

THE FUNDAMENTAL BASIS OF HDR: COMPARAMETRIC EQUATIONS

S. Mann^{*,†,‡}, M.A. Ali^{*}

*University of Toronto, Toronto, ON, Canada**

Rotman School of Management CDL, Toronto, ON, Canada[†]

Meta, Redwood City, CA, United States[‡]

CHAPTER OUTLINE

1.1 Introduction to High Dynamic Range Imaging	2
1.1.1 The Fundamental Concept of HDR Sensing and Metasensing.....	2
1.1.2 The Fundamental Principle of HDR: Dynamic Range and Dynamage Range.....	4
1.1.3 HDR Imaging Techniques	5
1.1.4 HDR From Multiple Exposures	5
1.2 Historical Motivation for HDR Imaging	5
1.3 Theory of HDR Imaging	6
1.3.1 The Wyckoff Principle and the Range of Light	7
1.3.2 What's Good for the Domain Is Good for the Range.....	8
1.3.3 Extending Dynamic Range and Improvement of Range Resolution by Combining Differently Exposed Pictures of the Same Subject Matter	8
1.3.4 The Photoquantity, q	9
1.3.5 The Camera as an Array of Light Meters	9
1.3.6 The Accidentally Discovered Componder	10
1.3.7 Why Stockham Was Wrong.....	12
1.3.8 The Value of Doing the Exact Opposite of What Stockham Advocated	13
1.3.9 Using Differently Exposed Pictures of the Same Subject Matter to Get a Better Estimate of q	14
1.3.10 Exposure Interpolation and Extrapolation.....	18
1.4 Comparametric Image Processing: Comparing Differently Exposed Images of the Same Subject Matter	18
1.4.1 Misconceptions About Gamma Correction	18
1.4.2 Comparametric Plots and Comparametric Equations	20
1.4.3 Zeta Correction of Images.....	21
1.4.4 The Affine Comparametric Equation and Affine Correction of Images.....	22
1.4.5 The Preferred Correction of Images.....	24
1.4.6 Some Solutions to Some Comparametric Equations That Are Particularly Illustrative or Useful	25
1.4.7 Properties of Comparametric Equations	25

1.5 Practical Implementations	27
1.5.1 Comparing Two Images That Differ Only in Exposure	27
1.5.2 Joint Histograms and Comparagrams.....	27
1.5.3 Comparametric Regression and the Joint Histogram.....	28
1.5.4 Comparametric Regression to a Straight Line.....	29
1.5.5 Comparametric Regression to the Preferred Model.....	31
1.6 Tone Mapping in HDR Systems	34
1.6.1 An Extreme Example With Spatiotonal Processing of Photoquantities	36
1.7 Analytical Solution of Comparametric Equations	38
1.7.1 Overview	38
1.7.2 Formal Solution by Scaling Operator	38
1.7.3 Solution by Ordinary Differential Equation	39
1.8 Compositing as Bayesian Joint Estimation	40
1.8.1 Example Joint Estimator	44
1.8.2 Discussion Regarding Compositing via the CCRF.....	46
1.8.3 Review of Analytical Comparametric Equations	47
1.9 Efficient Implementation of HDR Reconstruction	48
1.9.1 Implementation.....	50
1.9.2 Compression Performance	53
1.9.3 Conclusion.....	56
Acknowledgments	57
References	57

1.1 INTRODUCTION TO HIGH DYNAMIC RANGE IMAGING

High dynamic range (HDR) imaging originated in the 1970s through digital eyeglass (DEG) and wearable computing as a seeing aid and, more generally, wearable technology as a sensory aid, which includes also HDR metasensing (the sensing of sensors and the sensing of their capacity to sense). HDR video as a way of seeing included other senses as well, such as HDR radar for the blind (Mann, 2001), and as a way of seeing radio waves with augmented reality overlays (Mann, 1992, 2001) (<http://wearcam.org/swim/>). Wearable computing gives rise to a rich sensory landscape that includes video, audio, and radar, plus physiological signals such as electrocardiogram and electroencephalogram, all of which made use of HDR signal capture and processing (Mann, 2001). This work is part of the general field of research and practice known as “sousveillance” (undersight), defined as “wearing and implanting various sensors, effectors, and multimedia computation in order to redefine personal space and modify sensory perception computationally” (Mann, 2004; Shakhakarmi, 2014).

1.1.1 THE FUNDAMENTAL CONCEPT OF HDR SENSING AND METASENSING

We begin by introducing the fundamental concept of HDR sensing, which allows the *dynamic range* of a sensor to be increased toward its “*dynamage range*” — that is, for a sensor to capture the full



FIGURE 1.1

Left: World's earliest known photograph, taken in 1826 on a plate coated in bitumen (petroleum tar, the asphalt commonly used as roofing material). Image from Wikimedia Commons. Center: A typical rooftop with modified bitumen shingles. Bitumen is commonly covered in stones or granules to protect it from damage by sunlight. Nevertheless, the south-facing side of the roof shows extensive damage due to exposure to sunlight. Right: Closeup showing sunlight-damaged southern exposure.

range of signals all the way to its limit, as defined as the limit to what it can sense without permanent damage to it.

Photography has a long and interesting history. For many years, it has been known that objects exposed to bright sunlight fade. Of particular note is the fact that bitumen (petroleum tar, asphalt, pitch, or the like), commonly used on rooftops, is easily damaged and becomes hard, brittle, and fragile with prolonged exposure to light (Fig. 1.1, center and right). The world's first known photograph was captured, in 1826, on a plate coated in bitumen (Fig. 1.1, left). Subsequently, various improvements to photography were made to make materials more sensitive to light, but the dynamic range of photographs was typically less than that of human vision. Similarly, the invention of motion pictures and of television gave rise to video capture, but also with similar problems regarding dynamic range.

Early video cameras (Fig. 1.2, left) used image pickup tubes that could be easily damaged by overexposure. As a result, the user must be careful with the f-stop on the camera lens, not to open it too far, and many cameras used to have an iris control button to protect the camera from damage:

Iris control button: A feature that closes down the iris or aperture of the lens to protect the sensitive video camera tube when the camera is not in operation. Camera tubes that are exposed to overly bright light or sun develop a 'burn' that may become permanent" (Jack and Tsatsulin, 2002).

Also, "some lenses even have a 'C' setting after the highest f-stop which means the lens is completely closed, letting no light through at all" (Inman and Smith, 2009).

Because early video cameras were easily damaged by excessive light exposure, the user had to be careful with the f-stop on the camera lens, and open it up only enough to get a proper exposure.

HDR video was made possible by the invention of sensors (eg, camera image sensors) that can be overexposed without permanent damage. Unlike old vidicon cameras, for which there was less difference between the dynamic range and dynamage range, modern cameras have a much greater dynamage range than their dynamic range. This allows them to produce images that are massively



FIGURE 1.2

Left: Early video cameras (television cameras), such as this one at the 1936 Summer Olympics, used camera tubes that could be permanently damaged by overexposure. Image from Wikimedia Commons. Center and right: Early television camera lenses typically had an iris with a “C” setting, which means that the iris is completely closed. This was to protect the sensor from being damaged by light. When the camera was not in use, or was passing into a region of bright light, the lens could be closed to protect the sensor from damage. Generally, the camera operator had to be very careful to open up the iris only enough for a proper exposure, but not so far as to permanently damage the camera sensor.

overexposed, thus making it possible to capture images at an extreme range of exposures, and it is these extreme exposures that allow us to see extreme shadow detail (due to the massively overexposed images) and extreme highlight detail (due to having one or more massively underexposed images as well).

HDR allows us to increase the dynamic range, ideally, all the way up to being equal to the dynamage range.

1.1.2 THE FUNDAMENTAL PRINCIPLE OF HDR: DYNAMIC RANGE AND DYNAMAGE RANGE

HDR works whenever the *dynamage range* of a sensor exceeds its *dynamic range*.

Dynamic range and dynamage range are defined as follows:

Dynamic range is the ratio between the largest and smallest nonnegative quantity, such as magnitude, amplitude, energy, or the like, of sound, light, or the like, for which a small incremental difference in the quantity can still be sensed (ie, the range over which changes in the quantity remain discernible) (Mann et al., 2011, 2012).

Dynamage range is the ratio between the largest quantity that will not damage a sensor or device or receiver, and the smallest nonnegative quantity for which changes in the quantity remain discernible (Mann et al., 2011, 2012).

1.1.3 HDR IMAGING TECHNIQUES

HDR imaging is the set of techniques that computationally extend the usual or standard dynamic range of a signal. HDR imaging has arisen in multiple fields, such as computational photography, computer graphics, and animation. HDR signals may be produced in several ways: by the combining of multiple lower dynamic range signals for HDR reconstruction; synthetically, by simulation or raytracing; or by use of HDR sensors for data acquisition.

1.1.4 HDR FROM MULTIPLE EXPOSURES

According to [Robertson et al. \(2003\)](#), “the first report of digitally combining multiple pictures of the same scene to improve dynamic range appears to be ([Mann, 1993](#)).”

HDR imaging by reconstruction from multiple exposures is defined as follows:

Definition of HDR reconstruction: The estimation of at least one photoquantity from a plurality of differently exposed images of the same scene or subject matter ([Mann, 1993, 2000, 2001](#); [Mann and Picard, 1995a](#); [Ali and Mann, 2012](#); [Robertson et al., 2003](#); [Reinhard et al., 2005](#)).

Specifically, HDR reconstruction returns an estimate of a photoquantity (or sequence of estimates in the case of video), $q(x, y)$ (any possibly spatially or temporally varying q or $q(x)$, $q(x, y, z)$, $q(x, y, t)$, or $q(x, y, z, t)$), on the basis of a plurality of exposures $f_i = f(k_i q(x, y))$, at exposure settings k_i , where there is also noise in each of these exposures f_i , through a camera response function, f , which is often unknown (although it may also be known, or it may be linear, or it may be the identity map). The exposure settings k_i may also be unknown.

