

Chapter 1

Exposure Correction for Imaging Devices: an Overview

S. BATTIATO

Department of Mathematics and Computer Science
University of Catania
Catania, Italy

Email: battiato@dmi.unit.it

G. MESSINA

Advanced System Technology
ST Microelectronics
Catania, Italy

Email: giuseppe.messina@st.com

A. CASTORINA

Advanced System Technology
ST Microelectronics
Catania, Italy

Email: alfio.castorina@st.com

1.1 Introduction

One of the main problems affecting image quality, leading to unpleasant pictures, comes from improper exposure to light. Beside the sophisticated features incorporated in todays cameras (i.e., automatic gain control algorithms), failures are not unlikely to occur. Digital consumer devices make use of ad-hoc strategies and heuristics to derive exposure setting parameters. Typically such techniques are completely blind with respect to the specific content of the involved scene. Some techniques are completely automatic, cases in point being represented by those based on average/automatic exposure metering or the more complex matrix/intelligent exposure metering. Others, again, accord the photographer a certain control over the selection of the exposure, thus allowing space for personal taste or enabling him to satisfy particular needs. Inspite of the great variety of methods for regulating the exposure and the complexity of some of them, it is not rare for images to be acquired with a nonoptimal or incorrect exposure. This is particularly true for handset devices (e.g., mobile phones) where several factors contribute to acquire bad-exposed pictures: poor optics, absence of flashgun, not to talk about difficult input scene lighting conditions, and so forth.

There is no exact definition of what a correct exposure should be. It is possible to abstract a generalization and to define the best exposure that enables one to reproduce the most important regions (according to contextual or perceptive criteria) with a level of gray or brightness, more or less in the middle of the possible range. In any case if the dynamic range of the scene is sensibly "high" there is no way to acquire the overall involved details. One of the main issue of this chapter is devoted to give an effective overview of the technical details involved in:

- Exposure settings of imaging devices just before acquisition phase (i.e Pre-Capture phase) [1];
- Content-dependent enhancement strategies applied as post-processing [2];
- Advanced solution where multi-picure acquisition of the same scene with different exposure time allows to reproduce the Radiance map of the real world [3].

The rest of the chapter is organized as follows. The initial section will discuss in details traditional and advanced approaches related to the Pre-Capture phase (i.e., the sensor is read continuously and the output is analyzed in order to determine a set of parameters strictly related with the quality of the final picture [1]). The role of exposure setting will be analyzed also considering some case studies where, by making use of some assumptions about the dynamic range of the real scene, it is possible to derive effective strategies. Section 3 will describe the work presented in [2] where by using post-processing techniques an effective enhancement has been obtained just analyzing some content dependent features of the picture. Finally, in

Section 4 a brief review of advanced approaches devoted to improve acquiring capabilities by using multi-picture acquisition (i.e., bracketing) will be provided. In particular a brief overview of the main techniques able to reproduce effectively the salient part of a real scene after having computed a reliable HDR (High Dynamic Range) [3] will be detailed.

1.2 Exposure Metering Techniques

Metering techniques built into the camera are getting much better with the introduction of computer technology but limitations still remain. For example taking a picture on a snow scene or trying to photograph a black locomotive without overriding the camera calculated metering is very difficult. The most important aspect of the exposure duration is to guarantee that the acquired image falls in a good region of the sensors sensitivity range. In many devices, the selected exposure value is the main processing step for adjusting the overall image intensity that the consumer will see. Many of the first digital cameras used a separate metering system to set exposure duration, rather than using data acquired from the sensor chip. Integrating exposure-metering function into the main sensor (through-the-lens, or TTL, metering) may reduce system cost. The imaging community uses a measure called *exposure value* (EV) to specify the relationship between the f-number¹, F , and exposure duration, T :

$$EV = \log_2\left(\frac{F^2}{T}\right) = 2\log_2(F) - \log_2(T) \quad (1.1)$$

The exposure value (1.1) becomes smaller as the exposure duration increases, and it becomes larger as the f-number grows. Most Auto-exposure algorithms work this way:

1. Take a picture with a pre-determined exposure value (EV_{pre});
2. Convert the RGB values to brightness, B ;
3. Derive a single value B_{pre} (like center-weighted mean, median, or more complicated weighted method as in matrix-metering) from the brightness picture;
4. Based on linearity assumption and equation (1.1), the optimum exposure value EV_{opt} should be the one that permits a correct exposure. The picture taken at this EV_{opt} should give a number close to a pre-defined ideal value B_{opt} , thus:

$$EV_{opt} = EV_{pre} + \log_2(B_{pre}) - \log_2(B_{opt}) \quad (1.2)$$

The ideal value B_{opt} for each algorithm is typically selected empirically. Different algorithms mainly differ in how the single number B_{pre} is derived from the picture.

¹f-numbers, or aperture values, are measurement of the size of the hole that the light passes through int the rear of the lens, relative to the focal length. The smaller the f-number, the more light gets through the lens.

1.2.1 Classical Approaches

The metering system in a digital camera measures the amount of light in the scene and calculates the best-fit exposure value based on the metering mode explained below. Automatic exposure is a standard feature in all digital cameras. After having selected the metering mode, it is requested just pointing the camera and pressing the shutter release. The metering method defines which information of the scene is used to calculate the exposure value and how it is determined. Cameras generally allow the user to select between spot, center-weighted average, or multi-zone metering modes.

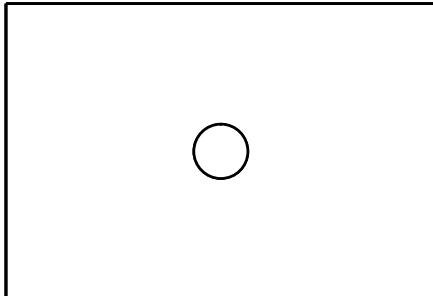
Spot Metering

Spot metering allows user to meter the subject in the center of the frame (or on some cameras at the selected AutoFocus (AF) point). Only a small area of the whole frame (between 1-5% of the viewfinder area) is metered while the rest of the frame is ignored. In this case B_{pre} (1.2) is the mean of the center area (see Figure 1.1(a)). This will typically be the effective centre of the scene, but some cameras allow the user to select a different off-center spot, or to recompose by moving the camera after metering. A few models support a Multi-Spot mode which allows multiple spot meter readings to be taken of a scene that are averaged. Both of those cameras and others also support metering of highlight and shadow areas. Spot metering is very accurate and is not influenced by other areas in the frame. It is commonly used to shoot very high contrast scenes. For example (see Figure 1.1(b)), if the subject's back is being hit by the rising sun and the face is a lot darker than the bright halo around the subject's back and hairline (the subject is "backlit"), spot metering allows the photographer to measure the light bouncing off the subject's face and expose properly for that, instead of the much brighter light around the hairline. The area around the back and hairline will then become over-exposed. Spot metering is a method upon which the zone system depends².

Partial Area Metering

This mode meters a larger area than spot metering (around 10-15% of the entire frame), and is generally used when very bright or very dark areas on the edges of the frame would otherwise influence the metering unduly. Like spot metering, some cameras can use variable points to take readings from (in general autofocus points), or have a fixed point in the centre of the viewfinder. In Figure 1.1(d) an example of partial metering on a backlight scene is shown; this method permits to equalize much more the global exposure.

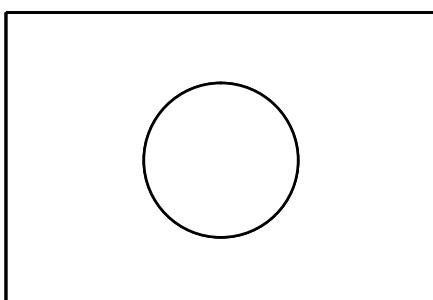
²The Zone System is a photographic technique for determining optimal film exposure and development, formulated by Ansel Adams and Fred Archer in 1941. The Zone System provides photographers with a systematic method of precisely defining the relationship between the way they visualize the photographic subject and the final results. Although it originated with black and white sheet film, the Zone System is also applicable to roll film, both black and white and color, negative and reversal, and to digital photography.



(a) Spot Area



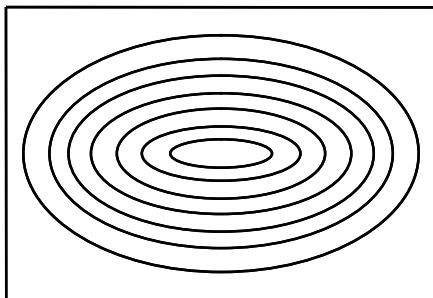
(b) Spot Example



(c) Partial Area



(d) Partial Area Example



(e) Center Weighted Area



(f) Center Weighted Example

Figure 1.1: Metering examples

Center-weighted Average Metering

This method is probably the most common metering method implemented in nearly every digital camera: it is also the default for those digital cameras which don't offer metering mode selection. In this system, as described in Figure 1.1(e), the meter concentrates between 60 to 80 percent of the sensitivity towards the central part of the viewfinder. The balance is then "feathered" out towards the edges. Some cameras allow the user to adjust the weight/balance of the central portion to the peripheral one. One advantage of this method is that it is less influenced by small areas that vary greatly in brightness at the edges of the viewfinder; as many subjects are in the central part of the frame, consistent results can be obtained. Unfortunately, if a backlight is present into the scene the central part results darker than the rest of the scene (Figure 1.1(f)), and unpleasant underexposed foreground is produced.

Average Metering

In this mode the camera will use the light information coming from the entire scene and averages for the final exposure setting, giving no weighting to any particular portion of the metered area. This metering technique has been replaced by Centred-Weighted metering, thus is only obsolete and present in older cameras only.

