to video straightforward. A common problem when going from single image to video frames processing is the appearance of flickering between frames. Since, our tone scale is based on the maximum of the low-pass filtered image, it is less subject to flickering than controlling the exposure using just the maximum of the image. If some flickering still occurs, computing the tone scale using more than one frame would further ameliorate the problem.

Other Applications Related to HDR Displays. Additional future work for HDR displays includes creation of new standards, compression algorithms, evaluation and development of image quality metrics, and adaptation of existing perception models to the new conditions of vision.

Appendix A

The Generation of Radiance Maps Representing an HDR Scene

Consumer-level cameras can not capture the full spectral content and dynamic range of all natural scenes. When the scene's dynamic range exceeds the capabilities of the capture device, the exposure setting (time and aperture) determines which part of the scene will be properly exposed in the captured image.

The dynamic range (DR) of a digital camera is defined by the charge capacity divided by the noise (A.1) (Nakamura 2006). The charge capacity is given by the maximum number of electrons that can be captured by a pixel, inducing a linear response. Beyond this limit, additional electrons falling on the sensor do not increase its response linearly. In this case, we say that the sensor is saturated. Current consumer-level cameras have a dynamic range of about 4000:1, which can numerically be represented with 12 bits. It follows that most of the camera's RAW output is encoded with 12 bits. The dynamic range can be increased by larger pixel size.

$$DR = \frac{\text{max \# of electrons}}{\text{read-noise}} \tag{A.1}$$

The exposed range of the captured scene depends on the dynamic range of the camera, on its flare, and on the exposure setting. A long exposure or large aperture provides detail visibility in the shadow areas but tends to over-expose the bright parts of the scene, which are clipped. With a short exposure or small aperture, less light is captured. The bright areas are exposed correctly at the expense of detail visibility in the shadow. This is illustrated in Figure A.1.

In the top image (long exposure time), details in the room are visible but the outdoor view through the window is over-exposed. In the bottom image (short exposure time), the buildings outside the window are visible but the objects in the room are under-exposed.
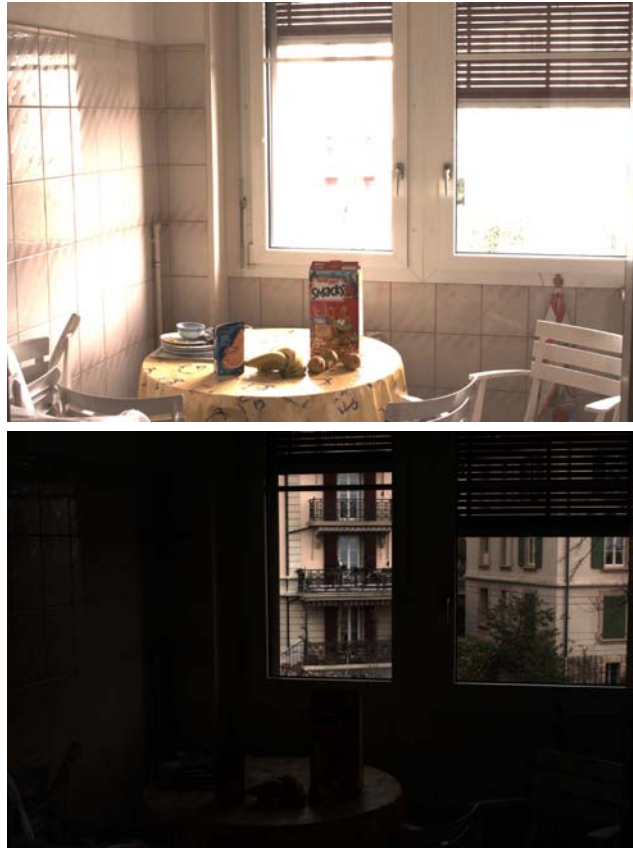

Figure A.1: Example of the same scene taken with different exposure settings. **Top:** Long exposure time. The objects in the room are visible but the window is over-exposed. **Bottom:** Short exposure time. The building outside the window is visible but the objects in the room are under-exposed.

A.1   Multiple Exposure Technique

Capturing the full dynamic range of any natural scene is however possible with conventional camera equipment using a multiple exposure technique (Mann and Picard 1995, Debevec and Malik 1997). This technique spans the range of scene radiances by taking several images with different exposure settings. The captured images are processed to compensate for the camera response curve,

the flare, and the exposure settings and then fused into a floating point image that is an accurate representation of the scene. We will call *radiance maps* the digital representation of an HDR scene as introduced by Debevec and Malik (1997).

Several software packages exist for the generation of radiance maps using this technique: Photosphere[1], HDRshop[2], or Photomatix [3]. Adobe Photoshop PCS2 also provides a script that automatically merges a set of images into a radiance map. Beside software solutions, the progress in developing HDR sensors has enabled the construction of cameras with larger dynamic range (Nayar and Branzoi 2003). Nayar and Mitsunaga (2000) also generate radiance maps using spatially varying pixel exposure.

Kang *et al.* (2003) propose a method to create videos representing HDR scenes from a sequence of alternating light and dark exposures. Each frame is transformed into a radiance map using neighboring frames. Their method includes a highly accurate motion estimation algorithm to compensate for the motion between frames.

Figure A.2 illustrates the construction of a radiance map. The left image is the result of the fusion of the three images on the right. Each of these images contributes to the radiance map differently depending on the pixel exposition: only properly exposed pixels are used for the construction of the radiance map. Pixels that are clipped or that have a small signal to noise ratio are excluded. In Figure A.2, the top image contributes to outdoor details while the middle and bottom image contribute to indoor details.

A.2 Radiance Map File Format

Radiance maps must be stored in a file format that can encode HDR data without loosing information. A file format is defined by an encoding technique, side information such as a header, and an optional compression method.

For all radiance maps used in this thesis, we use the "hdr" format originally known as *radiance*. This format was first developed for the computer graphic community but is now commonly used in digital photography. The pixel data come in a four byte rgbe encoding. The first three bytes r, g, and b share the same exponent e given by the fourth byte.

Let us denote v_{rgbe}, a vector of 4 entries representing a pixel value encoded in rgbe. The vector of 3 entries v_{RGB} representing the same pixel value encoded with RGB primaries is retrieved as follows:

---

[1]www.anyhere.com
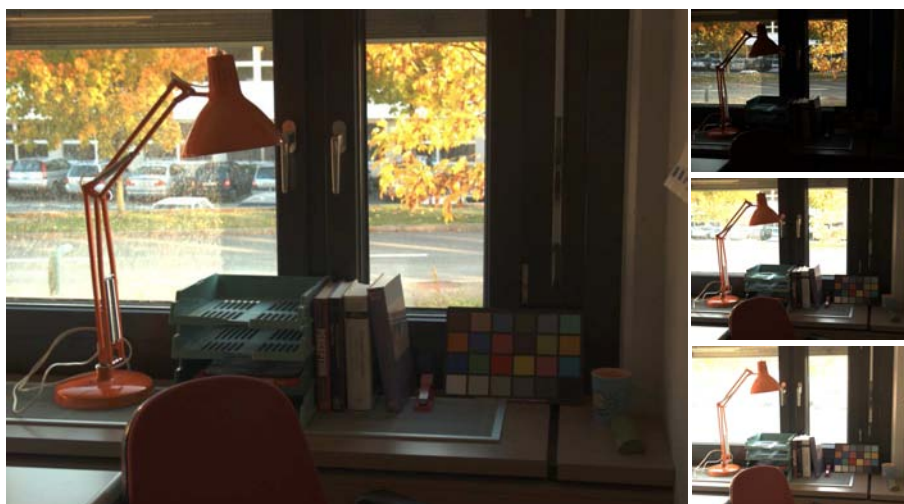[2]www.hdrshop.com
[3]www.hdrsoft.com

Figure A.2: Generation of a radiance map using multiple exposures of a scene. The left image shows the result of the fusion of the three images on the right.

$$v_{\text{RGB}}(i) = \frac{v_{\text{rgbe}}(i) + 0.5}{2^8} \cdot 2^{(v_{\text{rgbe}}(4) - 2^7)} \qquad (A.2)$$

where $i = 1, 2, 3$, indicating one of the R,G,B color channel.