A separate optional step of tone mapping the photoquantigraph, q , may be taken, if desired — for example, to produce an output image that can be printed or displayed on low dynamic range (LDR) output media. In situations where there is no need for a human-viewable HDR image (eg, HDR-based input to a computer vision system such as the wearable face-recognizer [Mann, 1996b](#)), the photoquantigraph may have direct use without the need to convert it to an LDR image.

A typical approach to generate q from f_i is to transform each of the input images f_i to estimates of that photoquantity, and then to combine the results with use of a weighted sum ([Mann, 1993, 2000, 2001](#); [Mann and Picard, 1995a](#); [Debevec and Malik, 1997](#); [Robertson et al., 2003](#)). Other approaches are probabilistic in nature, and typically use nonlinear optimization ([Ali and Mann, 2012](#); [Pal et al., 2004](#)).

1.2 HISTORICAL MOTIVATION FOR HDR IMAGING

HDR reconstruction from multiple exposures originated with author S. Mann (described as “the father of the wearable computer,” IEEE International Solid-State Circuits Conference, February 7, 2000), through a childhood fascination with sensing and metasensing that led to the invention of the DEG ([Fig. 1.3](#)). This includes the use of wearable sensors to process and mediate daily life, from wearable technologies as a photographic art form ([Mann, 1985](#); [Ryals, 1995](#)), to gesture-based augmented/augmented reality (AR) ([Mann, 1997b](#)) for the capture, enhancement, and rendering of



FIGURE 1.3

Author S. Mann with DEG and a DEG-based HDR welding helmet. Left: Original DEG, in comparison with a more recent commercial product. Right: Early DEG-based welding helmet prototype. The EyeTap principle was used to capture eyeward-bound light and process it with HDR video overlays, along with augmented reality: augmented in dark areas with 3D computer-generated overlays that adapt to appear where they are least distracting, while the vision is deliberately diminished in bright areas of the electric welding arc.

everyday experience, a generalized form of self-sensing known as “sousveillance” (“undersight”), in contrast to the more familiar and common practice of “surveillance” (“oversight”). Sousveillance, as a field of research, has recently expanded greatly, and been given a variety of new names — for example, “lifelogging,” “quantified self,” “self-quantifying,” “self-quantification,” “personal imaging,” “personal informatics,” “personal sensing,” “self-tracking,” “self-analytics,” “autoveillance,” “self-(sur/sous)veillance,” “body hacking,” “personal media analytics,” and “personal informatics.” Sousveillance also includes metaveillance. Metaveillance is the seeing of sight, visualization of vision, sensing of sensors, and sensing their capacity to sense (Fig. 1.4). According to Nicholas Negroponte, Founder, Director, and Chairman of the MIT Media Lab, “Steve Mann is the perfect example of someone... who persisted in his vision and ended up founding a new discipline.” From Bangor Daily News — Sep. 26, 1997; later appears in Toronto Star — Jul. 8, 2001.

1.3 THEORY OF HDR IMAGING

The theory and practice of quantigraphic image processing, with comparametric equations, arose out of the field of sousveillance (wearable computing, quantimetric self-sensing, etc.) within the context of mediated reality (Mann, 1997a) and personal imaging (Mann, 1997b). However, it has potentially much more widespread applications in image processing than just the wearable photographic and videographic vision systems for which it was developed. Accordingly, a general formulation that does not necessarily involve a wearable camera system will be given. This section follows very closely, if not identically, that given in Mann (2000) and the textbook *Intelligent Image Processing* (Mann, 2001).

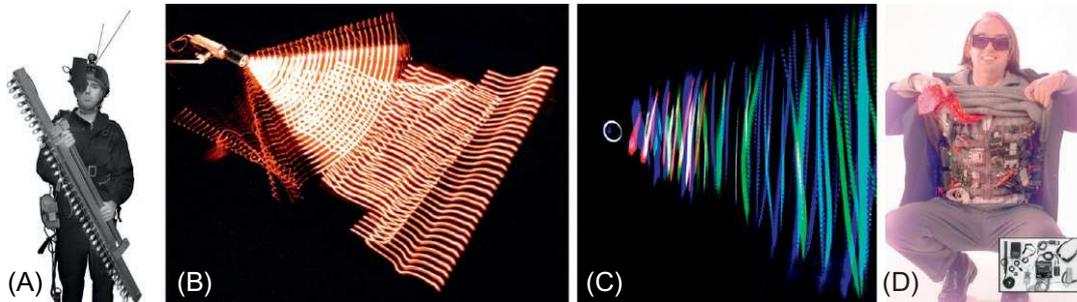


FIGURE 1.4

Sousveillance: (A) Sequential Wave Imprinting Machine developed by author S. Mann in the 1970s and early 1980s for making invisible fields such as sound waves and radio waves, visible, and also to sense sensing itself (Mann, 1992). In this example, a television receiver (“rabbit ears” antenna) picked up the video signal from a wireless surveillance camera and amplified the signal with enough strength to directly drive an array of 35 electric light bulbs. (B) Because of video feedback, the lamps illuminated when they were within the field of view of the surveillance camera. The effect was an AR display rendering the camera’s sightfield (Mann, 2014) visible to anyone watching the light wand being waved back and forth (no need for special eyewear). A problem with metasensing is the massive dynamic range present in the sightfield (not just the camera’s field/angle of view, but also the wide variation in visual acuity, so rendered: more light meant more sight, but the light saturated and made it difficult to see differences in the amount of sight present at each point in space). (C) To overcome this limitation, pseudocolor metasensing was used. Here multiple-exposure photographs were taken through various colored filters while a metasensory device was moved through the space around the sensor (camera or human eye). Recent work on veillometrics (Janzen and Mann, 2014) also uses pseudocolor to render HDR metasensory images from a large collection of highly accurate scientific measurements of camera or human sight. (D) Author S. Mann, in 1996, with video (electrovisuogram), electrocardiogram, electroencephalogram, respiration, skin conductivity, and various other sensors that provide a highly complete capture, processing, recording, and transmission of physiological body function plus surrounding environmental information.

From Mann, S., 2001. Intelligent Image Processing. John Wiley and Sons, New York, p. 384.

1.3.1 THE WYCKOFF PRINCIPLE AND THE RANGE OF LIGHT

The quantity of light falling on an image sensor array, or the like, is a real-valued function $q(x, y)$ of two real variables x and y . An image is typically a degraded measurement of this function, where degradations may be divided into two categories, those that act on the domain (x, y) and those that act on the range q . Sampling, aliasing, and blurring act on the domain, while noise (including quantization noise) and the nonlinear response function of the camera act on the range q .

Registering and combining multiple pictures of the same subject matter will often result in an improved image of greater definition. There are four classes of such improvement:

1. increased spatial resolution (domain resolution),
2. increased spatial extent (domain extent),
3. increased tonal fidelity (range resolution), and
4. increased dynamic range (range extent).

1.3.2 WHAT'S GOOD FOR THE DOMAIN IS GOOD FOR THE RANGE

The notion of producing a better picture by combining multiple input pictures has been well studied with regard to the *domain* (x, y) of these pictures. [Horn and Schunk \(1981\)](#), for example, provided a means of determining optical flow, and many researchers have used this result to spatially *register* multiple images in order to provide a single image of increased spatial resolution and increased spatial extent. Subpixel registration methods such as those proposed by [Irani and Peleg \(1991\)](#) and [Mann and Picard \(1994\)](#) attempt to increase *domain resolution*. These methods depend on a slight (subpixel) shift from one image to the next. Image compositing (mosaicing) methods such as those proposed by [Mann \(1993\)](#), [Mann and Picard \(1995c\)](#), and [Szeliski \(1996\)](#) attempt to increase *domain extent*. These methods depend on large shifts from one image to the next.

Methods that are aimed at increasing *domain resolution* and *domain extent* tend to also improve tonal fidelity, to a limited extent, by virtue of a signal-averaging and noise-reducing effect. However, we will see in what follows a generalization of the concept of signal averaging called “quantigraphic signal averaging.” This generalized signal averaging allows images of different exposure to be combined to further improve on tonal fidelity (*range resolution*), beyond improvements possible by traditional signal averaging. Moreover, the proposed method drastically increases dynamic range (*range extent*). Just as spatial shifts in the domain (x, y) improve the image, we will also see how exposure shifts (shifts in the range, q) can, with the proposed method, result in even greater improvements to the image.

1.3.3 EXTENDING DYNAMIC RANGE AND IMPROVEMENT OF RANGE RESOLUTION BY COMBINING DIFFERENTLY EXPOSED PICTURES OF THE SAME SUBJECT MATTER

The principles of quantigraphic image processing and the notion of the use of differently exposed pictures of the same subject matter to make a picture composite of extended dynamic range were inspired by the pioneering work of [Wyckoff \(1962, 1961\)](#), who invented so-called “extended response film.”

Most everyday scenes have a far greater dynamic range than can be recorded on a photographic film or electronic imaging apparatus. However, a set of pictures that are identical except for their exposure collectively show us much more dynamic range than any single picture from that set, and also allow the camera’s response function to be estimated, to within a single constant scalar unknown ([Mann, 1993, 1996a; Mann and Picard, 1995b](#)).