1.2.2 Advanced Approaches

Matrix or Multi-zone Metering

This mode is also called matrix, evaluative, honeycomb, segment metering, or esp (electro selective pattern) metering on some cameras. It was first introduced by the Nikon FA, where it was called Automatic Multi-Pattern metering. On a number of cameras this the default/standard metering setting. The camera measures the light intensity in several points of the scene, and then combines the results to find the settings for the best exposure. How they are combined/calculated deviates from camera to camera. The actual number of zones used varies wildly, from several to over a thousand. However performance should not be concluded on the number of zones alone, or the layout. As shown in Figure 1.2 the layout can change drastically from a manufacturer to another, also within the same company the use of different multi-zone metering can change due to several reason (e.g., the dimension of the final pixel matrix).

Many manufacturers are less than open about the exact calculations used to determine the exposure. A number of factors are taken into consideration, these include: AF point, distance to subject, areas in-focus or out-of-focus, colours/hues of the scene, and backlighting. Multi-zone tends to bias its exposure towards the autofocus point being used (while taking into account other areas of the frame too), thus ensuring that the point of interest has been properly exposed (it is also designed to avoid the need to use exposure compensation in most situations). A database of many thousands of exposures is pre-stored in the camera,

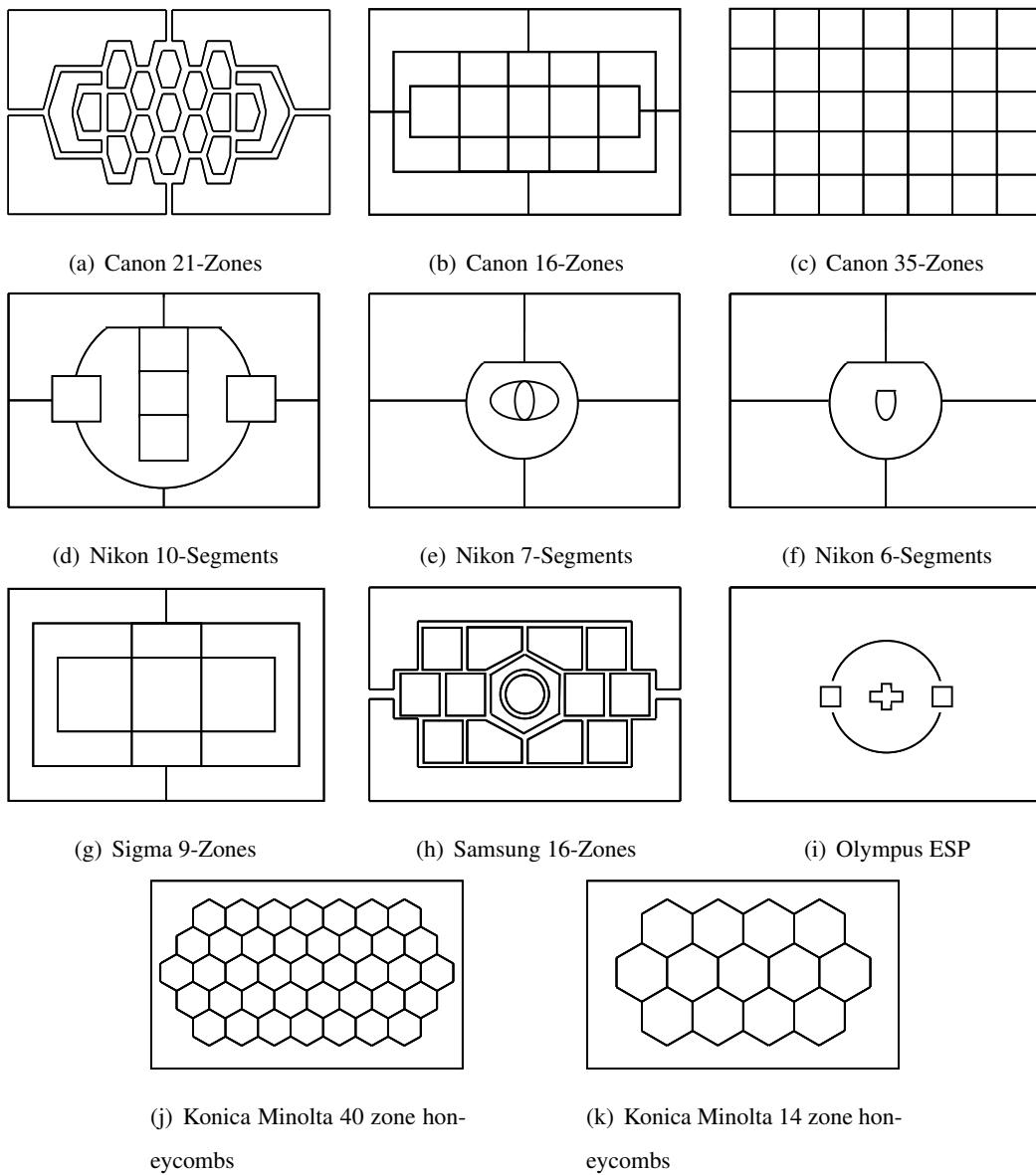


Figure 1.2: Examples of different kind of Multi-zone Metering mode used by several cameras manufacturers.

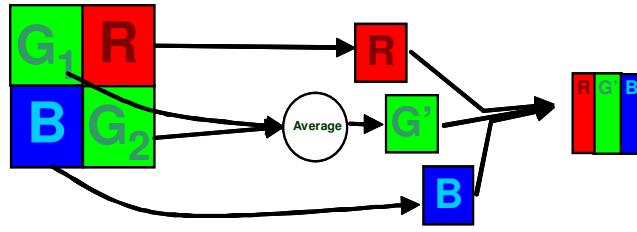


Figure 1.3: Bayer data sub-sampling generation.

and the processor can use a selective pattern to determine what is being photographed. Some cameras allow the user to link (or unlink) the autofocus and metering, giving the possibility to lock exposure once AF confirmation is achieved, AEL (Auto Exposure Lock). Using manual focus, and on many compacts cameras, the AF point is not used as part of the exposure calculation, in such instances it is common for the metering to default to a central point in the viewfinder, using a pattern based off of that area. Some users have problems with wide angle shots in high contrast, due to the large area which can vary greatly in brightness, it is important to understand that even in this situation, the focus point can be critical to the overall exposure.

1.3 Exposure Correction Content Dependent

As explained in section 1.2, it is possible to define the best exposure able to reproduce the most important regions (according to contextual or perceptive criteria) with a level of gray or brightness, more or less in the middle of the possible range. After acquisition phase typical post-processing techniques try to realize an effective enhancement by using global approaches: histogram specification, histogram equalization and gamma correction to improve global contrast appearance [4] only stretch the global distribution of the intensity. More adaptive criterions are needed to overcome such drawback. The exposure correction technique [2] described in this section is designed essentially for mobile sensors applications. This new element, present in newest mobile devices, is particularly harmed by “backlight” when the user utilizes a mobile device for video phoning. The detection of skin characteristics in captured images allows selection and properly enhancement and/or tracking of regions of interest (e.g., faces). If no skin is present in the scene the algorithm switches automatically to other features (such as contrast and focus) tracking for visually relevant regions. This implementation differs from the algorithm described in [5] because the whole processing can also be performed directly on Bayer pattern images [6], and simpler statistical measures were used to identify *information carrying* regions. The algorithm is defined as follows:

1. Luminance extraction. If the algorithm is applied on Bayer data, in place of the three full colour

planes, a sub-sampled (quarter size) approximated input data (Figure 1.3) is used.

2. Using a suitable features extraction technique the algorithm fixes a value to each region. This operation permits to seek visually relevant regions (for contrast and focus the regions are block based, for skin recognition the regions are associated to each pixel).
3. Once the ‘visually important’ pixels are identified (e.g., the pixels belonging to skin features) a global tone correction technique is applied using as main parameter the mean gray level of the relevant regions.

1.3.1 Features Extraction: Contrast and Focus

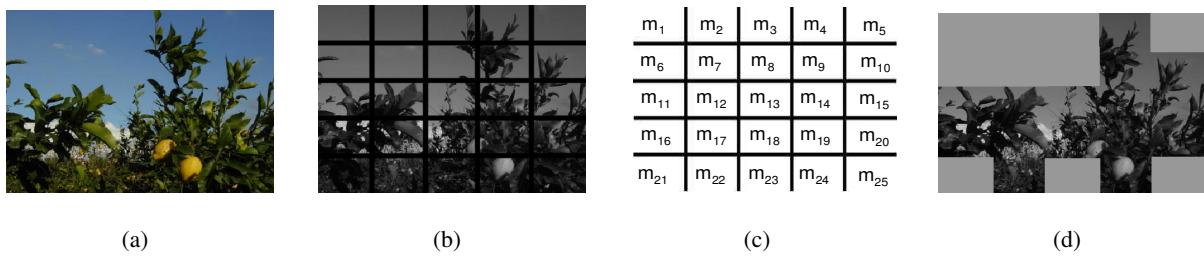


Figure 1.4: Features extraction pipeline (for focus and contrast with $N = 25$). Visual relevance of each luminance block (b) of the input image (a) is based on relevance measures (c) able to obtain a list of relevant blocks (d).

To be able to identify regions of the image that contain more information, the luminance plane is subdivided in N blocks of equal dimensions (in our experiments we employed $N=64$ for VGA images). For each block, statistical measures of “contrast” and “focus” are computed. Therefore it is assumed that well focused or high-contrast blocks are more relevant compared to the others. Contrast refers to the range of tones present in the image. A high contrast leads to a higher number of perceptual significance regions inside a block.