The rgbe encoding enables to encode a dynamic range sufficient for natural scenes using only 4 bytes per pixel. The "hdr" format is composed of a header followed by the rgbe encoded image. A lossless compression (run-length encoding) is then applied.

Another possible format to store radiance maps is "floating point tiff". Because it uses 96 bits per pixel (12 bytes), it can encode a very high dynamic range ($\sim$ 79 orders of magnitude) with a better precision than the radiance format. The drawback of "floating point tiff" is that it takes three times more storage room than radiance. More formats for radiance map encoding are described in (Reinhard *et al.* 2005), Chapter 4.

A.3   Image Sources used in this Thesis

Several databases of HDR images are available on the Internet. In addition to our own images, we used radiance maps provided by various authors. Reinhard put together images from Peter Shirley and Greg Ward.[4] Debevec used his implementation of the multiple exposure technique (Debevec and Malik 1997)

---

[4]http://www.cs.utah.edu/~reinhard/cdrom/hdr.html

and made available the resulting images.[5] Xiao et al. built a database of well-calibrated radiance maps (Xiao *et al.* 2002) that can be downloaded from their website.[6] The Munsell Color Science Laboratory (RIT/MCSL) created more than 50 radiance maps using Photosphere software.[7]

We also use RAW images representing HDR scenes taken with a Canon (EOS30, 300D, or Powershot G2) digital camera. Our image database is available on our webpage. [8]

---

[5]http://www.debevec.org/
[6]http://pdc.stanford.edu/hdri/
[7]http://www.cis.rit.edu/mcsl/icam/hdr/
[8]http://lcavwww.epfl.ch/∼lmeylan

Appendix B

Adapting the Multiple Exposure Technique for Scenes Containing a Moving Object

The problem of using a multiple exposure technique is that it can only be used to capture still scenes. Indeed, if objects move in the scene, they will appear at different positions in the captured images, which will create "ghosts" in the merged image. An example of a ghost created by a tennis ball moving in the scene is shown in Figure B.1. While motion of a few pixels can be treated as a pixel registration problem and solved using motion compensation (Kang *et al.* 2003), larger amplitude moving objects are still an issue.

B.1 A Method to Remove Ghosts in Radiance Maps

We developed a technique that detects and prevents ghosts in images constructed with a multiple exposure technique.[1] Its principle is to merge the still parts of the images and to use only a single exposure for the moving objects. This requires that the dynamic range of each moving object can be captured by a single exposure. We tested it for one moving object and propose an extension when several moving objects are present.

   The problem of ghost elimination in radiance maps has to our knowledge

---

[1]This work was done in collaboration with Olivier Hochreutiner (semester project, June 2005).

Figure B.1: Example of ghost caused by a moving object in an HDR scene captured by a multiple exposure technique.

only been described in (Reinhard *et al.* 2005). Their method is based on the same idea but differs in various ways. In particular, the motion detection and the computation of the motion mask differ. They are based on the variance at each pixel, while we use an image differencing algorithm.

B.1.1   Motion Mask Computation

The first step of our ghost removal method is to compute the motion mask, which defines the areas that must be excluded from the image fusion. Prior to any operation, we have to compensate for the difference of exposure during image capture. We assume that the device is perfectly linear and that images were taken by regularly changing the exposure time. Each image is then weighted by the inverse of its exposure time, using one image for the reference time (τ_{ref}).

$$Z_i(p) = \frac{\tau_{ref}}{\tau_i} \cdot I_i(p) \tag{B.1}$$

$I_i(p)$ is the captured image with exposure time τ_i. τ_{ref} is the reference exposure time and $Z_i(p)$ is the image originally taken with exposure time τ_i that has been compensated to the reference exposure time.

The motion mask is computed by the image differencing technique described in (Radke *et al.* 2005). Pairs of images are subtracted resulting in non-zero

values where motion occurs:

$$\Delta_{i,j}(p) = Z_i(p) - Z_j(p). \tag{B.2}$$

The results for all image pairs are added and thresholded to obtain a binary map $\mathcal{B}$ (Figure B.2 (a)):

$$\Delta(p) = \sum_{i=1}^{J} \sum_{j=1, i \neq j}^{J} \Delta_{i,j}(p), \tag{B.3}$$

where J is the number of captured images.

$$\mathcal{B}(p) = \begin{cases} 1 & \text{if } \Delta(p) \geq \text{threshold} \\ 0 & \text{otherwise} \end{cases} \tag{B.4}$$

Using a differencing technique for motion detection tends to emphasize the edges of the moving object, since the difference between the background and the object is likely to be larger than adjacent areas within the object. We can see in Figure B.2 (a) that the mask does not completely cover the area where motion occurs. Since the detected motion corresponds to the edges of the moving object across images, the convex hull of the detected moving pixels contains the moving object. The convex hull is defined by the smallest polygon containing a set of points in a two-dimensional space. In our case, the set of points is given by the pixels having a value of 1 (shown white in Figure B.2) in the binary map $\mathcal{B}(p)$. The convex hull is computed using the Graham scan algorithm (Graham 1972). Applying this algorithm on Figure B.2 (a) and filling in the shape defined by the convex hull results in Figure B.2 (b). We call this new binary map $\mathcal{B}_{hull}$.

The example of Figure B.2 has only one moving object. An extension to more than one moving object can be done by creating clusters of white pixels in the binary map $\mathcal{B}$. A maximum distance between pixels belonging to the same object is chosen and determines n, the number of moving objects. Each of the n clusters corresponds to one moving object and is put in a separate binary map. The convex hull algorithm is applied on each of these separately.

B.1.2 Using the Motion Mask in the Fusion

Now that the location of pixels belonging to moving objects is known, we can exclude them from the fusion of images. The best-exposed image, called Z_{best} is selected for each moving object and used in the radiance map at the object location. Z_{best} is defined by the image having the longest exposure time and no saturated pixel at the moving object location.
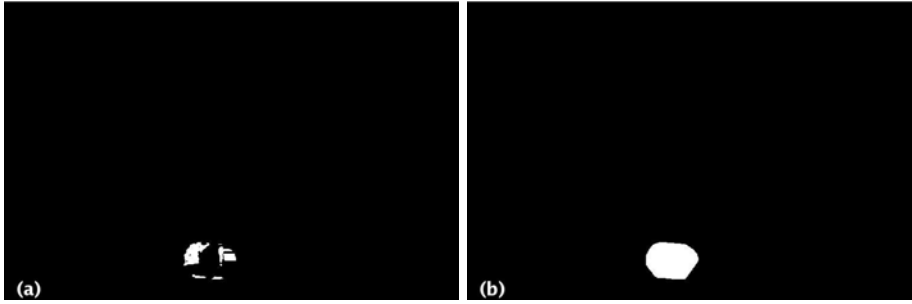
Figure B.2: Computation of the motion mask. **Left:** Motion mask resulting from the image differencing and thresholding operations ($\mathcal{B}$). **Right:** Motion mask after the computation of the convex hull ($\mathcal{B}_{hull}$).