A set of functions

$$f_i(\mathbf{x}) = f(k_i q(\mathbf{x})), \quad (1.1)$$

where k_i are scalar constants, is known as a Wyckoff set ([Mann, 1993, 1996a](#)). A Wyckoff set of functions $f_i(\mathbf{x})$ describes a set of images differing only in exposure when $\mathbf{x} = (x, y)$ is the continuous spatial coordinate of the focal plane of an electronic imaging array (or piece of film), q is the quantity of light falling on the array (or film), and f is the unknown nonlinearity of the camera’s (or combined film’s and scanner’s) response function. Generally, f is assumed to be a pointwise function (eg, invariant to \mathbf{x}).

1.3.4 THE PHOTOQUANTITY, q

The quantity, q , in Eq. (1.1), is called the “*photoquantigraphic* quantity (Mann, 1998),” or just the “photoquantity” (or “photoq”) for short. This quantity is neither radiometric (eg, neither *radiance* nor *irradiance*) nor photometric (eg, neither *luminance* nor *illuminance*). Most notably, because the camera will not necessarily have the same spectral response as the human eye, or, in particular, that of the photopic spectral luminous efficiency function as determined by the CIE and standardized in 1924, q is not brightness, lightness, luminance, or illuminance. Instead, quantigraphic imaging measures the quantity of light integrated over the spectral response of the particular camera system,

$$q = \int_0^{\infty} q_s(\lambda)s(\lambda) d\lambda, \quad (1.2)$$

where $q_s(\lambda)$ is the quantity of light falling on the image sensor and s is the spectral sensitivity of an element of the sensor array. It is assumed that the spectral sensitivity does not vary across the sensor array.

1.3.5 THE CAMERA AS AN ARRAY OF LIGHT METERS

The quantity q reads in units that are quantifiable (eg, linearized or logarithmic), in much the same way that a photographic light meter measures in quantifiable (linear or logarithmic) units. However, just as the photographic light meter imparts to the measurement its own spectral response (eg, a light meter using a selenium cell will impart the spectral response of selenium cells to the measurement), quantigraphic imaging accepts that there will be a particular spectral response of the camera, which will define the quantigraphic unit q . Each camera will typically have its own quantigraphic unit. In this way, the camera may be regarded as an array of light meters, each being responsive to the quantigral:

$$q(x, y) = \int_0^{\infty} q_{ss}(x, y, \lambda)s(\lambda) d\lambda, \quad (1.3)$$

where q_{ss} is the spatially varying spectral distribution of light falling on the image sensor.

Thus, varying numbers of photons of lesser or greater energy (frequency times Planck’s constant) are absorbed by a given element of the sensor array, and, over the temporal quantigral time of a single frame in the video sequence (or the exposure time of a still image) result in the photoquantity given by Eq. (1.3).

In the case of a color camera, or other color processes, $q(x, y)$ is simply a vector quantity. Color images may arise from as few as two channels, as in the old bichromatic (orange and blue) motion pictures, but more typically arise from three channels, or sometimes more as in the four-color offset printing, or even the high-quality Hexachrome printing process. A typical color camera might, for example, include three channels — for example, $[q_r(x, y), q_g(x, y), q_b(x, y)]$ — where each component is derived from a separate spectral sensitivity function. Alternatively, another space such as YIQ, YUV, or the like may be used, in which, for example, the Y (luminance) channel has full resolution and the U and V channels have reduced (eg, half in each linear dimension giving rise to one quarter the number of pixels) spatial resolution and reduced quantizational definition. Part III of this book covers representation and coding of HDR video in general.

In this chapter, the theory will be developed and explained for grayscale images, where it is understood that most images are color images, for which the procedures are applied either to the

separate color channels or by way of a multichannel quantigraphic analysis. Thus, in both cases (grayscale and color) the continuous spectral information $q_s(\lambda)$ is lost through conversion to a single number q or to typically three numbers, q_r, q_g, q_b . Although it is easiest to apply the theory in this chapter to color systems having distinct spectral bands, there is no reason why it cannot also be applied to more complicated polychromatic, possibly tensor, quantigrals.

Ordinarily, cameras give rise to noise — for example, there is noise from the sensor elements and further noise within the camera (or equivalently noise due to film grain and subsequent scanning of a film, etc.). Thus, a goal of quantigraphic imaging is to attempt to estimate the photoquantity q in the presence of noise. Because $q_s(\lambda)$ is destroyed, the best we can do is to estimate q . Thus, q is the fundamental or “atomic” unit of quantigraphic image processing.

1.3.6 THE ACCIDENTALLY DISCOVERED COMPANDER

In general, cameras do not provide an output that varies linearly with light input. Instead, most cameras contain a dynamic range compressor, as illustrated in Fig. 1.5. Historically, the dynamic range compressor in video cameras arose because it was found that televisions did not produce a linear response to the video signal. In particular, it was found that early cathode ray screens provided a light output approximately equal to voltage raised to the exponent of 2.5. Rather than build a circuit into every television to compensate for this nonlinearity, a partial compensation (exponent of

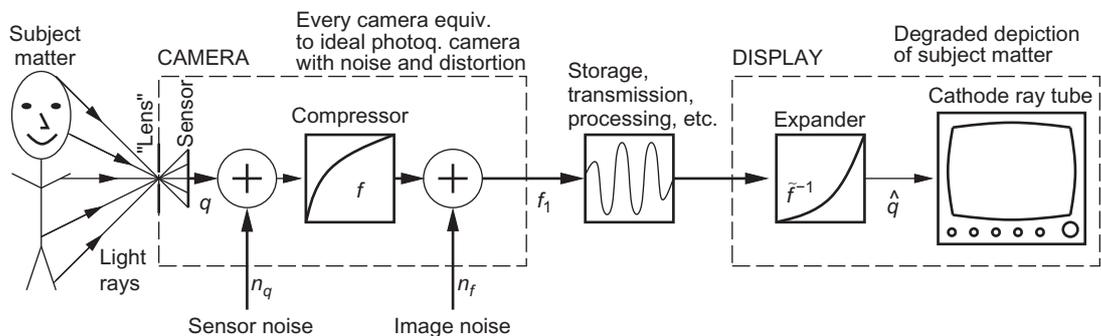


FIGURE 1.5

Typical camera and display. Light from subject matter passes through a lens (typically approximated with simple algebraic projective geometry, eg, an idealized “pinhole”) and is quantified in units “ q ” by a sensor array where noise n_q is also added, to produce an output which is compressed in dynamic range by a typically unknown function f . Further noise n_f is introduced by the camera electronics, including quantization noise if the camera is a digital camera and compression noise if the camera produces a compressed output such as a JPEG image, giving rise to an output image $f_1(x, y)$. The apparatus that converts light rays into $f_1(x, y)$ is labeled “CAMERA.” The image f_1 is transmitted or recorded and played back into a display system (labeled “DISPLAY”), where the dynamic range is expanded again. Most cathode ray tubes exhibit a nonlinear response to voltage, and this nonlinear response is the expander. The block labeled “expander” is generally a side effect of the display, and is not usually a separate device. It is depicted as a separate device simply for clarity. Typical print media also exhibit a nonlinear response that embodies an implicit “expander.”

1/2.22) was introduced into the television camera at much lesser total cost because there were far more televisions than television cameras in those days before widespread deployment of video surveillance cameras and the like. Indeed, the original model of television is suggested by the names of some of the early players: American Broadcasting Corporation (ABC), National Broadcasting Corporation (NBC), etc. Names such as these suggest that they envisioned a national infrastructure in which there would be one or two television cameras and millions of television receivers.

Through a very fortunate and amazing coincidence, the logarithmic response of human visual perception is approximately the same as the inverse of the response of a television tube (eg, human visual response is approximately the same as the response of the television camera) (Poynton, 1996). For this reason, processing done on typical video signals will be on a perceptually relevant tone scale. Moreover, any quantization on such a video signal (eg, quantization into 8 bits) will be close to ideal in the sense that each step of the quantizer will have associated with it a roughly equal perceptual change in perceptual units.

Fig. 1.6 shows plots of the compressor (and expander) used in video systems together with the corresponding logarithm $\log(q + 1)$ and antilogarithm $\exp(q) - 1$ plots of the human visual system and its inverse. (The plots have been normalized so that the scales match.)

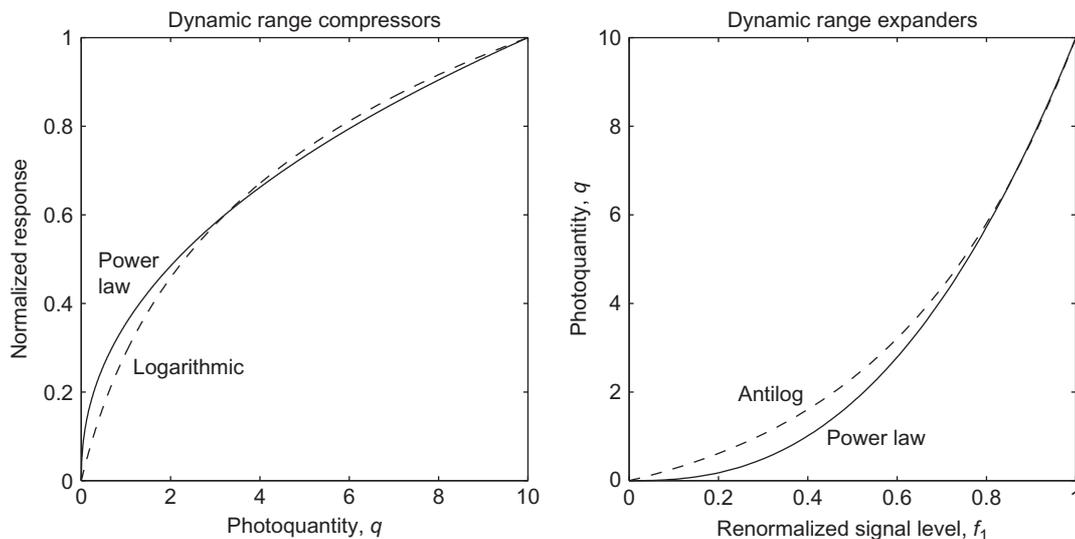


FIGURE 1.6

The power law dynamic range compression implemented inside most cameras has approximately the same shape of curve as the logarithmic function, over the range of signals typically used in video and still photography. Similarly, the power law response of typical cathode ray tubes, as well as that of typical print media, is quite similar to the antilogarithmic function. Therefore, the act of doing conventional linear filtering operations on images obtained from typical video cameras, or from still cameras taking pictures intended for typical print media, is, in effect, homomorphic filtering with an approximately logarithmic nonlinearity.

