Controlling illumination to increase information in a collection of images

Asla Medeiros e Sá

Abstract

The solution of several problems in *Computer Vision* benefits from analyzing collections of images, instead of a single image, of a scene that has variant and invariant elements that give important additional clues to interpret scene structure. If the acquisition is done specially for some vision task, then acquisition parameters can be set up so as to simplify the interpretation task. In particular, changes in scene illumination can significantly increase information about scene structure in a collection of images.

In this work, the concept of *active illumination*, that is, controlled illumination that interferes on the scene at acquisition time, is explored to solve some Computer Vision tasks. For instance, a minimal structured light pattern is proposed to solve stereo correspondence problem, while intensity modulation is used to help in foreground/background segmentation task as well as in image tone enhancement.

Resumo

A solução de vários problemas de *Visão Computacional* pode se beneficiar da análise de uma coleção de imagens, ao invés de utilizar uma única imagem, de uma cena que possui elementos variáveis e invariantes capazes de fornecer dicas adicionais importantes para interpretar a estrutura de uma cena. Se a aquisição das imagens for feita especialmente para uma determinada tarefa, então os parâmetros de aquisição podem ser escolhidos de forma a facilitar a interpretação dos dados para a dada tarefa. Em particular, variações nas condições de iluminação de uma cena podem aumentar significativamente a quantidade de informação de uma coleção de imagens sobre a estrutura da cena.

Neste trabalho, o conceito de *iluminação ativa*, isto é, iluminação controlada que interfere na cena em tempo de aquisição, é explorado para resolver algumas tarefas de Visão Computacional. Por exemplo, um padrão minimal de luz estruturada é proposto para a solução do problema de correspondência estéreo; enquanto que a modulação da intensidade da luz é usada para auxiliar a tarefa de segmentação figura/fundo, bem como para melhorar a representação tonal da imagem.

Acknowledgements

First of all I would like to thank Paulo Cezar Carvalho for the patience and constant participation in my academic career. Thanks to Luiz Velho for the enthusiastic discussions. Thanks to *Max-Planck Institut fur Informatik* for the opportunity to visit their Computer Graphics research group, and special thanks to Michael Goesele for his visit to *IMPA* and the valuable comments and suggestions on my work. Special thanks also to Marcelo Gattass and the support given by *Tecgraf*.

Special thanks to Marcelo Bernardes who implemented the video (V4D) based on the structured light coding scheme proposed in this thesis as well as the real-time active tone-enhancement video. Marcelo is also a co-author of the graph-cut segmentation application presented in Chapter 5. Thanks to Anselmo Montenegro who actively participated in discussions and implementation of Chapter 5. Thanks to Michele, the girl on the test images that patiently participated on our experiments. To my *Visgraf* colleagues, specially Victor Bogado, Roberto de Beauclair, Adelailson Peixoto, Esdras Soares, Moacyr Alvim Horta, Vinicius Mello and Serginho Krako. Special thanks also to Google.

Thanks to Patricia Gouvea and Simone Rodrigues for the opportunity to be at *Atelie da Imagem* learning photography. Still concerning photography I also owe many thanks to Cesar Barreto, Frank Scanner and Otavio Schipper.

I would also like to thank my dear friends: Aubin, Re, Ronaldo, Lola, Aline, Paulo e Paula... and Candi. My Yoga instructors Marcio Oliveira, Horivaldo Gomes, Tat Pinheiro, Mercedes Chavarria and to my Yoga colleagues. Rol, Jerome, Li, Marcio, Vitor e pai. E sobretudo ao Raina, nossa Clarice e nossa family.

Contents

Chapter 1

Introduction

Computer Graphics (CG) studies methods for creating and structuring graphics data as well as methods for turning these data into images. *Computer Vision* (CV) studies the inverse problem: given an input image or a collection of input images, obtain information about the world and turn it into graphics data. CG problems are usually stated as direct problems while in CV they are naturally stated as inverse problems. There are several ways to control input data acquisition acquisition in order to ease CV tasks. The knowledge and control on how images are acquired in many cases determines the approaches to be used to solve the CV problem at hand.

A single image of a scene can suffer from lack of information to infer world structure. Many CV systems benefit from analyzing a collection of images to increase information about the world and obtain important clues about scene structure, such as object movement, changes in camera view point, changes in shading, etc. Also CG image processing can benefit from collection of images to synthesize new images, as was widely explored in [ADA*04]. By using collections of images it is possible to identify invariant elements in the set of images; the detection of varying elements together with the knowledge of what caused the variation (camera movement, object movement, changes in lighting, changes in focus, etc.) is helpful to analyze data.

A significant application that benefits from analyzing collection of images in many different ways is 3D Photography, a problem that, to be solved, combines techniques from computer vision, image processing, geometric modeling and computer graphics into an unified framework. The reconstruction of three-dimensional objects from images, illustrated in Figure 1.1, can be used in a vast number of important application fields, ranging from Archeology, Cultural Her-

itage, Art and Education, to Electronic Commerce and Industrial Design.

The development of commodity hardware and consumer electronics makes it possible to build low-cost acquisition systems that are increasingly effective. Some applications demand precision in the acquisition of a scene's radiance properties that up-to-date off-the-shelf sensors can't provide. For instance, to visualize an object with illumination conditions different from the illumination of the ambient where the object was captured, the object capture has to satisfy some requisites that are beyond most sensors capabilities [Len03, Goe04].

In particular, *active illumination* is a powerful tool to increase image information at acquisition time. By active illumination we mean a controllable light source that can be modulated or moved in order to augment scene structure information in a sequence of images, either by controlling shading or by directly projecting information onto the scene. Examples of active illumination in action are shown in Figures 1.1, 1.2 and 1.3.

(a) (b) (c)

Figure 1.1: Images (a) and (b) are images acquired by a photographic digital camera that observes the scene illuminated by projected coded patterns. Stripe boundaries (c) and depth at boundary points can be recovered using structured light principles.

This thesis focuses on controlling illumination to increase image information at acquisition time, that is, to acquire additional information not present in a single shot, by changing scene illumination. By exploring the illumination control we go through different areas of recent research in *Computer Vision*, like shape acquisition from structured light, active image segmentation and tone enhancement.

1.1 Problem Statement

Digital photographic images are projections of a real scene through a lens system onto a photosensitive sensor. At acquisition time, image information can be increased by marking the scene with controlled illumination, this is the principle of *active illumination*. The standard active illumination setup uses a pair camera/projector where the camera produces images of a scene illuminated by the projector in a desired fashion.

(a) (b) (c)

Figure 1.2: Images (a) and (b) have been differently illuminated by varying the camera flash intensity between shots, (c) is the difference thresholded image that can be used to segment objects from non-illuminated backgrounds.

We are particularly interested in exploring the potential of camera/projector pairs. A digital camera can be seen as a non-linear photosensitive black box that acquires digital images, while a projector is another non-linear black box that emits digital images. Their non-linear behavior is a consequence of several technical issues ranging from techniques limitations to market demands to produce beautiful images.

In order for these black boxes to become measurement tools it is mandatory to characterize their non-linear behavior, that is, to perform a calibration step. In some cases, absolute color calibration relating devices to global world references is needed. However, in our case, we will be concerned with the relative calibration of a camera/projector pair, since we only want to guarantee a consistent communication between them.

We explore scene illumination to extract more information of a given scene. The digital projector is our standard active light source. It can project structured light onto the scene in order to recover geometric information (Figure 1.1), and modulate light intensity to help solving background/foreground segmentation (Figure 1.2), or improve tonal information (Figure 1.3).

(a) (b) (c)

Figure 1.3: Images (a) and (b) are two subsequent video input fields. In this experiment a video camera synchronized with a digital projector acquire images with modulated projected light intensity. In (c) it is shown the tonal-enhanced foreground produced from processing together both (a) and (b) frames.

1.2 Chapters Overview

The main reasoning that guides this work is active illumination. Active illumination will be used in different applications with different setups ranging from fine tunned setups of lab environments to cheap home-made setups. Traditionally active illumination is used in 3D photography for depth acquisition. In this work active illumination is used also to solve other problems in Computer Vision. The main overall concepts found in literature that will be useful to the entire work are introduced in Chapter 2.

Photometric calibration enhance the setup performance, and it is mandatory if the setup is to be used as a measurement tool. Calibration will be discussed in Chapter 3, where a basic setup is calibrated. The difference between projected and observed colors is clearly observed in calibration results as well as the non-linear projector intensity behavior. After setup description and calibration we turn into applications.

Coded structured light is a technique applied to recover depth maps from images. In Chapter 4 we propose the design of a minimal coding for structured light with respect to the restrictions imposed on the scene to be scanned. To achieve this minimal coding we revisit the usage of color in code design, we show that using complementary slides we achieve a robust decoding, in addition several reflective restrictions on the object can be removed. The classification of structured light coding strategies proposed in [JPB04] is simplified. We also show an application of the proposed code that permits to acquire depth maps together with scene colors at 30Hz using NTSC off-the-shelf hardware. As a consequence of acquisi-

tion of geometry and texture from the same data the texture-geometry registration problem is avoided.

In Chapter 5 the problem of foreground segmentation using active illumination and graph-cut optimization is discussed. The key idea is that light source positioning and intensity modulation can be designed to affect objects that are closer to the camera and let the background unchanged. Following this reasoning, a scene is lit with two different intensities of a controllable light source that we call segmentation light source. By capturing a pair of images with such illumination, we are able to produce a mask that distinguishes between foreground objects and scene background. The initial segmentation is optimized by graph-cut optimization.

The quality of the masks produced by the method is, in general, quite good. Some difficult cases may arise when the objects are highly specular, translucent or have very low reflectance. Because of its characteristics, the camera parameters settings chosen according to the situation in hand can strongly influence on the quality of the output mask.

In Chapter 6 the concept of relative tone values will be introduced. The fact that relative tones can be recovered, by varying illumination intensity, without knowledge about the camera response function is presented. In our approach, we illuminate the scene with an uncalibrated projector and capture two images of the scene under different illumination conditions. The output of our system is a segmentation mask, together with an image with enhanced tonal information for the foreground pixels. The segmentation and the visualization algorithms are implemented in real-time, and can be used to produce range-enhanced video sequences.

The system is implemented using two different setups. The first uses the same acquisition device built for the stereo correspondence application and is composed of a NTSC camera synchronized with a DLP projector. The second is a home made cheap version of the system that uses a web cam synchronized with a CRT monitor playing the role of the light source.

Although our implementation has been done in real time for video, the same idea could be used in digital cameras by programming flashes. There are many recent works [PAH*04, ED04] that explore the use of programmable flash to enhance image quality, but they do not introduce tone-enhancement concepts.

Conclusions and future work will be discussed in Chapter 7.

1.3 Main Contributions

We list below the main original contribution of this thesis, some of which have already been published:

- Relative photometric calibration of an active pair camera-projector.
- Proposal of (b,s)-BCSL (Figure 1.1), a minimal structured light coding for active stereo correspondence [SCV02, VSVC05].
- Light intensity modulation for active segmentation using graph-cuts (Figure 1.2).
- The concept of relative tones as a tool to tone enhance LDR images without HDR recovery (Figure 1.3) [SVCV05].

Chapter 2

Imaging Devices

Image capture devices measure the flux of photons that were emitted from a light source and interacted with the observed scene. Tasks in Computer Vision are heavily dependent on such devices. For that reason, we are interested in how light sources and the scene behave with respect to visible energy flux that reaches the imaging device. In this chapter we review the most relevant characteristics of light sources, imaging emitters, scene reflectivity properties and imaging capture devices.

2.1 Digital Image

The basic elements of a digital image are the pixel coordinates and the color information at each pixel. Pixel coordinates are related to image spatial resolution while color resolution is determined by how color information is captured and stored. If the instrument used to capture the image is a camera, we obtain a photographic image, that is, a projection of a real scene that passed through a lens system to reach a photosensitive sensor. A photographic image can be modeled as a function $f : U \subseteq R^2 \rightarrow C$ representing light intensity information at each measurement point $p \in U$. The measured intensity values depend upon physical properties of the scene being viewed and on the light sources distribution as well as on photosensitive sensor characteristics.

In order to digitize the image signal that reaches the sensor a *sampling* operation is carried on. *Sampling* is the task of converting the continuous incoming light into a discrete representation. The scene is sampled at a finite number of points, where the intensity function f takes on values in a discrete subset of the color

space C. Color space discretization is also called *quantization*. The *color resolution* of an image is usually expressed by the number of bits used to store the color information. Image sensors are physical implementations of signal discretization operators [GV97].

To visualize the image, a *reconstruction* operation recovers the original signal from samples. Ideally, the reconstruction operation should recover the original signal from the discretized information; however, the result of reconstruction frequently is only an approximation of the original signal. Imaging devices, such as digital projectors, monitors and printers, reconstruct the discrete image to be viewed by the observer. It should be noted that the observer's visual system also plays a role in the reconstruction process.

2.1.1 Modeling Light and Color

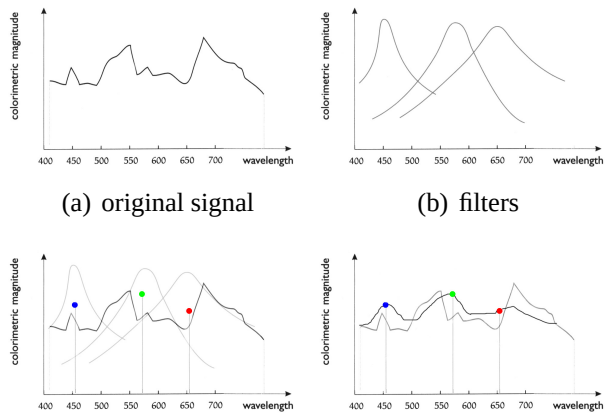
The physics of light is usually described by two different models, *Quantum Optics* describes photons behavior, and *Wave Optics*, models light as an eletromagnectic wave [Goe04]. The most relevant light characteristics that will be useful to us are well described by the eletromagnectic model; *Geometric Optics* is an approximation of wave optics. We will adopt the eletromagnectic approach and geometric optics when it is convenient.

Light sources radiate photons within a range of wavelengths. The energy of each photon is related to its wave frequency by the *Planck's* constant h, that is, $E = hf$. Once the frequency is determined, the associated wavelength λ is also known through the relation $c = \lambda f$. The emitted light can be characterized by its *spectral distribution* that associates to each wavelength a measurement of the associated radiant energy, as illustred in Figure 2.1(a). A source of radiation that emits photons all with the same wavelength is called *monochromatic* [GV97].

A photosensitive sensor is characterized by its *spectral response function* $s(\lambda)$. If a sensor is exposed to light with spectral distribution $C(\lambda)$, the resulting mensurable value is given by

$$w = \int_{\lambda} C(\lambda)s(\lambda)d\lambda$$

The human eye has three types of photosensors with specific spectral response curves. Light sensors and emitters try to mimic the original signal with respect to human perception. The standard solution adopted by industry is to use red, green and blue filters as *primary colors* to sample and reconstruct the emitted light, illustred in Figure 2.1(b). The interval of wavelengths perceived by the human eye is between the range of 380 nm to 780 nm, known as the *visible spectrum*.



(a) original signal (b) filters

(c) sampled signal (d) reconstructed signal

Figure 2.1: The original spectral distribution of an emitted light is shown in (a), the spectral response function of the three filters used to sample the signal are shown in (b), the sampled values are in (c) and the reconstructed signal is shown in (d).

The spectrum of emitted light together with sensor response curves defines the color perceived by an human observer or an imaging system.

Emitters work by superimposing different primary light sources characterized by their *spectral emitting function* $P(\lambda)$. Usually, these primary lights are produced by passing white light, with spectral distribution $C_W(\lambda)$, through red, green and blue filters. The filters are characterized by their spectral distribution $F_i(\lambda)$, where i indexes the different filters. Thus, the spectral emitting function is given by $P_i(\lambda) = F_i(\lambda)C_W(\lambda)$. The signal is also weighted by its emitted intensity value h. The reconstructed signal in trichromatic base is then given by the additive color formation principle:

$$C_r(\lambda) = \sum_{i=1}^{3} h_i P_i(\lambda)$$

2.1.2 Measuring Light

The intensity value registered by a sensor is a function of the incident energy reaching it. It corresponds to the integration of the eletromagnectic energy flux both in time and in a region of space that depends upon the shape of the object of interest, the optics of the imaging device and the characteristics of the light

sources.

Radiometry is the field that study electromagnetic energy flux measurements. The human visual system is only responsive to energy in a certain range of the electromagnetic spectrum, that is the *visible spectrum*. If the wavelength is in the visible spectrum, the radiometric quantities are also described in *photometric* terms. Photometric terms are simply radiometric terms weighted by the human visual system spectral response function [Gla94].

In the literature, the description of visible spectrum can come in radiometric quantities as well as in photometric quantities. This can be a source of confusion and a table relating both quantities is given below:

Quantity Description	**Radiometric Quantity [Unity]**	**Photometric Quantity [Unity]**
Q: basic quantity of energy transported by the wave	Radiant Energy $[Joule]$ $J = \frac{kg.m^2}{s^2}$	Luminous Energy $[talbot]$
Energy per unity of time $\Phi := \frac{dQ}{dt}$	Radiant Flux $[Watt]$ $W = \frac{J}{s}$	Luminous Flux $[lumens]$ $lm = \frac{talbot}{s}$
Flux per solid angle (ω) $I := \frac{d\Phi}{d\omega}$	Radiant Intensity $\frac{W}{sr}$	Luminous Intensity $[candelas]$ $cd = \frac{lm}{sr}$
Flux per area $u := \frac{d\Phi}{dA}$	Irradiance/ Radiosity $\frac{W}{m^2}$	Illuminance/ Luminosity $[lux]$ $lx = \frac{lm}{m^2}$
Flux through a small area from a certain direction $L := \frac{d^2\Phi}{cos\theta.dA.d\omega}$	Radiance $\frac{W}{m^2.sr}$	Luminance $[nit]$ $nit = \frac{cd}{m^2}$

Table 2.1: Radiometric quantities and their photometric counterparts [Goe04].

Most radiometric quantities can be measured in practice with lab instruments. If one intends to use digital cameras as measurement instruments its non-linear response to light intensity must be characterized.

2.2 Image Emitting Devices

Imaging emitters are a special type of light source capable to modulate light intensity spatially in order to reconstruct the digital image desired. In this work we adopt digital projectors to project information onto the scene. Digital projectors

will be modeled with the intention to understand its behavior and their technologic peculiarities will be observed.

2.2.1 Light Sources

Usual light sources contain various elements that shape its radiation pattern; diffusors, mirrors and lenses can be used to characterize, change direction and focus the emitted light. According to their emission, light sources can be classified into point and area light sources.

A *point light source* is characterized by the fact that all light is emitted from a single point in space. For an *uniform point light source* light is emitted equally in all directions. A *spot light* is a uniform point light source that emits light only within a cone of directions. For *textured point light sources* the intensity can vary freely with the emission direction. This model is useful in rendering but rare in real world. Finally, an *area light source* in contrast to point light source emit light from a region in space and is responsible for the presence of *soft shadows* in the scene consisting of *umbra* and *penumbra* regions [Goe04].

2.2.2 Digital Image Projectors

In the rendering context image projectors can be conveniently modeled as a textured spot light source. This model does not take into account the effects resultant of the presence of projector lenses. In this work, images of a real scene with projected patterns will be observed by the camera, so we need a more complete model, capable of a better description of real digital projectors behavior.

Real digital projectors usually are composed of a single lamp whose rays pass through an array of light intensity modulators in order to form an image. After that, light passes through a lens system to be focused at some plane of focus. If we consider that, after being modulated, each point in space is an independent spot light source, then a reasonable model of a general digital projector is to consider an array of spot light sources that passes through a single lens system. Each spot light source from the projector array will be referred as a *projector pixel*. With this model it is possible to simulate the plane of focus and the out-of-focus regions, as well as neighborhood spot light interaction.

The projector lamp is described by its spectral distribution $C_l(\lambda)$. For each projector pixel, a given digital intensity value ρ is to be projected. The actual emitted intensity value is given by the projector *characteristic emitting function* $h(\rho)$, dependent on the projector technology and other factors. To produce colored

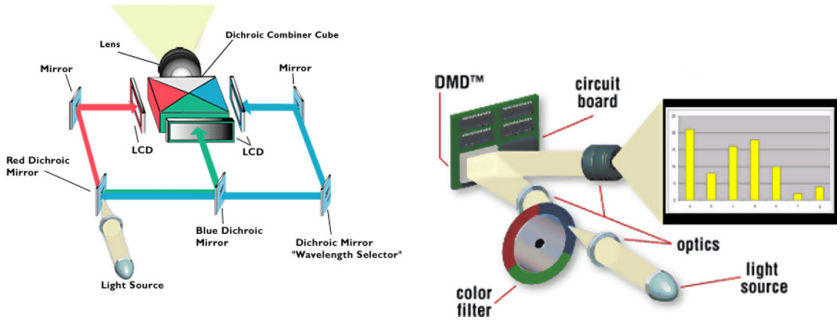
images, color filters with spectral distribution $F_i(\lambda)$ are used, where i indexes color channels. The resultant *spectral emitting function* per channel is $P_i(\lambda) = F_i(\lambda)C_l(\lambda)$. The actual emitted signal at each channel of a projector pixel is then:

$$C_i(\lambda, \rho) = h(\rho)P_i(\lambda)$$

Below, the two most common digital projectors technologies are described in more detail.

LCD vs. DLP technology

LCD (*Liquid Crystal Display*) projectors, illustrated in Figure 2.2 (a), usually contain three separate LCD glass panels, one each for red, green, and blue components of the image signal. As light passes through the LCD panels, individual pixels can be opened to allow light to pass or closed to block the light. This activity modulates the light and produces the image that is projected onto the screen [Powa].



(a) (b)

Figure 2.2: A LCD projector technology (a) compared to a DLP technology (b).

The DLP (Digital Light Processing) chip, Figure 2.2 (b), is a reflective surface made up of one tiny mirror for each pixel. Light from the projector's lamp is directed onto the surface of the DLP chip. The mirrors wobble back and forth, directing light either into the lens path to turn the pixel on, or away from the lens path to turn it off.

In DLP projectors, usually there is only one DLP chip (some have three chips, one per channel); in order to define color, there is a color wheel that filters incoming light. This wheel spins between the lamp and the DLP chip and alternates the

color of the light hitting the chip from red to green to blue. The mirrors tilt away from or into the lens path based upon how much of each color is required for each pixel. This activity modulates the light and produces the image that is projected onto the screen [Powa].

The use of a spinning color wheel to modulate the image has the potential to produce a unique visible artifact on the screen referred as the ”rainbow effect”, which is simply colors separating out in distinct red, green, and blue. Basically, at any given instant in time, the image on the screen is either red, or green, or blue. The technology relies upon human eyes not being able to detect the rapid color changes, what is not always true, especially with respect to frame rate variations.