The radiance map I_{HDR} resulting from the fusion is computed as follows. For one moving object:

$$I_{HDR}(p) = (\frac{1}{|P_i|} \sum_{i=1, p \in P_i}^{J} Z_i(p) \cdot (1 - \mathcal{B}_{hull}(p))) + Z_{best}(p) \cdot \mathcal{B}_{hull}(p), \quad \text{(B.5)}$$

where P_i is the subset of well-exposed (non-saturated) pixels used in the fusion.

For n moving objects, we have n binary maps $B_{hull,k}$ and n best exposed images $Z_{best,k}$, where k indicates one moving object:

$$I_{HDR}(p) = \frac{1}{|P_i|} \sum_{i=1, p \in P_i}^{J} Z_i(p) \cdot (1 - \mathcal{B}_{all}(p)) + \sum_{k=1}^{n} Z_{best,k}(p) \cdot \mathcal{B}_{hull,k}(p), \quad \text{(B.6)}$$

where

$$B_{all}(p) = \prod_{k=1}^{n} B_{hull,k}(p). \quad \text{(B.7)}$$

B.2   Results

Figure B.3 and B.4 show the result of our ghost removal algorithm for two scenes containing one moving object. The top row (a) illustrates the motion mask computation for each scene. The second row (b) shows the image obtained without the ghost removal algorithm. The bottom row (c) shows the result obtained with our algorithm.

Figure B.3: Result of our ghost removal algorithm. (a) Motion mask computation. (b) Result without ghost removal algorithm. (c) Result obtained with our ghost removal algorithm.
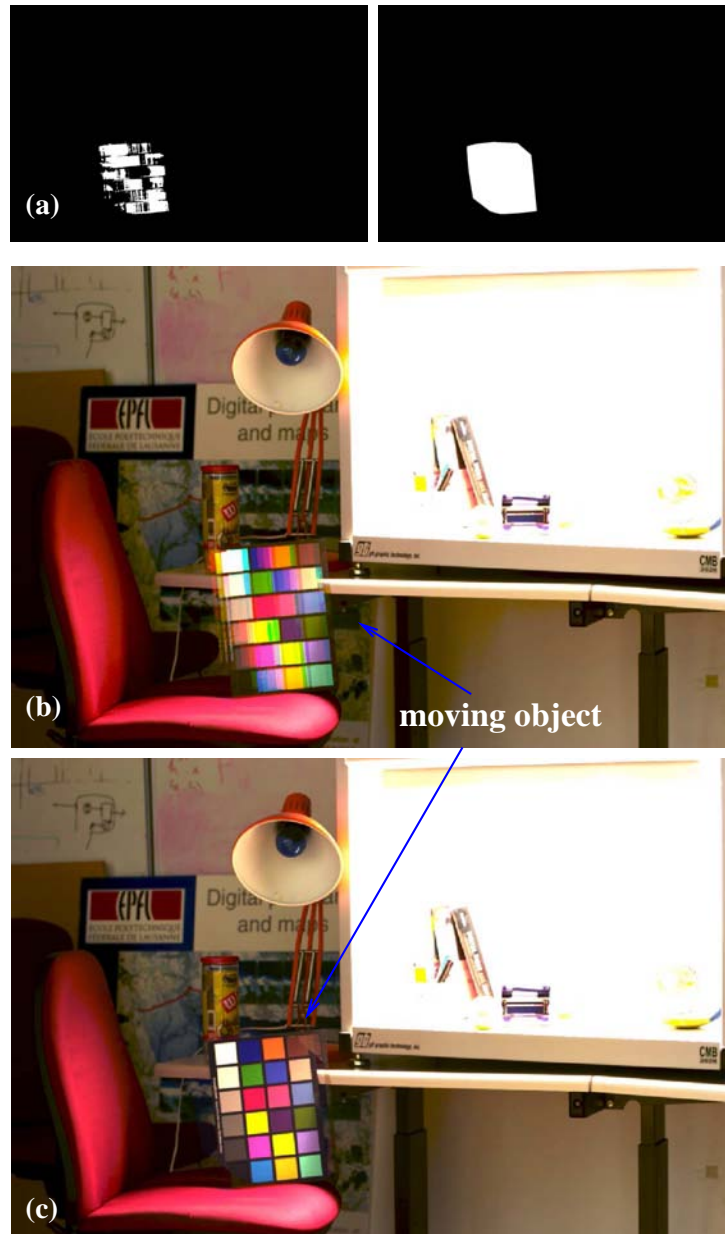
Figure B.4: Result of our ghost removal algorithm. (a) Motion mask computation. (b) Result without ghost removal algorithm. (c) Result obtained with our ghost removal algorithm.
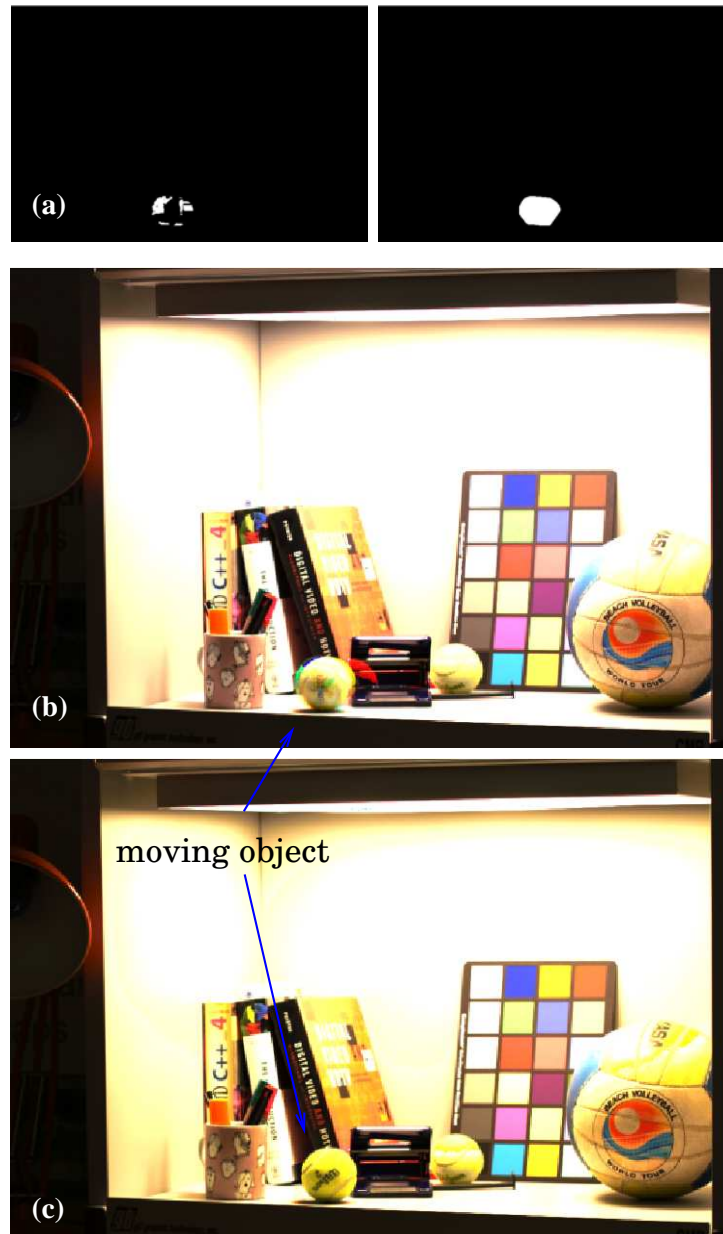
Appendix C

Software Correction Algorithm for High Dynamic Range Display

In an HDR display, the uniform backlight common to standard displays is replaced by an array of white LEDs, which can be controlled independently. The image formed by the LEDs projected through a diffusion screen serves as backlight to the LCD panel.