Focus characterizes the sharpness or edgeness of the block and is useful in identifying regions where high frequency components (i.e., details) are present. If the aforementioned measures were simply computed on highly underexposed images, then the regions having better exposure would always have higher contrast and edgeness compared to those that are obscured. In order to perform a visual analysis revealing the most important features regardless to lighting conditions, a new ‘visibility image’ is constructed by pushing the mean gray level of the input green Bayer pattern plane (or the Y channel for colour images) to 128. The push operation is performed using the same function that is used to adjust the exposure level and it will be described later. The contrast measure is computed by simply building a histogram for each block and then

calculating its deviation (1.4) from the mean value (1.5). A high deviation value denotes good contrast and vice versa. In order to remove irrelevant peaks, the histogram is slightly smoothed by replacing each entry with its mean in a ray 2 neighborhood. Thus, the original histogram entry is replaced with the gray-level $\tilde{I}[i]$:

$$\tilde{I}[i] = \frac{(I[i-2] + I[i-1] + I[i] + I[i+1] + I[i+2])}{5} \quad (1.3)$$

Histogram deviation D is computed as :

$$D = \frac{\sum_{i=0}^{255} |i - M| \cdot \tilde{I}[i]}{\sum_{i=0}^{255} \tilde{I}[i]} \quad (1.4)$$

where M is the mean value :

$$M = \frac{\sum_{i=0}^{255} i \cdot \tilde{I}[i]}{\sum_{i=0}^{255} \tilde{I}[i]} \quad (1.5)$$

The focus measure is computed by convolving each block with a simple 3x3 Laplacian filter.

In order to discard irrelevant high frequency pixels (mostly noise), the outputs of the convolution at each pixel are thresholded. The mean focus value of each block is computed as:

$$F = \frac{\sum_{i=1}^N \text{thresh}[\text{lapl}(i), \text{Noise}]}{N} \quad (1.6)$$

where N is the number of pixels and the $\text{thresh}()$ operator discards values lower than a fixed threshold Noise . Once the values F and D are computed for all blocks, relevant regions will be classified using a linear combination of both values. Features extraction pipeline is illustrated in Figure 1.4.

1.3.2 Features Extraction: Skin Recognition

As before a “visibility image” obtained forcing the mean gray level of the luminance channel to be about 128 is built. Most existing methods for skin colour detection usually threshold some sort of measure of the likelihood of skin colours for each pixel and treat them independently. Human skin colours form a special category of colours, distinctive from the colours of the most other natural objects. It has been found that human skin colours are clustered in various colour spaces ([7], [8]). The skin colour variations between people are mostly due to intensity differences. These variations can therefore be reduced by using chrominance components only. Yang et al. [9] have demonstrated that the distribution of human skin colours can be represented by a two-dimensional Gaussian function on the chrominance plane. The center of this distribution is determined by the mean vector $\vec{\mu}$ and its shape is determined by the covariance matrix Σ ; both values can be estimated from an appropriate training data set. The conditional probability $p(\vec{x}|s)$ of a block belonging to the skin colour class, given its chrominance vector \vec{x} is then represented by:

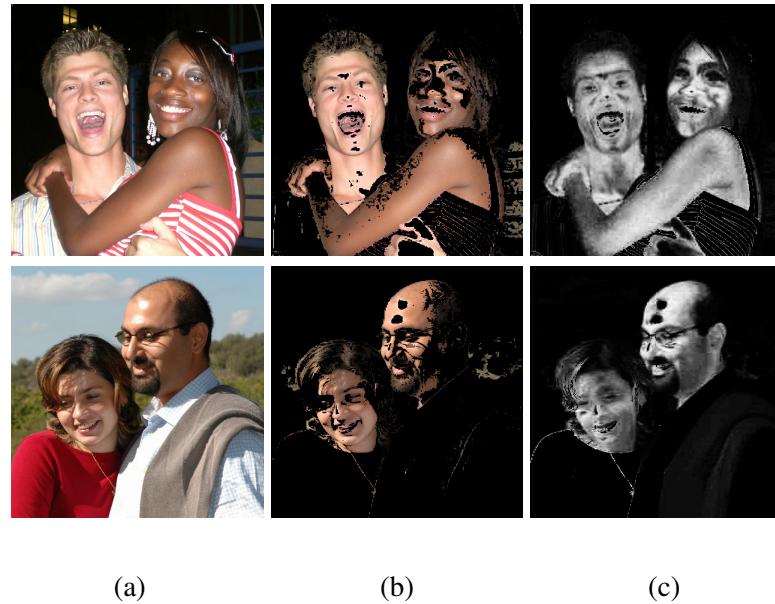


Figure 1.5: Skin recognition examples on RGB images: (a) original images compressed in JPEG format; (b) simplest threshold method output; and (c) probabilistic threshold output.

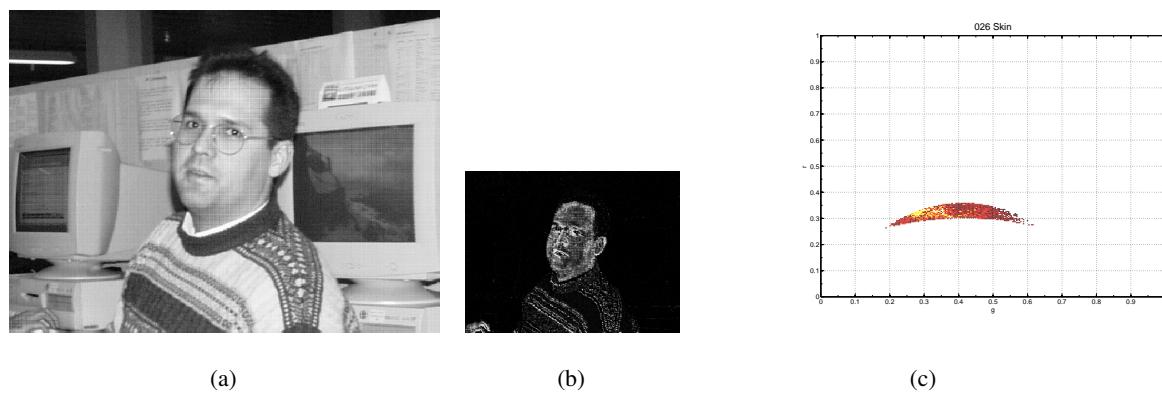


Figure 1.6: Skin recognition examples on Bayer pattern image: (a) original image in Bayer data; (b) recognized skin with probabilistic approach; and (c) threshold skin values on $r - g$ bidirectional histogram (*skinlocus*).

$$p(\vec{x}|s) = \frac{1}{2\pi} |\Sigma|^{-\frac{1}{2}} \exp \left\{ \frac{-[d(\vec{x})]^2}{2} \right\} \quad (1.7)$$

where $d(\vec{x})$ is the so-called Mahalanobis distance from the vector \vec{x} to the mean vector $\vec{\mu}$ and defined as:

$$[d(\vec{x})]^2 = (\vec{x} - \vec{\mu})' \Sigma^{-1} (\vec{x} - \vec{\mu}) \quad (1.8)$$

The value $d(\vec{x})$ determines the probability that a given block belongs to the skin colour class. The larger the distance $d(\vec{x})$, the lower the probability that the block belongs to the skin colour class s . Such class has been experimentally derived using a large dataset of images acquired at different conditions and resolution using CMOS-VGA sensor on “STV6500 - E01” Evaluation Kit equipped with “502 VGA sensor”[10]. Due to the large quantity of colour spaces, distance measures, and two-dimensional distributions, many skin recognition algorithms can be used. The skin colour algorithm is independent from exposure correction, thus we introduce two different alternative techniques aimed to recognize skin regions (as shown in Figure 1.5):

1. By using the input YCbCr image and the conditional probability (1.7), each pixel is classified as belonging to a skin region or not. Then a new image with normalized grayscale values is derived, where skin areas are properly highlighted (Figure 1.5(c)). The higher the gray value the bigger the probability to compute a reliable identification.
2. By processing an input RGB image, a 2D chrominance distribution histogram (r, g) is computed, where $r=R/(R+G+B)$ and $g=G/(R+G+B)$. Chrominance values representing skin are clustered in a specific area of the (r,g) plane, called “skin locus” (Figure 1.6.c), as defined in [11]. Pixels having a chrominance value belonging to the skin locus will be selected to correct exposure.

For Bayer data, the skin recognition algorithm works on the RGB image created by sub-sampling the original picture, as described in Figure 1.3.

1.3.3 Exposure Correction

Once the visually relevant regions are identified, the exposure correction is carried out using the mean gray value of those regions as reference point. A simulated camera response curve is used for this purpose. This function can be expressed by using a simple parametric closed form representation:

$$f(q) = \frac{255}{(1 + e^{-(Aq)})^C} \quad (1.9)$$

where q represents the ‘light’ quantity and the final pixel value is obtained by means of the parameters A , and C used to control the shape of the curve. q is supposed to be expressed in 2-based logarithmic

unit (usually referred as “stops”). These parameters could be estimated, depending on the specific image acquisition device or chosen experimentally, as better specified below (See Section 1.4). The offset from the ideal exposure is computed using the f curve and the average gray level of visually relevant regions avg , as:

$$\Delta = f^{-1}(Trg) - f^{-1}(avg) \quad (1.10)$$

where Trg is the desired target gray level. Trg should be around 128 but its value could be slightly changed especially when dealing with Bayer Pattern data where some post processing is often applied.