Consumers can decide on the preferred technology by analyzing its effects in practice. LCD usually delivers a somewhat sharper image than DLP at any given resolution. LCD projectors produce a visible pixelation, clearly reduced on DLPs. DLP technology can produce higher contrast video with deeper black levels than what is usually obtained with an LCD projector. Leading-edge LCD projectors are rated at 1000:1 contrast. Meanwhile, the latest DLP products are rated as high as 3000:1 [Powa].

There are also other technologies used in digital projectors, but DLP and LCD are the most commonly available and cheaper. We will restrict our discussion to them.

2.3 Scene Reflective Properties

The interaction of light and matter is a complex physical process. For graphics purposes, materials can be characterized by their reflective properties. When light reaches an object surface it is partially reflected back to the ambient. The surface *reflectance* is the fraction of the incident flux that is reflected, and it is a function of wavelength, position, time, incident and exitant directions and polarization. In almost all physical materials, surface scattering is linear, that is, energy arriving from each direction contributes independently to the reflection.

By assuming that some materials won’t be present in the scene of interest some useful simplifications can be made to characterize the reflective function. Polarization can be reasonably ignored. Different wavelengths can be assumed to be *decoupled*, that is, the energy at wavelength λ_1 is independent of the energy at λ_2. This excludes *fluorescent materials*, where energy is absorbed at one wavelength and reradiated at another. It can also be assumed that there is no time-dependent behavior, what excludes *phosphorescent materials*. Complex phenomena such as

subsurface scattering, where light is reflected also inside the surface generating multiple scattering events per incident ray, will be also ignored. Considering the above simplifications, the reflectance becomes, to each wavelength, a function of position and incident and exitant directions.

Without subsurface light transport, all light arriving at an object's surface is either reflected or absorbed at the incidence point. This behavior is usually described by the *bidirectional reflectance distribution function* (BRDF) that incorporates the simplifications above. The BRDF function $f_\lambda(p, \omega_i, \omega_o)$ is the ratio of the reflected radiance leaving the surface at a point p in direction ω_o to the irradiance arriving at the same point p from a direction ω_i. Other models incorporate the behaviors not described by BRDFs [Gla94, Goe04].

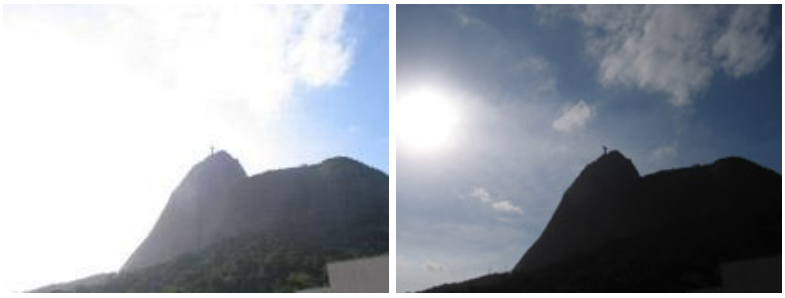
Further simplifications are much more restrictive in terms of real objects. For instance, reflectance of *homogeneous materials* is independent of position; for *isotropic materials* incoming and outgoing directions can be rotated around the surface normal without change, and for *diffuse materials* reflectance is independent of direction. Perfectly homogeneous materials as well as perfectly diffuse materials rarely occurs in real world.

If one have in hand the BRDF of the desired material, the rendering of a virtual object with such aspect is a direct problem treated in Computer Graphics. The acquisition of BRDFs of given real materials is measured in practice with lab instruments. Recently, digital cameras have been used as such instruments [Len03], and the problem is stated as a typical inverse problem heavily dependent on the quality of data acquisition.

Note that if scene, camera and light source are static, then for each camera pixel the surface position and incoming and outgoing directions are well defined. In this case additive behavior of light is preserved by surface reflectance; this property will be useful in Chapter 4.

2.4 Imaging Capture Devices

A digital camera is a device containing a sensor consisting of a grid of photosensitive pixels that convert incident radiance into digital values. A digital photography is acquired by exposing the camera sensor to light during a certain period of time, called *exposure time*. During exposure time, the sensor keeps collecting charge. At the end, the total electric charge collected is converted into digital brightness values. In Figure 2.3 the effect of varying exposure time while keeping all other camera parameters fixed is illustrated.



(a) overexposed sky (b) underexposed hill

Figure 2.3: This images illustrate the resultant acquired images when exposure time is changed. In (a) the sensor has been exposed longer than in (b).

The fundamental information stored in a digital image is the pixel *exposure*, that is, the integral of the incident radiance on the exposure time. A reasonable assumption is that incident radiance is constant during exposure time, specially when small exposure times are used. Thus exposure is given by the product of incident radiance by total exposure time. The incident radiance value w_{ij} is a function of the scene's radiance, optical parameters of the system and the angle between the light ray and system's optical axis. The most obvious way to control exposure is by varying exposure time, that is, by varying the total time that the sensor keeps collecting photons; but other camera parameters can also be controlled to alter exposure in different ways:

- controlling lens aperture;
- changing film/sensor sensitivity (ISO);
- using neutral density filters;
- modulating intensity of light source;

To vary time (as illustred in Figure 2.3) and lens aperture is easy and all professional and semi-professional cameras have these facilities. The disadvantages are related to limitations in applications since long exposures can produce motion blur while lens aperture affects the depth of focus, which can be a problem if the scene has many planes of interest. Film and sensor sensitivity can be altered but the level of noise and graininess are also affected. Density filters are common photographic accessories but its usage depends on an implementation in hardware,

or to be manually changed, which may not be practical. The use of controllable light source is tricky since intensity change depends on the distance of objects to the light source, that is, fails to change constantly on the scene and, in addition, produces shadows.

The photographic industry has been studying acquisition, storage and visualization of images for chemical emulsions since the beginning of the history of photography. Many concepts concerning image quality and accuracy were established since then [Ada80, Ada81, Ada83]. Technical information about photographic material is available for reference in data sheets provided by manufacturers. The standard information provided is storage, processing and reproduction information, as well as its technical curves, shown in Figure 2.4.

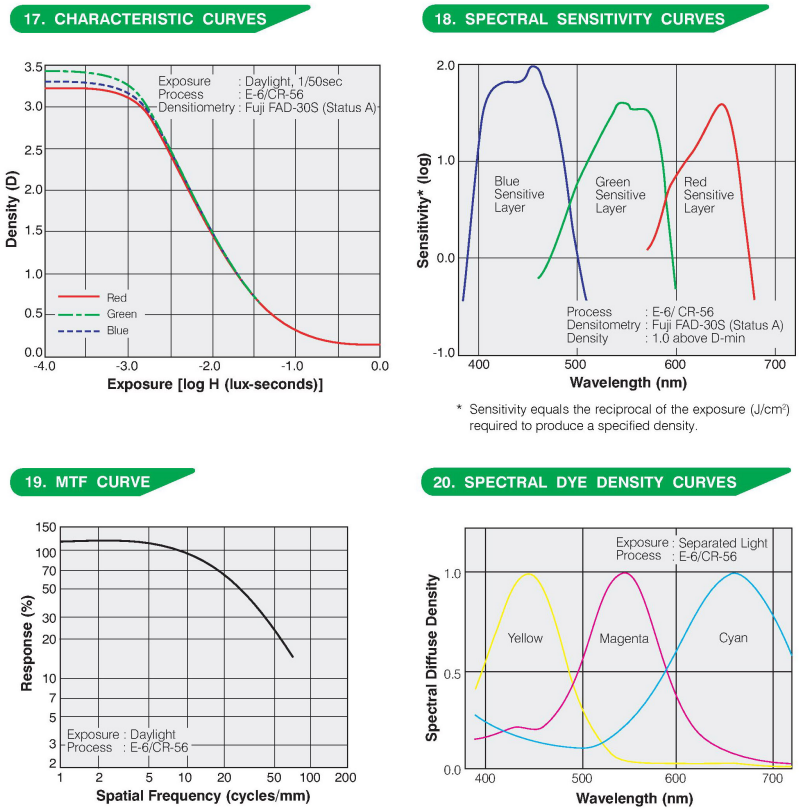


Figure 2.4: FujiChrome Provia 400F Professional [RHP III] data sheet, from FujiFilm.

Film technical curves guide the characterization of emulsions and are the technical base to chose an emulsion adequate for each scene and illumination situation. To characterize emulsion light intensity response, the useful curve is the *characteristic response curve*. *Spectral response curves* considers problems directly related to color reproduction. The *MTF curve* describe the spatial frequency resolution power of the sensible area.

The classical characterization of emulsions can also guide the study of digital sensors, although this information is usually not provided by digital sensors manufacturers. We turn now to the understanding of their role in digital image formation.

2.4.1 Tonal Range and Tonal Resolution

Considering light intensity, the behavior of an imaging sensor is described by its characteristic response function f. The distinct values registered by the sensor are the image *tones*.

In the classical photographic process, the film's photosensitive emulsion is exposed to light during exposure time. The film is then processed to transform the emulsion's latent image into *density* values. The concept of density is central in photography and relates the incoming and outcoming light; for films it is a transmission ratio $D_T = -\log_{10} T$ and for photo papers $D_R = \log_{10} 1/R$ is the reflection ratio, with both T and R in the interval $[0, 1]$ [Ada81]. The characteristic curve of a film is the curve that relates exposure and density. In Figure 2.5 the characteristic curves of different film emulsions are compared. Observe that the film sensitivity to light (ISO) is different for each emulsion.

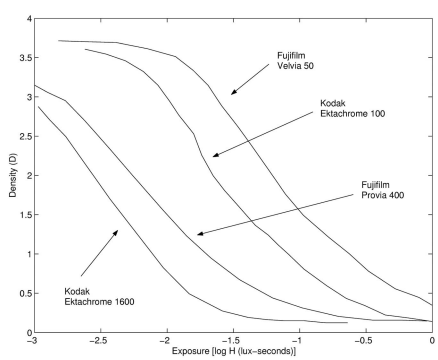
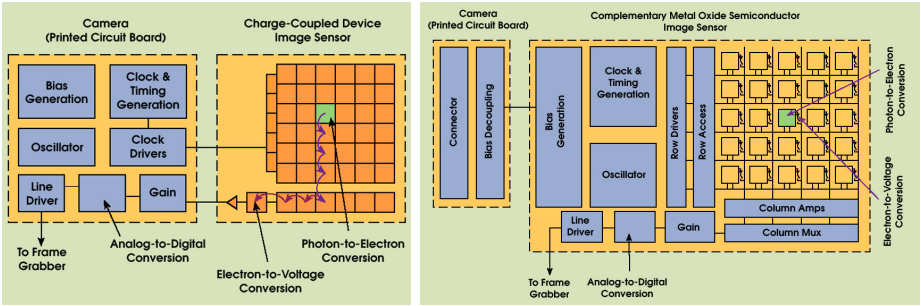


Figure 2.5: Film characteristic response curves compared.

In digital photography, the film emulsions are replaced by photosensitive sensors. The sensor behavior depends on its technology. Usually the stored electrical charge is highly linearly proportional to radiance values. In this case, if the sensor is capable to store a total number of d electrons, then the maximum number of distinct digital brightness values, that is, its *tonal resolution*, will potentially be equal to d. In practice, the digitization process influences on final image tonal resolution. Another important concept is that of *tonal range*, that is the difference between the maximum and the minimum exposure values registered by the sensor.

The leading sensors technologies are CCDs (Charge-Coupled Device) and CMOS (Complementary Metal Oxide Semiconductor) sensors. The main difference between them is that in CCDs every pixel charge is transfered through a very limited number of output nodes to be converted to voltage, as shown in Figure 2.6 (a). Differently, in CMOS sensors each pixel has its own charge-to-voltage conversion, shown in Figure 2.6 (b). This difference implies in several other differences ranging from noise level to manufacturing costs and sensor size [Lit].



(a) (b)

Figure 2.6: (a) CCD sensor and (b) CMOS sensor, from [Lit].

CMOS sensors sensitivity to light is decreased in low light conditions because part of each pixel photosensitive area is covered with circuitry that filters out noise and performs other functions. The percentage of a pixel devoted to collecting light is called the pixels *fill factor*. Most CCDs have a near 100% fill factor while CMOS usually have much less. To compensate for lower fill-factors, micro-lenses can be added to each pixel to gather light from the insensitive portions of the pixel and focus it down to the photosensitive area.

Although the sensor's natural behavior is linear, due to perceptive reasons the final brightness value stored in the image is non-linearly related to radiance. This

non-linear behavior is characterized by the camera response curve. Only some scientific cameras keep the sensor natural linear behavior to produce the final image.

The function $f : [E_{min}, E_{max}] \rightarrow [0, M]$ actually maps sensor exposure to brightness values, where E_{min} and E_{max} are respectively the minimum and the maximum exposure values measurable by the sensor, and M is the maximum digitized value. The function f is in the core of the image formation process. In most cases, f is non-linear and the application of f^{-1} is required to make meaningful comparisons between brightness values of differently exposed images.

Another important concept is that of *dynamic range*, it is the ratio of the highest to the lowest in a set of values; in the image context these values are light intensity values. The fact that the range is dynamic is due to the possibility to control exposure by varying camera parameters, thus changing the maximum and the minimum radiance values related to the same exposure range.

In photography, dynamic range – also referred as film or photopaper *latitude* – is given in terms of stops, which is a log_2 scale. Films produce a density range of about 7 stops (that is, 128:1, or two orders of magnitude in base 10). Photographic paper has a much lower dynamic range, equivalent to 4 or 5 stops (approximately 20:1). Several techniques are adopted in the printing process to overcome this gap. The design of photographic materials has evolved to the goal of optimal response for human viewing under a variety viewing conditions, and is well known that contrast plays a huge role in achieving good images.

Sensors convert an analog signal into a digital signal, so its characteristics define the signal discretization step. The dynamic range defines the range of tones that can be stored by the chosen media. Digital sensors and displays, independent of their accuracy, represent a discrete interval of the continuous infinite range of real luminances, so tonal resolution is influenced by the number of bits n used to describe it.

There is a subtle difference between tone resolution and the tonal range of radiances spanned in an image. Tonal range is related to the total size of the interval that can be perceived by a sensor, while the tone resolution is related to the sample frequency, that is, on how many tones are represented given a fixed interval. Tonal range can be changed without altering n while changing n not necessarily changes the total range; both changes have influence on the final resolution. Intuitively the total range is the maximum contrast reproduced by the media, while the resolution influences on the tonal smoothness reproduction.

2.4.2 Color Reproduction

As mentioned before, light sensors and emitters try to mimic the scene's light signal concerning human perception; it is the human perception that is important concerning colors reproduction. Inspired on the trichromatic base of the human eye, the standard solution adopted by industry is to use red, green and blue filters, referred as RGB base, to sample the input light signal and also to reproduce the signal using light based image emitters. Printers work on a different principle, a discussion about them is out of the scope of this work.

Photographic color films usually have three layers of emulsion, each with a different spectral curve, sensitive to red, green and blue light respectively. The RGB spectral response of the film is characterized by spectral sensitivity and spectral dye density curves (see Figure 2.4).

Electronic sensors are by nature, sensitive to the entire visible spectrum and also to infrared wavelengths. In order to sample the input signal in RGB tristimulus base, colored filters are attached to the sensors. To each sensor pixel only one filter is attached, this implies the adoption of solutions like the usage of a Bayer pattern as shown in Figure 2.7. In the Bayer pattern, the green channel is sampled twice more than the red and green channels. this design choice is also based on human perception.

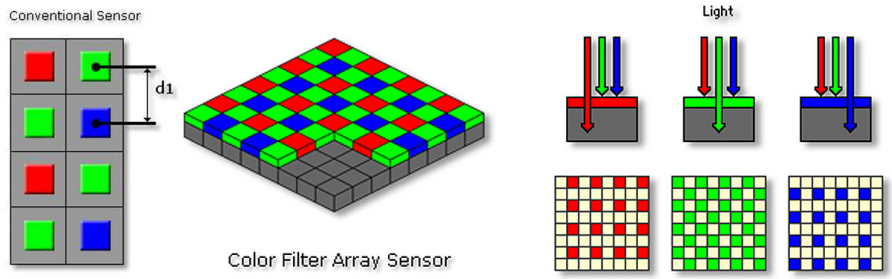


Figure 2.7: Bayer pattern (from http://www.dpreview.com/learn/ by Vincent Bockeart).

Other solutions can also be adopted. The most common alternative is to use three sensors, one for each channel; recently, a new sensor technology was proposed that mimics the behavior of a color film and captures RGB values in the same sample point. Both solutions are not cheap, and most consumer cameras use Bayer patterns to sample light in the RGB base.

2.4.3 Spatial resolution

The total size of the sensor, the size of an individual pixel and their spatial distribution determine the image spatial resolution and the resolution power of the sensor. The spatial resolution power is related to the image sampling process. A tool of fundamental importance in the study of spatial resolution and accuracy issues is the sampling theorem [GV97]:

Theorem 1 (The Shannon-Whittaker sampling theorem) *Let g be a band-limited signal and Ω the smallest frequency such that $\sup \hat{g} \subset [-\Omega, \Omega]$, where $\hat{g}$ is the Fourier transform of g. Then g can be exactly recovered from the uniform sample sequence $\{g(m\Delta t) : m \in Z\}$ if $\Delta t < 1/(2\Omega)$.*

In other words, if the signal is bandlimited to a frequency band going from 0 to ω cycles per second, it is completely determined by samples taken at uniform intervals at most $1/(2\Omega)$ seconds apart. Thus we must sample the signal at least two times every full cycle [GV97]. The sampling rate $1/(2\Omega)$ is known as the *Nyquist limit*. Any component of a sampled signal with a frequency above this limit is subject to *aliasing*, that is, a high frequency that will be sampled as a low frequency.

The number of sensor's pixels defines the image grid, that is, its spatial resolution in classical terms; but their physical size and spatial distribution also influences on the resolution power of the imaging device. Since the pixel spacing δ is uniform, the sensor Nyquist frequency is given by $1/(2\delta)$. Note that the adoption of Bayer pattern alters this δ value altering sensor Nyquist frequency for each color channel.

Modulation Transfer Function

Between light and sensor there is the camera lens system, which has its own resolution that influences on the final camera resolution. Lenses, including eye, are not perfect optical systems. As a result when light passes through it undergo a certain degree of degradation. The Modulation Transfer Function (MTF) (see Figures 2.8 and 2.4) shows how well a spatial frequency information is transfered from object to image. It is the Fourier transform of the point spread function (PSF) that gives the scattering response to an infinitesimal line of light and is instrumental in determining the resolution power of a film emulsion or a lens system.

Lens and film manufacturers provide the MTF curves of their lenses and film emulsions. It is useful for a photographer to interpret these curves, see Figure 2.9, in order to chose which is better for his requirements.

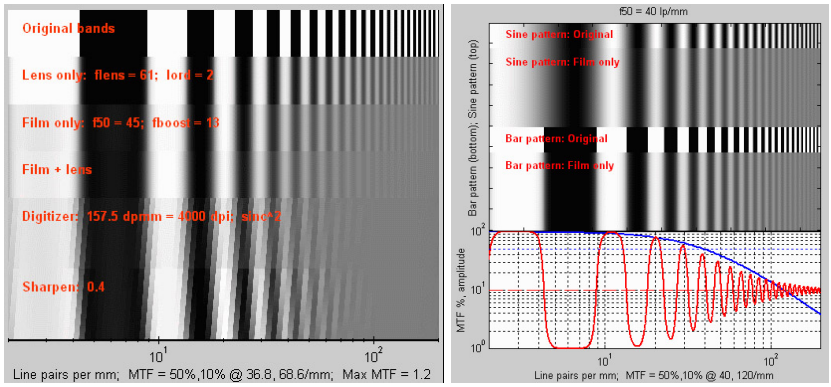


Figure 2.8: Images taken from http://www.normankoren.com/Tutorials/MTF.html illustrates the effects of MTF on the input target. The blue curve below the target is the film MTF, expressed in percentage; the red curve shows the density of the bar pattern.

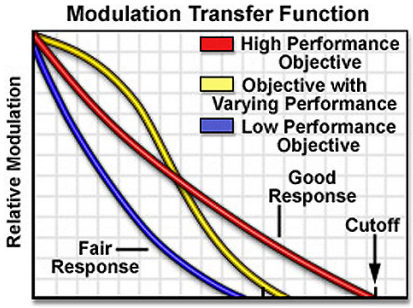


Figure 2.9: MTF of different lenses compared

In summary, image spatial resolution power is determined by imaging system lenses and sensor's pixels physical spacing. Usually, if the chosen lens is high quality, pixel dimension and spacing is the critical information to be considered to evaluate spatial resolution power.

2.4.4 Noise, Aberrations and Artifacts

Many distortions on measurement can be caused by electric phenomena like dark current, thermal noise, charge overflowing to neighboring sensors, etc. Dark current means that a pixel may exhibit non-zero measurements even when there is no incoming photon. The longer the exposure time is, the more dark current noise is

accumulated. Cooling the sensor can be of great help since noise can double with every increase in temperature of about 6 Kelvin [Goe04]. Bright spots can create large currents and the charge overflows to neighboring pixels leading to blooming artifacts.

Concerning pixel size, small pixels respond to fewer photons and can hold fewer electrons. Thus, although they allow for finer spacing, they suffer from increased noise, that is, poorer signal-to-noise ratio (SNR), reduced exposure range (fewer f-stops), and reduced sensitivity (lower ISO speed). Large pixels have good SNR, ISO speed and exposure range, but suffer from aliasing.

Concerning sensors total size, a well known issue for large format photography lovers, small sensors are affected by lens diffraction, which limits image resolution at small apertures – starting around f/16 for the 35mm format. At large apertures – f/4 and above– resolution is limited by aberrations; these limits are proportional to the format size. Large sensors are costly. For reference, sensor diagonal measurements are 43.3 mm for full frame 35mm film; up to 11 mm for compact digital cameras, and 22 mm and over for digital SLRs. Large format cameras are famous for they image resolution power.

Digital image post-processing can introduce artifacts. For instance, a side effect of Bayer pattern adoption is that the reconstruction of RGB values for each pixel uses information of neighboring pixels, the spatial measurement displacement can then introduce chromatic artifacts.

In this thesis the post-processing is reduced as much as allowed by the camera manufacturers. This is done intending to preserve as much as possible the original sensor measurement.

Chapter 3

Active Setup

An active setup is composed by a controllable light source that influences on scene illumination and a camera device. The most commonly used active setup is a pair camera/projector. Technical properties of devices are chosen according to the requirements of the scene of interest. In this work the focus is on objects with dimensions comparable with a vase or a person. In most cases the acquisition is done in a controlled ambient, which means that background and ambient light can be controlled.