With images in print media, there is a similarly expansive effect in which the ink from the dots bleeds and spreads out on the printed paper, such that the midtones darken in the print. For this reason, printed matter has a nonlinear response curve similar in shape to that of a cathode ray tube (eg, the nonlinearity expands the dynamic range of the printed image). Thus, cameras designed to capture images for display on video screens have approximately the same kind of built-in dynamic range compression suitable for print media as well.

It is interesting to compare this naturally occurring (and somewhat accidental) development in video and print media with the deliberate introduction of companders (compressors and expanders) in the audio field. Both the accidentally occurring compression and expansion of picture signals and the deliberate use of logarithmic (or mu-law) compression and expansion of audio signals serve to allow 8 bits to be used to often encode these signals in a satisfactory manner. (Without dynamic range compression, 12–16 bits would be needed to obtain satisfactory reproduction.)

Most still cameras also have dynamic range compression built into the camera. For example, the Kodak DCS-420 and DCS-460 cameras capture images internally in 12 bits (per pixel per color) and then apply dynamic range compression, and finally output the range-compressed images in 8 bits (per pixel per color). Recently, as digital cameras have become dominant, range compression of images is still performed; however, modern cameras typically either emulate photographic film or perform computational tone mapping.

1.3.7 WHY STOCKHAM WAS WRONG

When video signals are processed, with linear filters, there is an implicit homomorphic filtering operation on the photoquantity. As should be evident from Fig. 1.5, operations of storage, transmission, and image processing occur between approximately reciprocal nonlinear functions of dynamic range compression and dynamic range expansion.

Many users of image-processing methods are unaware of this fact, because there is a common misconception that cameras produce a linear output, and that displays respond linearly. In fact there is a common misconception that nonlinearities in cameras and displays arise from defects and poor-quality circuits, when in fact these nonlinearities are fortuitously present in display media and deliberately present in most cameras. While CMOS and CCD response to light (electron counts) is usually linear, nonlinearities are introduced because of the need to reduce bit depth or produce display-referred images.

Thus, the effect of processing signals such as f_1 in Fig. 1.5 with linear filtering is, whether one is aware of it or not, homomorphic filtering; most computer vision cameras or RAW images from digital SLR cameras are linear, so in this case the assumption of linear camera output is correct.

Stockham advocated a kind of homomorphic filtering operation in which the logarithm of the input image was taken, followed by linear filtering (eg, linear space invariant filters), followed by the taking of the antilogarithm (Stockham, 1972).

In essence, what Stockham did not appear to realize is that such homomorphic filtering is already manifest in the application of ordinary linear filtering on ordinary picture signals (whether from video, film, or otherwise). In particular, the compressor gives an image $f_1 = f(q) = q^{1/2.22} = q^{0.45}$ (ignoring noise n_q and n_f) which has the approximate effect of $f_1 = f(q) = \log(q + 1)$ (eg, roughly the same shape of curve, and roughly the same effect, eg, to brighten the midtones of the image before processing), as shown in Fig. 1.6. Similarly, a typical video display has the effect of undoing

(approximately) this compression — for example, darkening the midtones of the image after processing with $\hat{q} = \tilde{f}^{-1}(f_1) = f_1^{2.5}$.

Thus, in some sense what Stockham did, without really realizing it, was to apply dynamic range compression to already range-compressed images, then do linear filtering, then apply dynamic range expansion to images being fed to already expansive display media.

1.3.8 THE VALUE OF DOING THE EXACT OPPOSITE OF WHAT STOCKHAM ADVOCATED

There exist certain kinds of image processing for which it is preferable to operate linearly on the photoquantity q . Such operations include sharpening of an image to undo the effect of the point spread function blur of a lens. It is interesting to note that many textbooks and articles that describe image restoration (eg, deblurring an image) fail to take into account the inherent nonlinearity deliberately built into most cameras.

What is needed to do this deblurring and other kinds of quantigraphic image processing is an *antihomomorphic filter*. The manner in which an antihomomorphic filter is inserted into the image-processing path is shown in Fig. 1.7.

Consider an image acquired through an imperfect lens that imparts a blurring to the image. The lens blurs the actual spatio-spectral (spatially varying and spectrally varying) quantity of light $q_{ss}(x, y, \lambda)$, which is the quantity of light falling on the sensor array just before the light is *measured* by the sensor array:

$$\tilde{q}_{ss}(x, y, \lambda) = \int \int B(x - u, y - v) q_{ss}(u, v, \lambda) du dv. \tag{1.4}$$

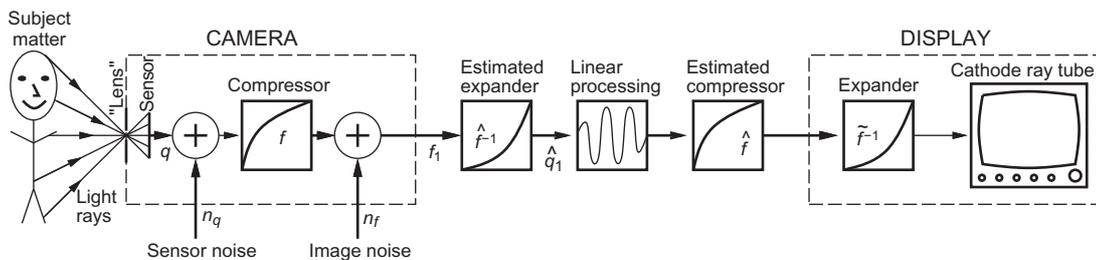


FIGURE 1.7

The antihomomorphic filter. Two new elements \hat{f}^{-1} and \hat{f} have been inserted as compared with Fig. 1.5. These are *estimates* of the inverse and forward nonlinear response functions of the camera. Estimates are required because the exact nonlinear response of a camera is generally not part of the camera specifications. (Many camera vendors do not even disclose this information if asked.) Because of noise in the signal f_1 , and also because of noise in the estimate of the camera nonlinearity f , what we have at the output of \hat{f}^{-1} is not q but, rather, an estimate, \tilde{q} . This signal is processed with linear filtering, and then the processed result is passed through the estimated camera response function, \hat{f} , which returns it to a compressed tone scale suitable for viewing on a typical television, computer, or the like, or for further processing.

This blurred spatio-spectral quantity of light $\tilde{q}_{ss}(x, y, \lambda)$ is then photoquantified by the sensor array:

$$\begin{aligned}
 q(x, y) &= \int_0^\infty \tilde{q}_{ss}(x, y, \lambda) s(\lambda) d\lambda \\
 &= \int_0^\infty \int_{-\infty}^\infty \int_{-\infty}^\infty B(x-u, y-v) q_{ss}(u, v, \lambda) s(\lambda) du dv d\lambda \\
 &= \int_{-\infty}^\infty \int_{-\infty}^\infty B(x-u, y-v) \left(\int_0^\infty q_{ss}(u, v, \lambda) s(\lambda) d\lambda \right) du dv \\
 &= \int_{-\infty}^\infty \int_{-\infty}^\infty B(x-u, y-v) q(u, v) du dv,
 \end{aligned} \tag{1.5}$$

which is just the blurred photoquantity q .

Thus the antihomomorphic filter of Fig. 1.7 can be used to undo the effect of lens blur better than traditional linear filtering, which simply applies linear operations to the signal f_1 and therefore operates homomorphically rather than linearly on the photoquantity q .

Thus, we see that in many practical situations there is an articulable basis for doing exactly the opposite of what Stockham advocated (eg, expanding the dynamic range of the image before processing and compressing it afterward as opposed to what Stockham advocated, which was to compress the dynamic range before processing and expand it afterward).

1.3.9 USING DIFFERENTLY EXPOSED PICTURES OF THE SAME SUBJECT MATTER TO GET A BETTER ESTIMATE OF q

Because of the effects of noise (quantization noise, sensor noise, etc.), in practical imaging situations, the Wyckoff set that describes a plurality of pictures that differ only in exposure (1.1) should be rewritten as follows:

$$f_i(\mathbf{x}) = f(k_i q(\mathbf{x}) + n_{q_i}) + n_{f_i}, \tag{1.6}$$

where each image has, associated with it, a separate realization of a quantigraphic noise process n_{q_i} and an image noise process n_{f_i} , which includes noise introduced by the electronics of the dynamic range compressor f and other electronics in the camera that affect the signal *after* its dynamic range has been compressed. In the case of a digital camera, n_{f_i} also includes quantization noise (applied after the image has undergone dynamic range compression). Furthermore, in the case of a camera that produces a data-compressed output, such as the Kodak DC260, which produces JPEG images, n_{f_i} also includes data-compression noise (JPEG artifacts, etc., which are also applied to the signal after it has undergone dynamic range compression). Refer again to Fig. 1.5.