The luminance value $Y(x,y)$ of a pixel (x,y) is modified as follows:

$$Y'(x,y) = f(f^{-1}(Y(x,y)) + \Delta) \quad (1.11)$$

Note that all pixels are corrected. Basically the previous step is implemented as a LUT (Lookup Table) transform.

1.3.4 Exposure Correction Results

The described technique has been tested using a large database of images acquired at different resolutions, with different acquisition devices, both in Bayer and RGB format. In the Bayer case the algorithm was inserted in a real-time framework, using a CMOS-VGA sensor on “STV6500 - EO1” Evaluation Kit equipped with “502 VGA sensor” [10]. Examples of skin detection by using real time processing are reported in Figure 1.7. In the RGB case the algorithm could be implemented as post-processing step. Examples of skin and contrast/focus exposure correction are respectively shown in Figure 1.8 and Figure 1.9. Results show how the features analysis capability of the proposed algorithm permits contrast enhancement taking into account some strong peculiarity of the input image. Major details and experiments can be found in [2].

1.4 Bracketing and Advanced Applications

In order to attempt to recover or enhance a badly exposed image, even if some kind of post-processing is possible, there are situations where this strategy is not possible or leads to poor results. The problem comes from the fact that badly captured data can be enhanced, but if no data exists at all there’s nothing to enhance. Today, despite the great advancements realized by digital photography, which has made available tremendous resolution even for mass market oriented products, almost all digital photo-cameras still deal with limited dynamic range and inadequate data representation, which make critical lighting situations, and the real world has tons of them, difficult to handle. This is where multiple exposure capture stands as a

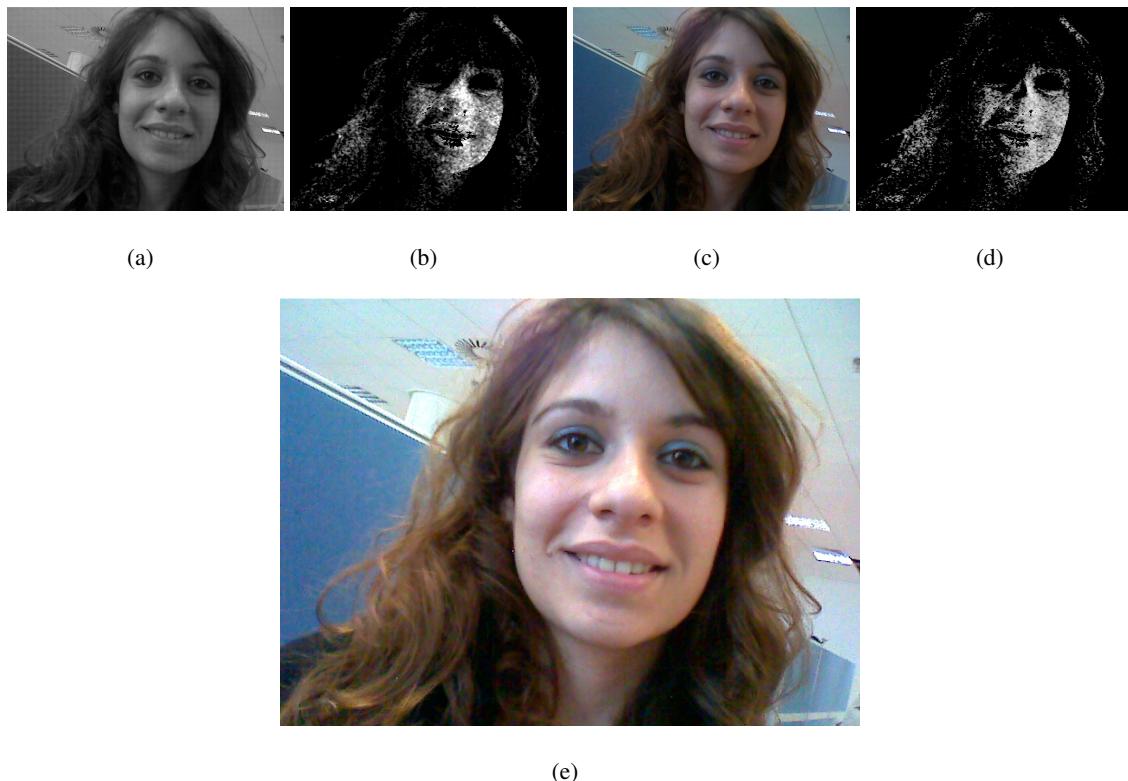


Figure 1.7: Exposure Correction results by real-time and post processing: (a) Original Bayer input image; (b) Bayer Skin detected in real-time; (c) Color Interpolated Image from Bayer Input; (d) RGB Skin detected in post processing; (e) Exposure Corrected Image obtained from RGB image.

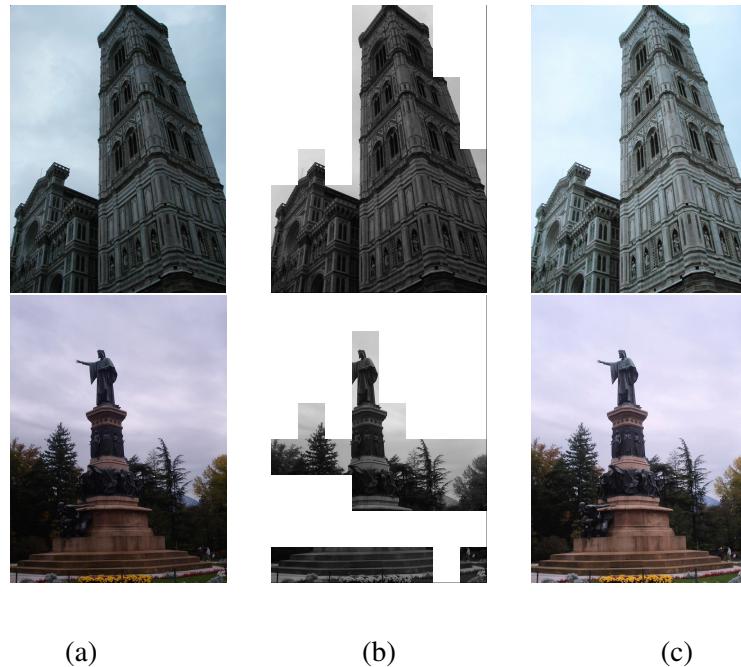


Figure 1.8: Exposure Correction results by post processing: (a) Original Color input image; (b) Contrast and focus visually significant blocks detected; (c) Exposure Corrected Image obtained from RGB image.

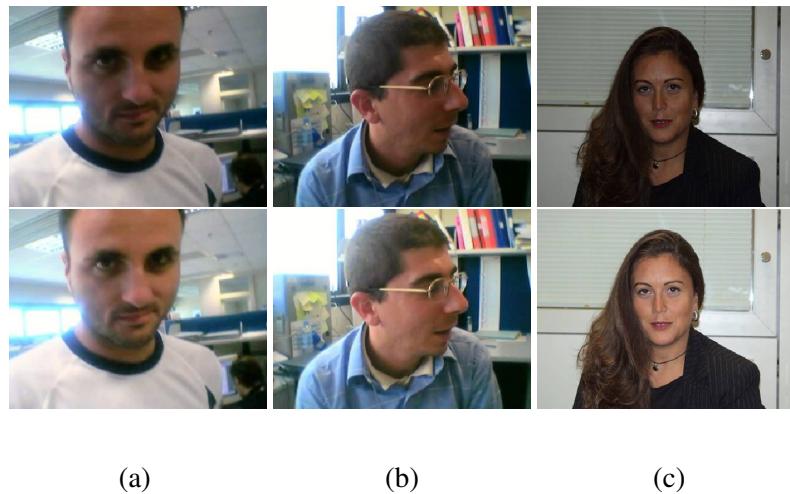


Figure 1.9: Exposure Correction results: in the first row the original images (a) and (b) acquired by Nokia 7650 VGA sensor compressed in Jpeg format, (c) picture acquired with CCD Sensor (4,1 Mega Pixels) Olympus E-10 Camera; in the second row the Corrected output.

useful alternative to overpass actual technology limits. Even if the idea of combining multiple exposed data is just recently receiving great attention, the methodology itself is very old. In the early sixties, well before the advent of digital image processing Charles Wyckoff [12] was able to capture high dynamic range images by using photographic emulsion layers of different sensitivity to light. The information coming from each layer was printed on paper using different colors, thus obtaining a pseudo-color image depiction.

1.4.1 The Sensor Versus The World

Table 1.1: Typical world luminance levels.

Scene	Illumination
<i>Starlight</i>	$10^{-3} cd/m^2$
<i>Moonlight</i>	$10^{-1} cd/m^2$
<i>Indoorlight</i>	$10^2 cd/m^2$
<i>Sunlight</i>	$10^5 cd/m^2$

Dynamic range refers to the ratio of the highest and lowest sensed level of light. For example, a scene where the quantity of light ranges from $1000 cd/m^2$ to $0.01 cd/m^2$, has a dynamic range of $1000/0.01=100,000$. The simultaneous presence in real world scenes poses great challenges on image capturing devices, where usually the available dynamic range is not capable to cope with that coming from the outside world. High dynamic range scenes are not uncommon; imagine a room with a sunlit window, environments presenting opaque and specular objects and so on. Table 1.1 shows typical luminance values for different scenes, spanning a very wide range from starlight to sunlight. On the other side dynamic range (*DR*) of a digital still camera (DSC) is defined as the ratio between the maximum charge that the sensor can collect (*full well capacity, FWC*), and the minimum charge that is just above sensor noise (*noise floor,NF*). This quantity is usually expressed in logarithmic units.

$$DR = \log_{10} \left[\frac{FWC}{NF} \right]. \quad (1.12)$$

This dynamic range, which is seldom in the same order of magnitude of those coming from real world scenes, is further affected by errors coming from analogue to digital conversion (*ADC*) of sensed light values. Once the light values are captured, they are properly quantized to produce digital codes, that usually for common 8-bit data fall in the $[0 : 255]$ range. This means that a sampled, coarse representation of the continuously varying light values is produced.