Digital cameras and projectors devices act like non-linear black-boxes that convert light signal into digital images and vice-versa. For these devices to become measurement instruments their non-linear behavior must be characterized.

The characterization of device behavior is a calibration process. To calibrate a device is basically to compare its behavior to some global reference values. In the case of geometric calibration, for example, the calibration is performed to find the device spatial coordinates relatively to a world coordinate system. Analogously, color calibration is usually performed using test targets as global reference values, the task is to classify the device behavior according to these global references.

In some cases global references are more than what is needed, and it is enough to situate the device behavior relatively to some other device. This is the case of projector geometric calibration for active stereo applications: what matters is the projector position relatively to the camera position; its world coordinates are less important.

In this chapter the devices calibration process and the obtained results of calibration of an specific setup is discussed.

3.1 Camera Calibration

Geometric Calibration: Camera geometric position in space can be derived from images by observing the projective deformations of geometric calibration targets and exploring principles of projective geometry. The knowledge of devices geometric position is fundamental in some applications like depth information recovery from photographs, 3D scanning, etc. In this work devices geometric position are not required for the studied applications.

Photometric Calibration: Considering devices photometric behavior, it has been already mentioned that, given a sensor's *spectral response function* $s(\lambda)$, if a sensor is exposed to light with spectral distribution $C(\lambda)$, the actual incoming value to be registered by the sensor is given by

$$w = \int_{\lambda} C(\lambda)s(\lambda)d\lambda.$$

It is also known that sensors pixels ij respond to exposure values

$$E_{ij} = w_{ij}\Delta t,$$

where Δt is the exposure time. Consequently, the actual digitized value d_{ij} is a function of the values w_{ij}. Thus, a full sensor photometric calibration should characterize the response function

$$d_{ij} = f(w_{ij}\Delta t)$$

as well as the RGB filters spectral functions $s(\lambda)$.

Note that the signal $C_{ij}(\lambda)$ cannot be recovered unless the calibration is done to each monochromatic wavelength λ and $s(\lambda)$ is known. In addition, it is possible that different input signals $C_{ij}(\lambda)$ at pixel ij produces equal responses w_{ij}, that is, the signals are sensor's *metameric* signals.

Noise: Sensor noise also influences on the image formation process. In this work we referred to technical references of the adopted devices to choose parameters that minimize sensors noise. In Figure 3.1 the behavior of noise respect to the chosen ISO sensitivity for different camera models is illustrated.

In this thesis no noise reduction post-processing is applied.

Spatial Resolution Power: To complete the characterization of a camera device, its spatial resolution power should be considered. This issue is related to the characterization os its MTF curve. In Figure 3.2 the image of a test target used to

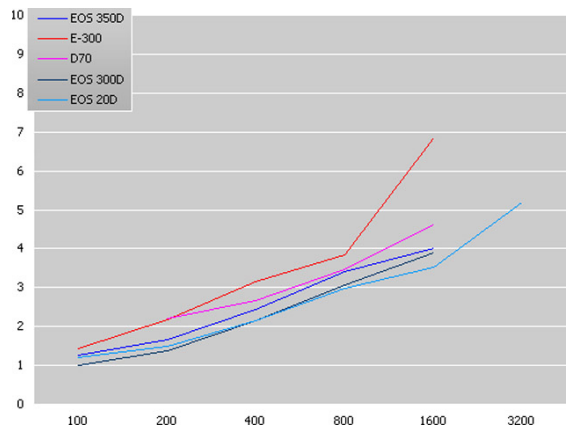


Figure 3.1: Some cameras noise compared. Indicated ISO sensitivity is on the horizontal axis of this graph, standard deviation of luminosity (normalized image) on the vertical axis. Image from the site http://www.dpreview.com.

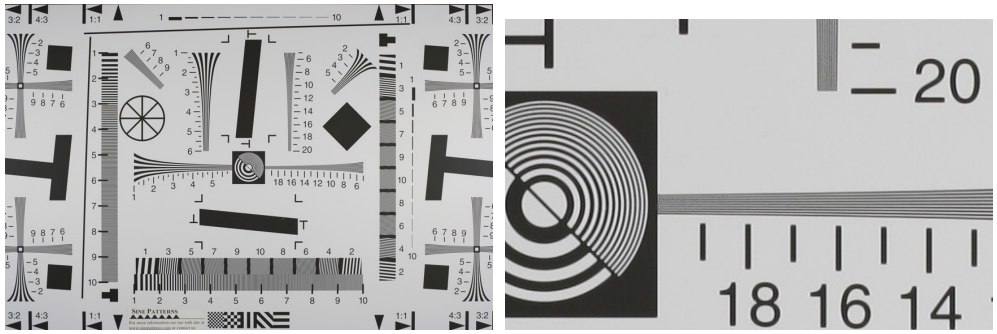
analyze the camera resolution power is shown. It can be observed that the finest spatial frequencies are not well reproduced in the acquired image.

In this work issues related to spatial resolution power are not required for the desired applications and we leave the discussion to a future work. We turn now to the discussion of camera photometric calibration.

3.1.1 Intensity Response Function

Intensity response calibration is responsible for the characterization of the response function f. As the d_{ij} values are non-linearly related to scene radiance values w_{ij}, it is mandatory to recover the characteristic sensor response function f in order to linearize data and perform meaningful comparisons between differently exposed d_{ij} values. As f is reasonably assumed to be monotonically increasing, thus its inverse f^{-1} is well defined. The recovery of f from observed data has been extensively studied in recent years. Most methods are based on the usage of a collection of differently exposed images of a scene as input. [DM97, GN03a, GN04, GHS01]

A collection of N differently exposed pictures of a scene acquired with known variable exposure times Δt_k gives a set of d_{ij}^k values for each pixel ij, where k is the index on exposure times. Although f is modeled as a continuous function, what can be observed are its discrete values registered by the sensor. The discrete response function $\hat{f}$ associated to f includes in its modeling important sensors



(a) (b)

Figure 3.2: (a) Camera spatial resolution power test target. (b) A detail of the same image. The camera used in this test was a Canon EOS 350D and the image is from the site http://www.dpreview.com.

characteristics such as noise.

Considering sensor's noise η_{ij}, the actual value to be digitized is given by $z_{ij}^k = E_{ij}^k + \eta_{ij} = w_{ij}\Delta t_k + \eta_{ij}$. As the digitization function is discrete, if $z_{ij}^k \in [I_{m-1}, I_m)$, where $[I_{m-1}, I_m)$ is an irradiance interval, then $d_{ij}^k = \hat{f}(z_{ij}^k) = m$. The discrete response function $\hat{f}$ is then:

$$\hat{f}(z) = \begin{cases} 0 & if \quad z \in [0, I_0), \\ m & if \quad z \in [I_{m-1}, I_m), \\ 2^n & if \quad z \in [I_{2^n-1}, \infty) \end{cases}$$

where $m = 0, ..., 2^n$, with n the number of bits used to store the information (in practice, the maximum is not required to be equal to 2^n, but here we will consider this for notation simplicity). The monotonically increasing hypothesis imposes that $0 < I_0 < ... < I_m < ... < I_{2^n-1} < \infty$. Thus an inverse mapping can be defined by $\hat{f}^{-1}(m) = I_m$.

If $\hat{f}(z_{ij}^k) = m$ then $\zeta_{ij} = I_m - z_{ij}^k$ is the quantization error at pixel ij, thus:

$$\begin{aligned} \hat{f}^{-1}(m) &= z_{ij}^k + \zeta_{ij} \\ &= w_{ij}\Delta t_k + \zeta_{ij} \\ \hat{f}^{-1}(m) - \zeta_{ij} &= w_{ij}\Delta t_k \\ w_{ij} &= \frac{\hat{f}^{-1}(m) - \zeta_{ij}}{\Delta t_k} \end{aligned}$$

If enough different irradiance values are measured – that is, at least one meaningful digital value is available for each mapped irradiance interval – then $\hat{f}^{-1}$

mapping can be recovered for the discrete m values. To obtain f in all its continuous domain some assumptions must be imposed on the function such as continuity or smoothness restrictions. In some cases parameterized models are used but it can be too restrictive and some real-world curves may not match the model.

At this point, a question can be posed: *What is the essential information necessary and sufficient to obtain cameras characteristic response functions from images?*

In [GN03a] the authors define the intensity mapping function $\tau : [0,1] \rightarrow [0,1]$ as the function that correlates the measured brightness values of two differently exposed images. This function is defined at several discrete points by the accumulated histograms H of the images, given by $\tau(d) = H_2^{-1}(H_1(d))$ [1], and expresses the concept that the m brighter pixels in the first image will be the m brighter pixels in the second image for all m [2]. Then the following theorem is derived:

Theorem 2 (Intensity Mapping [GN03a]) *The histogram h_1 of one image, the histogram h_2 of a second image (of the same scene) is necessary and sufficient to determine the intensity mapping function τ.*

The referred function τ is given by the relation between two corresponding tones in a pair of images:

Let

$$\begin{aligned} d_{ij}^1 &= f(w_{ij}\Delta t_1) \\ d_{ij}^2 &= f(w_{ij}\Delta t_2) \end{aligned} \tag{3.1}$$

then

$$\begin{aligned} d_{ij}^1 &= f\left(\frac{f^{-1}(d_{ij}^2)}{\Delta t_2}\Delta t_1\right) \\ &= f(\gamma f^{-1}(d_{ij}^2)) \end{aligned} \tag{3.2}$$

where $\gamma = \frac{\Delta t_1}{\Delta t_2}$

that is

$$\begin{aligned} d_{ij}^1 &= f(\gamma f^{-1}(d_{ij}^2)) \\ &= \tau(d_{ij}^2) \end{aligned} \tag{3.3}$$

The answer to the posed question is that τ, together with the exposure times ratio are necessary and sufficient to recover f. Here the conclusions were derived

[1] Supposing that all possible tones are represented in the input image, the respective accumulated histogram H is monotonically increasing, thus the H^{-1} inverse mapping is well defined.

[2] H and τ are considered as continuous functions although in practice they are observed at discrete points and their extension to continuous functions deserves some discussion.

from an ideal camera sensor without noise. But note that τ is obtained observing the accumulated histograms, that are less sensitive to noise than the nominal values.

Response curve f from observed data

Different approaches can be used to recover the f function from observed data. In what follows some methods will be described. Each of these methods has its advantages and drawbacks.

In [DM97], the continuous f is recovered direct from the observed data by finding a smooth $g(d_{ij}^k) = lnf^{-1}(d_{ij}^k) = lnw_{ij} + ln\Delta t_k$ using an optimization process. Only a subset of correspondent image pixels from a set of differently exposed images are used. The selection of a subset of pixels is needed otherwise the formulated optimization problem would be too large with a lot of redundant information. Then, f^{-1} is applied to recover the actual scene irradiance by applying $w_{ij} = \frac{f^{-1}(d_{ij}^k)}{\Delta t_k}$.

In [GN03a], the images accumulated histograms are used to obtain the intensity mapping function τ. Then, a continuous f^{-1} is obtained assuming that it is a sixth order polynomial, and solving the system given by equation $f^{-1}(\tau(d)) = \gamma f^{-1}(d)$ on the coefficients of the polynomial. Two additional restrictions are imposed: that no response is observed if there is no light, that is, $f^{-1}(0) = 0$, assuming also that $f : [0, 1] \rightarrow [0, 1]$, $f^{-1}(1) = 1$ is fixed, which means that the maximum light intensity leads to the maximum response. Note that there is no guarantee that the obtained f is monotonically increasing.

The usage of accumulated histograms has some advantages: all the information present in the image is used instead of a subset of pixels, the images need not to be perfectly registered since spatial information is not present on histograms, and they are less sentive to noise. Note that, if an histogram approach is used, the spatial registration between image pixels is not necessary to find f, but pixel correspondences cannot be neglected when reconstructing the Radiance Map.

The same authors, in [GN04], study the space of camera response curves. It is observed that, although the space $W_{RF} := \{f|f(0) = 0, f(1) = 1$ and f is monotonically increasing$\}$ of normalized response functions is of infinite-dimension, only a reduced subset of them arise in practice. Based on this observation it is created a low-parameter empirical model to the curves, derived from a database of real-world camera and film response curves.

In [RBS99] the discrete version of the problem is solved by an iterative optimization process, an advantage is that $\hat{f}$ do not need to be assumed to have a

shape described by some previously defined class of continuous functions. The irradiance values w_{ij} and the function $\hat{f}$ are optimized at alternated iterations. As a first step, the quantization error $\zeta_{ij}^k = I_m - z_{ij}^k = I_m - w_{ij}\Delta t_k$ is minimized with respect to the unknown w using

$$O(I, w) = \sum_{(i,j),k} \sigma(m)(I_m - w_{ij}\Delta t_k)^2 \tag{3.4}$$

The function $\sigma(m)$ is a weighting function chosen based on the confidence on the observed data. In the original paper $\sigma(m) = \exp\left(-4\frac{(m-2^{n-1})^2}{(2^{n-1})^2}\right)$.

By setting the gradient $\nabla O(w)$ to zero, the optimum w_{ij}^* at pixel ij is given by

$$w_{ij}^* = \frac{\sum_k \sigma(m)\Delta t_k I_m}{\sum_k \sigma(m)\Delta t_k^2} \tag{3.5}$$

In the initial step f is supposed to be linear, and the I_m values are calculated using f. The second step iterate f given the w_{ij}. Again the objective function 3.4 is minimized, now with respect to the unknown I. The solution is given by:

$$I_m^* = \frac{\sum\limits_{((i,j),k)\in\Omega_m} w_{ij}\Delta t_k}{\#(\Omega_m)} \tag{3.6}$$

where $\Omega_m = \{((i,j),k) : d_{ij}^k = m\}$ is the index set and $\#(\Omega_m)$ is its cardinality. A complete iteration of the method is given by calculating 3.5 and 3.6, then scaling of the result. The process is repeated until some convergence criterion is reached.

We observe that, as originally formulated, there is no guarantee that the values I_m obtained in 3.6 are monotonically increasing. Especially in the presence of noise this assumption can be violated. If I_m^* are not increasing, then the new w_{ij}^* can be corrupted, and the method does not converge to the desired radiance map. The correct formulation of the objective function should include the increasing restrictions:

$$\begin{aligned} O(I, w) \quad &= \sum_{(i,j),k} \sigma(m)(I_m - w_{ij}\Delta t_k)^2 \\ s.a \quad & 0 < I_0 < \cdots < I_m < \cdots < I_{2^n-1} < \infty \end{aligned} \tag{3.7}$$

This new objective function is not easily solved to the unknown I as the original one. In the next section the iterative optimization method is applied to recover the f response function of the cameras used in the proposed experiments. The approaches used to deal with the increasing restrictions will then be discussed.

Another observation is that although the I_m values were modeled as the extreme of radiance intervals, the calculated I_m^* are an average of their correspondent radiance values.

3.1.2 Spectral Calibration

The RGB values recorded for a color patch depends not only on light source spectral distribution and scenes reflective properties but also on spectral response of the filters attached to camera sensors. To interpret meaningfully the RGB values each spectral distribution should be characterized separately.

An absolute spectral response calibration is the complete characterization of RGB filters spectral behavior, that is, to recover $s(\lambda)$. It would be possible to characterize $s(\lambda)$ if measurements at each monochromatic wavelength λ were done separately, but that is not possible with commonly available light sources.

To the graphical arts and printing industry, color values have to be comparable in order to achieve consistent colors throughout the processing flow. What is done in practice is to adopt a color management systems (CMS) to ensure that colors remain the same regardless of the device or medium used. The role of a CMS is to provide a profile for the specific device of interest that allows to convert between its color space and standard color spaces. Usually standard color charts are used to characterize the device color space, in this case, a photograph of the given chart is taken and based on the registered RGB values the device color space is inferred. The core information of a device profile is in most cases a large lookup table which allows to encode a wide range of transformations, usually non-linear, between different color spaces [Goe04].

Another issue related to color calibration is the compensation of light source spectral distribution. If scene illumination is different from white, what occurs with the most common illuminants like incandescent and fluorescent lighting, then the measured values will be biased by the illuminant spectral distribution.

The human eye has a chromatic adaptation mechanism that preserves approximately the colors of the scene despite the differences caused by illuminants. Digital imaging systems can not account for these shifts in color balance, and the measured values should be transformed to compensate for illuminant chromatic distortions.

Many different algorithms can perform color balancing. A common approach usually referred as *white balance* is a normalize-to-white approach. There are several versions of white balance algorithm, but the basic concept is to set at white (W_R, W_G, W_B) a point or a region that should be white in the real scene. One ver-

sion of the white balance algorithm sets values $(max(R), max(G), max(B))$, the maximum values of the chosen white region, at a reference white (W_R, W_G, W_B). The colors in the image are then transformed using:

$$(R', G', B') = (\frac{W_R}{max(R)}R, \frac{W_G}{max(G)}G, \frac{W_B}{max(B)}B)$$

In the case of this thesis, the interest is on a relative correlation between a projector color space and a camera color space, thus no color charts are used. The calibration is done based on projected color information. Regarding white balance, we chose to minimize the post-processing of acquired data, thus white balancing is not performed. We also assume that in calibration phase, ambient light is not present. In the case of applications where this assumption is not valid, the adoption of white balance may be reconsidered.

3.2 Projector Calibration

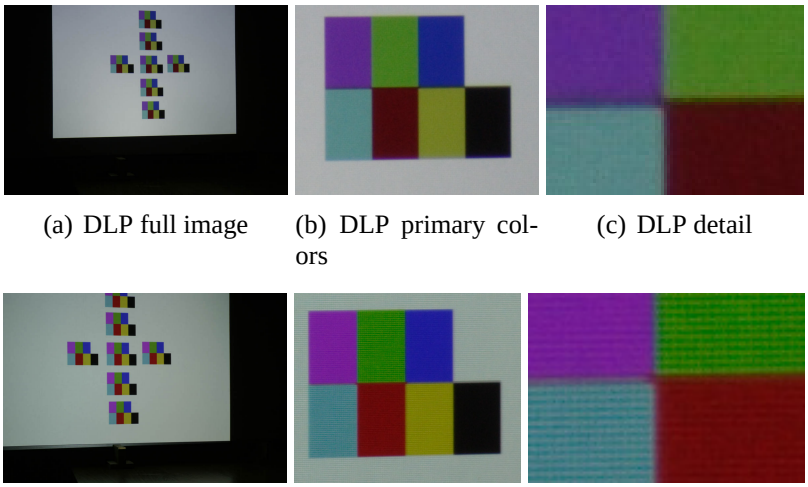
All measurements of projected colors are to be done through the camera. Thus projector calibration becomes an indirect problem dependent on camera calibration. The camera calibration errors are then propagated to projector calibration.

Geometric Calibration: Light source geometric position can be recovered relative to camera position. If the light source is not included in the camera scene composition, the use of mirrors or reflective spheres to localize its position in the ambient is useful [Len03]. Projector geometric calibration can benefit from projective principles. A calibration pattern can be projected allowing the recovery of projector position by observing projective deformations on the pattern []. In this work we are not concerned with geometric calibration.

Photometric Calibration: The actual value emitted by the light source relative to the projected nominal value ρ is given by its characteristic emitting function $h(\rho)$, that is reasonably assumed to be monotonically increasing. It is also known that to project light in RGB basis, the projector has color filters with characteristic spectral emitting function $P(\lambda)$. Thus, a full projector photometric calibration should characterize the emitting function $h(\rho)$ as well as the RGB filters spectral emitting function $P(\lambda)$. The characterization of $P(\lambda)$ for each wavelength requires specific measurement instruments and cannot be done by using common photographic cameras.

Spatial Resolution Power and Noise: To complete the projector behavior characterization, issues related to spatial resolution power and noise should also be

analyzed. To illustrate the issue two projector technologies, a DLP and an LCD projectors, are compared in Figure 3.3. It is possible to observe the spatial intensity variation as well as noise in the cropped detail.



(a) DLP full image (b) DLP primary colors (c) DLP detail

(d) LCD full image (e) LCD primary colors (f) LCD detail

Figure 3.3: DLP vs. LCD spatial resolution and noise behavior.

In this work the discussion on spatial resolution and level of noise is left as a future work. Instead, we analyze the average of regions uniformly illuminated by the projector to perform projector photometric calibration. Recall that, in the case of this work, projector calibration is always dependent on the camera characteristics. We turn now to the discussion of projectors photometric calibration.

3.2.1 Intensity Emitting Function

The actual projected intensity $h(\rho)$ is a monotonically increasing function of the projected nominal value ρ. The non-linear relation of ρ to the observed camera value is given by $d(h(\rho)) = f(w(h(\rho))\Delta t)$. The value $w(h(\rho))$ that reaches the camera sensor is a result of the projector lamp spectral distribution passing through both camera and projector color filters and is described by:

$$w(h(\rho)) = \int_\lambda C_{h(\rho)}(\lambda)s(\lambda)d\lambda \tag{3.8}$$

where $C_{h(\rho)}(\lambda) = h(\rho)P(\lambda)$ for spectral emitting function $P(\lambda)$, thus

$$w(h(\rho)) \quad = \quad \int_\lambda h(\rho)P(\lambda)s(\lambda)d\lambda \tag{3.9}$$

It is reasonable to assume that $h(\rho)$ is not dependent on λ. By observing the projectors technologies we know that a single light source with fixed spectral distribution passes through RGB pre-defined color filters, this implies that the intensity modulation proportioned by ρ should act like a neutral density filter and alters the whole signal in the same way, consequently:

$$w(h(\rho)) \quad = \quad h(\rho) \int_\lambda P(\lambda)s(\lambda)d\lambda \tag{3.10}$$

For an RGB based system the camera has three spectral response curves $s_q(\lambda)$, where $q = R, G$ or B, as well as projector has three spectral emitting curves $P_r(\lambda)$ where $r = R, G$ or B. This gives rise to nine $S_r^q(\lambda)$ combined spectral curves that characterize the pair camera/projector, in addition, as the spectral functions are fixed, nine constant factors arise:

$$\kappa_r^q = \int_\lambda P_q(\lambda)s_r(\lambda)d\lambda = \int_\lambda S_r^q(\lambda)d\lambda$$

For an ideal camera/projector pair $\kappa_r^q = 0$ if $q \neq r$. Assuming that ambient light is set to zero, the projector become the only scene illuminant. It is reasonable to assume that h is the same for all the three channels by observing the projectors technologies described in previous Chapter. For each emitted intensity $h(\rho)$, there is a correspondent $w(h(\rho))$ value, both have three channels of information, that is, the system that relates the actual projected intensity to the intensity values that reaches the sensor is linear and given by:

$$\underbrace{\begin{bmatrix} w^R \\ w^G \\ w^B \end{bmatrix}}_{w} = \underbrace{\begin{bmatrix} \kappa_R^R & \kappa_G^R & \kappa_B^R \\ \kappa_R^G & \kappa_G^G & \kappa_B^G \\ \kappa_R^B & \kappa_G^B & \kappa_B^B \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} h(\rho^R) \\ h(\rho^G) \\ h(\rho^B) \end{bmatrix}}_{h(\rho)}$$

The matrix K characterizes the spectral behavior of the pair camera/projector, and it will be referred as the *spectral characteristic matrix*. It is expected that K is near diagonal, that is, $\kappa_r^q \approx 0$ if $q \neq r$, and all its entries are nonnegative, in addition, its diagonal entries should be strictly positive. For ideal pairs camera/projector K is the identity.