Thus, displaying images on an HDR monitor requires some processing. The input image must be split into a low resolution image sent to the LEDs and a high resolution color image sent to the LCD such that what is seen on the screen represents accurately the input image.

The process applied to the input images before display is illustrated in Figure C.1. It is also described in (Seetzen *et al.* 2004). In short, it consists in resizing the input image to screen resolution, computing the values to drive the LEDs, simulating the backlight image generated by the LEDs seen through the diffusion screen, and computing the LCD image using this backlight image. Each of these steps is described in this appendix.

C.1 Scaling to Screen Resolution

First, the input image is resized to the display resolution (1280×1024 for the model described in (Seetzen *et al.* 2004)). If the aspect ratio is different than 5/4, zero padding is added where necessary. This is illustrated in Figure C.2.
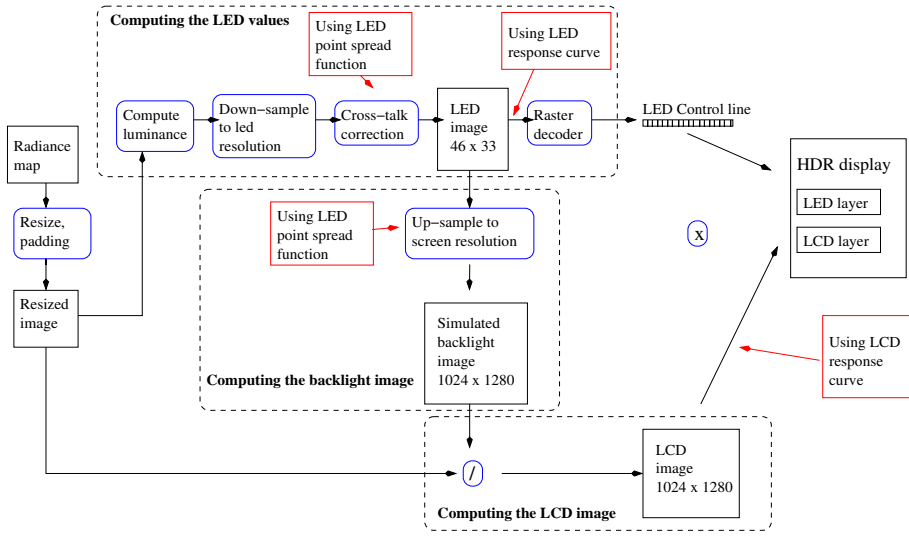
Figure C.1: Global processing applied to the input image to compute the LED driving values and the LCD image.

C.2   LED Control Line Computation

The next step is to compute the control line that drives the LEDs. Let us call I_s the scaled image obtained at the previous step. Since the LED image will multiply with the LCD image, $\sqrt{I_s}$ is used for the LEDs contribution. We compute its luminance Λ'_s:

$$\Lambda'_s = 0.265 * \sqrt{I_{s,1}} + 0.670 * \sqrt{I_{s,2}} + 0.065 * \sqrt{I_{s,3}}, \qquad (C.1)$$

where $I_{s,1}$, $I_{s,2}$, and $I_{s,3}$ are the red, green and blue color channels of I_s. Then, Λ'_s is downsampled to the LED array resolution (46×33).

$$\lambda = \Lambda_s \downarrow_{n,m}, \qquad (C.2)$$

where λ is the downsampled image called *LED image*. n and m are the downsampling factor so that λ is of size 46×33.

In order to accommodate the fact that the LEDs are arranged on the screen in a hexagonal pattern, the LED image has twice the number of columns than the horizontal dimension of the physical LED array. This is illustrated in Figure C.3, which presents the process to compute the LED driving values.

After downsampling, the *cross-talk correction* step corrects the LED image pixel values to compensate for the light contribution of neighboring LEDs on the physical position of one considered LED. It is assumed that all LEDs have

Figure C.2: Scaling and padding. The input image is scaled to the display resolution. Zero-padding is used if the aspect ratio of the source differs from that of the display.

the same point spread function, i.e. the measured luminance decreases similarly when increasing the distance from the center of the light source, and that their light contribution only affects direct neighbors.

The matrix in (C.3) shows the six neighbors of one LED at position (x, y) that are used for cross-talk correction. Due to the physical arrangement of the LEDs, they are at equal distance d. The symbol '$\times$' shows a physical position in the grid where there is no LED. '0' corresponds to an existing LED, whose light contribution is considered to be null at position (x,y).

$$\begin{bmatrix} 0 & \times & (x, y+2) & \times & 0 \\ \times & (x-1, y+1) & \times & (x+1, y+1) & \times \\ 0 & \times & (\mathbf{x}, \mathbf{y}) & \times & 0 \\ \times & (x-1, y-1) & \times & (x+1, y-1) & \times \\ 0 & \times & (x, y-2) & \times & 0 \end{bmatrix} \quad (C.3)$$

The pixel values of the cross-talk corrected image λ_{cross} are determined by subtracting the light contribution of the six neighbors for each LED position.

$$\lambda_{cross}(x, y) = \begin{cases} 0 \text{ if x and y are odd or x and y are even} \\ a_1 \cdot \lambda_{(x,y)} - a_2 \cdot (\lambda_{(x-1,y+1)} + \lambda_{(x+1,y-1)} + \\ +\lambda_{(x+1,y+1)} + \lambda_{(x-1,y-1)} + \lambda_{(x,y+2)} + \lambda_{(x,y-2)}) \quad \text{otherwise} \end{cases}$$
$$(C.4)$$

We measured the luminance $L(d)$ emitted by one LED at distance d of its center and compute a_1 and a_2 by solving a linear system (C.5). Since the LEDs' point spread function is linear with input voltage, a_1 and a_2 only have to be computed once.