Limited dynamic range and quantization thus irremediably lead to loss of information and to inadequate data representation. This process is synthetically shown in Fig. 1.10, where the dynamic range of a scene

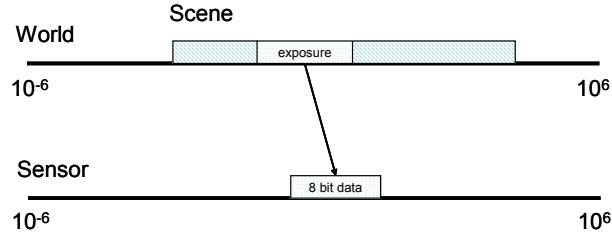


Figure 1.10: Due to limited camera dynamic range, only a portion, depending of exposure settings, of the scene can be captured and digitized.

is converted to the digital data of a *DSC*: only part of the original range is captured, the remaining part is lost. The portion of the dynamic range where the loss occurs depends to employed exposure settings. Low exposure settings, by preventing information loss due to saturation of highlights, allow to capture highlight values, but lower values will be easily overridden by sensor noise. On the other side, high exposures settings allow a good representation of low light values, but the higher portion of the scene will be saturated. Once again a graphical representation gives a good explanation of the different scenarios.

Fig. 1.10 shows a high exposure capture. Only the portion of the scene under the green area is sensed with a very fine quantization (for simplicity only 8 quantization levels, shown with dotted lines, are supposed), the other portion of the scene is lost due to saturation which happens at the luminance level corresponding to the end of the green area.

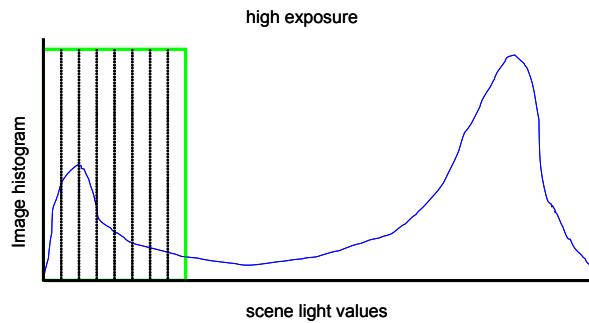


Figure 1.11: Information loss for high exposure. A limited dynamic range is captured due to saturation. The captured data is finely quantized.

Fig. 1.12 shows a low exposure capture. This time since saturation, which happens at the light level corresponding to the end of the red area, is less severe due to low exposure settings and apparently all the scene is captured (the red area). Unfortunately, due to very widely spanned sampling intervals, quality of captured data is damaged by quantization noise and errors.

To bring together data captured by different exposure settings allows to cover a wider range, and reveal

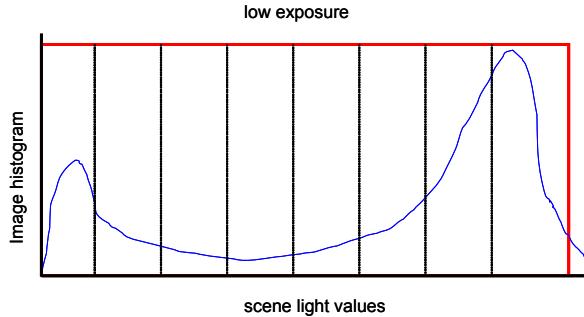


Figure 1.12: Information loss for low exposure.

more details than those that would have been possible by a single shot. The process is usually conveyed by different steps:

1. camera response function estimation;
2. high dynamic range construction;
3. tone mapping to display or print medium.

1.4.2 Camera Response Function

In order to properly compose a high dynamic range image, using information coming from multiple low dynamic range (*LDR*) images, the camera response function must be known. This function describes the way the camera reacts to changes in exposures, thus providing digital measurements.

Camera exposure X , which is the quantity of light accumulated by the sensor in a given time, can be defined as follows:

$$X = I \cdot t \quad (1.13)$$

where I is the irradiance and t the integration time.

When a pixel value Z is produced, it is known that it comes from some scene radiance I sensed for a given time t , mapped into the digital domain through some function f . Even if most CCD and CMOS sensors are designed to produce electric charges that are strictly proportional to the incoming amount of light (up to the near saturation point, where values are likely to fluctuate), the final mapping is seldom linear. Nonlinearities can come from the *ADC* stage, sensor noise, gamma mapping and specific processing introduced by the manufacturer. In fact often DSC camera have a built-in nonlinear mapping to mimic a film-like response, which usually produces more appealing images when viewed on low dynamic displays.

The full pipeline, from the scene to the final pixel values is shown in Fig. 1.13 where prominent nonlinearities can be introduced in the final, generally unknown, processing.

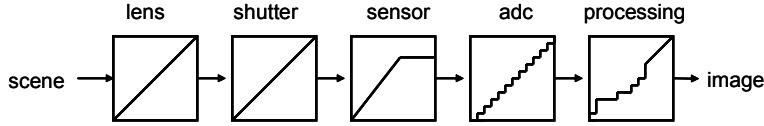


Figure 1.13: The full pipeline from scene to final digital image. The main problem behind assembling the high dynamic range from multiple exposures, lies in recovering the function synthesizing the full process.

The most obvious solution to estimate the camera response function, is to use a picture of uniformly lit different patches, such as the Macbet Chart [13] and establish the relationship between known light values and recorded digital pixel codes. However this process requires expensive and controlled environment and equipment. This is why several chartless techniques have been investigated. One of the most flexible algorithms have been described in [14], which only requires an estimation of exposure ratios between the input images. Of course exposure ratios are at hand given the exposure times, as produced by almost all photo-cameras. Given N digitized *LDR* pictures, representing the same scene and acquired with timings $t_j : j = 1, \dots, N$, exposure ratios $R_{j,j+1}$ can be easily described as

$$R_{j,j+1} = \frac{t_j}{t_{j+1}}. \quad (1.14)$$

Thus the following equation relates the i 'th pixel of the j 'th image, $Z_{i,j}$, to the underlying unknown radiance value I_i

$$Z_{i,j} = f(I_i \cdot t_j) \quad (1.15)$$

which is the aforementioned camera response function. The principle of high dynamic range compositing is the estimation for each pixel, of the radiance values behind it, in order to obtain a better and more faithful description of the scene that has originated the images. This means that we are interested in finding the inverse of Eq. 1.14: a mapping from pixel value to radiance value is needed!

$$g(Z_{i,j}) = f^{-1}(Z_{i,j}) = I_i \cdot t_j. \quad (1.16)$$

The nature of the function g is unknown, the only assumption is that it must be monotonically increasing. That's why a polynomial function of order K is supposed.

$$I \cdot e = g(Z) = \sum_{k=0}^K c_k \cdot Z^k. \quad (1.17)$$

The problem thus becomes the estimation of the order K and the coefficients c_k appearing in Eq. 1.17. If the ratios between successive image pairs $(j, j + 1)$ are known, the following relation holds:

$$\frac{I_i \cdot t_j}{I_i \cdot t_{j+1}} = \frac{g(Z_{i,j})}{g(Z_{i,j+1})} = R_{j,j+1} \quad (1.18)$$

Using Eq. 1.18, parameters are estimated by minimizing the following objective function:

$$O = \sum_{j=1}^N \sum_{i=1}^P \left[\sum_{k=0}^K c_k \cdot Z_{i,j}^k - R_{j,j+1} \cdot \sum_{k=0}^K c_k \cdot Z_{i,j+1}^k \right]^2 \quad (1.19)$$

where N is the number of images and P the number of pixels. The system can be easily solved by using the least square method. The condition $g(1) = 1$ is enforced to fix the scale of the solution, and different K orders are tested. The K value that better minimizes the system is retained.

To limit the number of equations to be considered not all pixels of the images should be used and some kind of selection is advised by respecting the following rules:

1. pixels should be well spatially distributed;
2. pixels should sample the input range;
3. pixels should be picked from low variance (homogenous) areas.

A different approach for feeding the linear system in (1.19) could be done by replacing pixel values correspondences by *comparagram* pairs. *Comparagrams* have been well described in [15] and provide an easy way to represent how pixels of one image are mapped to the same image with different exposure. This mapping is usually called brightness transfer function (*BTF*).

It's worth noting that if direct access to raw data is available, and known to be linear, the response curve estimation step could be avoided, since in this case the function equals a simple straight line normalized in the range $[0, \dots, 1]$. Fig. 1.14 shows 10 images captured at different exposure settings, from $\frac{1}{1600}$ sec to $\frac{1}{4}$ sec, while Figure 1.15 shows the recovered response curve on both linear (left) and logarithmic units.

1.4.3 High Dynamic Range Image Construction

Once the response function, estimated or at priori known, is at hand the high dynamic range image, usually referred as *radiance map* and composed of floating point values having greater range and tonal resolution than usual *low dynamic range (LDR)* data, can be assembled. The principle is that each pixel in each image, provides a more or less accurate estimation of the radiance value of the scene in the specific position. For example, very low pixel values coming from low exposure images are usually noisy, and thus not reliable, but the same pixels are likely to be well exposed in images acquired with higher exposure settings.