If it were not for noise, we could obtain the photoquantity q from any one of a plurality of differently exposed pictures of the same subject matter — for example as

$$q = \frac{1}{k_i} f^{-1}(f_i), \tag{1.7}$$

where the existence of an inverse for f follows from the semimonotonicity assumption. Semimonotonicity follows from the fact that we expect pixel values to either increase or stay the same with increasing

quantity of light falling on the image sensor.¹ However, because of noise, we obtain an advantage by capturing multiple pictures that differ only in exposure. The dark (“underexposed”) pictures show us highlight details of the scene that would have been overcome by noise (eg, washed out) had the picture been “properly exposed.” Similarly, the light pictures show us some shadow detail that would not have appeared above the noise threshold had the picture been “properly exposed.”

Each image thus provides us with an estimate of the actual photoquantity q :

$$q = \frac{1}{k_i} (f^{-1}(f_i - n_{f_i}) - n_{q_i}), \quad (1.8)$$

where n_{q_i} is the quantigraphic noise associated with image i , and n_{f_i} is the image noise for image i . This estimate of q , \hat{q}_i , may be written as

$$\hat{q}_i = \frac{1}{\hat{k}_i} \hat{f}^{-1}(f_i), \quad (1.9)$$

where \hat{q}_i is the estimate of q based on our considering image i , and \hat{k}_i is the estimate of the exposure of image i based on our considering a plurality of differently exposed images. The estimated \hat{q}_i is also typically based on an estimate of the camera response function f , which is also based on our considering a plurality of differently exposed images. Although we could just assume a generic function $f(q) = q^{0.45}$, in practice, f varies from camera to camera. We can, however, make certain assumptions about f that are reasonable for most cameras, such as the fact that f does not decrease when q is increased (that f is semimonotonic), and that it is usually smooth, and that $f(0) = 0$. In what follows, it will be shown how k and f are estimated from multiple differently exposed pictures. For the time being, let us suppose that they have been successfully estimated, so that we can calculate \hat{q}_i from each of the input images i . Such calculations, for each input image i , give rise to a plurality of estimates of q , which in theory would be identical, were it not for noise. However, in practice, because of noise, the estimates \hat{q}_i are each corrupted in different ways. Therefore, it has been suggested that multiple differently exposed images may be combined to provide a single estimate of q which can then be turned into an image of greater dynamic range, greater tonal resolution, and lesser noise (Mann, 1993, 1996a). In particular, the criteria under which collective processing of multiple differently exposed images of the same subject matter will give rise to an output image which is acceptable at every point (x, y) in the output image, are summarized as follows:

The Wyckoff signal/noise criteria: $\forall (x_0, y_0) \in (x, y), \exists k_i q(x_0, y_0)$ such that

1. $k_i q(x_0, y_0) \gg n_{q_i}$ and
2. $c_i(q(x_0, y_0)) \gg c_i\left(\frac{1}{k_i} f^{-1}(n_{f_i})\right)$.

The first criterion indicates that for every pixel in the output image, at least one of the input images provides sufficient exposure at that pixel location to overcome sensor noise, n_{q_i} . The second criterion states that at least one input image provides an exposure that falls favorably (eg, is neither overexposed nor underexposed) on the response curve of the camera, so as not to be overcome by camera noise n_{f_i} .

The manner in which differently exposed images of the same subject matter are combined is illustrated, by way of an example involving three input images, in Fig. 1.8.

¹Except in rare instances where the illumination is so intense as to damage the imaging apparatus — for example, when the sun burns through photographic negative film and appears black in the final print or scan.

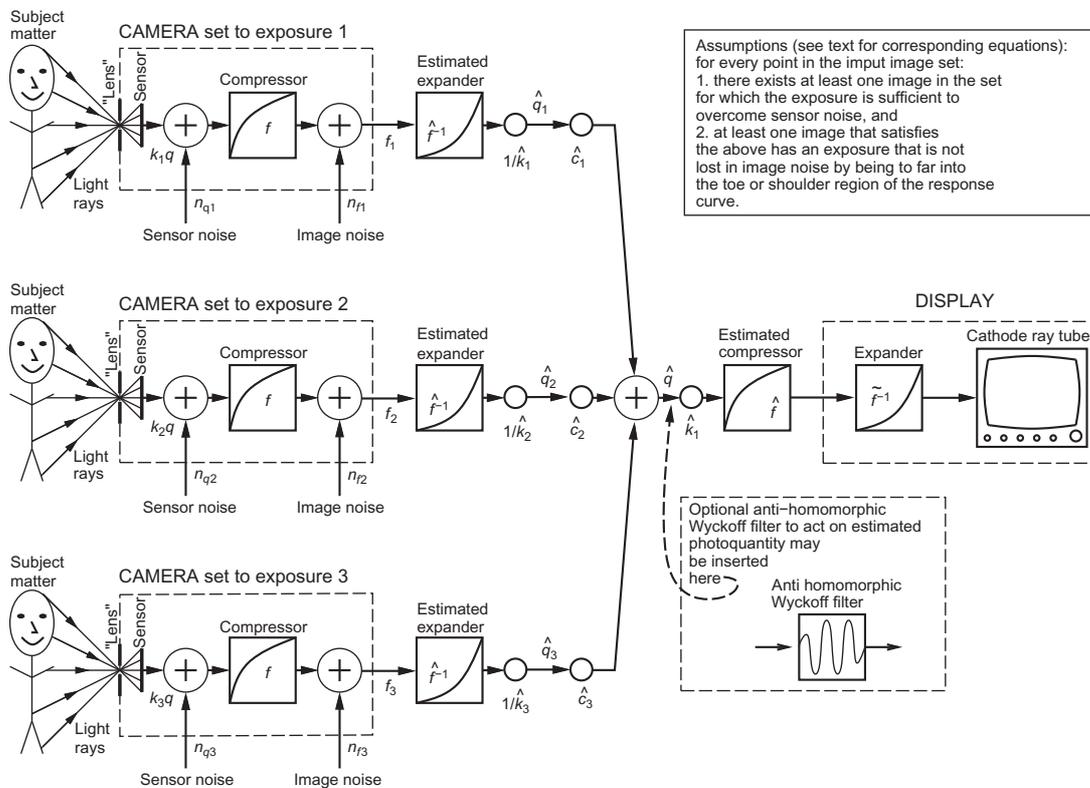


FIGURE 1.8

The Wyckoff principle. Multiple differently exposed images of the same subject matter are captured by a single camera. In this example there are three different exposures. The first exposure (CAMERA set to exposure 1), gives rise to an exposure $k_1 q$, the second exposure gives rise to an exposure $k_2 q$, and the third exposure gives rise to an exposure $k_3 q$. Each exposure has a different realization of the same noise process associated with it, and the three noisy pictures that the camera provides are denoted f_1 , f_2 , and f_3 . These three differently exposed pictures constitute a noisy Wyckoff set. To combine them into a single estimate, the effect of f is undone with an estimate \hat{f} that represents our best guess of what the function f is. While many video cameras use something close to the standard $f = kq^{0.45}$ function, it is preferable to attempt to estimate f for the specific camera in use. Generally, this estimate is made together with an estimate of the exposures k_i . After the dynamic ranges have been reexpanded with \hat{f}^{-1} , the inverse of the estimated exposures $1/\hat{k}_i$ is applied. In this way, the darker images are made lighter and the lighter images are made darker, so that they all (theoretically) match. At this point the images will all appear as if they were taken with identical exposure, except that the pictures that were brighter to start with will be noisy in lighter areas of the image and those that were darker to start with will be noisy in dark areas of the image. Thus, rather than application of ordinary *signal averaging*, a weighted average is taken. The weights are the spatially varying *certainty functions*, $c_i(x, y)$. These certainty functions are the derivative of the camera response function shifted up or down by an amount k_i . In practice, because f is an estimate, so is c_i , so it is denoted \hat{c}_i in this figure. The weighted sum is $\hat{q}(x, y)$, the estimate of the photoquantity $q(x, y)$. To view this quantity on a video display, it is first adjusted in exposure, and may be adjusted to an exposure level different from any of the exposure levels used in the taking of the input images. In this figure, for illustrative purposes, it is set to the estimated exposure of the first image, \hat{k}_1 . The result is then range-compressed with \hat{f} for display on an expansive medium (DISPLAY).

Moreover, it has been shown (Mann and Picard, 1995b) that the constants k_i as well as the unknown nonlinear response function of the camera can be determined, up to a single unknown scalar constant, given nothing more than two or more pictures of the same subject matter, in which the pictures differ only in exposure. Thus, the reciprocal exposures used to tonally register (tonally align) the multiple input images are estimates, $1/\hat{k}_i$, in Fig. 1.8. These exposure estimates are generally made by application of an estimation algorithm to the input images, either while f is simultaneously estimated or as a separate estimation process (because f has to be estimated only once for each camera but the exposure k_i is estimated for every picture i that is taken).

Owing to the large dynamic range that some Wyckoff sets can cover, small errors in f tend to have adverse effects on the overall estimate \hat{q} . Thus, it may be preferable to estimate f as a separate process (eg, by the taking of hundreds of exposures with the camera under computer program control). Once f is known (previously measured), k_i can be estimated for a particular set of images.

The final estimate for q , depicted in Fig. 1.8, is given by

$$\hat{q}(x, y) = \frac{\sum_i \hat{c}_i \hat{q}_i}{\sum_i \hat{c}_i} = \frac{\sum_i \frac{\hat{c}_i(\hat{q}(x, y))}{\hat{k}_i} \hat{f}^{-1}(f_i(x, y))}{\sum_i \hat{c}_i(\hat{q}(x, y))}, \quad (1.10)$$

where \hat{c}_i is given by

$$\hat{c}_i(\log(q(x, y))) = \frac{df_i(x, y)}{d \log \hat{q}(x, y)} = \frac{d\hat{f}(\hat{k}_i \hat{q}(x, y))}{d \log \hat{q}(x, y)}, \quad (1.11)$$

from which we can see that $c_i(\log q)$ are just shifted versions of $c(\log q)$ — for example, dilated versions of $c(q)$. While this analysis is useful for insight into the process, the certainty and uncertainty functions in this form ignore other sources of noise (eg, photon noise, readout noise), which are dominant for modern cameras. See Granados et al. (2010) for details on handling camera noise.