Input image

Extract luminance
Down−sample to LED
image resolution
46 x 33

LED image

Cross−talk
correction

Cross−talk corrected
image

LED driving value
image

LED response
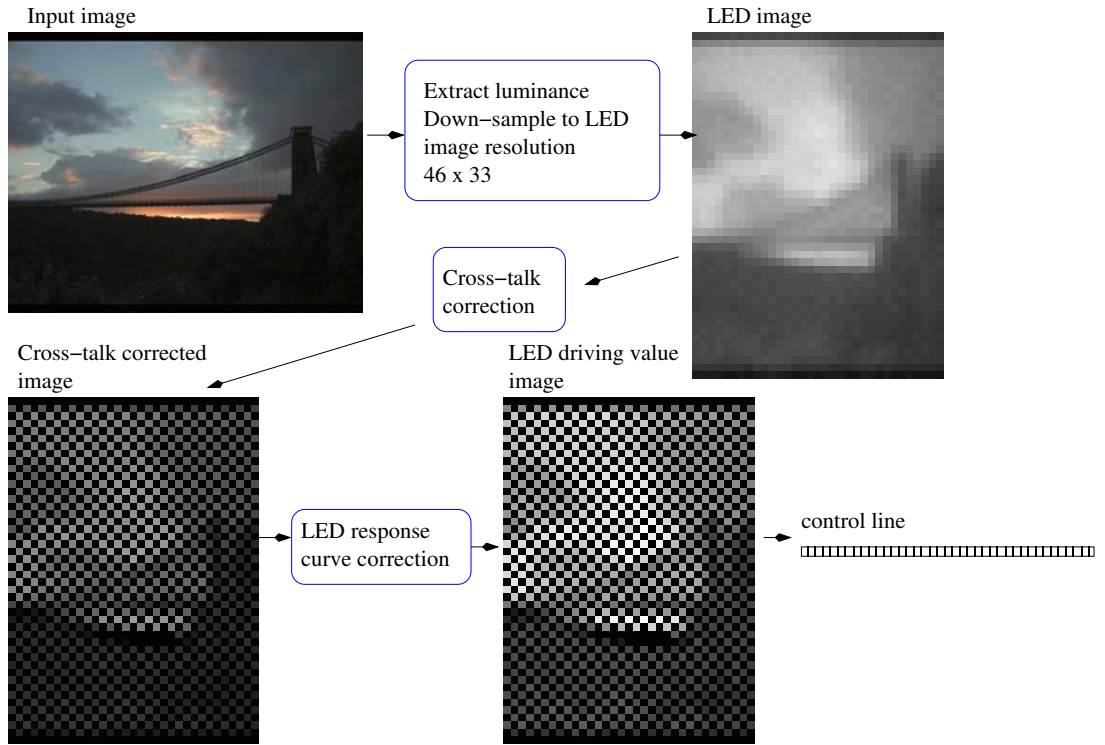curve correction

control line

Figure C.3: Computation of the LED driving values. The luminance image is first downsampled to the resolution of the LED array. Then, compensation for cross-talk between LEDs is applied as well as compensation for the LED response curve. The LED driving values are transformed into a control line by removing pixels which do not correspond to an LED location.

$$
\begin{bmatrix}
\lambda_{(x-1,y-1)} \\
\lambda_{(x+1,y-1)} \\
\lambda_{(x-1,y+1)} \\
\lambda_{(x+1,y+1)} \\
\lambda_{(x,y+2)} \\
\lambda_{(x,y-2)} \\
\lambda_{(x,y)}
\end{bmatrix}
=
\begin{bmatrix}
1 & L(d) & ... & & & & L(d) \\
L(d) & 1 & L(d) & ... & & & ... \\
... & L(d) & 1 & L(d) & ... & & \\
& ... & L(d) & 1 & L(d) & ... & \\
& & ... & L(d) & 1 & L(d) & \\
& & & ... & L(d) & 1 & L(d) \\
L(d) & ... & & & ... & L(d) & 1
\end{bmatrix}
*
\begin{bmatrix}
\lambda_{cross(x-1,y-1)} \\
\lambda_{cross(x+1,y-1)} \\
\lambda_{cross(x-1,y+1)} \\
\lambda_{cross(x+1,y+1)} \\
\lambda_{cross(x,y+2)} \\
\lambda_{cross(x,y-2)} \\
\lambda_{cross(x,y)}
\end{bmatrix},
$$

(C.5)

where λ's are the ideal values that would be used to drive the LEDs if there was no cross-talk between them. λ_{cross} are the LED driving values after cross-talk correction, which are computed with (C.4). Let us call $\mathcal{L}$, the central matrix in (C.5). The two coefficients a_1 and a_2 are obtained by inversing $\mathcal{L}$ as

shown in (C.6).

$$\mathcal{L}^{-1} = \begin{bmatrix} a_1 & a_2 & ... & & & a_2 \\ a_2 & a_1 & a_2 & ... & & ... \\ ... & a_2 & a_1 & a_2 & ... & \\ & ... & a_2 & a_1 & a_2 & ... \\ & & ... & a_2 & a_1 & a_2 \\ a_2 & ... & & ... & a_2 & a_1 \end{bmatrix} \tag{C.6}$$

Then, a compensation for the LED response curve is performed by simply applying a power function. The function exponent is based on physical measurements.

These steps define the input voltage values that will drive the LEDs. They are integrated in the image sent to the HDR display under the form of a control line. Each pixel of the line controls one LED. The step of making a correspondence between control line and spatial position in the LED array is called *raster decoding* (Figure C.1) and is performed by a simple look-up table.

C.3   Simulating the Backlight Image

Knowing the LED driving values (λ_{cross}), we can simulate the backlight image that lights up the LCD panel through the diffusion screen. First, the cross-talk corrected image computed in the previous step is upsampled to the screen resolution. The point spread function of the LED seen through the diffusion screen is approximated by an addition of two Gaussian, which is then used to filter the upsampled image. The simulated backlight image is shown in Figure C.4.

C.4   Computing the LCD Front Image

The LCD image is computed by dividing each color channel of the input image with the simulated backlight image (Figure C.4). A small compensation for LED's yellow color cast is also performed.

The LCD image is sent to the LCD panel and backlighted by the LEDs. For large luminance patches, the resulting image at the screen is an accurate representation of the input image. For small spatial luminance variations, the cross-talk correction is not sufficient, as discussed in Section 5.5.
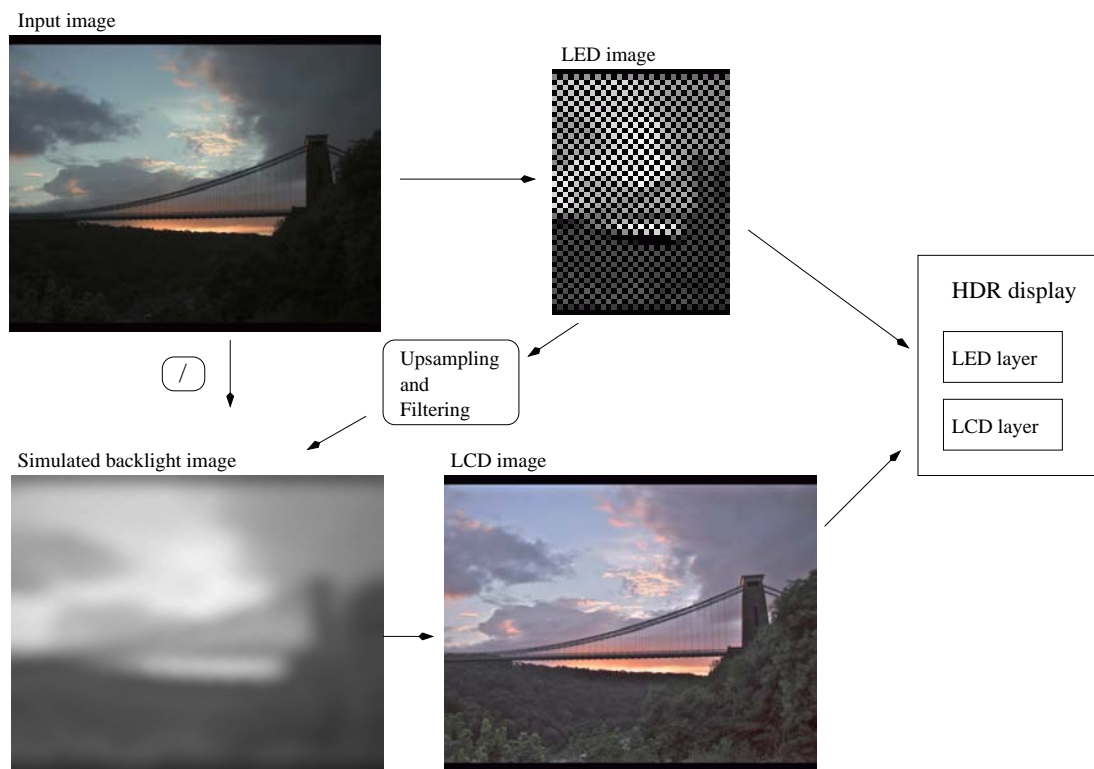
Figure C.4: The simulated backlight is computed from the LED array image. It is then used to compute the LCD image that is sent to the LCD layer of the HDR display.