Given N images, with exposure ratios $e_i : i = 1 : N$ and considering Eq. 1.16 the sequence $\left[\frac{g(Z_{i,1})}{t_1}, \frac{g(Z_{i,2})}{t_2}, \dots, \frac{g(Z_{i,N})}{t_N} \right]$ of estimates for a pixel in position i is obtained. Different estimates should be assembled by means of a weighted average taking into account reliability of the pixel itself. Of course, the weight should completely discard pixels that appear as saturated and assign very low weight to pixels whose value is below some noise floor, since they are unable to provide decent estimation.

One possible weighting function could be a hat or gaussian shaped function centered around mid-gray pixel values, which are far from noise and saturation. As a thumb of rule, for each pixel there should be at least one image providing an useful pixel (e.g. that is not saturated, nor excessively noisy). Given the weighting function $w(Z)$ the radiance estimate for a given position i is given by:

$$I_i = \frac{\sum_{j=1}^N w(Z_{i,j}) \cdot \frac{g(Z_{i,j})}{t_j}}{\sum_{j=1}^N w(Z_{i,j})}. \quad (1.20)$$

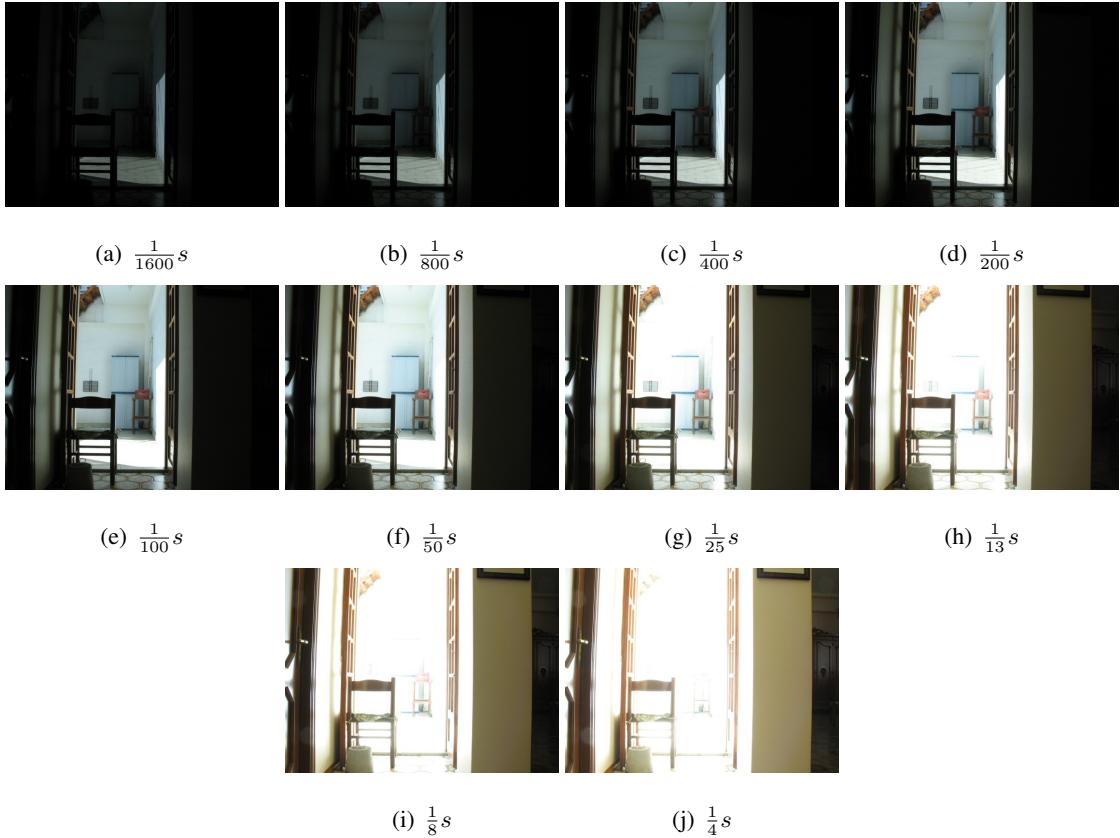


Figure 1.14: A sequence of 10 images, captured at iso 50, f-6.3, and exposures ranging from $\frac{1}{1600}$ to $\frac{1}{4}$ sec.

1.4.4 The Scene Versus the Display Medium

Once the high dynamic range image has been assembled, what's usually required is a final rendering on the display medium, such as a CRT display or a printer. The human eye is capable of seeing a huge range of luminance intensities, thanks to its capability to adapt to different values. Unfortunately this is not the way most image rendering systems work. Hence they are usually not capable to deal with the full dynamic range contained into images that provide an approximation of real world scenes. Indeed most CRT display have an useful dynamic range in the order of nearly 1:100. It's for sure that in the next future, high dynamic reproduction devices will be available, but for the moment they are well far from mass market consumers.

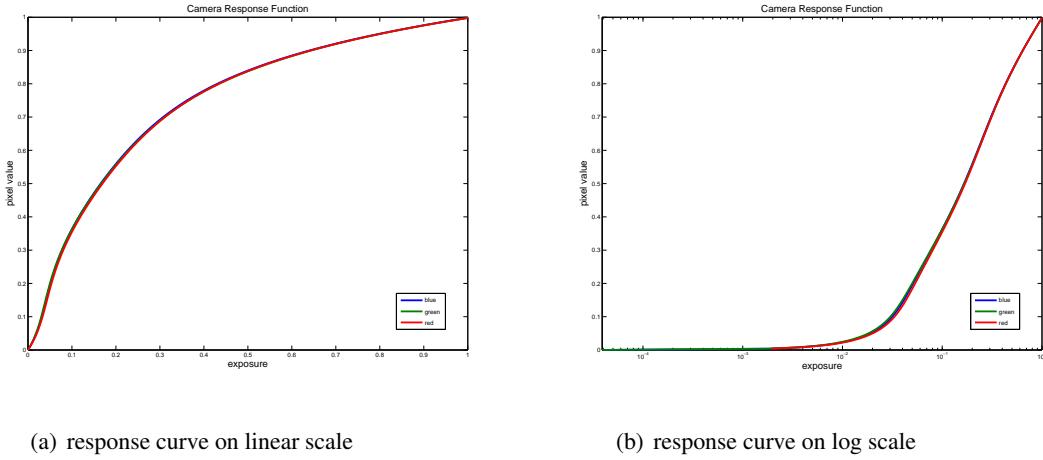


Figure 1.15: Response Curves derived from images depicted in Fig. 1.14

Simply stated, *tone mapping* is the problem of converting an image containing a large range of numbers, usually expressed in floating point precision, into a meaningful number of discrete gray levels (usually in the range 0, ..., 255), that can be used by any imaging device. So, we can formulate the topic as that of the following quantization problem:

$$Q(val) = \lfloor (N - 1) \cdot F(val) + 0.5 \rfloor \quad (1.21)$$

$$F : [L_{w_{min}} : L_{w_{max}}] \rightarrow [0 : 1]$$

where $[L_{w_{min}} : L_{w_{max}}]$ is the input range, N the number of allowed quantization levels, and F the tone mapping function. A simple linear scaling usually leads to the loss of a high amount of information on the reproduced image. Figure 1.16, shows the result obtained by linearly scaling an high dynamic range image, constructed from the sequence of Fig. 1.14 using the techniques described above. As it can be seen, only a portion of the scene is visible, so better alternatives for F are needed.

Two different categories of tone mapping exist:

1. Tone Reproduction Curve (TRC): the same function is applied for all pixels;
2. Tone Reproduction Operator (TRO): the function acts differently depending on the value of a specific pixel and its neighbors.

In what follows, several of such techniques will be briefly described and applied on the input HDR image, assembled from the sequence in Figure 1.14. The recorded input was in the range 0.00011 : 32.

Histogram Adjustment (TRF)

The algorithm described in [16], by G. Ward et al., is based on ideas coming from image enhancement techniques, specifically histogram equalization. While histogram equalization is usually employed to ex-



Figure 1.16: An HDR image built from the sequences of Figure 1.14, linearly scaled in the $[0, \dots, 1]$ range and quantized to 8 bits.

pand contrast images, in this case it is adapted to map the high dynamic range of the input image within that of the display medium, while preserving the sensation of contrast. The process starts by computing a downsampled version of the image, with a resolution that equals to 1 degree of visual angle. Luminance values of this, so called *fovea* image, are then converted in the *brightness* domain, which can be approximated by computing logarithmic values. For the logarithmically valued image, an histogram is built, where values between minimum and maximum bounds $L_{w_{min}}$ and $L_{w_{max}}$ (of the input radiance map) are equally distributed on the logarithmic scale. Usually employing around 100 histogram bins each having a size of $\Delta b = \frac{\log(L_{w_{max}}) - \log(L_{w_{min}})}{100}$ provides sufficient resolution. The cumulative distribution function, normalized by the total number of pixels T , is defined as:

$$\begin{aligned} P(b) &= \sum_{b_i < b} f(b_i)/T \\ T &= \sum_{b_i} f(b_i) \end{aligned} \quad (1.22)$$

where $f(b_i)$ is the frequency count for bin i . The derivative of this function can be expressed as

$$\frac{\partial P(b)}{\partial b} = \frac{f(b)}{T\Delta b}. \quad (1.23)$$

Applying a histogram equalization on the input, the result is an image where all brightness values have equal probability. The equalization formula, which provides a way to map luminance values to display values, can be expressed as:

$$\log(L_d(x, y)) = \log(L_{d_{min}}) + (\log(L_{d_{max}}) - \log(L_{d_{min}})) \cdot P(\log \cdot L_w(x, y)) \quad (1.24)$$

where $L_{d_{min}}$ and $L_{d_{max}}$ stay for minimum and maximum display values. This means that the equalized brightness is fit into the available display dynamic range. Unfortunately naive equalization, tends to over-exaggerate contrast in correspondence of highly populated bins (histogram peaks) leading to undesirable

effects. To prevent this, a ceiling procedure is applied on the histogram, imposing that contrast should never exceed those obtained by a linear mapping. The ceiling can be written in terms of the derivative of the mapping (which is indicative of contrast):