Ambient light can be added to the model by summing up its contribution:

$$w \quad = \quad Kh(\rho) + c \tag{3.11}$$

Emitting curve f from observed data

It is easy to see that if K is known, then the *characteristic emitting function* $h(\rho)$ is recovered from observations by solving the system $w(h(\rho)) = Kh(\rho)$. The problem is that K is also unknown, and the complete calibration process should recover the emitting function $h(\rho)$ as well as the spectral characteristic matrix K. In addition, h is not necessarily linear, thus the problem on the unknowns K and h is non-linear.

The solution can be iteratively approximated by minimizing error solving a non-linear least squares problems given by $err = Kh(\rho) - w$. An initial solution to the problem can be produced solving its linear version, that is, $w = K\rho$.

3.3 Calibration in Practice

The calibration of an active setup involve the camera calibration and the light source calibration, in our case, a projector. In applications different set-ups were used, in what follows our photographic setup will be described and calibrated. This set-up uses a photographic camera and two types of digital projectors. *Setup a*: Camera plus LCD projector (illustred in Figure 3.4). *Setup b*: Camera plus DLP projector.

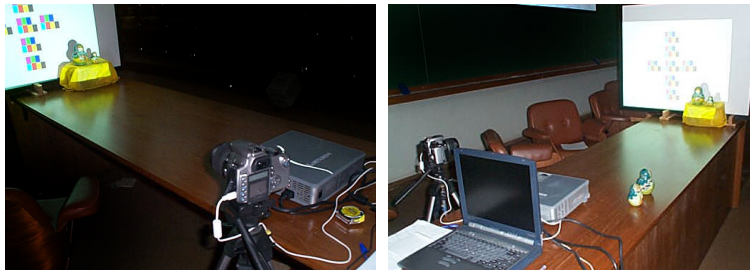


Figure 3.4: Our photographic setup.

A diffuse white screen was used to project images during the calibration process.

3.3.1 Camera Calibration

The digital camera used in our calibration tests is a Canon EOS D350. In the experiments we vary image exposure by controlling acquisition time or by con-

trolling illumination, while all other parameters were kept fixed. The camera parameters were set to:

- lens aperture = 22F,
- ISO speed = 200,
- focal distance = 41 mm,
- image size = 3456×2304 pixels - RAW.

All other parameters were turned off to minimize image processing.

Images were captured in RAW 12 bits proprietary camera format and converted to TIF 16 bits image by the *Digital Photo Professional 1.6* software attempting to turn off all unnecessary additional processing.

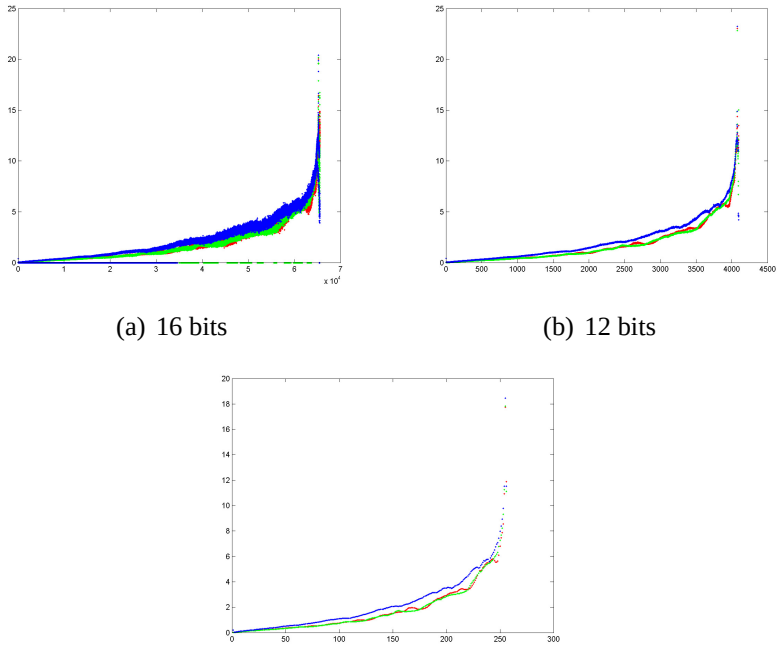
The camera characteristic response curve was obtained applying the iterative optimization method described in previous section. The input images were acquired by varying exposure time, three of them are shown in Figure 3.5.

(a) $\Delta t = 1/30sec$ (b) $\Delta t = 1/8sec$ (c) $\Delta t = 1/2sec$

Figure 3.5: Input scene.

In Figure 3.6 the computed f function is plotted. The difference between graphics (a),(b) and (c) is that in (a) the input TIF images with 16 bits of precision were used to run the method; in (b) the input images were reduced to 12 bits of precision that is the native sensor precision; in (c) the precision is reduced once again to 8 bits.

Note that when the TIF images with 16 bits of precision were used, many zeros were obtained as I_m values (Figure 3.6 (a)) by the iterative method. This is because when the TIF 16 bits was created out of the original RAW many bins remain empty, that is, $\Omega_m = \emptyset$. This problem is solved turning back to 12 bits and working with this channel depth resolution. Note also that the produced f aren't monotonically increasing, in addition, a high frequency can be observed on the



(a) 16 bits

(b) 12 bits

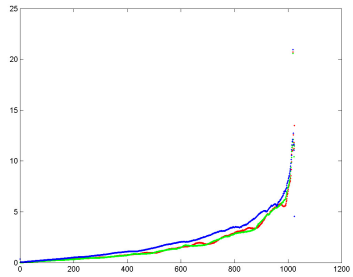
(c) 8 bits

Figure 3.6: Output f. Reducing the number of bits that encodes each color channel.

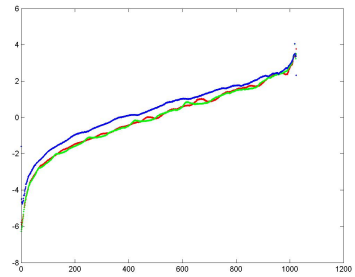
output f with 16 bits and 12 bits due to noise on the input images (Figure 3.6 (a) and (b)). The effect of noise is reduced when we reduce bits depth.

Based on these results we chose to work in 10 bits of precision, since it is reasonable to assume that 2 bits of our 12 bits of information is noise, given the conditions of our experiments. Figure 3.7 shows in (a) the produced f when the original algorithm was applied; in (b) its $\log 2$ values were plotted.

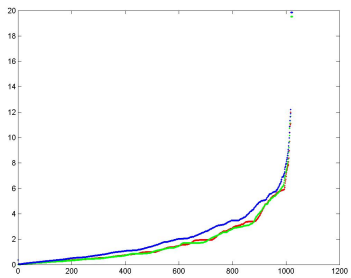
As expected, using the original formulation of the algorithm the obtained function is not monotonically increasing. Specially where the input data is poor the obtained f function is likely to be non-monotonically increasing. The heuristic adopted to guarantee that the function is monotonically increasing is very simple, it is based on linear interpolation, we simply ignore the values were some descent is observed and recalculate the values by linear interpolation considering the first non descent occurrence. Figure 3.7 (c) shows the final monotonically increasing f obtained from (a). To apply the linear interpolation we work on $\log 2$ of the data that is more reasonably assumed to be well interpolated by linear parts.



(a) 10 bits



(b) 10 bits - log2



(c) monotonically increasing f

Figure 3.7: Output f 10 bits of pixel depth.

3.3.2 Projector Calibration

Not only different cameras register different brightness values for the same input exposure, projectors emission characteristics also depends on projector technology, model and time of use. The projectors used in our experiments were a LCD Mitsubishi SL4SU and a DLP InFocus LP70. We now analyze our projectors by calibrating them respect to the previously calibrated camera.

The camera parameters were fixed after photometering the white screen with a constant gray pattern being projected. The screen plane was initially focused using the camera auto-focus facility and then the auto-focus was turned off and kept fixed during the experiment. The camera characteristic function f^{-1} was applied to the nominal camera values to obtain the linearized w values.

To recover the characteristic emitting function $h(\rho)$ at some specific values ρ, and the spectral characteristic matrix K, Projected intensity was modulated for the primary colors and the values registered by the camera observed. In Figure 3.8 the projected green values for the DLP projector are shown.

(a) $\rho = 64$ (b) $\rho = 128$ (c) $\rho = 192$ (d) $\rho = 255$

Figure 3.8: Green values observed by the camera related to modulated ρ intensity of projected green. DLP projector.

To solve the system $w = Kh(\rho)$, samples of the projected patterns were used. Green, Red, Blue and Gray full screen were subsequentially projected with ρ values equal to 64, 128, 192 and 256; additionally a Black screen was also projected. The non-linear system was then solved for both projectors to find K and $h(\rho)$ for the projected ρ. Without loss of generality we define $h(64) = 1$. For our DLP projector we obtain:

$$K_{DLP} = \begin{bmatrix} 0.0712 & 0.0607 & 0.0137 \\ 0.0032 & 0.1789 & 0.0401 \\ 0.0051 & 0.0987 & 0.2627 \end{bmatrix}$$

$h(0) = 0.04, h(64) = 1, h(128) = 4.11, h(192) = 11.39, h(256) = 14.40$

For the LCD projector we get:

$$K_{LCD} = \begin{bmatrix} 0.1664 & 0.0456 & 0.0249 \\ 0.0093 & 0.2071 & 0.0415 \\ 0.0100 & 0.0337 & 0.3589 \end{bmatrix}$$

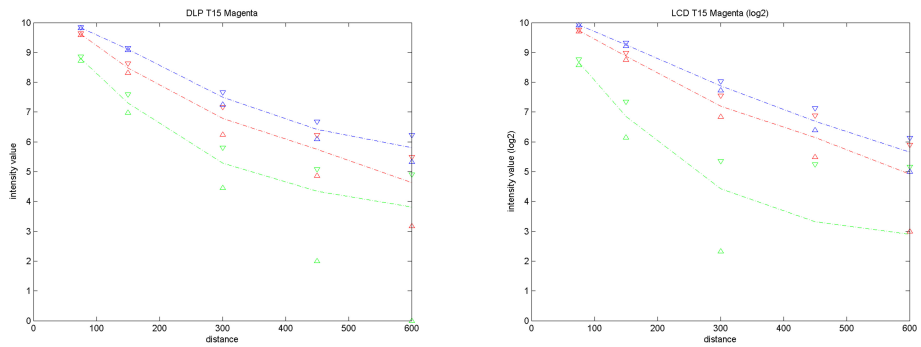
$h(0) = 0.05, h(64) = 1, h(128) = 3.26, h(192) = 6.80, h(256) = 9.96$

The non-linearity of h is clear. The fact that DLPs projectors produce higher contrast then LCDs is confirmed by the obtained h values. Another interesting observation is that for our DLP projector $\kappa_R^R = 0.0712$ and $\kappa_R^G = 0.0607$, this means that the response of camera Red channel is similar when projector projects Red or Green information, that is, if pure Green is projected, the camera red channel register an undesired high response. We turn back to this problem in Chapter 4.

The camera linearized values are: $w = Kh(\rho)$, where ρ is the nominal projected color. As K is near diagonal its inverse can be used to isolate the non-linear emitting function: $h(\rho) = K^{-1}w$. Thus a value ρ for which $h(\rho)$ is known can be used to obtain the nominal projected ρ. Then a linear transformation can be applied to simulate any other projector illumination.

Intensity decay with distance

In this experiment we verify the intensity decay with the increase of the screen distance. The purpose here is to define a working volume in the sense that the projector light source affects scene illumination within the defined volume. In Figure 3.9 the camera response to a uniformly projected magenta region is observed.



(a) DLP projector (b) LCD projector

Figure 3.9: Camera nominal $\log 2$ values decay with distance, in cm, of a projected uniform Magenta region. Maximum, minimum and average values were plotted.

As we are observing a projected Magenta, the Green channel would be expected to be equal to zero, but the effects of channel contamination given by matrix K can be clearly observed. By observing the maximum and minimum values it is evident the increasing noise with distance, this is expected since scene luminance decay. The conclusion that we can derive from this data is that for each 2 meters increased in screen distance, we observe a reduction of 2 bit of nominal information. This can be used to define thresholds in application of Chapters 5 and 6.

Chapter 4

Stereo Correspondence

Shape acquisition is one of the fundamental tasks in 3D photography. An object can be thought of as made up of a collection of surfaces which in turn have geometric properties such as curvature and features as well as photometric properties such as color, texture and material reflectance. In the last two decades the problem of accurately capturing an object's geometry was extensively studied, while the acquisition of high-quality textures, an equally important problem, only in recent years has become subject of research.

The recovery of an object's 3D shape is an inverse problem usually subdivided in several subproblems: *depth acquisition, alignment of views, mesh reconstruction*, etc.

The *depth acquisition* is the recovery of a depth map given an image or a set of images of a scene; it is heavily dependent on the hardware set-up chosen to acquire the images. The *alignment of views* is the problem of given a set of depth maps acquired from distinct points of view, construct a cloud of points that consistently represents the object; it is dependent on the knowledge of camera positions and on initial solutions. Texture information can help in this step. The *mesh reconstruction* is responsible for given a cloud of points construct a mesh that describes the object. This step can be substituted by the direct visualization of a cloud of points.

This chapter focus on *depth acquisition* techniques. The reasoning that allow depth recovery is based on the observation of how depth influences on the image formation. Observing how a controlled light source produces shadows we get shape from shading algorithms, observing the behavior of image focus we get depth from focus, observing images from different points of view we get stereo techniques. A classification of shape acquisition methods is given in Figure 4.1.

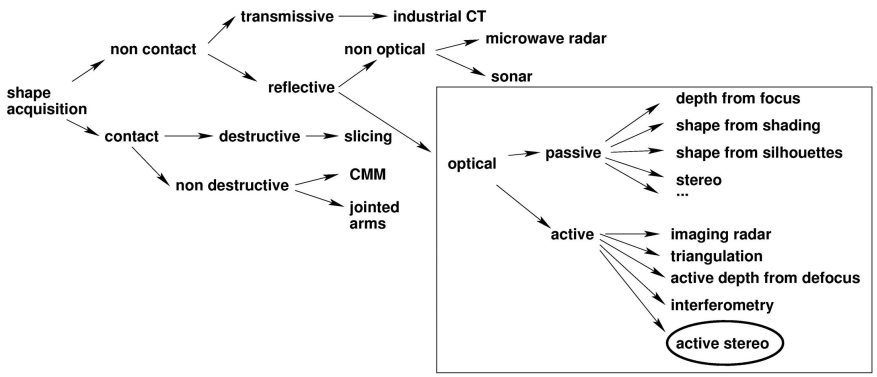


Figure 4.1: Shape acquisition methods.

We will concentrate on approaches that allows using off-the-shelf hardware, reducing significantly the cost of the scanner.

The highlighted branch in figure 4.1 shows the classification of acquisition techniques based on optical sensors, that is, what is being measured is the light intensity after its interaction with the scene to be measured. To choose among these techniques one has to consider their advantages and limitations depending on the application, such as resolution, accuracy, hardware and software to be used, etc.

One of the basic principles that allows obtaining depth maps from images is stereo vision, that is, if two known cameras observe the same scene point X then its position can be recovered by intersecting the rays corresponding to the projection in each image as illustrated in Figure 4.2. This processes is called *stereo triangulation*.

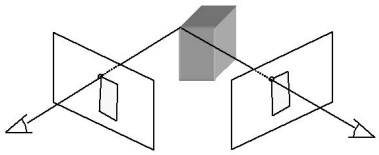


Figure 4.2: Stereo Triangulation principle

The so called *passive stereo* methods try to recover depth using two images acquired from different points of view, the main challenge of these methods to

recover 3D shape lies in the difficulty of automatically matching points in the two images. In order to avoid this problem, the *passive stereo* methods can be replaced by *active stereo* techniques, where one of the cameras is replaced by a calibrated and well defined light source, that mark the scene with some known pattern. The active approach helps to solve the stereo correspondence problem, which is a difficult task in passive methods. Active stereo methods is a typical technique that benefits from controlled illumination and we will focus our attention on this technique in the present chapter.

Recently an hybrid approach has been proposed that employs a pair of calibrated cameras and a projector that do not need to be calibrated with the system [].

In summary, the basic steps in recovering depth maps employing stereo vision techniques are the following:

- System geometric calibration;
- Establishing correspondences between points in the stereo pair;
- Construction of the depth map using stereo triangulation.

Limitations of techniques based on optical sensors includes that it acquires only visible portions of the surface and it is sensible to surface's reflectance properties.

4.1 Active Stereo

Shape from structured light is an active stereo vision technique used in establishing correspondences for stereo triangulation. Measurement of depth values is carried out with a system that resembles a two-camera stereo system, except that a projection unit is used instead of the second camera. A very simple technique to achieve depth information with the help of structured light is to scan a scene with a projected laser plane and detect the location of the reflected stripe in the camera image. Assuming that the projected laser can be seen by the camera, and both are calibrated, the depth information is then computed by stereo triangulation using the known correspondences.

For instance, laser-based systems direct a laser beam (contained in a known plane) to the scene and detect the beam position in the image. By intersecting the ray corresponding to each point with the known plane, one can compute the position of the points as shown in Figure 4.3.

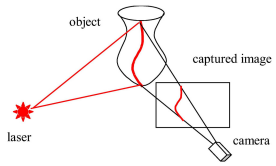
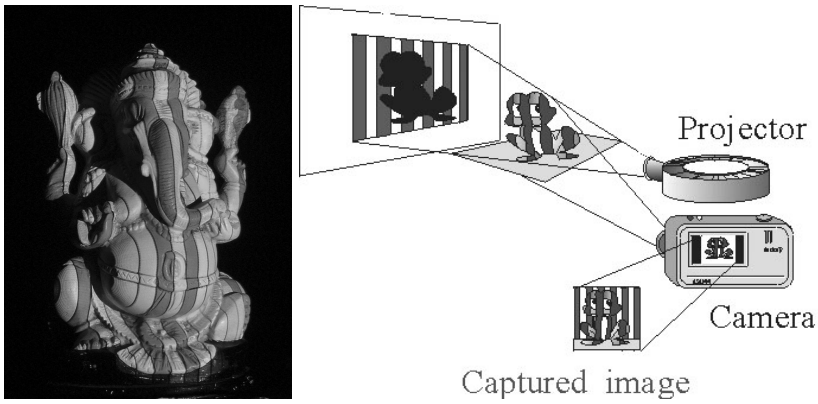


Figure 4.3: Laser beam projected on an object and its captured image.

In order to get dense range information, the laser plane has to be moved in the scene (or, equivalently, the object has to be rotated). Structured light methods improve the speed of the capturing process by projecting a slide containing multiple stripes onto the scene, as depicted in Figure 4.4. To distinguish between different stripes, they must be coded appropriately, in such a way that the projector coordinates are determined without ambiguity.

Coded Structured Light (CSL) techniques consists of illuminating the object with one or more slides with patterns coded according to certain schemes. There are many ways to code structured light. Early research on CSL methods was done in the 80's [JM82, KA87, PA90, JM90]. The idea of this work was to create empirically light patterns to capture the geometry of scanned 3D objects. In the 80's, pioneer tests with light coding were proposed and the main concepts were conceived. At the time, the methods were limited by restrictions imposed by ex-



(a) ganesh statue (b) schematic view of a scanner device

Figure 4.4: Example of structured light projected on a statue and an illustration of the scanner device.

isting hardware and software. During the 90's theoretical results in coding were obtained and improvements in processing and error analysis were achieved. Implementation of CSL methods to be applied in dynamic scenes have been the main recent contribution.

4.1.1 Coding Principles

The goal of a structured light code is to encode the projector coordinates using slide patterns and transmit them throughout the scene - where they suffer interference from the object's surface - that will be observed by a camera responsible for redigitizing the transmitted signal. The main task to decode the position of a projector coordinate is to recover the projected code from a sequence of images.

An widely adopted idea used to encode projector position is to sequentially project a black and white pattern corresponding to the binary digits of a code; in 1982, [JM82] proposed a binary temporal coding, while in 1984 [SF84] proposed to replace it by a more robust Gray binary code illustred in Figure 4.5. Binary codes produce 2^n coded stripes when a sequence of n slides are projected and the spatial scan resolution increases as the number of slides increases. The main problem of binary temporal code is the large number of slides that have to be projected to achieve the desired resolution and its restriction to static scenes.

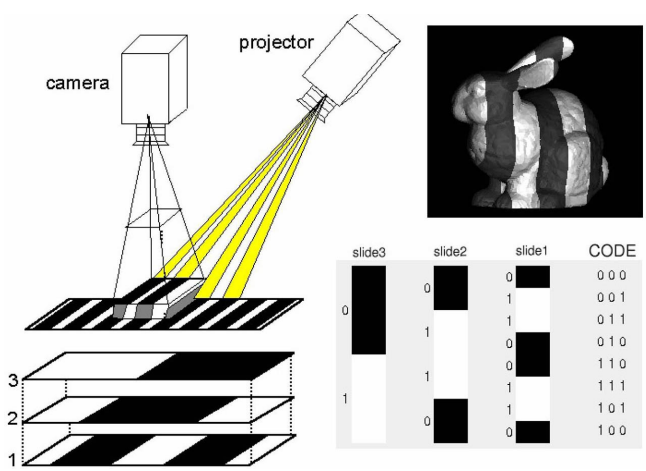
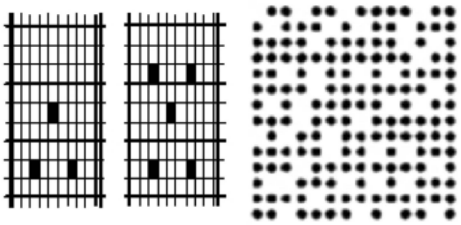


Figure 4.5: Temporal coding (Gray Code)

Note the natural analogy between CSL and a digital communication system. At each pixel of the camera image, a noisy transmission is received and needs to

be decoded. The transmission channel is the object's surface and the transmitted message is the encoded position of the projector coordinate. Considering this analogy, two main issues are to be studied: limitations of the transmission channel, related to materials properties; and projector coordinates coding scheme that leads to restrictions on the class of objects suitable to be robustly scanned.