The intuitive significance of the certainty function is that it captures the slope of the response function, which indicates how quickly the output (pixel value or the like) of the camera varies for given input. In the case of a noisy camera, especially a digital camera, where quantization noise is involved, generally the output of the camera will be most reliable where it is most sensitive to a fixed change in input light level. This point where the camera is most responsive to changes in input is at the peak of the certainty function, c . The peak in c tends to be near the middle of the camera's exposure range. On the other hand, where the camera exposure input is extremely large or small (eg, the sensor is very much overexposed or very much underexposed), the change in output for a given input is much less. Thus, the output is not very responsive to the input, and the change in output can be easily overcome by noise. Thus, c tends to fall off toward zero on either side of its peak value.

The certainty functions are functions of q . We may also write the uncertainty functions, which are functions of pixel value in the image (eg, functions of gray value in f_i), as

$$U(x, y) = \frac{dF(f_i(x, y))}{df_i(x, y)}, \quad (1.12)$$

and its reciprocal is the certainty function C in the domain of the image (eg, the certainty function in pixel coordinates):

$$C(x, y) = \frac{df_i(x, y)}{dF(f_i(x, y))}, \quad (1.13)$$

where $F^{-1} = \log f$. Note that C is the same for all images (eg, for all values of image index i), whereas c_i was defined separately for each image. For any i , the function c_i is a shifted (dilated) version of any other certainty function, c_j , where the shift (dilation) depends on the log exposure, K_i (the exposure k_i).

The final estimate of q (1.10) is simply a weighted sum of the estimates from q obtained from each of the input images, where each input image is weighted by the certainties in that image.

1.3.10 EXPOSURE INTERPOLATION AND EXTRAPOLATION

The architecture of this process is shown in Fig. 1.9, which depicts an image acquisition section (in this illustration, of three images), followed by an analysis section (to estimate q), followed by a resynthesis section to generate an image again at the output (in this case four different possible output images are shown).

The output image can look like any of the input images, but with improved signal-to-noise ratio, better tonal range, better color fidelity, etc. Moreover, an output image can be an interpolated or extrapolated version in which it is lighter or darker than any of the input images. This process of interpolation or extrapolation provides a new way of adjusting the tonal range of an image. The process is illustrated in Fig. 1.9. The image synthesis portion may also include various kinds of deblurring operations, as well as other kinds of image sharpening and lateral inhibition filters to reduce the dynamic range of the output image without loss of fine details, so that it can be printed on paper or presented to an electronic display in such a way as to have optimal tonal definition.

1.4 COMPARAMETRIC IMAGE PROCESSING: COMPARING DIFFERENTLY EXPOSED IMAGES OF THE SAME SUBJECT MATTER

As previously mentioned, comparison of two or more differently exposed images may be done to determine q , or simply to tonally register the images without determining q . Also, as previously mentioned, tonal registration is more numerically stable than estimation of q , so there are some advantages to comparometric analysis and comparometric image processing in which one of the images is selected as a reference image and others are expressed in terms of this reference image, rather than in terms of q . Typically the dark images are lightened and/or the light images are darkened so that all the images match the selected reference image. In such lightening and darkening operations, full precision is retained for further comparometric processing. Thus, all but the reference image will be stored as an array of floating point numbers.

1.4.1 MISCONCEPTIONS ABOUT GAMMA CORRECTION

So-called *gamma correction* (raising the pixel values in an image to an exponent) is often used to lighten or darken images. While gamma correction does have important uses, such as lightening or darkening images to compensate for incorrect display settings, it will now be shown that when one uses gamma correction to lighten or darken an image to compensate for incorrect exposure that, whether one is aware of it or not, one is making an unrealistic assumption about the camera response function.

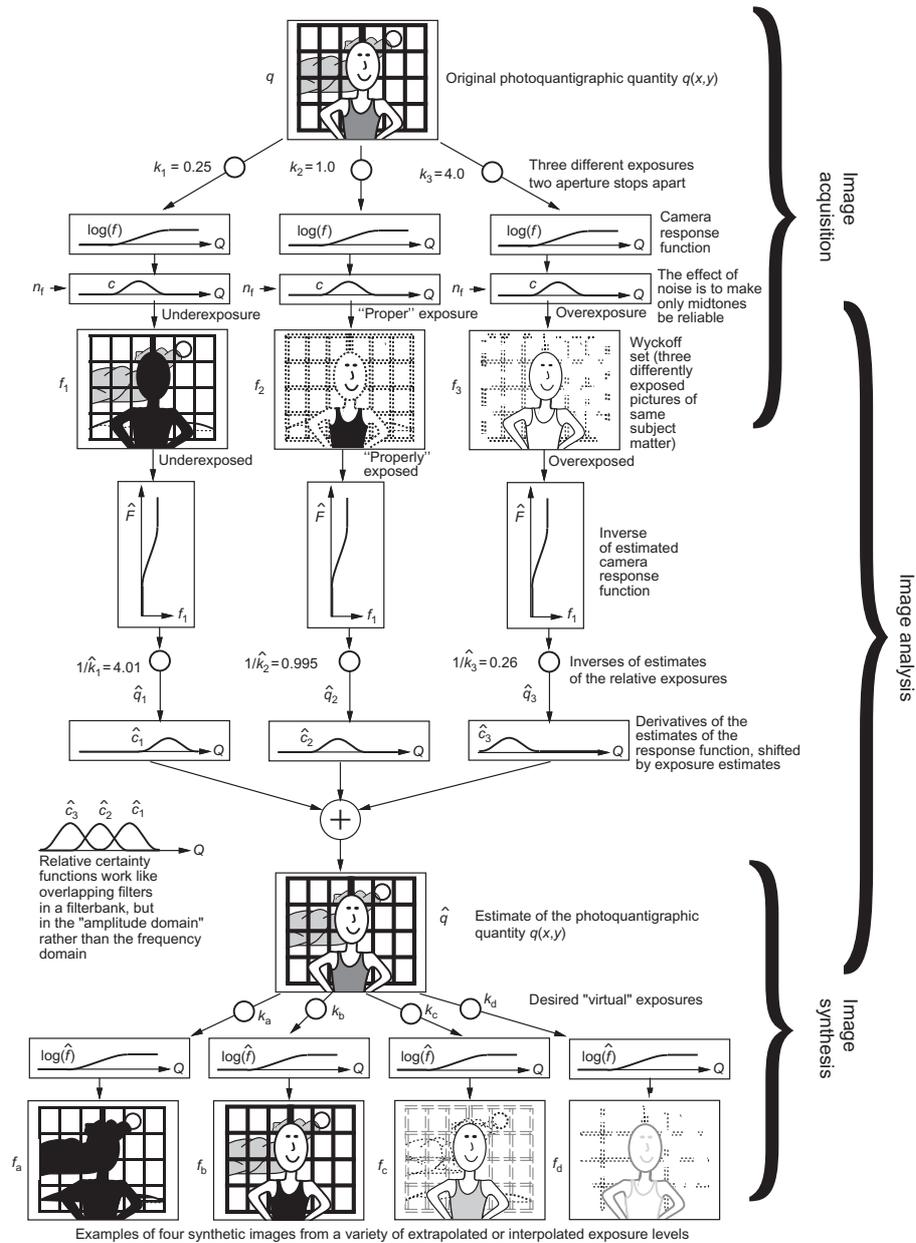


FIGURE 1.9

Quantigraphic exposure adjustment on a Wyckoff set. Multiple (in this example, three) differently exposed images are acquired. Estimates of q from each image are obtained and combined by weighted sum. Weights are estimates of the certainty function shifted along the exposure axis by an amount given by the estimated exposure for each image. From the estimated photoquantity \hat{q} , one or more output images may be generated by multiplication by the desired synthetic exposure and the passing of the result through the estimated camera nonlinearity. In this example, four synthetic pictures are generated. These are extrapolated and interpolated versions of the input exposures. The result is a "virtual camera" (Mann, 1999) where a picture can be generated as if the user were free to select the original exposure settings that had been used on the camera originally taking the input images.

Proposition 1.4.1. *Tonally registering differently exposed images of the same subject matter by gamma correcting them with exponent $\gamma = k^\Gamma$ is equivalent to assuming that the nonlinear response function of the camera is $f(q) = \exp(q^\Gamma)$.*

Proof. The process of gamma correcting an image may be written as

$$g(q) = f(kq) = (f(q))^\gamma, \quad (1.14)$$

where f is the original image and g is the lightened or darkened image. Solving for f , the camera response function, we obtain

$$f(q) = \exp q^\Gamma. \quad (1.15)$$

□

We see that the response function (1.15) does not pass through the origin — $f(0) = 1$, not 0. Because most cameras are designed so that they produce a signal level output of zero when the light input is zero, the function $f(q)$ does not correspond to a realistic or reasonable camera response function. Even a medium which does not record a zero at zero exposure (eg, film) is ordinarily scanned in such a way that the scanned output is zero for zero exposure, assuming that d_{\min} (minimum density for the particular emulsion being scanned) is properly set in the scanner. Therefore, it is inappropriate and incorrect to use gamma correction to lighten or darken differently exposed images of the same subject matter when the goal of this lightening or darkening is tonal registration (making them look the “same,” apart from the effects of noise, which will be accentuated in the shadow detail of the images that are lightened and the highlight detail of images that are darkened).