Bibliography

Adams, A. (1971*a*), *Camera and Lens*, Morgan & Morgan, Dobbs Ferry, NY.

Adams, A. (1971*b*), *The Print*, Morgan & Morgan, Dobbs Ferry, NY.

Adams, A. (1972), *The Negative*, Morgan & Morgan, Dobbs Ferry, NY.

Alleysson, D. and Süsstrunk, S. (2002), On adaptive non-linearity for color discrimination and chromatic adaptation, *in* Proc. IS&T First European Conference on Color in Graphics, Image, and Vision, Poitiers, France, pp. 190–195.

Ashikhmin, M. (2002), A tone mapping algorithm for high contrast images, *in* Proc. EUROGRAPHICS 2002, Pisa, Italy, pp. 1–11.

Barnard, K. and Funt, B. (1998), Investigations into multi-scale Retinex, *in* Proc. Colour Imaging in Multimedia '98, Derby, UK, pp. 9–17.

Blake, A. and Bülthoff, H. (1991), Shape from specularities: computation and psychophysics, *Philosophical Transactions of the Royal Society of London, B: biological sciences* **331**(1260), 237–252.

Brainard, D. and Wandell, B. (1986), Analysis of the Retinex theory of color vision, *Journal of the Optical Society of America A* **3**(10), 1651–1661.

Braun, G. J. and Fairchild, M. D. (1999), Image lightness rescaling using sigmoidal contrast enhancement functions, *Journal of Electronic Imaging* **8**(4), 380–393.

Buchsbaum, G. and Gottschalk, A. (1983), Trichromacy, opponent colours coding and optimum colour information transmission in the retina, *in* Proc. Royal Society of London, Series B, biological Sciences, Vol. 220, pp. 89–113.

Canny, J. (1986), A computational approach to edge detection, *IEEE Transactions on pattern analysis and machine intelligence* **8**(6), 679–698.

CIE 15:2004 (2004), Colorimetry.

Ciurea, F. and Funt, B. (2004), Tuning Retinex parameters, *Journal of Electronic Imaging* **13**(1), 58–64.

Debevec, P. E. and Malik, J. (1997), Recovering high dynamic range radiance maps from photographs, *in* Proc. ACM SIGGRAPH 97, Annual Conference on Computer Graphics, Los Angeles, CA, pp. 369–378.

DiCarlo, J. M. and Wandell, B. (2000), Rendering high dynamic images, *in* Proc. IS&T/SPIE Electronic Imaging 2000. Image Sensors, Vol. 3965, San Jose, CA, pp. 392–401.

Drago, F., Martens, W. L., Myszkowski, K. and Chiba, N. (2003), Design of a tone mapping operator for high dynamic range images based upon psychophysical evaluation and preference mapping, *in* Proc. IS&T/SPIE Electronic Imaging 2003. The Human Vision and Electronic Imaging VIII, Vol. 5007, San Jose, CA, pp. 321–331.

Durand, F. and Dorsey, J. (2002), Fast bilateral filtering for the display of high-dynamic-range images, *in* Proc. ACM SIGGRAPH 2002, Annual Conference on Computer Graphics, San Antonio, CA, pp. 257–266.

Elad, M., Kimmel, R., Shaked, D. and Keshet, R. (2003), Reduced complexity Retinex algorithm via the variational approach, *Journal of visual communication and image representation* **14**(1), 369–388.

Engeldrum, P. G. (2000), *Psychometric scaling: A toolkit for imaging systems development*, Imcotek press, Winchester, MA.

Fairchild, M. D. and Johnson, G. M. (2004), iCAM framework for image appearance differences and quality, *Journal of Electronic Imaging* **13**(1), 126–138.

Fattal, R., Lischinski, D. and Werman, M. (2002), Gradient domain high dynamic range compression, *in* Proc. ACM SIGGRAPH 2002, Annual Conference on Computer Graphics, San Antonio, CA, pp. 249–256.

Finlayson, G., Hordley, S. and Drew, M. (2002), Removing shadows from images using Retinex, *in* Proc. IS&T/SID Tenth Color Imaging Conference: Color Science, Systems, and Application, Scottsdale, AR, pp. 73–79.

Foster, D. and Nascimento, S. (1994), Relational colour constancy from invariant cone-excitation ratios, *in* Proc. Royal Society of London, Series B, biological Sciences, Vol. 257, London, pp. 115–121.

Frankle, J. and McCann, J. (1983), Method and appartus for lightness imaging. US patent #4,384,336.

Fredembach, C. and Finlayson, G. (2005), Hamiltonian path based shadow removal, *in* Proc. The 16th British Machine Vision Conference (BMVC), pp. 970–980.

Funt, B., Barnard, K., Brockington, M. and Cardei, V. (1997), Luminance-based multi-scale Retinex, *in* Proc. AIC Colour 1997 - 8th Congress of the International Colour Association, Vol. I, Kyoto, Japan, pp. 330–333.

Funt, B., Ciurea, F. and McCann, J. (2004), Retinex in matlab, *Journal of Electronic Imaging* **13**(1), 48–57.

Graham, R. L. (1972), An efficient algorithm for determining the convex hull of a finite planar set, *Information Processing Letters* **1**, 73–82.

Green, P. J. (2003), A colour engineering toolbox, http://www.digitalcolour.org/toolbox.htm.

Grigorescu, C., Petkov, N. and Westenberg, M. A. (2003), Contour detection based on nonclassical receptive field inhibition, *IEEE Transactions on Image Processing* **12**(7), 729–739.

Grigorescu, C., Petkov, N. and Westenberg, M. A. (2004), Contour and boundary detection improved by surround suppression of texture edges, *Image and Vision Computing* **22**, 609–622.

Holm, J. (1996), Photographic tone and colour reproduction goals, *in* Proc. CIE Expert Symposium'96 on Colour Standards for Image Technology, Vienna, Austria, pp. 51–56.

Horn, B. K. P. (1974), Determining lightness from an image, *Computer Graphics and Image Processing* **3**, 277–299.

Hunt, R. W. G. (1995), *The reproduction of colour*, Fountain Press, England.

Hurlbert, A. (1986), Formal connections between lightness algorithms, *Journal of the Optical Society of America A* **3**, 1684–1692.