The desire to acquire dynamic scenes and to reduce the number of projected slides leads to codes conveyed by a single slide. The only way to code position in a single slide is by increasing the number of distinct projected patterns, in such a way that there are enough patterns to achieve the desired resolution. A possible way to do so is to use the neighborhood of a pixel, known as spatial coding, or to modulate the projected light as a function of projector position (to be discussed in subsequent sections).



(a) grid pattern (b) M-array code

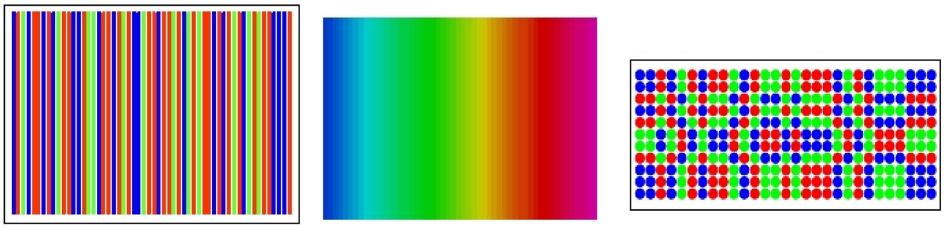
Figure 4.6: Examples of spatial codes (from [Powb]).

In figure 4.6 we show some schemes for spatial coding. Spatial coding imposes limitations on object discontinuities: the code array/window cannot include discontinuity regions in order to be decoded. But these are local restrictions, while for grid patterns the restriction is global. The main challenge is to recover projected patterns deformed by the object's surface.

In order to code a pixel in a single slide without neighborhood information one can modulate light intensity as a function of the projector pixel. This approach is sensitive to noise and surface properties can interfere in the signal in such a way that decoding is not robust. To alleviate this problem an additional white pattern can be projected and the difference of projected intensities is used to recover code; this method was proposed in [CH85].

The usage of color was introduced in the late 80's due to technological advances in capturing color images. The basic improvement was the possibility to use 3 channels in codes rather than one, but light source color is altered by the object's color, thus restricting the usage of this kind of code to neutral colored

scenes.



(a) horizontal codeword (b) wavelength modulation (c) cardeal neighbors

Figure 4.7: Examples of color based codes (from [Powb]).

Colored codes are shown in figure 4.7. Vertical slits are coded by its sequence of colors in [KA87] – figure 4.7(a)– , while modulation of wavelength in a rainbow pattern – figure 4.7(b) – was proposed in [JM90]. In the 90's, the great improvement in hardware and software permitted a more accurate and extensive research in coding light.

Several works were published in this decade attempting to explore the main ideas of coding in their full potential. New codes, improving the known ones, were proposed. Also, existing codes were re-implemented and had their results enhanced. Some of the representative works are [BMS98, Paj95, VP96, Mon94]. An excellent survey on coding techniques can be found in [JPB04], in their work the authors derive an exhaustive classification of coding schemes proposed in literature.

The application of CSL methods to dynamic scenes is dependent on the adopted coding scheme. Several patterns proposed in literature are coded in more then one slide, in the presence of movement the transmitted codeword can loose it's structure, this can also happens with some spatially coded schemes, leading to errors in decoding.

Recent work on structured light implements face detection for video [ZSCS04]. In general, the pattern of light projected defines the capturing features. The method proposed in [PGG] uses a self-adaptive, one-shot pattern for real-time 3D capture at 20fps. Most of the proposed algorithms cannot perform in real-time without some kind of hardware acceleration. In this context, a recent trend is to take advantage of programmable GPUs, [RM03]. Another option is to use multiple fixed cameras and scene analysis as the basis for visual hull and photo hull methods [MBR*00].

The current configuration of our system is similar to the one proposed in

[OHS01], [SM02], however, our implementation for video is more efficient due to the (6,2(2))-BCSL color code, that allows to robustly obtain a 3D video stream with texture and depth information at 30fps [VSVC05]. The data processing module extracts depth information from structured light code. The visualization module renders 3D video using the geometry induced by the color code used.

Texture recovery is usually done as a separate acquisition step, after acquisition the geometry and texture needs to be registered to each other. Acquiring both at the same time the registration problem is avoided.

4.1.2 Taxonomy

As we saw in previous sections, the three ways to code projected light are the use of chromatic, spatial or temporal modulation in the illumination intensity pattern. A taxonomy of CSL proposed in [OHS01] considering differences between patterns and the restrictions that they impose on the scene to be scanned. Chromatic modulation imposes restrictions on the allowable colors in the scene. When spatial coding is used, local continuity of the scene is necessary for recovering the transmitted code. Conversely, temporal coding restricts motion of the scene.

To encode information we have to decide how to describe the information in terms of the signal to be transmitted. The nature of the signal will define possible *letters of an alphabet* while the rule for concatenating letters defines *codewords*. The main tools to design the code are the number of distinct symbols (basic signals) available, the size of the codeword made by these symbols and, in the case of spatial coding, the structure of the neighborhood used to define a codeword.

In light coding the levels of intensity modulation and the number of channels considered (usually 1 or 3 channels) forms the alphabet letters, while codewords can be formed by temporal or spatial concatenation of letters. Accurate transmission of this alphabet requires that the material properties of objects present on the scene do not distort intensities or chroma too much, while the correct transmission of the codeword structure depends on scene's properties, that is, spatial concatenation of codewords can be lost if discontinuities are present on surface on which one is projecting the code, while robust decoding of temporal concatenation restricts scene's movement.

This reasoning leads to a simple *a priori* classification of codes, that is, based on decision made in designing the code since it imposes restrictions on the scene to be scanned, in contrast to the usual *a posteriori* classification of coding, like the complete classification proposed in [JPB04]. Our classification is based on the following observations of the proposed code design:

- available alphabet symbols → restricts scene's reflectivity
  - levels of light intensity modulation
  - number of channels used
- rules for letters concatenation: codeword structure
  - temporal → restricts scene's changes in time
  - spatial → restricts scene's depth discontinuities

Comparing our classification to the one given in [JPB04], we associate their *scene applicability* as a consequence of the usage of temporal coding, their *pixel depth* is related to the choice of the alphabet for coding (number of channels used and levels of intensity modulation), and finally the *coding strategy* is a consequence of the numbers of codeword generated by the codeword schema. Strictly temporal coding (as Gray code) is then applicable on static scenes but imposes no spatial nor reflectance restrictions on scanned objects; the codeword produced is binary and produces 2^s words where s is the number of projected slides. Spatio-temporal coding, as proposed in [OHS01], attempt to reduce the number of projected slides using weak spatial restrictions.

We classify CSL methods by observing the code design since it is the code design that imposes restrictions on the scene (or transmission channel) to be scanned. That is, if we use spatial coding, the scene has to preserve the spatial structure; otherwise, there will be loss of information. In the scene this can be translated as local continuity. The main characteristics of code design are the number of distinct symbols (basic signals), the size of the word made by symbols and, in the case of spatial coding, the geometry used. The following table classifies some codes proposed in literature.

We observe that code spatial structure of coding will be lost if discontinuities are present on projected surface area. Also, it is necessary to recover accurately intensity modulations, which requires that the surface does not distort intensities or chroma too much. Finally, the size of the codeword imposes that movement is not allowed while the codeword is not completed.

An improvement in spatial coding is proposed in [OHS01], where a CSL scheme is based on stripe boundaries. The codes are associated with pairs of stripes, instead of with the stripes themselves as in traditional methods. Boundary

method	num. slides (codeword size)	intensity modulation/channel (no. of characters)	neighborhood (character's region)	resolution (alphabet size)
Gray Code (fig.4.5)	n	binary(2) monochromatic	single pixel	2^n per line
colored Gray (fig.4.8)	n	binary(2)/ RGB	single pixel	$2^{3\times n}$ per line
rainbow pattern (fig.4.7(b))	2	2^8/ RGB	single pixel	$(2^8)^3$ per line
dot matrix (fig.4.7(c))	1	3 (R, G or B)		3^5
coded window fig.4.6(c)	1	binary	4 pixels 2×3 window	4^6

Table 4.1: Main characteristics of the code design of some codes proposed in literature.

coding has several advantages: it gives higher spatial precision and requires less slides (that is, features better temporal coherence).

In order to allow the greatest possible variations in scene reflectance, the scheme of [OHS01] is based on black and white stripes. This option leads to an undesirable problem: "ghost" boundaries (i.e., black-to-black and white-to-white transitions) must be allowed.

Recently, [LBS02] proposed using dynamic programming techniques to compute an optimal surface given a projected pattern and the observed image. The camera and projector correspondence is obtained up to one pixel resolution and a post-processing step is carried out to achieve sub-pixel accuracy.

The concept of using a colored boundary code is present in [LBS02] but the option to use a one-shot code implies in considering a subsequence of consecutive stripes to guarantee uniqueness of codewords with the desired resolution. Increasing the size of the basis used in coding complicates the decoding step. The price of adopting a one-shot code is that requirements on spatial coherence cannot be minimized, and some information will be lost due to discontinuities in the scene.

4.2 Minimal Code Design

When designing a CSL we have two conflicting objectives: one is to generate enough distinct codewords to encode projector coordinates without ambiguity, what is achieved augmenting spatial and temporal neighborhood or increasing the number of alphabet symbols; the other is to reduce the restrictions imposed on the scene to be scanned. We are then challenged to find a minimal code in terms of restrictions while getting a maximal possible number of distinct codewords. What is minimal in this sense? The answer is: a temporal neighborhood $s = 2$, a spatial neighborhood that considers only one neighbor, that is achieved coding boundaries, and finally using a binary light intensity modulation that guarantees a robust recovery of the transmitted information. The coding proposed in [OHS01] is one step in this direction except for the fact that their temporal neighborhood is $s = 4$. Our proposal, to be described in the following sections, is the usage of the 3 color channels in coding then reducing to $s = 2$ and achieving a minimal configuration.

4.2.1 Minimal Robust Alphabet

In this section we discuss the definition of a minimal robust alphabet for light coding. Our proposal is to use binary intensity exploring all the three color channels. In the following discussion we assume to have an ideal pair camera/projector, that is, the spectral characteristic matrix K is the identity. The generalization to real active pairs is postponed to the following section.

Binary Intensity Modulation

We will firstly consider a monochromatic image. The emitted intensity ρ of a projector light beam is scattered from the object surface and read by the camera sensor. As K is the identity, $w = u + rh(\rho)$. The digitized intensity is then given by $d_{ij} = f^{-1}(u_{ij} + r_{ij}h(\rho)_{ij})$ where u is the ambient light contribution and r is the local intensity reflectance factor, mainly determined by local surface properties [Mal84]. In the following we assume that pixels are not under nor overexposed.

Parameters u and r can be robustly estimated if we fix projector, sensor and object in relative positions, and produce sequential projected patterns varying ρ. The usage of binary levels for the ρ values has many advantages on the signal

recovery phase, in practice this is achieved projecting complementary slides, that is, if $\rho_{ij} = 0$ on first slide then $\rho_{ij} = 1$ on second:

$$I_{ij} = \begin{cases} u_{ij} & \text{when } \rho_{ij} = 0 \\ u_{ij} + r_{ij} & \text{when } \rho_{ij} = 1 \end{cases}$$

The non zero value $h(0)$ can be incorporated to the ambient light component. The maximum value at pixel (i, j) comparing the values of complementary slides, given by $\max(I_{ij}, I'_{ij})$, is equivalent to the maximum intensity coming from the projector, that is, $\rho = 1$; while the minimum value per pixel, given by $\max(I_{ij}, I'_{ij})$, is related to the ambient light contribution, that is, $\rho = 0$.

From the signal transmission point of view the usage of complementary slides introduces redundancy on the transmitted message that is replicated in both slides. This is a good procedure since it reduces the probability of errors on the received message.

Using Three Channels for Coding

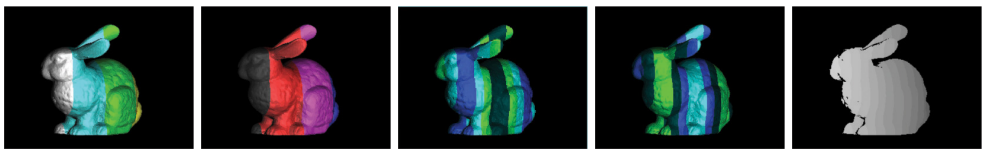
Instead of think in colors as traditionally, we will consider that a color is the visual result of three monochromatic signals projected together as separated wavelengths. As we prefer to work only with binary levels for each channel, we are restricted to project the primary colors R, G and B, their corresponding complementary colors C, M and Y, as well as black (K) and white (W). By processing each channel separately and extending the concept of complementary slides to each channel, we are able to recover the color of the projected light as well as ambient contribution and objects colors.

The set of basic signals (R,G,B,W,C,M,Y,K) is minimal and robust for decoding. We avoid projecting black (0 in all channels) since it may be confused with shadowed areas; for symmetry reasons, we do not use white, either. Thus, the number of different colors adopted as an alphabet is $b = 6$, if black and white are used for coding, then $b = 8$, where b is the base length of the code.

The traditional use of color in coding restricts the object surface reflectivity, because projected color is changed by the color of objects surface in an unknown way. By projecting complementary slides, however, the reflectivity restrictions are eliminated [SCV02].

To give an example of this reasoning we show the usage of a colored Gray code, where each channel of a colored slide corresponds to one slide of the black and white Gray pattern. This approach divides by three the number of slides

required to achieve the same resolution required by classic Gray pattern. We have tested this code in our virtual environment and the results are shown in figure 4.8.



(a) slide1 (b) slide1-neg (c) slide2 (d) slide2-neg (e) decoding

Figure 4.8: Two slides showing colored Gray code (a,c) and their respective negatives (b,d) projected on bunny. The number of recovered stripes after decoding is represented in (e) as gray levels.

The colored Gray code is, as the traditional Gray code, strictly temporal, it imposes no spatial restrictions on the object to be scanned and if complementary slides are projected color restriction are eliminated; it produces 2^{3s} codewords where s is the number of projected slides.

4.2.2 Codeword design - (6,2)-BCSL

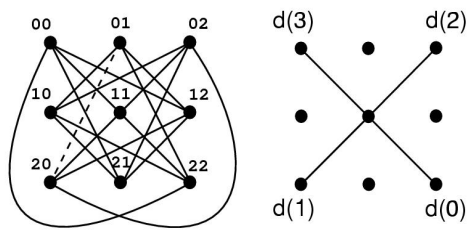
Considering that we have already chosen the alphabet of a maximum of $b = 6$ primary colors, we have to define a way to concatenate this basic signals to form a codeword. If a strictly temporal concatenation is adopted using a number of s slides we get a total of b^s different codewords. Adopting a boundary coding, that is a minimal spatial neighborhood to be considered, we increase the number of different codewords to $[b(b-1)]^s$ (we assume that two successive stripes may not have the same color, to avoid ghost boundaries). We call this general boundary-coded scheme a (b, s)-BCSL.

Schemes having $s = 1$ are purely spatial codings, imposing no restrictions on object movement. For $s > 1$, we have spatio-temporal codings. Among these, the case $s = 2$ reduces the need for time coherence to a minimum, namely that objects in the scene move slowly enough so that the displacement between the two captures is smaller than the width of the projected stripe. Using $b = 6$ and $s = 2$, leads to 900 coded boundaries, we will call this the (6,2)-BCSL.

Coding

The problem of generating a sequence of b-colored stripes for each of the s slides can be modeled as a the problem of finding an eulerian path in a suitable graph G.

G has b^s vertices, each corresponding to a possible assignment of the b colors at a given position for each of the s slides. For instance, if $b = 3$ and $s = 2$, G has 9 vertices, each labeled by a 2-digit number in base b, as shown in figure 4.9(a). For example, vertex 01 corresponds to projecting color 0 in the first slide and color 1 in the second, at a given stripe position.



(a) graph (3,2) (b) neighborhood

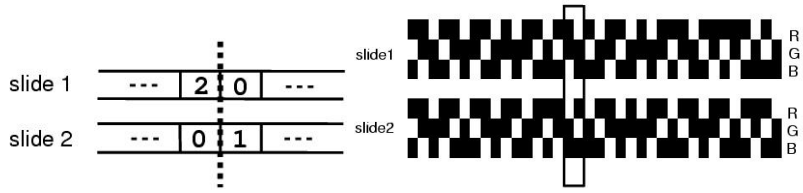
Figure 4.9: (3,2)-BCSL Encoding.

The edges of G correspond to the possible transitions for two consecutive stripe positions. The forbidden transitions are those that repeat the same color at the same slide, in order to disallow ghost boundaries. For instance, in 4.9(a), there isn't an edge connecting vertex 01 to vertex 02, since that would mean that two consecutive stripes in the first slide would use color 0. On the other hand, there is an edge connecting vertex 20 to vertex 01. This situation is illustrated in figure 4.10: at the same border position, we go from color 2 to color 0 in the first slide and from color 0 to color 1 in the second.

For the $(3, 2)$ case, the neighborhood structure is shown in figure4.9(b), with each vertex having 4 possible neighbors. For the general (b, s) scheme, there are $(b - 1)^s$ possible neighbors for each vertex, leading to a regular graph where each of the b^s vertices has degree $(b - 1)^s$. It is more appropriate, however, to think of G as a directed graph where each vertex has $(b - 1)^s$ incoming arcs and $(b - 1)^s$ outgoing arcs, since the same pair of vertices correspond to two distinct transitions, one for each ordering.

Possible color stripe schemes correspond to paths with no repeated edges (meaning that each multi-slide transition occurs only once) in the directed graph G. The maximum number of stripes is achieved by an eulerian path, i.e., a path that visits once each edge of G. This path certainly exists since every vertex in G has even degree and G is connected (for $b \geq 3$) ([D.B96]).

In fact, there is a very large number of different Eulerian paths in G, and an optimization problem can be formulated to search for the best path according to



(a) Boundary code (b) (3,2)-BCSL

Figure 4.10: (a)Example of boundary code for dashed edge in Figure 4.9(a). (b)(3,2)-BCSL using R, G, B as base. The outlined boundary has code $20|01$

some desired criteria. Horn et al. [EN99] formulated the problem of designing optimal signals for digital communication, they consider more important to encode with very distinct codewords points which are spatially distant, than to encode neighbor points with similar codewords, using this as a criterion to evaluate the path quality. We could also adopt an image processing perspective, using information about the photometric properties of the scene to be scanned as the criteria to generate an adaptive best code.

In some cases there is no need to use the complete Eulerian path, since it suffices to use a path of length equal to the maximum resolution handled by the cameras or projectors used and the path can be truncated. Figure 4.10 shows a particular (3,2)-BCSL correspondent to an Eulerian path in the graph.

Decoding

Once a codeword is identified in the captured images, that is, the colors on both sides of a projected boundary are known, we employ a decoding table to obtain the position of the projected boundary on the pattern.

The decoding table allows computing in constant time the projector coordinates of a given stripe boundary. This decoding table is shown in Table 4.2 for the $(3, 2)$ case. Each row of the table corresponds to a node v of G represented in base b. Each column corresponds to an edge connecting the node to its neighbor, ordered according to the pattern shown in figure 4.9(b). Each one of the neighbors can be conveniently expressed by means of arithmetic operations modulo-b, exploiting the regularity of the adjacency relationships, as shown in detail in [Hsi01] and in [SCV02].

Each entry of the table gives the position of the transition from the vertex associated with the row to the neighbor associated with the column in the path.

G nodes	d(0)	d(1)	d(2)	d(3)
V(00)	0	3	6	9
V(01)	14	17	19	11
V(02)	28	34	22	24
V(10)	26	29	18	21
V(11)	1	31	33	35
V(12)	15	4	8	13
V(20)	**16**	23	32	12
V(21)	27	5	7	25
V(22)	2	10	20	30

Table 4.2: Decoding table for (3,2)-BCSL.

For example, the arc that begins at vertex 11 and ends at vertex 02, which is neighbor $d(2)$ of 11 (see fig. 4.9), is the 33rd arc on path, and the 33rd stripe transition in the pattern.

4.3 Receiving and Cleaning the Transmitted Code

In previous sections the basic signals to be used and the rule to concatenate them forming codewords have been defined. Images corresponding to the codes were constructed to be projected onto the scene. Although the code is defined by s slides, complementary slides needed for the robust detection of the transmitted message doubles the number of projected patterns leading to the projection of a $(b, 2s)$-BCSL pattern. We refer to the two coded projected slides as P_1 and P_2, their respective complementary slides are P_1' and P_2'. The images captured are referred respectively as (I_1, I_1', I_2, I_2').

To recover the projector coordinates we need to identify the transmitted basic signals, then consult the decoding table to obtain the desired projector coordinates. The identification of the basic signals is an image processing step, in our case our goal is to find the stripe boundaries and recover the color projected in each side of the boundary on both coded slides.

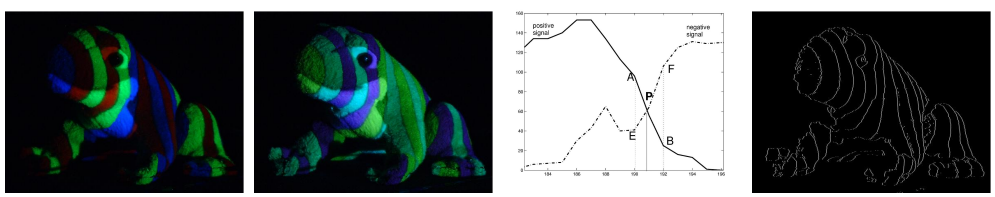
4.3.1 Boundary Detection

The boundary detection technique is based on pairs of complementary color stripes. Since complementary colors are projected at the same position, the boundaries

are given by the locations with zero-crossings of at least one color channel of the difference image $D_{ij} = I_{ij} - I'_{ij}$. In order to reduce false detections, the zero-crossings should be examinated only in the direction perpendicular to the direction of projection and have high slope, i.e., high local intensity variation. In our case, for a fixed row j the set of stripe boundaries T is the set of subpixel locations (i, j) for which

$$D_{ij} = 0 \text{ and } |D_{(i-\delta)j} - D_{(i+\delta)j}| \geq t_s,$$

for at least one color channel, where δ is the considered neighborhood in pixels and t_s is the slope threshold. The δ parameter is usually small and depends on the width of projected stripes as well as on how accurate is the reproduction of the projected boundary in the image (see Figure 4.11).