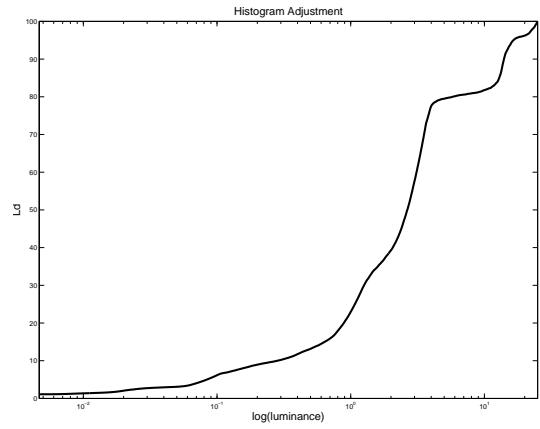
$$\frac{\partial L_d}{\partial L_w} \leq \frac{L_d}{L_w}. \quad (1.25)$$

By putting together Eq. 1.23 and Eq. 1.24 the final histogram constraint is obtained:

$$f(b) \leq \frac{T\Delta b}{(\log(L_{d_{max}}) - \log(L_{d_{in}}))}. \quad (1.26)$$



(a) histogram adjustment



(b) histogram adjustment mapping plot

Figure 1.17: Histogram Adjustment mapping.

Thus in order to prevent excessive contrast, histogram values are repetitively cut to satisfy (1.26). The operator has been further refined by the authors to include more sophisticated ceiling procedure, simulation of color and contrast sensitivity, according to some features of the human visual system (*HVS*). Figure 1.17 (a) shows the example radiance map, tonemapped to display using Ward's operator in its basic implementation. Figure 1.17 (b) plots the resultant mapping.

Chiu's local operator (TRO)

One of the first works performing spatially varying local processing, for visualization of high dynamic range image data, was described in [17] whose importance is mainly due to historical reasons. The idea is to apply at each pixel position a specific scaling, $s(x, y)$. The output image, to be rendered on the medium, is then obtained by multiplying the input by the scaling function:

$$L_d(x, y) = s(x, y) \cdot L_w(x, y). \quad (1.27)$$

The scaling function $s(x, y)$ is defined as

$$s(x, y) = \frac{1}{k \cdot L_{blur}(x, y)} \cdot L_w(x, y) \quad (1.28)$$

where $L_{blur}(x, y)$ is a gaussian filtered version of $L_w(x, y)$ and k a user parameter. A value of $k = 2$ brings values near the local average (given by the low pass filtered data) to a value of 0.5. This doesn't leave too much room for the mapping of brighter pixels, hence a value of $k = 8$ is suggested. The main problem with this technique, and which is main concern of all local operators, is the appearance of so called *halo artifacts* which easily manifest itself around areas of relevant brightness transition. This is due to the fact that the local average for pixels around transition zones, carries unwanted "information" which is not related to the luminance value of the specific pixel. For example, for a bright pixel near a very dark region, the local average $L_{blur}(x, y)$ will be very low due to the influence of dark pixels, leading to a poor scaling of the bright pixel and thus to the appearance of a *bright halo*. On the other side, a *dark halo* is likely to appear for dark pixels near bright regions, where the scaling will be excessive. Fig. 1.18 shows the result of Chiu's algorithm (with $k = 8$ and a gaussian filtering with a pixel width of 15) where some of the halo artifacts are highlighted by red squares.



Figure 1.18: Image mapped with Chiu's algorithm. Some halo artifacts are highlighted.

Bilateral Filtering (TRO)

Durand et al. in [18] consider the input image as separable into two different layers: a *base layer* and a *detail layer*. The first is related to the low frequency content of the image and the second to the high frequency content. Thus an input image $L_w(x, y)$ can be expressed as the multiplication of its two layers: $L_w(x, y) = Base(x, y) \cdot Detail(x, y)$. In order to properly scale the high dynamic range data, the base layer is feed to a compressive function, while the detail layer is leaved unchanged. This helps the preservation of subtle local contrast, and is also related to the concept that the base layer represents the influence of lightning

conditions (and thus the scene dynamic range). All the processing is done in the logarithmic domain, where the two layers are separated, processed and recombined. The basic steps of the algorithm are:

1. express the data in the logarithmic domain $l(x, y) = \log(L_w(x, y))$
2. compute the base layer $Base(x, y)$
3. compute the detail layer $Detail(x, y) = Base(x, y) - l(x, y)$
4. compress the base layer obtaining $comp(Base(x, y))$
5. recombine the layers and exponentiate the result, to produce the final image,

$$L_d(x, y) = \exp(comp(Base(x, y)) + Detail(x, y)).$$

The compression of the base layer is simply done by scaling it by a multiplicative factor m , such that its range equals a desired contrast c :

$$comp(Base(x, y)) = m \cdot Base(x, y), \quad (1.29)$$

such that

$$m \cdot (\max(Base(x, y)) - \min(Base(x, y))) = c. \quad (1.30)$$

The most relevant feature of the algorithm is the way in which the base layer is computed, which should be a low pass filtered version of the image, but without the unwanted issues of the gaussian filtering employed by the aforementioned Chiu's algorithm. In other words the low pass filtering process should not consider for each pixel, those luminance values that are far from the luminance of the pixel itself. To achieve this, a *bilateral filter* is considered. Bilateral filtering $bil(x, y)$ is obtained by adding to usual gaussian filter with a spatial kernel g , a further gaussian weighting function w , whose weights decrease as the difference in luminance value between the central pixel and its neighbors in a surround Ω increase.

$$bil(x, y) = \frac{1}{k(x, y)} \sum_{(u, v) \in \Omega} g(x - u, y - v) \cdot l(u, v) \cdot w(d(l(u, v), l(x, y))), d(u, v) = |l(u, v) - l(x, y)| \quad (1.31)$$

where $k(x, y)$ is the normalization term. Since bilateral filtering in the spatial domain can be computationally very slow, the authors have developed a very fast approximation in the frequency domain. Fig. 1.19, shows the result of the algorithm ($c = \log(50)$) on the input image.

Photographic Tone Reproduction

Reinhard et al. [19] have developed an operator based on some simple photographic principles such as *automatic exposure* and *dodge and burning*, where the first provides a global mapping of the image and the



Figure 1.19: Image mapped with bilateral filtering.

latter exploits some local features. The global part of the operator analyzes the concept of *scene's key*, which is measure of how overall dark or bright the images is. This quantity is approximated with the log average value \bar{L}_w of the image luminance values. According to photographic principles, where the key is usually printed (or displayed) to have the 18% reflectance of the medium, an initial global mapping is performed using the following equation:

$$L_m(x, y) = \frac{0.18}{\bar{L}_w} \cdot L_w(x, y). \quad (1.32)$$

In this way a kind of automatic exposure setting is provided for the scene, even it is done *ex post facto*, since the scene has already been captured by the camera (but, since radiance maps provide a floating point description of the initial scene, this allow us to do such virtualizations of the photographic process). Even if in Eq. 1.32, the scene's key value is linearly mapped to the value of 0.18, different values could be used depending on the specific image content (eg. a nightlife picture should be scaled to a very low value). No matter what the dynamic range of the initial scene is, the luminance values exposed by means of Eq. 1.32 are forced to fit inside the medium dynamic range (which is here supposed to vary within [0, .., 1]) using a compressive function, which is particularly effective on very high luminance values:

$$L_d(x, y) = \frac{L_m(x, y)}{1 + L_m(x, y)}. \quad (1.33)$$

This function scales input values differently, according to their magnitude: small values, usually $\ll 1$ are almost leaved unchanged, while very high values , usually $\gg 1$, are scaled by a very large amount (the quantity $\frac{1}{L_m(x, y)}$).

The scaling function, is further refined to include some local processing, similar to *dodge and burning* procedures, where a dark value in a bright surround is heavily compressed (burned), and a bright pixel on a dark surround is only mildly compressed (dodged). To exploit these local properties, filtered versions at

different scales $s = 1, \dots, S$ of L_m are produced as

$$L_{blur_s} = L_m \circ R_s \quad (1.34)$$

and in Eq. 1.33 the quantity $L_m(x, y)$ on the denominator, is replaced by

$$L_d(x, y) = \frac{L_m(x, y)}{1 + L_{blur_s}(x, y)} \quad (1.35)$$

where \circ equals to the convolution operator, and R_s are different gaussian kernels, having different pixel widths $w(s)$, for $s = 1..S$

$$w(s) = e^{\log(w_{min}) + \frac{s}{S} \cdot (\log(w_{max}) - \log(w_{min}))}. \quad (1.36)$$

w_{max} and w_{min} are the maximum and minimum allowed pixel widths, and are fixed respectively to 1 and 43. Thus the smallest scale for a pixel in position (x, y) equals to the pixel itself. To avoid halo artifacts, for each pixel the largest scale $s_{max} : |V_{s_{max}}(x, y)| < \epsilon$, where V_s is the difference between two successive scales, computed as:

$$V_s(x, y) = \frac{L_{blur_s} - L_{blur_{s+1}}}{2^\Phi \cdot 0.18/s^2 + L_{blur_s}}. \quad (1.37)$$

According to the authors a value of $\Phi = 8$ is used. Practically Eq. 1.37 is the search of the largest surround across a pixel in position (x, y) whose value is reasonably similar to that of the pixel. This avoids the appearance of severe halo artifacts, similar to those seen by the application of Chiu's algorithm. Fig. 1.20 shows the result of the algorithm of Reinhard et. al on our example radiance map, where the parameters $S = 8, \Phi = 8$ have been used.