IEC 61966-2-1 (1999), Multimedia systems and equipment - Colour measurement and management - Part2-1:Colour management - Default RGB colour space - sRGB.

Interrante, V., Fuchs, H. and Pizer, S. M. (1997), Conveying the 3d shape of smoothly curving transparent surfaces via texture, *IEEE Transactions on Visualization and Computer Graphics* **3**(2), 98–116.

ISO 22028-1 (2004), Photography and graphic technology - Extended colour encodings for digital image storage, manipulation and interchange. Part 1: Architecture and requirements.

Jobson, D., Rahman, Z.-U. and Woodell, G. (1997), A multiscale Retinex for bridging the gap between color images and the human observation of scenes, *IEEE Transactions on Image Processing* **6**(7), 965–976.

Johnson, G. M. (2005), Cares and concerns of CIE TC8-08: Spatial appearance modeling and HDR rendering, *in* Proc. IS&T/SPIE Electronic Imaging 2005, Vol. 5668, San Jose, CA, pp. 148–156.

Kaiser, P. K. (1996), The joy of visual perception: A web book, *http : //www.yorku.ca/eye/*.

Kang, S. B., Uyttendaele, M., Winder, S. and Szeliski, R. (2003), High dynamic range video, *ACM Transactions on Graphics* **22**(3), 319–325.

Kendall, M. G. and Smith, B. B. (1940), On the method of paired comparisons, *Biometrika* **31**(3/4), 324–345.

Kimmel, R., Elad, M., Shaked, D., Keshet, R. and Sobel, I. (2003), A variational framework for Retinex, *International Journal of Computer Vision* **52**(1), 7–23.

Klinker, G. J., Shafer, S. A. and Kanade, T. (1987), Using a color reflection model to separate highlights from object color, *in* Proc. First International Conference on Computer Vision (ICCV), London, UK, pp. 145–150.

Klinker, G. J., Shafer, S. A. and Kanade, T. (1988), The measurement of highlights in color images, *International Journal of Computer Vision* **2**(1), 7–32.

Kuang, J., Yamaguchi, H., Johnson, G. M. and Fairchild, M. D. (2004), Testing HDR image rendering algorithms, *in* Proc. IS&T/SID Twelfth Color Imaging Conference: Color Science, Systems, and Application, Scottsdale, AR, pp. 315–320.

Land, E. (1964), The Retinex, *American Scientist* **52**(2), 247–264.

Land, E. (1977), The Retinex theory of colour vision, *Scientific American* **237**(6), 108–128.

Land, E. (1986), An alternative technique for the computation of the designator in the Retinex theory of color vision, *in* Proc. National Academy of Sciences of the United States of America, Vol. 83, pp. 3078–3080.

Land, E. and McCann, J. (1971), Lightness and Retinex theory, *Journal of the Optical Society of America A* **61**(1), 1–11.

Ledda, P., Chalmers, A., Troscianko, T. and Seetzen, H. (2005), Evaluation of tone mapping operators using a high dynamic range display, *in* Proc. ACM SIGGRAPH 2005, Annual Conference on Computer Graphics, Los Angeles, CA, pp. 640–648.

Ledda, P., Santos, L. P. and Chalmers, A. (2004), A local model of eye adaptation for HDR images, *in* Proc. Third International Conference on Computer Graphics, Virtual Reality, Visualisation, AFRIGRAPH, Cape Town, South Africa, pp. 151–160.

Lee, H.-C. (1986), Method for computing the scene-illuminant chromaticity from specular highlights, *Journal of the Optical Society of America A* **3**(10), 1694–1698.

Mann, S. and Picard, R. W. (1995), On being 'undigital' with digital cameras: extending dynamic range by combining differently exposed pictures, *in* Proc. IS&T's 48th Annual Conference, Cambridge, MA, pp. 422–428.

Marini, D. and Rizzi, A. (1993), Colour constancy and optical illusions: a computer simulation with Retinex theory, *in* Proc. ICIAO93, the 7th International Conference on Image Analysis and Processing, Monopoli (Bari) Italy, pp. 657–660.

McCann, J. (1999), Lessons learned from mondrians applied to real images and color gamuts, *in* Proc. IS&T/SID Seventh Color Imaging Conference: Color Science, Systems, and Applications, Scottsdale, AZ, pp. 1–8.

Meylan, L. and Süsstrunk, S. (2004*a*), Bio-inspired image enhancement for natural color images, *in* Proc. IS&T/SPIE Electronic Imaging 2004. The Human Vision and Electronic Imaging IX, Vol. 5292, San Jose, CA, pp. 46–56.

Meylan, L. and Süsstrunk, S. (2004*b*), Color image enhancement using a Retinex-based adaptive filter, *in* Proc. IS&T Second European Conference on Color in Graphics, Image, and Vision (CGIV 2004), Aachen, Germany, pp. 359–261.

Meylan, L. and Süsstrunk, S. (2006), High dynamic range image rendering with a Retinex-based adaptive filter, to be published in IEEE Transactions on Image Processing.

Moroney, N. (2004), Method and system of local color correction using background luminance mask. US patent #6,741,753.

Nakamura, J., ed. (2006), *Image sensors and signal processing for digital still cameras*, CRC press, Taylor & Francis Group, Boca Raton, FL.

Nayar, S. and Branzoi, V. (2003), Adaptive dynamic range imaging: Optical control of pixel exposures over space and time, *in* Proc. Ninth International Conference on Computer Vision (ICCV), Vol. 2, Nice, France, pp. 1168–1175.

Nayar, S. K. and Mitsunaga, T. (2000), High dynamic range imaging: Spatially varying pixel exposures, *in* Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2000), Vol. 1, Head Island, SC, pp. 472–479.

Osadchy, M., Jacobs, D. and Ramamoorthi, R. (2003), Using specularities for recognition, *in* Proc. Ninth International Conference on Computer Vision (ICCV), Vol. 2, Nice, France, pp. 1512–1519.

Park, J. B. and Kak, A. C. (2003), A truncated least squares approach to the detection of specular highlight in color images, *in* Proc. International Conference on Robotics and Automation, Taipei, Taiwan, pp. 1397–1403.

Pattanaik, S. N., Ferwerda, J. A., Fairchild, M. D. and Greenberg, D. P. (1998), A multiscale model of adaptation and spatial vision for realistic image display, *in* Proc. ACM SIGGRAPH 98, Annual Conference on Computer Graphics, Orlando, FL, pp. 287–298.

Pattanaik, S. N., Tumblin, J., Yee, H. and Greenberg, D. P. (2000), Time-dependent visual adaptation for fast realistic image display, *in* Proc. ACM SIGGRAPH 2000, Annual Conference on Computer Graphics, New Orleans, LO, pp. 47–54.

Petkov, N. and Westenberg, M. A. (2003), Suppression of contour perception by band-limited noise ans its relation to nonclassical receptive field inhibition, *Biological Cybernetics* **88**, 236–246.

Poynton, C. (2003), *Digital video and HDTV, algorithms and interfaces*, Morgan Kaufmann publishers, San Francisco, CA.

Radke, R. J., Andra, S., Al-Kofahi, O. and Roysam, B. (2005), Image change detection algorithms: A systematic survey, *IEEE Transactions on Image Processing* **14**(3), 294–306.