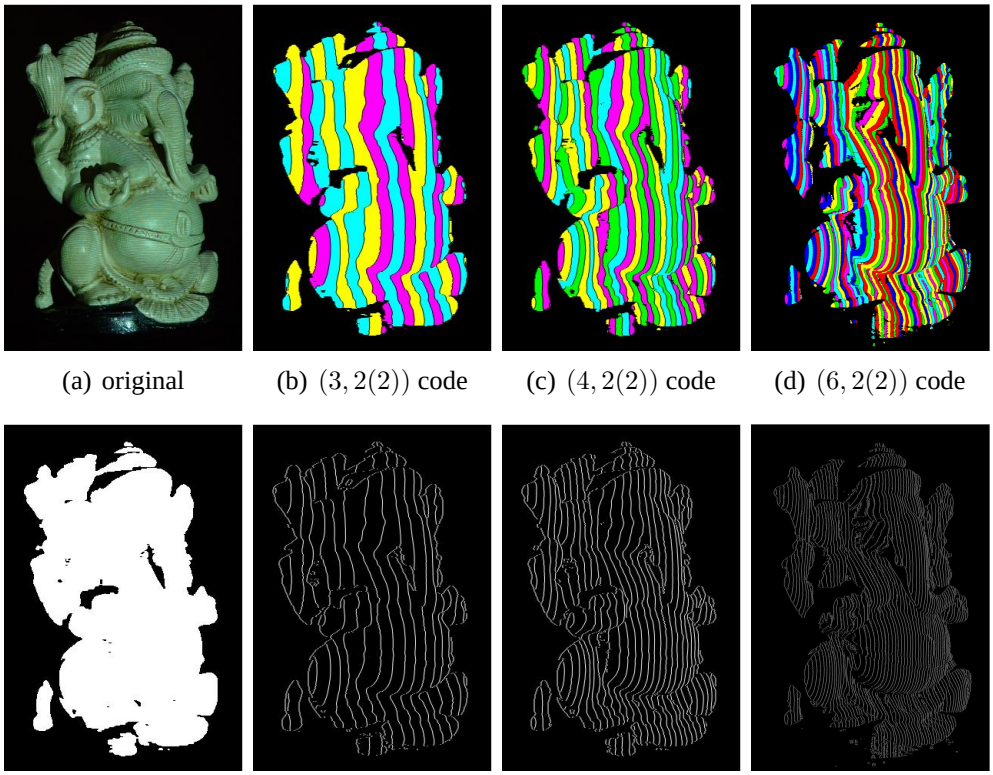


(a) Slide I_1 (b) Slide I'_1 (c) Signal crossing of an specific row (d) Boundaries detected

Figure 4.11: Boundary from complementary patterns.

Shadow regions are also detected by analyzing the difference image D, we consider that a point is in shadow if the absolute value in the three channels of the difference image are all bellow a threshold, as shown in Figure 4.12(e). For these areas, the stripe pattern is not processed and no range values are obtained.

Figure 4.12(a) we show a Ganesh statue, which has a detailed geometry and homogeneous surface properties. We used three different $(b, 2(s))$ codes with s fixed equal 2 and an increasing resolution, the stripe width is 18 pixels for the (3,2(2)) code, 10 pixels for the (4,2(2)) code and 5 pixels for the (6,2(2)) code. The $(b, 2(s))$ slides were generated with 600×480 pixels. Figures 4.12(b), (c) and (d) show the recovered colors for the (3,2(2)), (4,2(2)) and (6,2(2)) codes, respectively, and Figures 4.12(f), (g) and (h) show the corresponding stripe boundaries. Figure 4.12(e) shows the mask for background and shadow areas.



(a) original (b) $(3, 2(2))$ code (c) $(4, 2(2))$ code (d) $(6, 2(2))$ code

(e) detected shadows (f) $(3, 2(2))$ edges (g) $(4, 2(2))$ edges (h) $(6, 2(2))$ edges

Figure 4.12: Ganesh statue – Augmenting geometry resolution

4.3.2 Colors and Texture Recovery

In this section we work on the linearized camera values since values of two images will be compared with the intention to define original projected colors as well as objects texture. The projected colors image C is recovered by verifying the sign of the difference image D in each color channel [SCV02]:

$$C_{ij} = \begin{cases} 0 & \text{when } D_{ij} > 0 \\ 1 & \text{otherwise} \end{cases}$$

The obtained image C is illustred in Figure 4.12(b),(c), (d) and in Figure 4.13. The proposed projected colors recovery assumes that the characteristic spectral matrix K is the identity. As we have seen in Chapter 3 this is not the case for real projectors. For a robust color detection projector calibration must be considered.

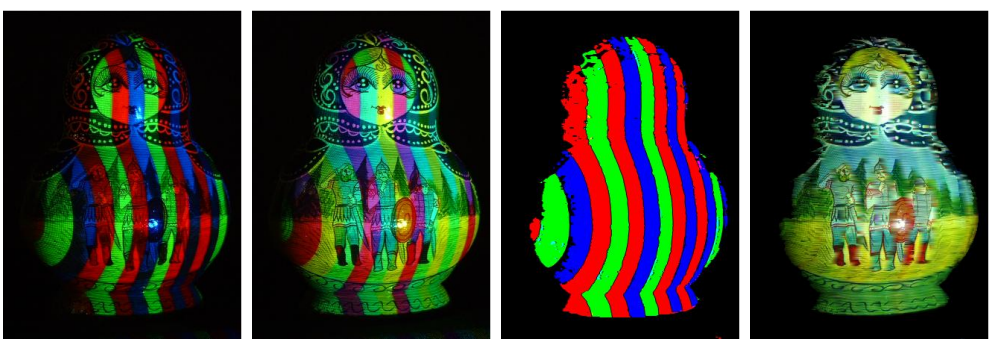
In theory the concept of complementary light colors means that if they are summed up the resulting signal is white light. Thus, in the case of primary colors, the complementary of Red $= [100]$ is Cyan $= [011]$. The considerations about projector calibration are crucial in texture recovery.

For ideal projector/camera pairs the recovery of objects texture is obtained by combining the information of the three channels of the input images to generate two new images:

$$W = (\max(I_R, I'_R), \max(I_G, I'_G), \max(I_B, I'_B)),$$
$$K = (\min(I_R, I'_R), \min(I_G, I'_G), \min(I_B, I'_B))$$

where W is equivalent to the projection of white light, $P = [1\ 1\ 1]$, while K is the ambient light , $P = [0\ 0\ 0]$. The projected pair of complementary slides can be interpreted as white light decomposed in time, since summing the complementary pair we get the triple $[1\ 1\ 1]$ at projector pixels. Thus, the object's texture is ideally given by $W - K$.

The considerations above are not valid for pixels at stripe boundaries not even in the ideal case. In these pixels, colors are recovered by interpolating neighborhood information at both sides of the boundary.



(a) Slide I_1 (b) Slide I'_1 (c) Projected colors (d) Texture

Figure 4.13: Matrioska – Recovering colors.

Figure 4.13 is a Matrioska doll, it has a simple shape and a highly reflective complex painted texture on its surface. Figures 4.13(a) and (b) show one of the slide pairs, (c) shows the stripe codes and (d) shows the recovered texture. Figures 4.11,4.12 and 4.13 were generated using setup consisting of a FujiPix 2400Z digital camera and a Sony VPL-CS10 LCD projector. The acquired images have a resolution of 1024×768 in JPEG format.

For real camera/projector pairs, channels color contamination can invalidate the above procedure if calibration is not considered. In that respect, color fidelity can be improved by a color correction pre-processing step considering projector-camera calibration.

4.4 Video Implementation

The proposed coded structured light scheme was implemented for video to work in real time [VVSC04, VSVC05]. The proposed setup produces one texture image per frame while each frame is correlated to the previous one to obtain depth maps. Thus both texture and geometry are reconstructed at 30Hz using NTSC (Fig. 4.15). Crucial steps that influences on depth map accuracy are the calibration of the system, poorly calibrated cameras or projectors cause error propagation in depth measurements.

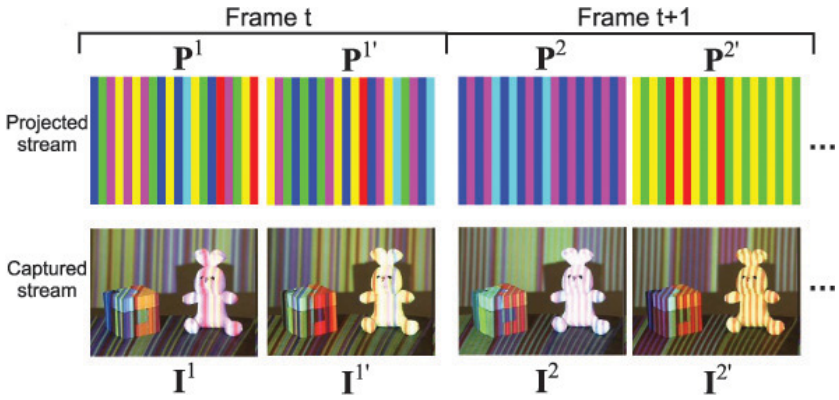


Figure 4.14: The sequence of color pattern frames and the captured images as a result of their projection onto a scene.

The camera/projector synchronization guarantees that one projected frame will correspond to one captured frame, illustred in Figure 4.14. Patterns P_t, P_{t+1} are coded with their corresponding complements $P_t{}', P_{t+1}{}'$ as fields in a single frame. Each 640×480 video frame in NTSC standard is composed by two interlaced 640×240 fields. Each field is exposed/captured in $1/59.54s$. The projector used in this set-up was the DLP InFocus LP70. The DLP projector was the only possible choice due to the long latency in switching colors for LCD projectors. Ambient light is minimized during acquisition.

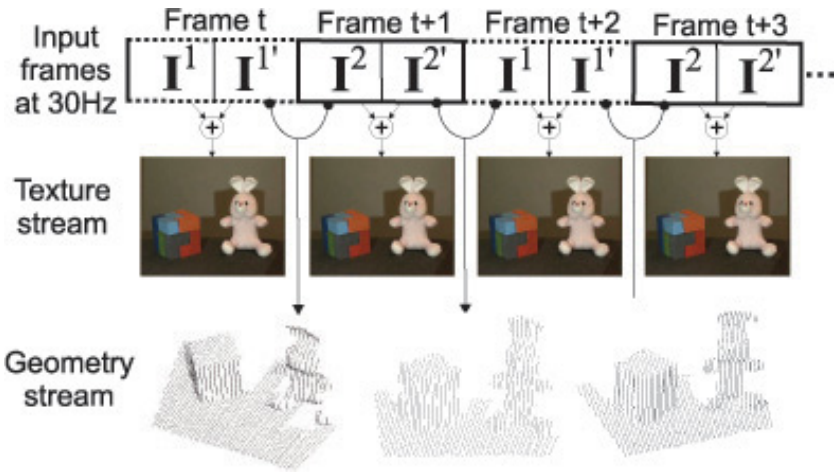


Figure 4.15: Input video frames and the texture and geometry output streams. The output rate is also 30Hz.

The input images are processed as described in the previous section. The quality of color detection is enhanced by the camera/projector color calibration. As only one projector intensity ρ is used, the calibration can be simplified to a linear problem where $h(\rho)$ is a constant implicit on the characteristic spectral matrix K.

To avoid errors in decoding, only pixels having at least two channels with absolute difference greater than a threshold t_c are used. Pixels not satisfying this condition or having white value $[1\ 1\ 1]$ are set to black $[0\ 0\ 0]$ and thus invalidated. This is necessary since as we saw, for the DLP projector the contamination of channels is critical leading to systematic errors in decoding. Another criterion used to invalidate a detected boundary, is if the left $c_l = C_{(i-\delta)j}$ and right $c_r = C_{(i+\delta)j}$ detected colors do not have two distinct valid colors. For analog video, $\delta = 3 - 6$ pixels are adequate for masking-off the noisy region around the boundary.

To use the code in dynamic scenes, boundaries are tracked between frames to compensate possible movements. To find the correlated boundary in time, all stripe boundaries detected in frame t need to be correlated to the boundaries detected in frame $t+1$. To do so, we look for the nearest point in subsequent frames which combined gives a valid (6,2)-BCSL code assuming the space-time coherence on the decoded data (Fig. 4.16). Each frame is decoded using the tuple which gives more valid stripe positions. This tuple can be easily predicted from the previous one. Note that the decoding color order depends on the last projected

pattern sequence.

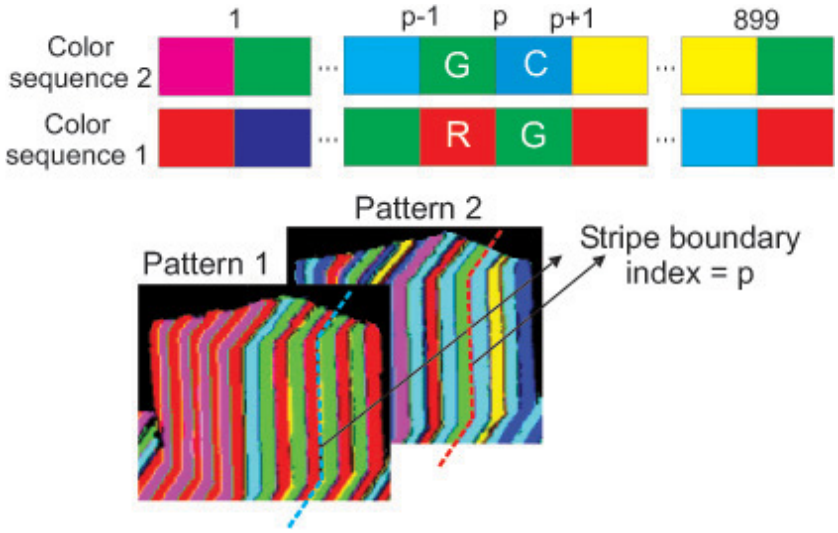


Figure 4.16: Decoding stripe transitions using (6,2)-BCSL.

The boundary correlation is processed in a 7×7 neighborhood of a boundary pixel. This is sufficient even for reasonably fast motion. The reason is that, while objects move, the stripe remains stationary. Since high discontinuities in depth are unlikely to occur in most scene regions, the boundaries are likely to be near each other.

With the boundaries and their estimated projector coordinates in hand, the real 3D points in camera reference system are obtained using the camera and projector geometric calibration.

Texture is retrieved by simply adding both input complementary fields. The influence of scene motion is perceived as an image blurring. Assuming that the motion is small compared to framerate we do not adopt any deblurring strategy. Analog video corrupts colors around stripe boundaries what results in a bad reconstruction of colors in that regions, this is also observed in the presence of boundary movement.

Video results are available at `http://www.impa.br/~mbvieira/video4d`.

Chapter 5

Image Segmentation

Image segmentation is an important problem in Computer Vision. A special case of the general segmentation problem is the foreground - background image segmentation, in which a binary classification is applied to an image that has a perceptual background/foreground separation.

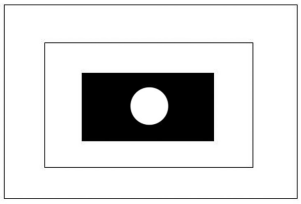


Figure 5.1: Ambiguity: the decision on whether the central circle is background or foreground depends on the interpretation.

Given a single image, some additional knowledge has to be given to the system as an initial clue of what is background or foreground. Figure 5.1 shows the intrinsic ambiguity in determining the foreground for a general image. Other examples of ambiguity are scenes with more then two distinct planes of information, where the classification of each plane as background or foreground is a matter of interpretation.

In practice, the information is usually disambiguated either by user interaction or by the acquisition of additional information about the scene such as previous calibration of the background, defocusing, analysis of movement and other techniques most of them benefiting from analyzing a collection of images. In this

Chapter the active illumination approach is applied to help in the solution of the foreground - background segmentation problem.

5.1 Foreground - Background Segmentation

There are two main branches of research in foreground - background segmentation: one assumes that image acquisition can be controlled to produce images that are automatically segmented, while the other analyzes a given image without any assumption about image formation. In the first approach the clues to be analyzed are decided a priori and determine acquisition prerequisites; the segmentation is then automatic. In the latter the initial clues for solving the problem are inserted by the user a posteriori and no knowledge on acquisition conditions is assumed. One example of a priori information used for foreground segmentation widely used in practice is chroma key [SB96].

Recently much work has been done in proposing segmentation methods where clues are inserted a posteriori intending to minimize user intervention. In most cases the user has to indicate coarsely the foreground and the background pixels [RKB04, WBC*05] as initial restrictions for a minimization process.

Image Segmentation via graph cut minimization

The image segmentation problem is a special case of a pixel labeling problem. It can be modeled as an optimization problem, which consists of computing the best image segmentation among all possible image segmentations satisfying a set of predefined restrictions.

In pixel labeling problems the goal is to find a labeling $f : \mathcal{P} \mapsto \mathcal{L}$, mapping a set of pixels $\mathcal{P}$ to a set of labels $\mathcal{L}$, that minimizes some energy function. This energy function typically has the form

$$E(f) = \sum_{p \in \mathcal{P}} D_p(f_p) + \sum_{p,q \in \mathcal{N}} V_{p,q}(f_p, f_q),$$

where $\mathcal{N} \in \mathcal{P} \times \mathcal{P}$ is a neighborhood system on $\mathcal{P}$. $D_p(f_p)$ is a function based on the observed data that measures the cost of assigning lable f_p to p and $V_{p,q}(f_p, f_q)$ is an spatial smoothness term that measures the cost of assigning labels f_p and f_q to adjacent pixels p and q.

Energy functions like E are, in general, very difficult to minimize, as they are non convex functions in large dimensional spaces. When these energy functions

have special characteristics, it is possible to find their exact minimum using dynamic programming. Nevertheless, in the general case, it is usually necessary to rely on general minimization techniques.

Many approaches found in the literature model the problem of image segmentation as an energy minimization problem. Recently, Graph-Cut Minimization became widely used in image segmentation [ADA*04, RKB04, BVZ01].

An interesting property of a graph cut C is that it can be related to a labeling f, mapping the set of vertices $\mathcal{V} - \{s, t\}$ of a graph $\mathcal{G}$ to the set $\{0, 1\}$, where $f(v) = 0$, if $v \in S$, and $f(v) = 1$, if $v \in T$. This labeling defines a *binary partitioning* of the vertices of the graph.

The optimality of graph cut minimization methods, considering labeling problems, depends on the number of labels and the exact form of the smoothness term V. In [GPS89]is proved that the method yields global minimum solutions for binary labeling problem. In [IG98] it is proved that, for labeling problems with arbitrary number of labels, if the smoothness term is restricted to a convex function, it is possible to compute global minima.

In most image segmentation applications, it is desirable to preserve boundary discontinuities. This is not possible by using a convex function as the smoothness penalty term in the energy function. In general, the minimization for energy functions that preserve discontinuities by graph cut minimization can only produce approximate solutions. In [BVZ01] a graph cut based algorithm is proposed that is able to compute a local minimum for discontinuity preserving energy functions. The authors also proved that the obtained local minimum lies within a small multiplicative factor(equal to 2) of the global minimum.

The early proposals that used graph cut optimization as a method for energy minimization required the construction of a specific graph for each particular problem. In [KZ04] is introduced a general scheme for graph cut minimization of energy functions that belong to the class of regular functions.

5.2 Active Segmentation

Active illumination can be combined with graph-cut optimization to perform the segmentation of foreground regions. We call this *active segmentation*, meaning the use of a light sources to illuminate only the objects to be segmented leaving the background essentially unchanged. Thus, the light source works as a substitute to the user, acting on the scene to indicate object and background seed elements automatically. This pre-segmentation provides the color distribution of

each region, that can be used in a graph-cut optimization step to obtain the final segmentation.

In recent works, camera flash was explored to enhance image quality. [ED04], [PAH*04] have proposed the use of bilateral filter to decompose a flash/non-flash pair of images and then recombine them appropriately; both authors have to deal with shadows produced by the flash. Multiple images with flash positioned in different locations among them is used to extract object borders in [RTF*04] and a non-photorealistic rendering is then applied to the images. Our work differs from these previous works since it explores a light source, specially positioned to illuminate only the objects to be segmented, that stays in a fixed position between shots.

5.2.1 Active illumination with Graph-Cut Optimization

By modulating segmentation light intensity in subsequent images (by projecting intensity ρ_i) we get I_{ρ_i} (illustrated in Figure 5.2). We assume that the ambient light does not change between two shots and that all camera parameters are fixed. Since we are not actively illuminating the background pixels p with the segmentation light source, we have $I_{\lambda_2}(p) \approx I_{\lambda_1}(p)$.

Figure 5.2: (left) and (center) are the input images differently illuminated by varying the camera flash intensity between shots, (right) is the difference thresholded image.

The luminance difference is used to build a likelihood function for background membership. The higher the difference, the lower the likelihood that the pixel belongs to the background. Thus, pixels for which the luminance difference is greater than a given threshold are likely to belong to the object and are used to define the initial seed to the optimization method.

The initial seed gives important clues about the regions that are likely to belong to the background and foreground regions. Based on these clues, it is possible to

compute the desired segmentation by minimizing an energy function. If the energy function is chosen in such a way that some regularity properties hold, then it is possible to minimize it efficiently by graph cut optimization methods.

As in [BVZ98] and [BJ01], a discontinuity preserving energy function is adopted. It is defined in terms of a set of pixels $\mathcal{P}$, a set of pairs of neighboring pixels in a neighborhood system $\mathcal{N}$ and a binary vector $A = (A_1, A_2, .., A_{|p|})$, where A_p is the assignment of pixel p either to 0 (background) or 1 (foreground).

The energy function has the form of the following cost function:

$$E(A) = \lambda \sum_{p \in \mathcal{P}} R_p(A_p) + \sum_{\{p,q\} \in \mathcal{N}} B_{\{p,q\}} \cdot \delta(A_p, A_q) \tag{5.1}$$

This energy function is defined in terms of a *regional term*, that measures the fitness of a region to the background or foreground of the scene, and a *boundary term*, which penalizes discontinuities in the label assignment while preserving those that are associated to features of the image. The first term is a *regional term* that measures how the intensities of the pixels of the image fit into intensity models (for example, obtained by a histogram) of the background and foreground. The second term is a *boundary term*, which penalizes discontinuities in the label assignment while preserving those that are associated to features of the image. Coefficient $B_{\{p,q\}} > 0$ can be interpreted as a penalty for spatial discontinuity of the labels assigned to neighboring pixels p and q. $B_{\{p,q\}}$ should be large when pixels p and q are similar, and close to zero when p and q are very different, so that feature discontinuities are preserved. The $\lambda \geq 0$ constant is used to specify the relative importance of the regional term versus the boundary properties term.

The proposed energy function is then minimized by a graph cut optimization algorithm that follows the scheme proposed in [KZ04]. The regions determined by the active illumination thresholding are used as seeds to the graph cut optimization. However, their labels can be modified as the process is executed. Note that we work in the *Lab* color space.

5.3 The objective function

The objective function is based on probability distributions of color values in three regions: background, object and boundary. They are defined assuming the following:

- Most actively illuminated pixels belongs to the foreground objects. The

influence of active illumination on the background can lead to wrong overall segmentation;

- The actively illuminated regions capture the object features, that is, they contain all color information necessary to distinguish foreground objects from the background;

- Regions corresponding to moving objects in the scene represent a small fraction of the scene;

- Color differences in Lab space are sufficient to define relevant object - background boundaries.