Figure 1.20: Photographic Tone Reproduction mapping.

Gradient Compression (TRO)

The last technique, belonging to the family of TRO, that we are going to describe was developed by Fattal et al. [20], and it's far more sophisticated than those that have been described hereof. Even if sometimes output

images can have an unnatural appearance, in most cases results can look very appealing. This algorithm doesn't operate directly on the spatial domain, but instead computes the *gradient field* of the input image and after manipulating it, reconstructs by means of Poisson integration the image having the new gradients. This derives from the observation that an image exhibiting an high dynamic range, will be characterized by gradients of large magnitude around zones of brightness transition. Hence attenuating those gradients seems like a viable way for building a LDR depiction of the scene, suitable to be viewed on a common display. Similarly to the pipeline of the algorithm based on bilateral filtering, gradient compression works on logarithmic data, and so just before producing the output image, the result undergoes exponentiation. Indicating with $l(x, y)$ the data in the logarithmic domain, the gradient field $\nabla l(x, y)$ is computed as follows:

$$\nabla l(x, y) = (l(x + 1, y) - l(x, y), l(x, y + 1) - l(x, y)) \quad (1.38)$$

Attenuation of the gradient field is obtained by multiplication with an proper scaling function $\Phi(x, y)$:

$$G(x, y) = \nabla l(x, y) \cdot \Phi(x, y). \quad (1.39)$$

The attenuated gradient field $G(x, y)$ is then inverted by solving the Poisson equation

$$\nabla^2 \tilde{l}(x, y) = \operatorname{div} G(x, y). \quad (1.40)$$

Since edges (and thus gradients) exist at multiple resolution levels, a gaussian pyramid representation $< l_0, l_1, \dots, l_S >$ is constructed, and for each level the gradient field is computed. The attenuation function is then computed on each level and reported to the upper level in *bottom to top* fashion. The attenuation function at the top level is the one that will effectively used in (1.39). Attenuation function at each level s is computed as follows:

$$\Psi_s(x, y) = \frac{\alpha}{\|\nabla l_s(x, y)\|} \cdot \left(\frac{\|\nabla l_s(x, y)\|}{\alpha} \right)^\beta. \quad (1.41)$$

The α parameter in 1.41 determines which gradient magnitudes are leaved untouched, while the β exponent amplifies magnitudes greater than α . Suggested values are $\alpha = 0.1 \cdot (\text{average gradient magnitude})$ and $\beta = 0.9$. Since the attenuation function is computed for each resolution level s , the propagation to full resolution is done by scaling the attenuation function from level $s - 1$ to s , and accumulating the values to obtain the full resolution attenuation function $\Phi(x, y)$ that will be effectively used (authors claim that by using the attenuation function just at full resolution halo artifacts are mostly invisible). This can be expressed by the following equations:

$$\Phi_d(x, y) = \Psi_d(x, y) \quad (1.42)$$

$$\Phi_k(x, y) = L(\Phi_{k+1})(x, y) \cdot \Psi_d(x, y)$$

$$\Phi(x, y) = \Phi_0(x, y)$$



Figure 1.21: Gradient Compression mapping.

where the d is the smallest resolution level and L is the bilinear up-sampling operator. Fig. 1.21 shows the result of applying the gradient compression operator on our sample hdr image. The operator looks computationally more complicated than others that have been described but as it can be seen the mapped image looks far more impressive, in terms of high-light and low-light visibility, than the previous renderings.

1.5 Conclusion

The problem of the proper exposure settings for image acquisition is of course strictly related with the dynamic range of the real scene. In many cases some useful insights can be achieved by implementing ad-hoc metering strategies. Alternatively it is possible to apply some tone correction methods that enhance the overall contrast of the most salient regions of the picture. The limited dynamic range of the imaging sensors doesn't allow to recover the dynamic of the real world; in that case only by using "bracketing" and acquiring several pictures of the same scene with different exposure timing a final good rendering can be realized.

In this work we have presented a brief review of automatic digital exposure correction methods trying to report the specific peculiarities of each solution. Just for completeness we report that recently, Raskar et al. [21] have proposed a novel strategy devoted to "flutter" the camera's shutter open and closed during the chosen exposure time with a binary pseudo-random sequence. In this way high-frequency spatial details can be recovered especially when movements with constant speed are present. In particular a robust deconvolution process is achieved just considering the so-called coded-exposure that makes the problem well-posed. We think that the Raskar's technique could be used also in multi picture acquisition just to limit the overall number of images needed to reconstruct a reliable HDR map.

Bibliography

- [1] M. Mancuso and S. Battiato, “An introduction to the digital still camera technology,” *ST Journal of System Research*, vol. 2, no. 2, pp. 1–9, 2001.
- [2] S. Battiato, A. Bosco, A. Castorina, and G. Messina, “Automatic image enhancement by content dependent exposure correction,” *EURASIP Journal on Applied Signal Processing*, vol. 12, pp. 1849–1860, 2004.
- [3] S. Battiato, A. Castorina, and M. Mancuso, “High dynamic range imaging for digital still camera: an overview,” *Journal of Electronic Imaging*, vol. 12, pp. 459–469, July 2003.
- [4] R. Gonzalez and R. Woods, “Digital image processing 2nd edition,” *Prentice Hall*.
- [5] S. Bhukhanwala and T. Ramabadran, “Automated global enhancement of digitized photographs,” *IEEE Transactions on Consumer Electronics*, vol. 40, no. 1, 1994.
- [6] B. Bayer, “Color imaging array,” *U.S. Patent 3,971,065*, 1976.
- [7] S. Phung, A. Bouzerdoum, and D. Chai, “A novel skin color model in ycbcr color space and its application to human face detection,” vol. 1, pp. 289–292, 2002.
- [8] B. Zarit, B. Super, and F. Quek, “Comparison of five colour models in skin pixel classification,” *Proc. Of Int. Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, IEEE Computer Society*, pp. 58–63, 1999.
- [9] J. Yang, W. Lu, and A. Waibel, “Skin-colour modeling and adaptation,” *Technical Report CMU-CS-97-146 School of Computer Science, Carnegie Mellon University*, 1997.
- [10] STMicroelectronics, “Colour sensor evaluation kit vv6501.” Edinburgh - www.edb.st.com/products/image/sensors/501/6501evk.htm.
- [11] M. Soriano, B. Martinkauppi, and M. L. S. Huovinen, “Skin color modeling under varying illumination conditions using the skin locus for selecting training pixels,” *RISA2000 - Real-time Image Sequence Analysis*, 2000.
- [12] C. Wyckoff, “An experimental extended response film,” *SPIE Newsletter*, July 1962.
- [13] Y.-C. Chang and J. F. Reid, “Rgb calibration for analysis in machine vision,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 1414–1422, October 1996.
- [14] T. Mitsunaga and S. Nayar, “Radiometric self calibration,” *Computer Vision and Pattern Recognition*, vol. 2, pp. 374–380, 1997.

- [15] S. Mann, “Comparametric equations with practical applications in quantigraphic image processing,” *IEEE Trans. Image Proc.*, vol. 9, pp. 1389–1406, August 2000.
- [16] G. W. Larso, H. Rushmeier, and C. Piatko, “A visibility matching tone reproduction operator for high dynamic range scenes,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 3, pp. 291–306, 1997.
- [17] B. Chiu, M. Herf, P. Shirley, S. Swamy, C. Wang, and K. Zimmerman, “Spatially nonuniform scaling functions for high contrast images,” *Proceedings of Graphics Interface 93*, pp. 245–253, 1993.
- [18] F. Durand and J. Dorsey, “Fast bilateral filtering for the display of high dynamic range images,” *Proceedings of SIGGRAPH 2002*, pp. 257–266, 2002.
- [19] E. Reinhar, M. Stark, P. Shirley, and J. Ferweda, “Photographic tone reproduction for digital images,” *Proceedings of SIGGRAPH 2002*, pp. 267–276, 2002.
- [20] R. Fattal, D. Lischinski, and M. Werman, “Gradient domain high dynamic range compression,” *Proceedings of SIGGRAPH 2002*, pp. 249–256, 2002.
- [21] R. Raskar, A. Agrawal, and J. Tumblin, “Coded exposure photography:motion deblurring using flattered shutter,” in *SIGGRAPH - International Conference on Computer Graphics and Interactive Techniques*, (Boston), July-August 2006.

Index

Average Metering, 6
Bilateral Filtering, 25
Bracketing, 13
Camera Response Function, 18
Center-weighted Average Metering, 6
Chiu's local operator, 24
Contrast and Focus, 9
Dynamic range, 16
Exposure Correction Content Dependent, 8
Gradient Compression, 28
Histogram Adjustment, 22
Matrix or Multi-zone Metering, 6
Partial Area Metering, 4
Photographic Tone Reproduction, 26
Skin Recognition, 10
Spot Metering, 4
Zone System, 4