Rahman, Z.-U., Jobson, D. D. and Woodell, G. A. (2004), Retinex process-
ing for automatic image enhancement, *Journal of Electronic Imaging*
13(1), 100–110.

Reinhard, E., Stark, M., Shirley, P. and Ferwerda, J. (2002), Photographic tone
reproduction for digital images, *in* Proc. ACM SIGGRAPH 2002, Annual
Conference on Computer Graphics, San Antonio, CA, pp. 267–276.

Reinhard, E., Ward, G., Pattanaik, S. and Debevec, P. (2005), *High Dynamic
Range Imaging. Acquisition, Display, and Image-Based Lighting*, Morgan
Kaufmann Publishers, San Francsisco, CA.

Rizzi, A., Marini, D., Rovati, L. and Docchio, F. (2003), Unsupervised cor-
rections of unknown chromatic dominants using a brownian-path-based
Retinex algorithm, *Journal of Electronic Imaging* **12**(3), 431–440.

Seetzen, H., Heidrich, W., Stuerzlinger, W., Ward, G., Whitehead, L., Trenta-
coste, M., Ghosh, A. and Vorozcov, A. (2004), High dynamic range display
systems, *ACM Transactions on Graphics (special issue SIGGRAPH 2004)*
23(3), 760–768.

Sobol, R. (2004), Improving the Retinex algorithm for rendering wide dynamic
range photographs, *Journal of Electronic Imaging* **13**(1), 65–74.

Tan, R. T., Nishino, K. and Ikeuchi, K. (2004), Color constancy through
inverse-intensity chromaticity space, *Journal of the Optical Society of
America A* **21**(3), 321–334.

Tumblin, J. and Rushmeier, H. (1993), Tone reproduction for realistic images,
IEEE Transactions Computer Graphics and Applications **13**(6), 42–48.

Tumblin, J. and Turk, G. (1999), LCIS: A boundary hierarchy for detail-
preserving contrast reduction, *in* Proc. ACM SIGGRAPH 99, Annual
Conference on Computer Graphics, Los Angeles, CA, pp. 83–90.

Wandell, B. (1995), *Foundations of Vision*, Sinauer Associates, Inc, Sunder-
land, MA.

Ward, G., Rushmeier, H. and Piatko, C. (1997), A visibility matching tone
reproduction operator for high dynamic range scenes, *IEEE Transactions
on Visualization and Computer Graphics* **3**(4), 291–306.

Webster, M. A. (1996), Human colour perception and its adaptation, *Network:
Computation in Neural Systems* **7**(4), 587–634.

West, G. and Brill, M. (1982), Necessary and sufficient conditions for von Kries chromatic adaptation to give color constancy, *Journal of Mathematical Biology* **15**, 249–258.

White, M., Zakia, R. and Lorenz, P. (1977), *The new zone system manual*, Morgan & Morgan, Dobbs Ferry, NY.

Wolff, L. (1994), On the relative brightness of specular and diffuse reflection, *in* Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 1994), Seattle, WA, pp. 369–376.

Xiao, F., DiCarlo, J., Catrysse, P. and Wandell, B. (2002), High dynamic range imaging of natural scenes, *in* Proc. IS&T/SID Tenth Color Imaging Conference: Color Science, Systems, and Application, Scottsdale, AZ, pp. 337–442.

Yoshida, A., Blanz, V., Myszkowski, K. and Seidel, H.-P. (2005), Perceptual evaluation of tone mapping operators with real-world scenes, *in* Proc. IS&T/SPIE Electronic Imaging 2005. The Human Vision and Electronic Imaging X, Vol. 5666, San Jose,CA, pp. 192–203.

Curriculum Vitae

Laurence Meylan

Audiovisual Communications Laboratory II
Ecole Polytechnique Fédérale de Lausanne (EPFL)
1015 Lausanne, Switzerland
`laurence.meylan@a3.epfl.ch`

Personal

Date of birth:	September 26, 1978.
Nationality:	Swiss.
Civil status:	Single.

Education

2002 - 2006 PhD candidate, Computer & Communication Sciences

Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Research topics: high dynamic range images and displays, tone reproduction and color imaging

1997 - 2002 Master's degree, Computer Engineering

Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

1992 - 1996 Science high school diploma with honors

Lycée Rousseau, Neuchâtel, Switzerland

Exchange with other universities:

2001 - 2002 University of Stanford, California, USA (5 months)

Master's thesis as a visiting student in the Computer Science Robotics Laboratory: "Automatic Tuning of a Physical Model through Video and Force Feedback Information"

1999 - 2000 University of Waterloo, Ontario, Canada (2 semesters)

Undergraduate exchange student in Computer Engineering

Professional Experience

Internships:

2004-2005 Sharp Laboratories of Americas, Camas, WA (2 × 4 months)
- Developed Matlab applications for a high dynamic range display prototype, including image and video processing and psychophysical experiments.
- Implemented motion estimation algorithms in C++

2001 Orange Communications, Lausanne, Switzerland (3 months)
- Development of a client server application in JAVA

2000 Cambridge Technology Partners, Geneva, Switzerland (3 months)
- Functional modeling of a reporting application, implemented using SQL Database, Oracle Server, PHP

Academic:

2002-2006 Student project supervision, EPFL
- Seven bachelor's and master's student projects (image processing)

2002-2006 Teaching Assistant, EPFL
- Color Imaging for master's students
- Digital Photography for master's students
- C++ Programming for bachelor's students
- Introduction to Computer Science for first-year students
- Ada Programming for first-year students

PATENTS

L. Meylan and S. Daly, Generating High Dynamic Range Image Data from Low Dynamic Range Image Data by the Use of Spatial Operators, US Serial No. 11/233,747 (filed on 9/22/05).

PUBLICATIONS

L. Meylan, S. Daly and S. Süsstrunk, The Reproduction of Specular Highlights on High Dynamic Range Displays, submitted to CIC05, IS&T/SID 14th Color Imaging Conference: Color Science, Systems, and Application.

D. Alleysson, S. Süsstrunk and L. Meylan, HDR CFA Image Rendering, accepted to Proc. Eurasip 14th European Signal Processing Conference, 2006.

L. Meylan and S. Süsstrunk, High Dynamic Range Image Rendering Using a Retinex-Based Adaptive Filter, accepted to IEEE Transactions on Image Processing, September 2005.

L. Meylan and S. Süsstrunk, The Influence of Luminance on Local Tone Mapping, Proc. IS&T/SID 13th Color Imaging Conference, pp. 276-282, 2005.

L. Meylan and S. Süsstrunk, Color Image Enhancement Using a Retinex-Based Adaptive Filter, Proc. IS&T Second European Conference on Colour in Graphics, Imaging and Vision (CGIV 2004), Vol. 2, pp. 359-363, 2004.

L. Meylan and S. Süsstrunk, Bio-Inspired Image Enhancement for Natural Color Images, Proc. IS&T/SPIE Electronic Imaging 2004: Human Vision and Electronic Imaging IV, Vol. 5292, pp. 46-56, 2004.

LANGUAGES

Native French
Fluent English (More than 2 years working, studying and living in North-America)
Conversational German

COMPUTER SKILLS

Programming Languages: Matlab, C/C++, JAVA, PHP, SQL
Operating systems: Windows, Linux, Unix
Others: Latex, CVS

ACTIVITIES & INTERESTS

Snowboarding and Skiing (certified snowboard instructor)
Volleyball (10 years of competition)
Running and Hiking
Traveling