Figure 5.3: Input images differently illuminated.

The challenge is to define probability distributions that approximate the real distribution of the expected segmentation regions. For the background region, we employ *a priori* distributions of the luminance difference $L_{I_2} - L_{I_1}$. Color histograms from the seed regions are used to build a color distribution function for the foreground region.

These distributions, together with a boundary likelihood function based on distances in Lab space, are the basis of the cost function to be proposed.

5.3.1 Composing the cost functions

The goal is to find the labels $\mathbf{X} = \{x_p, \ \ p \in I_1\}$, where x_p is 0 if p is background or 1 if p is foreground, that minimize an objective function $E(\mathbf{X})$. Inspired in Information Theory, the regional term of the energy function is defined as:

$$C(x_p) = \begin{cases} -\log(\mathrm{p}_O(p)), & \text{if } x_p \text{ is 1 (foreground)} \\ -\log(\mathrm{p}_B(p)), & \text{if } x_p \text{ is 0 (background)} \end{cases} \tag{5.2}$$

The boundary term, for neighboring pixels p, q is $-|x_p - x_q| \log \mathrm{p}_R(p, q)$. Thus the final objective function is

$$E(\mathbf{X}, \sigma_L, \sigma_C) = \sum_{p \in I_1} C(x_p) - \sum_{p,q \in I_1} |x_p - x_q| \cdot \log \mathrm{p}_R(p, q), \tag{5.3}$$

where points q are those in the 8-connected neighborhood of p.

We turn now to the definition of $\mathrm{p}_B(p)$, $\mathrm{p}_O(p)$ and $\mathrm{p}_R(p, q)$. We start by discussing how to infer foreground sites from the input data. With the above assumptions, high values of the luminance difference $|L_{I_2}(p) - L_{I_1}(p)|$ indicate foreground pixels p, where $L_{I_1}(p)$ and $L_{I_2}(p)$ are the luminance channels of the transformed images I_1 and I_2. However, it cannot be stated that low values of that difference indicate background pixels since there may be parts of foreground objects that are not actively illuminated (like shadow areas). Thus, the luminance difference does not characterize completely foreground and background elements.

Luminance difference for background pixels can be modeled by a gaussian distribution, with density

$$\mathrm{p}_B(p) = \frac{1}{\sqrt{2\pi}\sigma_L} \exp\left(\frac{-|L_{I_2}(p) - L_{I_1}(p)|^2}{2\sigma_L^2}\right), \tag{5.4}$$

where σ_L is the standard deviation of the luminance differences.

High $\mathrm{p}_B(p)$ values do not necessarily indicate that p is background but pixels with small $\mathrm{p}_B(p)$ values are likely to belong to the foreground. The set of foreground pixels are then defined as $O = \{p \mid \mathrm{p}_B(p) < t\}$, where t is a small threshold. We fix $t = 0.05$ since the parameter σ_L can be adjusted.

The color histogram of the foreground pixels determine the object probability function. For simplicity, we use a 3D histogram for the Lab components with uniform partition. Let nb_L, nb_a and nb_b be the number of predefined bins for each lab component. All points $p \in O$, with normalized color components $L_1(p)$, $a_1(p)$ and $b_1(p)$, are assigned to a bin k with coordinates

$$(\lfloor L_1(p) * nb_L) \rfloor, \lfloor a_1(p) * nb_a \rfloor, \lfloor b_1(p) * nb_b \rfloor)$$

The object distribution function is then defined as

$$\mathrm{p}_O(p) = \frac{n_k}{n_O} \tag{5.5}$$

Figure 5.4: Images of background probabilities. Darker pixels have smaller probabilities (left) $\sigma_L = 15$ (center) $\sigma_L = 25$ (right) $\sigma_L = 35$.

where n_k is the number of pixels assigned to the bin k and n_O is the number of pixels in the object region O.

To construct the histogram information only one image is considered and it will depend on each situation. In our experiments $L_1(p)$, $a_1(p)$ and $b_1(p)$ are the color components from the image correspondent to the lowest projected ρ value. Another consideration is that it may be difficult to determine the number of bins for each component. The bins that distinguish relevant color groups when the partition is uniform. If the number of bins is too small, wide ranges are mapped in few bins. If there are too many bins, frequencies tend to be small everywhere. In our experiments, the object pixels are sufficient to populate a histogram with n_L=32, n_a=64 and n_b=64 bins.

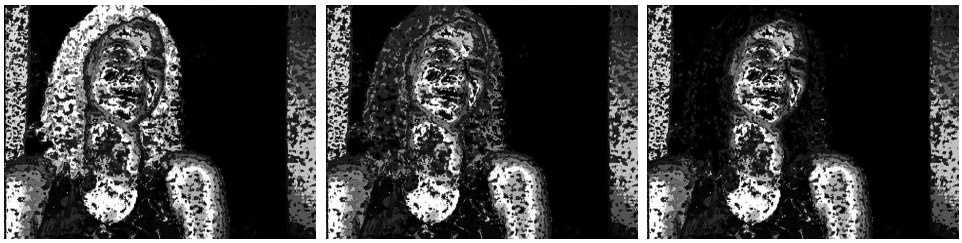


Figure 5.5: Image of object probabilities. Darker pixels have smaller probabilities (left) $\sigma_L = 15$ (center) $\sigma_L = 25$ (right) $\sigma_L = 35$.

Finally, the likelihood function for neighboring boundary pixels is given by

$$\mathrm{p}_R(p,q) = 1 \; - \; \exp\left(\frac{-(||Lab(p) - Lab(q)||)^2}{2\sigma_C^2}\right), \tag{5.6}$$

where $Lab(p)$ denotes the color at point p and σ_C is the standard deviation of the L^2-norm of the color difference. the effect of this term is that, if the colors of

pixels p and q are close in the Lab space, their connection is unlikely to cross a foreground-background border.

Figure 5.6: Image of boundary probabilities taking the maximum value of a 8-connected neighborhood. Darker pixels have smaller probabilities (left) $\sigma_C = 5$ (center) $\sigma_C = 15$ (right) $\sigma_C = 25$.

According to [KZ04], an energy function of the form $E(x_1, x_2, \cdots, x_n) = \sum_p E^p(x_p) + \sum_{p \neq q} E^{p,q}(x_p, x_q)$, where each x_p is a 0-1 variable, can be minimized by means of a minimum graph-cut when it is regular, that is, satisfies the inequality

$$E^{p,q}(0,0) + E^{p,q}(1,1) \leq E^{p,q}(0,1) + E^{p,q}(1,0).$$

In our case, we have $E^{p,q}(0,0) = E^{p,q}(1,1) = E^{p,q}(0,1) = E^{p,q}(1,0) = 0$ when $x_p = x_q$. On the other hand, if $x_p \neq x_q$, then $E^{p,q}(0,1) = E^{p,q}(1,0) = 0$ and $E^{p,q}(0,1) = E^{p,q}(1,0) = -\log(p_R(p,q) \geq 0$, since $0 \leq p_R(p,q) \leq 1$. Hence, the proposed energy function is regular.

5.4 Method and Results

The main steps of our active segmentation method are illustrated in Fig. 5.7. Two input images are acquired. We apply a low-pass filter to the input images in order to reduce noise. Next, the input colors are transformed into the Lab color system. This perception-based color space is desirable for two reasons: we need to cluster regions with small perceived color variations and we want to explore the orthogonality between luminance and chromaticity information in Lab space. Our goal is to have perceptually homogeneous regions, with the segmentation boundaries preferably located where high Lab color differences occur.

Active illumination is explored to attribute weights to the pixels that are used in the energy minimization by graph cuts. For the optimization step, a graph where the nodes are the pixels and the edges form a 8-connected neighborhood is created.

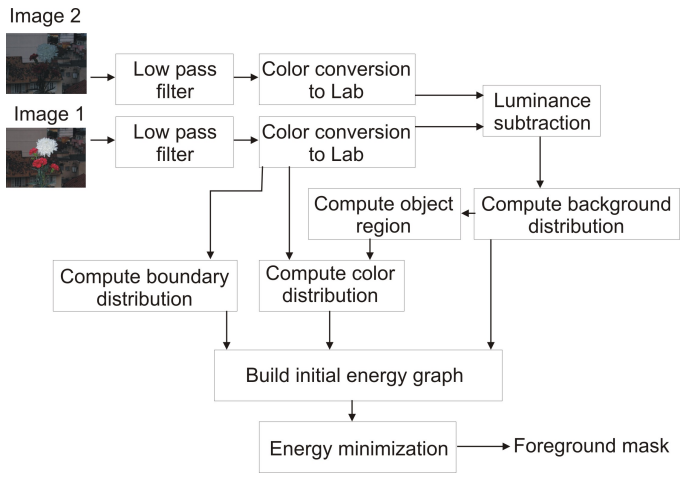


Figure 5.7: The proposed active foreground extraction method.

The object and background color distributions are used to compute the cost of assigning object or background label to each node. The boundary distribution is used to compute the cost of having an object - background transition for each edge.

Initially, all points belonging to the estimated object region are labeled as foreground. All other points are labeled as background. The min-cut/max-flow algorithm [BVZ01] is used to find the global minimum

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \; E(\mathbf{X}, \sigma_L, \sigma_C) \tag{5.7}$$

Only some object regions can be determined before the optimization step. Usually, seeds for both the foreground and the background are used, which implies that histograms for both classes are available. The seed pixels are not allowed to have their label changed. In our case, there is no guarantee that the estimated object region is correct. Furthermore, the *a priori* background distribution (Eq. 5.4) is not fully precise. Equation 5.7 is then defined in such a way that the original labels may be changed during optimization.

A modified version of the energy minimization software available at [BVZ01] was used in our implementation. Basically, the constraint that the original seeds must be kept was removed. The result is an image mask for foreground pixels.

The parameter σ_L determines the number of pixels in the initial object region. Lower values result in more pixels as shown in Fig. 5.4. Depending on the object material, it is remarkable that even small variations of the illumination can be detected by luminance differences. On the other hand, objects with highly specular,

Figure 5.8: Segmentation results of the images in Fig. 5.3 for several values of σ_L with $\sigma_C = 10$. Compare with Fig. 5.4. (left) $\sigma_L = 15$ (center) $\sigma_L = 25$ (right) $\sigma_L = 35$.

transparent or with complex structure materials are hard to be detected. This is the case of the hair in the images of Fig. 5.3. The object membership probabilities in the hair region (Fig. 5.5) show how the number of initial pixels affects the color distribution.

Fig. 5.8 shows the segmentation results for several values of $\sigma_L \in [15, 20]$. It is harder to segment the hair as σ_L assumes higher values. This happens when the pixels with high probability of this region are not enough to populate the histogram.

Figure 5.9: Segmentation results for the images in Fig. 5.3 for several values of σ_C with $\sigma_L = 20$. Compare with Fig. 5.6. (left) $\sigma_C = 5$ (center) $\sigma_C = 15$ (right) $\sigma_C = 25$.

Parameter σ_C controls how the image borders constrain the expansion or contraction of the object clusters during optimization. If its value is low, the difference of probability between the highest and lowest gradient values is high. As a result, the segmentation tends to be more fragmented and well aligned with high color variation areas. If the value of σ_C is high, the probabilities tend to vary more slowly, resulting in a smoother segmentation.

Segmentation results varying σ_C are shown in Fig. 5.9. Note that small back-

ground regions appear like holes inside the hair when $\sigma_C = 5$. Higher values of σ_C tend to classify these holes as foreground. As explained above, high values of σ_C smooth the segmented clusters.

Figure 5.10: (left) Boundary probabilities with $\sigma_C = 5$ and segmentation results for values of (center) $\sigma_L = 25,\ \sigma_C = 10$, (right) $\sigma_L = 15,\ \sigma_C = 5$.

In Figure 5.10 the segmentation result related to the input images shown in Figure 5.2 are presented. Observe the difference in the segmentation continuity when we vary the system parameters. In Figure 5.11 another example of input images and its segmentation is illustrated.

Figure 5.11: Input images and the final segmentation mask.

Chapter 6

Tonal Range and Tonal Resolution

In Chapter 2 the camera characteristic response function f, that characterizes sensors behavior respect to pixel exposure values was presented. In Chapter 3, f was recovered from a collection of differently exposed images. Then f^{-1} was used to linearize data captured by the camera, as well as to recover real scenes radiance values.

In this Chapter, image tonal range and resolution is discussed. The distinction between tonal range and tonal resolution is subtle and it is important to the understanding of some simple operations that can be done to enhance images tonal quality without using the powerful tools of HDR concept. The main question answered in this chapter is: *What can be done in terms of tonal enhancement of an image without knowledge about the camera characteristic f function and image radiance values?*

We start with a brief review on the recent HDR images research. Then the concept of relative tones is introduced and applied to obtain real-time tone enhanced video sequences.

6.1 HDRI reconstruction: absolute tones

The research on *High Dynamic Range Images* (HDRI) is looking forward to overcome sensors tonal range limitations. The goal is to achieve better representation and visualization of images, as well as to recover the scenes actual radiance values. We will refer to the scenes radiance values as *absolute tones*, since they are related to a physical real quantity. The usual images acquired by cameras with limited dynamic range are the *Low Dynamic Range Images* (LDRI).

An important observation when studying HDRI is that the input light $C(\lambda)$ is read in an interval around a reference value and the output C is discrete. What is being questioned is that C discretization should not be imposed by sensors nor displays limitations, but it should be adapted to the scene's luminance range. Instead of being driven by the device representation, the discretization should adapt to scenes tonal information.

6.1.1 HDRI Acquisition

Algorithms that recover HDRI by software usually combine collections of LDRIs. As discussed in Chapter 3, sensors characteristic response curve is recovered from the input images and then applied to obtain radiance values, as illustred in Figure 6.1.

To generate the collection of differently exposed images, several exposure camera parameters can be controlled. In the reconstruction phase the knowledge on which parameter has been changed should be considered since they affect image formation in different ways, e. g. introducing motion blur, defocus, etc.

To correctly reconstruct the radiance values, pixels correspondence between images is crucial. The correspondence depends on scenes features and temporal behavior as well as on which camera parameter is being changed to vary exposure. The pixel correspondence problem is usually stated as an optimization problem. It can be badly defined and hard to solve. A fast, robust, and completely automatic method for translational alignment of hand-held photographs is presented in [War03].

The f recovery was discussed in Chapter 3. With f in hand, the actual scene radiance values is obtained applying its inverse f^{-1} to the set of correspondent brightness values d_{ij}^k observed in the differently exposed images, where k is an index on the differently exposed images and ij are pixel coordinates. Different weights can be given according to the confidence on d_{ij}^k. If the pixel is almost over or underexposure, a lower weight is given to it, augmenting the influence of the middle of the f curve, where sensors (and films) are well behaved. It is required that at least one meaningful digital value is available for each pixel, that is, at least one pixel value in a set of correspondent pixel has a good confidence on the measured data.

In case one has an HDR sensor, the knowledge of its characteristic function is necessary to recover the w values (we recall that w are the radiance values), but the correlation step is often unnecessary since only one exposure is enough to register the whole range of intensities present in the scene (at least it is what is ex-

(a) HDRI recovery (b) Linear TMO (c) Histogram TMO

Figure 6.1: In (a) differently exposed LDR pictures from the same scene are processed to compose a HDR Radiance Map of the input images. A TMO is then applied to visualize the HDR image; figure (b) is visualized using linear TMO and figure (c) using histogram adjustment TMO.

pected from such devices). Scenes of high latitude range can be also synthetically generated by physically-based renderers like RADIANCE [War94].

6.1.2 HDRI Visualization

Usually HDR data is to be visualized on low dynamic range displays. The reduction of the range of image radiances to display brightness in a visual meaningful way is known as the tone mapping problem. Tone Mapping Operators (TMO) have been firstly proposed to solve the problem of visualization of synthetic images generated by physically based renderers before HDR from photographs became popular [LRP97].

In [DCWP02] the tone mapping research is reviewed and TMOs are classified in three main classes: *spatially uniform time independent*, *spatially uniform time dependent* and *spatially varying time independent* operators. Another survey in TMO research is [MNP97]. Recently, specific TMOs for HDR video visualization has been proposed. Below, some simple TMOs are described as well as the main reasoning that guides their intuition.

- *Linear Mapping*: the simplest way to reduce the range of an HDRI is by

using linear tone mapping, obtained as follows:

$$d = \frac{(w - w_{min})}{R_w} R_d + d_{min} \tag{6.1}$$

where $R_w = (w_{max} - w_{min})$
and $R_d = (d_{max} - d_{min})$.

- *Histogram Adjustment*: in [LRP97] is proposed the histogram adjustment heuristic, it is inspired on the fact that in typical scenes luminance levels occur in clusters rather than being uniformly distributed throughout the dynamic range. The algorithm proposed is:

$$d = P(w) R_d + d_{min} \tag{6.2}$$

where

$$\begin{aligned} P(B_w) &= \frac{[\sum_{b_i < w} h(b_i)]}{[\sum_{b_i < w_{max}} h(b_i)]} \\ &= \frac{H(w)}{H(w_{max})} \end{aligned} \tag{6.3}$$

- *Imaging system simulation*: techniques adopted in photographic printing process are a source of inspiration to create TMOs. A possible approach is the application of an specific film characteristic curve intending to simulate its look. More complex systems can also be simulated considering film development and enlargement techniques used in photographic laboratory and they are a source of inspiration to create TMOs, for instance, see [RSSF02, GM03, TR93].

- *Human eye simulation*: the behavior of human vision is not the same for all illumination conditions, specially in very dim scenes or when there are abrupt illumination changes in time varying scenes. The reality of visualized images is achieved only if human vision behavior is modeled. In [LRP97] the histogram adjustment algorithm is extended in many ways to model human vision. Also in [THG99] this approach is explored.

In Figure 6.1 the difference of image visualisation depending on the chosen TMO is illustred; after reconstruct the church HDR image, linear TMO was applied to visualize (b) and histogram adjustment was used to visualize (c). Comparison between TMOs is mainly perceptual, see [DMMS02]. In [Mac01] perception based image quality metrics are presented.

The Tone Mapping problem can be thought in image color domain as the problem of adjusting the range of image luminances to display luminances maintaining color chrominances. Thus it is a quantization problem in radiance domain and TMOs available in literature can be interpreted from this point of view. The uniform quantization algorithm is comparable to linear TMO, the populosity quantization is similar to histogram adjustment. Reasoning like that, a family of new TMOs derived from quantization algorithms can be proposed. The main difference between color quantization and tone mapping is that human eye is more sensitive to spatial changes in luminances than in color. This explains the focus of several TMOs in working on spatial domain, like in human eye perception simulation and in dodging-and-burning [RSSF02].

Another branch of research in HDRI visualisation is on HDR displays that are able to represent all tones encoded in an HDR file format [SHS*04].

6.1.3 HDRI Encoding

HDR data requires accurate tonal representation. Since the digital format can only encode discrete quantities, quantization is inevitable, the key point is to control the error of a high range of values. By simply using more bits to encode pixel values the number of possibly represented tones augment, however, such option can be highly inefficient.

The two approaches used to efficiently represent radiance data are floating point encoding and Log encoding. In floating point encoding a value v is represented using exponential notation as $x \cdot 2^y$, with x being the mantissa and y the exponent. In Log encoding a value v is represented as $a \cdot (\frac{b}{a})^v$, with a and b being the minimum and maximum values of the output range. The Log encoding representation is naturally correlated to the notation of stops and density values widely used in photography.

The error introduced by encoding in the two cases is different. It is easy to see that adjacent values in the Log encoding differ by $(\frac{b}{a})^N$, where N is the number of quantization steps of the discretization. Therefore, the quantization error is constant throughout the whole range. This is in contrast with the floating point encoding, which does not have equal step sizes.

A very comprehensive analysis of image encodings for HDR Images is available at `http://www.anyhere.com/gward/hdrenc/hdr_encodings.html`. The main HDRI available formats are: the Radiance RGBE, SGI LogLuv, ILM's OpenEXR and the scRGB encoding.

6.2 Partial reconstruction: relative tones

As we have seen, the camera characteristic f function is the tool that correlates radiance values to camera brightness values, and it is necessary to recover absolute radiance values from captured images. Here we introduce the conceptual difference between absolute and relative tone values.

We remind that image histograms are necessary and sufficient to recover the intensity mapping function τ, as shown in Chapter 3, useful to reconstruct the radiance map. The image histogram comparison expresses the concept that the m brighter pixels in the first image will be the m brighter pixels in the second image for all m discrete values assumed by the image.

We observe that a simple summation of two images preserves the information present on the histogram of the original images. The sum operation potentially doubles the number of distinct tones in the resulting image, consequently it requires one bit more to be stored. In Figure 6.2 we show an example of LDRI histograms, and the combined information present in the image sum.

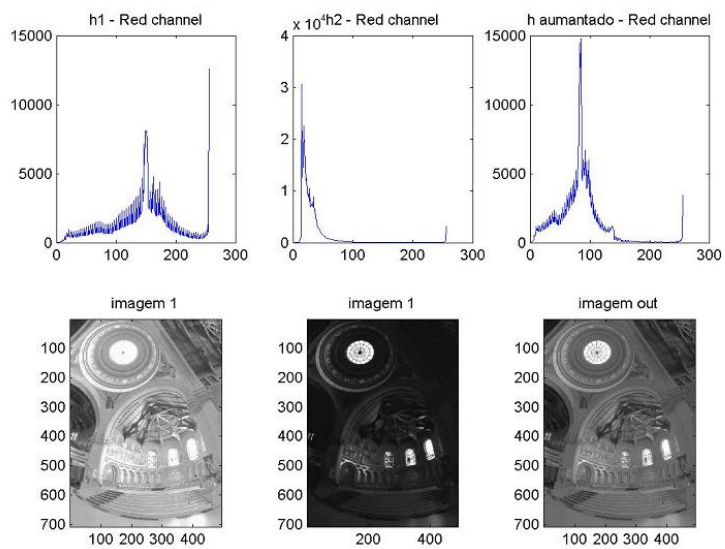


Figure 6.2: Example of two different exposed images with correspondent histograms, and the summation image with the combined histogram. To be visualized, the image sum was linearly tone mapped.

We then pose the following question: *what can be recovered from the sum of the images, if one does not know neither the exposures nor the response curve?*

Surprisingly, the answer is: many things!

Of particular interest to our discussion is the following theorem [GN03b]:

Theorem 3 (Simple Summation) *The sum of a set of images of a scene taken at different exposures includes all the information in the individual exposures.*

This means that, if one knows the exposure times used to obtain the images and the response curve of the camera, the radiance values and the individual images can be recovered from the image sum. In [GN03b], the authors use this result to optimize the camera acquisition parameters.