1.4.2 COMPARAMETRIC PLOTS AND COMPARAMETRIC EQUATIONS

To understand the shortcomings of gamma correction, and to understand some alternatives, the concept of comparametric equations and comparametric plots will now be introduced.

Eq. (1.14) is an example of what is called a *comparametric equation* (Mann, 1999).

Comparametric equations are a special case of the more general class of equations called *functional equations* (Aczél, 1966), and *comparametric plots* are a special case of the more general class of plots called *parametric plots*.

The notion of a parametric plot is well understood. For example, the parametric plot $(r \cos q, r \sin q)$ is a plot of a circle of radius r . It does not depend explicitly on q so long as the domain of q includes at least all points on the interval from 0 to 2π , modulo 2π .

A comparametric plot is a special kind of parametric plot in which a function f is plotted against itself, and in which the parameterization of the ordinate is a linearly scaled parameterization of the abscissa.

More precisely, the comparametric plot is defined as follows:

Definition 1.4.1. A plot along coordinates $(f(q), f(kq))$ is called a *comparametric plot* (Mann, 1999) of the function $f(q)$.

A function $f(q)$ has a family of comparametric plots, one for each value of the constant k , which is called the *comparametric ratio*.

Proposition 1.4.2. *When a function $f(q)$ is monotonic, the comparametric plot $(f(q), f(kq))$ can be expressed as a monotonic function $g(f)$ not involving q .*

Thus, the plot in Definition 1.4.1 may be rewritten as a plot $(f, g(f))$ not involving q . In this form, the function g is called the *comparametric function*, and expresses the range of the function $f(kq)$ as a function of the range of the function $f(q)$ independently of the domain, q , of the function f .

The plot g defines what is called a *comparametric equation*:

Definition 1.4.2. Equations of the form $g(f(q)) = f(kq)$ are called comparametric equations (Mann, 1999).

A better understanding of comparametric equations may be had if we refer to the following diagram:

$$\begin{array}{ccc}
 & & k \\
 & q & \longrightarrow & kq \\
 f & \downarrow & & \downarrow & f \\
 & f(q) & \longrightarrow & f(kq) & \\
 & & & g &
 \end{array} \tag{1.16}$$

wherein it is evident that there are two equivalent paths to follow from q to $f(kq)$,

$$g \circ f = f \circ k. \tag{1.17}$$

Eq. (1.17) may be rewritten as

$$g = f \circ k \circ f^{-1}, \tag{1.18}$$

which provides an alternative definition of a *comparametric equation* to that given in Definition 1.4.2.

Eq. (1.14) is an example of a comparametric equation, and Eq. (1.15) is a solution of Eq. (1.14).

It is often preferable that comparametric equations be on the interval from 0 to 1 in the range of f . Equivalently stated, we desire comparametric equations to be on the interval from 0 to 1 in the domain of g and the range of g . In this case, the corresponding plots and equations are said to be *unicomparametric*. (Actual images typically range from 0 to 255 and must thus be rescaled so that they range from 0 to 1, for unicomparametric image processing.)

Often we also impose a further constraint that $f(0) = 0$, and the constraint that $g(0) = 0$ and $g(1) = 1$.

Solving a comparametric equation is equivalent to determining the unknown camera response function from a pair of images that differ only in exposure when the comparametric equation represents the relationship between gray values in the two pictures and the comparametric ratio, k , represents the ratio of exposures (eg, if one picture was taken with twice the exposure of the other, then $k = 2$).

1.4.3 ZETA CORRECTION OF IMAGES

An alternative to *gamma correction* is proposed. This alternative, called *zeta correction*, will also serve as another example of a comparametric equation.

For zeta correction, we simply adjust the exponential solution (1.15) of the comparametric equation given by traditional gamma correction so that the solution passes through the origin: $f(q) = \exp(q - 1)$. (For simplicity, and without loss of generality, γ has been set to k , the comparametric exposure ratio.)

Thus, $g = (bf + 1)^\gamma = (bf + 1)^k$. Preferably (to be unicomparametric) we would like to have $g(1) = 1$, so we use the response function

$$f(q) = \frac{e^q - 1}{\sqrt[k]{2} - 1}, \quad (1.19)$$

which is a solution to the corresponding comparametric equation:

$$g = ((\sqrt[k]{2} - 1)f + 1)^k - 1, \quad \forall k \neq 0. \quad (1.20)$$

The comparametric equation (1.20) forms the basis for zeta correction of images:

$$g = \begin{cases} ((2^\zeta - 1)f + 1)^{\frac{1}{\zeta}} - 1, & \forall \zeta \neq 0. \\ 2^f - 1, & \text{for } \zeta = 0. \end{cases} \quad (1.21)$$

Implicit in zeta correction of images is the assumption of an exponential camera response function, which, although not realistic (given that the exponential function expands dynamic range, and most cameras have compressive response functions rather than expansive response functions), is preferable to gamma correction because of the implicit notion of a response function for which $f(0) = 0$.

With standard IEEE arithmetic, values of ζ can range from approximately -50 to $+1000$.

1.4.4 THE AFFINE COMPARAMETRIC EQUATION AND AFFINE CORRECTION OF IMAGES

In this section, one of the two most useful (in the authors' opinion) comparametric equations is introduced, giving rise to *affine correction* of images. Affine correction is an improvement over zeta correction (which itself was an improvement over gamma correction).

First consider the classic model

$$f(q) = \alpha + \beta q^\gamma \quad (1.22)$$

used by photographers to characterize the response of a variety of photographic emulsions, including so-called *extended response* film (Wyckoff, 1962). It is well known that Eq. (1.22) becomes the equation of a straight line when expressed in logarithmic coordinates if we subtract α (as many scanners such as Photo CD scanners attempt to do by prompting the user to scan a piece of blank film from the film trailer before scanning the rest of the roll of film):

$$\log(f(q) - \alpha) = \gamma \log q + \beta. \quad (1.23)$$

It is an interesting coincidence that the comparametric plot of this function (1.22) is also a straight line:

Proposition 1.4.3. *The comparametric plot corresponding to the standard photographic response function (1.22) is a straight line. The slope is k^γ , and the intercept is $\alpha(1 - k^\gamma)$.*

Proof. $g(f(kq)) = f(kq) = \alpha + \beta(kq)^\gamma$. Rearranging the equation to eliminate q gives $g = k^\gamma(\alpha + \beta q^\gamma) + \alpha(1 - k^\gamma)$, so

$$g = k^\gamma f + \alpha(1 - k^\gamma). \quad (1.24)$$

□

The constant β does not appear in the comparametric equation and thus we cannot determine β from the comparametric equation. The physical (intuitive) interpretation is that we can determine the nonlinear response function of a camera only up to a single unknown scalar constant.

Eq. (1.14) looks quite similar in form to Eq. (1.22), and in fact is identical if we set $\alpha = 0$ and $\beta = 1$. However, one must recall that Eq. (1.14) is a comparametric equation and Eq. (1.22) is a solution to a (different) comparametric equation; thus, we must be careful not to confuse the two. The first corresponds to gamma correction of an image, while the second corresponds to the camera response function that is implicit in the application of Eq. (1.24) to lighten or darken the image. To make this distinction clear, application of Eq. (1.24) to lighten or darken an image will be called *affine correction* (eg, correction by the modeling of the comparametric function with a straight line). The special case of *affine correction* when the intercept is equal to zero will be called *linear correction*.

Preferably affine correction of an image also includes a step of clipping values greater than 1 to 1, and values less than 0 to 0, in the output image:

$$g = \min(\max(k^\gamma f + \alpha(1 - k^\gamma), 0), 1). \quad (1.25)$$

If the intercept is 0 and the slope is greater than 1, the effect, disregarding noise, of Eq. (1.25) is to lighten the image in a natural manner that properly simulates the effect of the image having been exposed with greater exposure. In this case, the effect is theoretically identical to that which would have been obtained by the use of a greater exposure on the camera, assuming the response function of the camera follows the power law $f = q^\gamma$, as is the case for many cameras. Thus, it has been shown that the correct way to lighten an image is to apply linear correction, not gamma correction (apart from correction of an image to match an incorrectly adjusted display device or the like, where gamma correction is still the correct operation to apply).

Here we have worked forward, starting with the solution (1.22) and deriving the comparametric equation (1.24), of which Eq. (1.22) is a solution. It is much easier to generate comparametric equations from their solutions than it is to solve comparametric equations.

The earlier comparametric equation is both useful and simple. The simplicity is in the ease with which it is solved, and in the fact that the solution happens to be the most commonly used camera response model in photography. As we will later see, when images are being processed, the comparametric function can be estimated by the fitting of a straight line through data points describing the comparametric relation between images. However, there are two shortcomings to affine correction:

1. It is not inherently unicomparametric, so it must be clipped to 1 when it exceeds 1 and to 0 when it falls below 0, as shown in Eq. (1.25).
2. Its solution, $f(q)$ describes the response of cameras only within their normal operating regime. Because the art of quantigraphic image processing involves a great deal of image processing done on images that have been deliberately and grossly overexposed or underexposed, there is a need for a comparametric model that captures the essence of cameras at both extremes of exposure (eg, both overexposure and underexposure).

1.4.5 THE PREFERRED CORRECTION OF IMAGES

Although affine correction was an improvement over zeta correction, which itself was an improvement over gamma correction, affine correction still has the two shortcomings listed earlier. Therefore, another form of image exposure correction is proposed, and it will be called the *preferred correction*. This new exposure correction is unicomparametric (bounded in normalized units between 0 and 1) and also has a parameter to control the softness of the transition into the *toe* and *shoulder* regions of the response function, rather than the hard clipping introduced by Eq. (1.25).