We then define the *relative tones* m as the values present in the summation image, while *absolute tones* w are the real correspondent radiances. The relative m values are unique indices to real radiance values. Thus, with the response function f and the exposure camera parameters in hand a look-up table can be generated mapping q to w values, i.e., $F_{f,\Delta_t} : [0, 2] \rightarrow [E_{min}, E_{max}]$. In Figure 6.3 we illustrate the relation between the quantization levels m and the absolute tone values w. We observe that, assuming that f is monotonically increasing, this mapping F is 1-1.

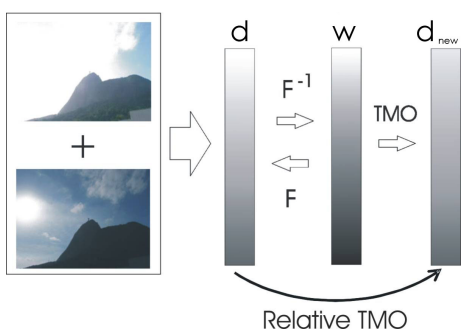


Figure 6.3: Absolute vs. Relative tone values.

Absolute tones are directly related to physical quantities while relative tones preserve absolute order but do not preserve magnitude.

Usually, TMOs are described in terms of absolute tones. However, since there is a 1-1 mapping between relative and absolute tone values, we conclude that TMOs to be applied directly to the relative tones can be proposed.

6.2.1 Active range-enhancement

When generating differently exposed images, instead of varying the exposure time, alternative ways to vary pixel exposure can be explored. By controlling light intensity, while keeping all other camera parameters fixed, the pixel irradiance w_{ij} is altered and hence its correspondent exposure value. This is not usually done, once the illumination intensity varies irregularly throughout the scene. However, the concept of relative tone values introduced here makes possible the use of active illumination in recovering relative tones to enhance tonal resolution on the actively illuminated area.

Changing light intensity the tonal partial ordering on actively illuminated areas is preserved once reflectance is proportional to incident radiance. Thus, relative tones are naturally recovered by simple summation in those regions.

The fact that illumination mostly affects the foreground pixels can be explored to perform foreground-background segmentation. The algorithm discussed in Chapter 5 can be used to produce a segmentation mask M, where 1 is assigned to the foreground pixels and 0 to the background, defining the actively illuminated regions.

Using active illumination, the foreground objects not only can be extracted but also can be tone-enhanced using relative tones concept. Note that the background cannot be tone-enhanced since there is no exposure variation in these regions. The summation image is given by $S(d_{ij}) = L(d_{ij}) + L(d_{ij})$, with $S(d_{ij}) \in [0, 2]$. As exposure values are different in two subsequent images tonal resolution is enhanced by simple summation.

To visualize the foreground tones Larson's histogram adjustment is applied, $\text{TMO}_{sv} : [0, 2] \rightarrow [0, 1]$, to the actively illuminated region. Chromaticities of both images are linearly combined to attenuate problems in over or under-exposed pixels.

The final image is obtained by a simple image composition:

$$C_{ij} = M_{ij}F_{ij} + (1 - M_{ij})B_{ij}$$

Where F_{ij} are the tone-enhanced foreground pixels and B_{ij} the background pixels that remain unchanged. A low-pass filter is applied to image M to avoid the perception of discontinuities in transition of segmented regions.

All discussion above assumes that one knows the pixel correspondence in differently exposed images. This is trivial for static scenes captured by cameras mounted on a tripod. For moving scenes, pixel correspondence has to be solved before applying the method outlined above.

6.3 Real-time Tone-Enhanced Video

For video implementation, a set-up composed by a video camera synchronized with a projector is used. A video signal, where each field has constant gray color values ρ_1 and ρ_2 with $\rho_1 \neq \rho_2$ is projected onto the scene. This signal is connected to the camera *gen*-lock pin, which guarantees projection/capture synchronization. Each light exposure lasts for $1/59.54s$ using the NTSC standard. The fields I^1 and I^2 of each captured frame represent the same object illuminated differently. This is sufficient for actively segment and tone-enhance the image pair.

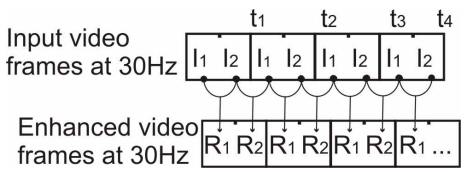


Figure 6.4: Two consecutive input fields result in one frame.

Input video images are in the Yuv color space. Thus, the processing is performed using the luminance defined as $L(p_{ij}) = Y(d_{ij})$, where Y is the video luminance channel. In this scheme, the tone-enhancement can be applied to any two consecutive fields. This produces an output video stream with the same input framerate, as shown in Fig. 6.4.

We assume that the framerate is high compared to the object's movement, thus, the effects of moving objects are small between a pair of fields. In the segmentation step, graph-cuts optimization is not applied, only the initial seed is used. To minimize undesirable effects of shadow regions, the projector is positioned very close to the camera. This implies that background should be far enough to not be affected by active illumination. The working volume can be determined by analyzing the projector intensity decay with distance (see Chapter 3).

Figures 6.5, 6.6 and 6.7 shows two consecutive video fields with different illumination, and their respective resulting tone enhancemed foreground. The projected gray values used were $\rho_1 = 0.35$ and $\rho_2 = 1$. The luminance threshold L_{min} used is 0.08. One can notice that our method has some trouble segmenting low-reflectance objects, such as hair. However, the resulting tone-enhanced images are still quite satisfactory.

A home made cheap version of the system was also implemented, it is composed by a web-cam synchronized with a CRT monitor playing the role of active illuminant. Some results are available at `http://www.impa.br/~asla/ahdr`.

(a) (b) (c)

Figure 6.5: Images (a) and (b) are the video input fields, while in (c) it is shown the tonal-enhanced foreground.

(a) (b) (c)

Figure 6.6: Images (a) and (b) are the video input fields, while in (c) it is shown the tonal-enhanced foreground.

(a) (b) (c)

Figure 6.7: Images (a) and (b) are the video input fields, while in (c) it is shown the tonal-enhanced foreground.

Chapter 7

Conclusion

This work was guided by the concept of active illumination, that was applied in several different contexts. The first problem that we have tackled was coded structured light methods for 3D photography purposes. We revisited the usage of color in code design and proposed a new minimal coding scheme. We also simplified the classification of structured light coding strategies proposed in [JPB04].

In this context, spatial variation of projected light is required, and hence, digital projectors are conveniently adopted as the active light source. The basic setup for structured light is a camera synchronized with a digital projector.

The proposed coding scheme was implemented for video and it is capable of acquiring depth maps together with scene colors at 30Hz using NTSC-based hardware. As a consequence of acquisition of both geometry and texture from the same data, the texture-geometry registration problem is avoided.

During the implementation of the video setup we observed that the hardware used to project slides and capture images has a direct influence on measurement accuracy. In a more subtle way, scene illumination conditions and the object's surface features also play an important role.

A crucial step that influences on depth map accuracy is the calibration of the system: poorly calibrated cameras or projectors cause error propagation in depth measurements. Unappropriate camera models (for instance, using a pinhole model when lens distortion is relevant) can also generate systematic errors.

We also concluded that the camera/projector photometric calibration is of great importance and that, if not applied, it may lead to decoding failures. We then studied more deeply the photometric calibration problem and proposed a procedure to calibrate the projector relatively to the camera used in the setup. There is still a lot to learn about projectors' photometric calibration, but we were

able to formalize the problem, proposing a non-linear model to determine both the projector *spectral characteristic matrix* and its *characteristic emitting function*.

Once the active setup was disponible and working for the depth recovery application, we started to explore other applications of the same setup. In particular, we propose to use active intensity variation for two different applications: background/foreground segmentation and, using the introduced concept of relative tones, image tone-enhancement.

The segmentation method is based on active illumination and employs graph-cut optimization to enhance the inition foreground mask obtained exploring the active illumination.

The key idea exploited in our method is that light variations can be designed to affect objects that are closer to the camera. In this way, a scene is lit with two different intensities of an additional light source that we call *segmentation light source*. By capturing a pair of images with such illumination, we are able to distinguish between objects in the foreground and the scene background.

Several light sources and different illumination schemes can be used to mark the foreground. The proposed method is fully automatic and does not require user intervention to label the image. Moreover, the energy function has only two parameters that must be specified: the standard-deviations of the normal background and boundary distributions. These parameters can be tuned only once for a wide variety of images with similar light setup.

The main technical contributions of this work are the concept of foreground / background segmentation by active lighting and the design of a suitable energy function to be used in the graph-cut minimization.

The quality of the masks produced by our method is, in general, quite good. Some difficult cases may arise when the objects are highly specular, translucent or have very low reflectance. Because of its characteristics, this method is well suited for applications in which the user can control the scene illumination; for example, in studio situations and/or using a flash/no-flash setup.

This method can be naturally extended to active segmentation of video sequences. All that it is required for this purpose is a synchronized camera/projector system.

Joining HDR images concepts with active light intensity variation we got aou third application: the image tone-enhancement. The additional information resulting from capturing two images of the same scene was used to extend the dynamic range and the tonal resolution of the final image. This technique is made possible by the concept of relative tone values introduced in this work. We remark that relative tones is a key concept in HDR theory and has many other applications,

which we intend to exploit in future work.

The synchronized camera and projector setup already implemented was then adopted to produce tonal-enhanced video. This system has many advantages, such as good cost performance, compatibility and various options of distribution channels. The data processing of our system can be easily incorporated into the pipeline of a video production environment.

Although our implementation has been done in real time for video, the same idea could be used in digital cameras, by exploiting flash - no flash photography. There are many recent works [PAH*04, ED04] that explore the use of programmable flash to enhance image quality, but they do not use HDR concepts. Our work gives a contribution to this new area of computational photography.

For both applications, segmentation and tone-enhancement, spatial variation of the active light source was not assumed. Therefore, although a digital projector can be used, much simpler controlled light sources can replace the projector in the implementation of the proposed methods. We have implemented the proposed segmentation using intensity variable flashes and a home made setup using a monitor as light source for tone enhancement.

The main limitation of using active light in different contexts is that the active light should be the main light source present in the scene or, at least, strong enough to be distinguished from other light sources. This is a very important consideration when planning scene illumination as well as the active light positioning.

7.1 Future Work

As future work we point ou some natural extensions to the proposed applications as well as deeper exploitation of projector calibration potentials:

- Model and calibration of radial decay of projector intensity.
- Better decoding and texture recovery by exploring projector calibration to enhance the coded structured-light results.
- Graph-cut segmentation for video, benefiting from temporal coherence of subsequent frames; better if working in real time.
- Tone-enhanced matting; better if working in real time.

Bibliography

[Ada80] ADAMS A.: *The Camera, The Ansel Adams Photography series*. Little, Brown and Company, 1980.

[Ada81] ADAMS A.: *The Negative, The Ansel Adams Photography series*. Little, Brown and Company, 1981.

[Ada83] ADAMS A.: *The Print, The Ansel Adams Photography series*. Little, Brown and Company, 1983.

[ADA*04] AGRAWALA A., DONCHEVA M., AGRAWALA M., DRUCKER S., COLBURN A., CURLESS B., SALESIN D., COHEN M.: Interactive digital photomontage. In *Computer Graphics Proceedings ACM SIGGRAPH* (2004), pp. 294–302.

[BJ01] BOYKOV Y., JOLLY M.: Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Proceedings of ICCV* (2001).

[BMS98] BATLLE J., MOUADDIB E., SALVI J.: Recent progress in coded structured light as a technique to solve the correspondence problem: A survey. *Pattern Recognition 31*, 7 (1998), 963–982.

[BVZ98] BOYKOV Y., VEKSLER O., ZABIH R.: Markov random fields with efficient approximation. In *IEEE Conference on Computer Vision and Pattern Recognition* (1998), pp. 648–655.

[BVZ01] BOYKOV Y., VEKSLER O., ZABIH R.: Fast approximate energy minimization via graph cuts. *IEEE Transactions on PAMI 23* (2001), 1222–1239.

[CH85] CARRIHILL B., HUMMEL R.: Experiments with the intensity ratio depth sensor. *Comput. Vision Graphics Image Process 32* (1985), 337–358.

[D.B96] D.B.WEST: *Introduction to graph theory*. Prentice Hall, 1996.

[DCWP02] DEVLIN K., CHALMERS A., WILKIE A., PURGATHOFER W.: Star: Tone reproduction and physically based spectral rendering. In *State of the Art Reports, Eurographics 2002* (September 2002), pp. 101–123.

[DM97] DEBEVEC P., MALIK J.: Recovering high dynamic range radiance maps from photographs. In *Proc. ACM SIGGRAPH '97* (1997), pp. 369–378.

[DMMS02] DRAGO F., MARTENS W., MYSZKOWSKI K., SEIDEL H.: Perceptual evaluation of tone mapping operators with regard to similarity and preference. In *Tech. Repport MPI-I-2202-4-002* (2002), p. ?

[ED04] EISEMANN E., DURAND F.: Flash photography enhancement via intrinsic relighting. *Computer Graphics Proceedings ACM SIGGRAPH* (2004), ?

[EN99] E.HORN, N.KIRYATI: Toward optimal structured light patterns. *Image and Vision Computing 17*, 2 (1999), 87–97.

[GHS01] GOESELE M., HEIDRICH W., SEIDEL H.: Color calibrated high dynamic range imaging with ICC profiles. In *Proc. of the 9th Color Imaging Conference Color Science and Engineering: Systems, Technologies, Applications, Scottsdale* (November 2001), pp. 286–290.

[Gla94] GLASSNER A.: *Principles of Digital Image Synthesis*. Morgan Kaufmann Publishers Inc., 1994.

[GM03] GEIGEL J., MUSGRAVE F.: A model for simulating the photographic development process on digital images. In *Comp. Graph. Proc.* (2003), pp. 135–142.

[GN03a] GROSSBERG M., NAYAR S.: Determining the camera response from images: What is knowable? *IEEE Trans.PAMI 25*, 11 (Nov. 2003), 1455–1467.

[GN03b] GROSSBERG M., NAYAR S.: High dynamic range from multiple images: Which exposures to combine? In *?* (2003), pp. ?–?

[GN04] GROSSBERG M., NAYAR S.: Modeling the space of camera response functions. *IEEE Trans.PAMI 26*, 10 (Oct. 2004), 1272–1282.

[Goe04] GOESELE M.: *New acquisition Techniques for Real Objects and Light Sources in Computer Graphics*. Verlag, 2004.

[GPS89] GREIG D., PORTEOUS B., SEHEULT A.: Exact maximum a posteriori estimation for binary images. *J. Royal Statistical Soc.* (1989), 271–279.

[GV97] GOMES J., VELHO L.: *Image Processing for Computer Graphics*. Springer, 1997.

[Hsi01] HSIEH Y. C.: Decoding structured light patterns for three-dimensional imaging systems. *Pattern Recognition 34*, 2 (2001), 343–349.

[IG98] ISHIKAWA H., GEIGER D.: Oclusions, discontinuities, and epipolar lines in stereo. In *Fifth European Conference on Computer Vision, (ECCV'98)* (Freiburg, Germany, 2-6 June 1998).

[JM82] J.L.POSADAMER, M.D.ALTSCHULER: Surface measurement by space-encoded projected beam systems. *Comput. Graphics Image Process 18* (1982), 1–17.

[JM90] J.TAJIMA, M.IWAKAWA: 3-d data acquisition by rainbow range finder. In *Proc. Int. Conf. on Pattern Recognition* (1990), pp. 309–313.

[JPB04] J.SALVI, PAGES J., BATLLE J.: Pattern codification strategies in structured light systems. *Pattern Recognition 37* (2004), 827–849.

[KA87] K.L.BOYER, A.C.KAK: Color-encoded structured light for rapid active ranging. *IEEE Trans. Pattern Anal. Mach. Intell. 9*, 1 (1987), 14–28.

[KZ04] KOLMOGOROV V., ZABIH R.: What energy functions can be minimized via graph cuts? *Proc. IEEE Transactions on Pattern Analysis and Machine Inteligence 26* (2004), 147–159.

[LBS02] L.ZHANG, B.CURLESS, S.M.SEITZ: Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *Proc. Symp. on 3D Data Processing Visualization and Transmission (3DPVT)* (2002).

[Len03] LENSCH H.: *Efficient, Image-Based Appearance Acquisition of Real World Objects*. MPI PhD Thesis, 2003.

[Lit] LITWILLER D.: Ccd vs. cmos: Facts and fiction. *http://www.dalsa.com*.

[LRP97] LARSON G., RUSHMEIER H., PIATKO C.: A visibility matching tone reproduction operator for high dynamic range scenes. *IEEE Trans. on Vis. and Comp. Graph. 3*, 4 (1997), 291–306.

[Mac01] MACNAMARA A.: Visual perception in realistic image synthesis. *Computer Graphics Forum 20*, 4 (2001), 221–224.

[Mal84] MALZ R. W.: *Handbook of computer vision and applications vol. 1, cap 20 3D sensors for high-performance surface measurement in reverse engineering*. 1984.

[MBR*00] MATUSIK W., BUEHLER C., RASKAR R., GORTLER S. J., MCMILLAN L.: Image based visual hulls. In *Proc. SIGGRAPH 2000* (2000).

[MNP97] MATKOVIC K., NEUMANN L., PURGATHOFER W.: *A Survey of Tone Mapping Techniques*. Tech. Rep. TR-186-2-97-12, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, apr 1997. human contact: technical-report@cg.tuwien.ac.at.

[Mon94] MONKS T.: *Measuring the shape of time-varying objects*. PhD thesis, Department of Eletronics and Computer Science, University of Southampton, 1994.

[OHS01] O.HALL-HOLT, S.RUSINKIEWICZ: Stripe boundary codes for real-time structured-light range scanning of moving objects. In *Proc. ICCV* (2001), pp. 13–19.

[PA90] P.VUYLSTEKE, A.OOSTERLINCK: Range image acquisition with a single binary-encoded light pattern. *IEEE Trans. PAMI 12*, 2 (1990), 148–164.

[PAH*04] PETSCHNIGG G., AGRAWALA M., HOPPE H., SZELISKI R., COHEN M., , TOYAMA K.: Digital photography with flash and no-flash image pairs. *Computer Graphics Proceedings ACM SIGGRAPH* (2004), 664–672.

[Paj95] PAJDLA T.: *BCRF - Binary Illumination Coded Range Finder: Reimplementation*. Tech. Rep. KUL/ESAT/MI2/9502, Katholieke Universiteit Leuven, Belgium, 1995.

[PGG] P.KONINCKX T., GRIESSER A., GOOL L. V.: Real-time range scanning of deformable surfaces by adaptively coded structured light. In *In Fourth International Conference on 3-D Digital Imaging and Modeling - 3DIM03*, p. 293300.

[Powa] POWELL E.:. http://www.ProjectorCentral.com.

[Powb] POWELL E.: Coded structured light course. http://eia.udg.es/ qsalvi/iitap/curs2001/tema3_structured_light/sld001.htm.

[RBS99] ROBERTSON M., BORMAN S., STEVENSON R.: Dynamic range improvement through multiple exposures. In *Proceedings of the IEEE International Conference on Image Processing* (Kobe, Japan, Oct. 1999), vol. 3, IEEE, pp. 159–163.

[RKB04] ROTHER C., KOLMOGOROV V., BLAKE A.: Grabcut - interactive foreground extraction using iterated graph cuts. In *Computer Graphics Proceedings ACM SIGGRAPH* (2004), pp. 309–314.

[RM03] R.YANG, M.POLLEFEYS: Multi-resolution real-time stereo on commodity graphics hardware. In *Proc. CVPR* (2003).

[RSSF02] REINHARD E., STARK M., SHIRLEY P., FERWERDA J.: Photographic tone reproduction for digital images. In *Proc. ACM SIGGRAPH '02* (2002), pp. ?–?

[RTF*04] RASKAR R., TAN K., FERIS R., YU J., TURK M.: Non-photorealistic camera: Depth edge detection and stylized rendering

using multi-flash imaging. *Computer Graphics Proceedings ACM SIGGRAPH* (2004), 679–688.

[SB96] SMITH A., BLINN J.: Blue screen matting. *Computer Graphics Proceedings ACM SIGGRAPH* (1996), 259–268.

[SCV02] SÁ A., CARVALHO P., VELHO L.: (b, s)-bcsl: Structured light color boundary coding for 3d photography. In *Proc. of 7th International Fall Workshop on Vision, Modeling, and Visualization* (2002).

[SF84] S.INOKUCHI K., F.MATSUDA: Range-imaging for 3d object recognition. In *Proc. Int. Conf. on Pattern Recognition* (1984), pp. 806–808.

[SHS*04] SEETZEN H., HEIDRICH W., STUERZLINGER W., WARD G., WHITEHEAD L., TRENTACOSTE M., GHOSH A., VOROZCOVS A.: High dynamic range display systems. In *Proc. ACM SIGGRAPH '04* (2004), pp. ?–?

[SM02] S.RUSINKIEWICZ O.-H., M.LEVOY: Real-time 3d model acquisition. In *Proc. SIGGRAPH 2002* (2002), p. ?

[SVCV05] SÁ A., VIEIRA M., CARVALHO P., VELHO L.: Range-enhanced active forground extraction. In *Proc. ICIP* (2005).

[THG99] TUMBLIN J., HODGINS J., GUENTER B.: Two methods for display of high contrast images. *ACM Transactions on Graphics 18*, 1 (1999), 5694.

[TR93] TUMBLIN J., RUSHMEIER H.: Tone reproduction for realistic images. *IEEE Computer Graphics and Applications 13*, 6 (1993), 4248.

[VP96] V.SMUTNY, PAJDLA T.: *Rainbow Range Finder and its Implementation at the CVL*. Tech. Rep. K335-96-130, Czech Technical University, Prague, 1996.

[VSVC05] VIEIRA M., SÁ A., VELHO L., CARVALHO P.: A camera-projector system for real-time 3d video. In *IEEE International Workshop on Projector-Camera Systems (PROCAMS)* (2005).

[VVSC04] VIEIRA M., VELHO L., SÁ A., CARVALHO P.: Real-time 3d video. In *SIGGRAPH 2004 Visual Proceedings* (2004).

[War94] WARD G.: The radiance lighting simulation and rendering system. In *Proc. ACM SIGGRAPH '04* (1994), pp. 459–472.

[War03] WARD G.: Fast, robust image registration for compositing high dynamic range photographcs from hand-held exposures. *Journal of Graphics Tools 8*, 2 (2003), 17–30.

[WBC*05] WANG J., BHAT P., COLBURN R., AGRAWALA M., , COHEN M.: Interactive video cutout. *Computer Graphics Proceedings ACM SIGGRAPH* (2005), ?

[ZSCS04] ZHANG L., SNAVELY N., CURLESS B., SEITZ S. M.: Spacetime faces: high resolution capture for modeling and animation. vol. 23, p. 548558.