As with affine correction, the preferred correction will be introduced first by its solution, from which the comparometric equation will be derived. The solution is

$$f(q) = \left(\frac{e^b q^a}{e^b q^a + 1} \right)^c, \quad (1.26)$$

which has only three parameters. Thus, no extra unnecessary degrees of freedom (which might otherwise capture or model noise) have been added over and above the number of degrees of freedom in the previous model (1.22).

An intuitive understanding of Eq. (1.26) can be better had if we rewrite it:

$$f = \begin{cases} \exp \left[\frac{1}{(1 + e^{-(a \log q + b)})^c} \right]^c, & \forall q \neq 0, \\ 0, & \text{for } q = 0, \end{cases} \quad (1.27)$$

where the soft transition into the toe (region of underexposure) and shoulder (region of overexposure) is evident by the shape of this curve on a logarithmic exposure scale.

This model may, at first, seem like only a slight improvement over (1.22), given our common intuition that most exposure information is ordinarily captured in the central portion, which is linear on the logarithmic exposure plot. However, it is important that we unlearn what we have been taught in traditional photography, where incorrectly exposed images are ordinarily thrown away rather than used to enhance the other images! It must be emphasized that comparometric image processing differs from traditional image processing in the sense that in comparometric image processing (following the Wyckoff principle, as illustrated in Fig. 1.8) the images typically include some that are *deliberately* underexposed and overexposed. In fact this overexposure of some images and underexposure of other images is often deliberately taken to extremes. Therefore, the additional sophistication of the model (1.26) is of great value in capturing the essence of a set of images where some extend to great extremes in the *toe* or *shoulder* regions of the response function.

Proposition 1.4.4. *The comparometric equation of which the proposed photographic response function (1.26) is a solution is given by*

$$g(f) = \frac{f k^{ac}}{(\sqrt[c]{f}(k^a - 1) + 1)^c}, \quad (1.28)$$

where $K = \log(k)$.

Function (1.28) gives rise to *the preferred correction* of images (eg, the preferred recipe for lightening or darkening an image). Again, $g(f)$ does not depend on b , which is consistent with our knowledge that the comparametric equation captures the information of $f(q)$ up to a single unknown scalar proportionality constant.

1.4.6 SOME SOLUTIONS TO SOME COMPARAMETRIC EQUATIONS THAT ARE PARTICULARLY ILLUSTRATIVE OR USEFUL

Some examples of comparametric equations and their solutions are summarized in Table 1.1.

1.4.7 PROPERTIES OF COMPARAMETRIC EQUATIONS

As stated previously, the comparametric equation provides information about the actual photoquantity only up to a single unknown scalar quantity — for example, if $f(q)$ is a solution of comparametric equation g then so is $f(\beta q)$. In general we can think of this as a coordinate transformation from q to

Table 1.1 Illustrative or Useful Examples of Comparametric Equations and Their Solutions	
Comparametric Equation $g(f(q)) = f(kq)$	Solution (Camera Response Function) $f(q)$
$g = f^\gamma$	$f = \exp q^\Gamma, \quad \gamma = k^\Gamma$
$g = k^\gamma f$	$f = q^\gamma$
$g = af + b, \forall a \neq 1, \text{ or } b = 0$	$f = \alpha + \beta q^\gamma, \quad a = k^\gamma, b = \alpha(1 - k^\gamma)$
$g = f + b$	$f = B \log(\beta q), \quad b = B \log k$
$g = (\sqrt[\zeta]{f} + \log k)^\gamma$	$f = \log^\gamma q$
$g = \begin{cases} ((2^\zeta - 1)f + 1)^{\frac{1}{\zeta}} - 1, & \forall \zeta \neq 0 \\ 2^f - 1, & \text{for } \zeta = 0 \end{cases}$	$\frac{e^q - 1}{2^\zeta - 1}$
$g = e^{bf^a} = e^{\alpha(1-k^\gamma)f^{k^\gamma}}$	$\log f = \alpha + \beta q^\gamma$
$g = \exp((\log f)^{k^b})$	$f = \exp(a^{q^b})$
$g = \exp(\log^k f)$	$f = \exp(a^{bq})$
$g = \frac{2}{\pi} \arctan(k \tan(\frac{\pi}{2}f))$	$f = \frac{2}{\pi} \arctan q$
$g = \frac{1}{\pi} \arctan(b\pi \log k + \tan((f - 1/2)\pi)) + \frac{1}{2}$	$f = \begin{cases} \frac{1}{\pi} \arctan(b\pi \log q) + \frac{1}{2}, & \forall q \neq 0 \\ 0, & \text{for } q = 0 \end{cases}$
$g = \left(\frac{\sqrt[\zeta]{f} k^a}{\sqrt[\zeta]{f(k^a - 1) + 1}}\right)^c$	$f = \left(\frac{e^b q^a}{e^b q^a + 1}\right)^c = \begin{cases} \left(\frac{1}{1 + e^{-(a \log q + b)}}\right)^c, & \forall q \neq 0 \\ 0, & \text{for } q = 0 \end{cases}$
$g = \exp\left(\left(\frac{\sqrt[\zeta]{\log f} k^a}{\sqrt[\zeta]{\log f(k^a - 1) + 1}}\right)^c\right)$	$f = \exp\left(\left(\frac{e^b q^a}{e^b q^a + 1}\right)^c\right) = \begin{cases} \exp\left(\left(\frac{1}{1 + e^{-(a \log q + b)}}\right)^c\right), & \forall q \neq 0 \\ 0, & \text{for } q = 0 \end{cases}$

Notes: The third from the top and second from the bottom were found to describe a large variety of cameras and have been used in a wide variety of quantigraphic image-processing applications. The second one from the bottom is the one that is most commonly used by the authors.

βq , in the domain of f . More generally, the comparametric plot $(f(q), g(f(q)))$ has the same shape as the comparametric plot $(f(h(q)), g(f(h(q))))$, for all bijective h . From this fact we can construct a property of comparametric equations in general:

Proposition 1.4.5. *A comparametric equation $\check{g}(f(q)) = g(f(h(q))) = g(\check{f}(q))$ has solution $\check{f}(q) = f(h(q))$ for any bijective function h .*

Likewise, we can also consider coordinate transformations in the range of comparametric equations, and their effects on the solutions:

Proposition 1.4.6. *A comparametric equation $g(f) = h(\check{g})$ has solution $f = h(\check{f})$, where $\check{g}(q) = \check{f}(kq)$ is a comparametric equation with solution $\check{f}(q)$*

Properties of comparametric equations related to their coordinate transformations are presented in Table 1.2.

Some simple but illustrative examples of the use of coordinate transformation properties to solve comparametric equations are now provided:

Example 1. Let $\check{g} = a\check{f} + b$, which we know has solution $\check{f} = \alpha + \beta q^\gamma$, with $a = k^\gamma$ and $b = \alpha(1 - a)$. Let $h() = () + n_f$ (eg, h is a transformation which consists in simply adding noise.) Thus, $g(f) = h(\check{g}) = \check{g} + n_f = a\check{f} + b + n_f$, so $g = a(f - n_f) + b + n_f$ has solution $f = h(\check{f}) = \alpha + \beta q^\gamma + n_f$.

Example 2. From Table 1.1, observe that the comparametric equation $\check{g} = a\check{f} + b$ has solution $\check{f} = \alpha + \beta q^\gamma$. Let $h() = \exp()$. We can thus solve $g(f) = h(\check{g}) = \exp(a\check{f} + b) = \exp(a \log(f) + b) = e^b f^a$ by noting that $f = h(\check{f}) = \exp(\check{f}) = \exp(\alpha + \beta q^\gamma)$.

This solution also appears in Table 1.1. We may also use this solution to seed the solution of the comparametric equation second from the bottom in Table 1.1 by using $h(x) = x/(x + 1)$. The equation second from the bottom in Table 1.1 may then be further coordinate-transformed into the equation at the bottom of Table 1.1 by use of $h(x) = \exp(x)$. Thus, properties of comparametric equations, such as those summarized in Table 1.2, can be used to help solve comparametric equations, such as those listed in Table 1.1.

Comparametric Equations	Solutions (Camera Response Functions)
$g(f(q)) = f(kq)$	$\check{f}(q)$
$\check{g}(f) = g(\check{f})$, where $\check{g}(f(q)) = g(f(h(q)))$	$\check{f}(q) = f(h(q))$, \forall bijective h
$g(f) = \check{g}(f)$, where $\check{g}(f(q)) = \check{f}(\beta q)$	$f(q) = \check{f}(\beta q)$
$g(f) = g(h(\check{f})) = h(\check{g})$, where $\check{g}(q) = \check{f}(kq)$	$f = h(\check{f})$
$h^{-1}(g) = \check{g}(f)$	$f = h(\check{f})$
<i>Notes: This table could be extended over several pages, much like an extensive table listing properties of Laplace transforms, or a table of properties of Fourier transforms, or the like.</i>	

1.5 PRACTICAL IMPLEMENTATIONS

This section pertains to the practical implementation of the theory presented in previous sections.

1.5.1 COMPARING TWO IMAGES THAT DIFFER ONLY IN EXPOSURE

Without loss of generality, consider two differently exposed pictures of the same subject matter, f_1 and f_2 , and recognize that in the absence of noise the relationship between the two images would be

$$\frac{1}{k_1}f^{-1}(f_1) = q = \frac{1}{k_2}f^{-1}(f_2), \quad (1.29)$$

so

$$f_2 = f(kf^{-1}(f_1)) = F^{-1}(F(f_1) + K), \quad (1.30)$$

where $k = k_2/k_1$, $K = \log(k_2) - \log(k_1)$, and $F = \log(f^{-1})$. It is evident that Eq. (1.30) is a comparametric equation.

This process (1.30) of “registering” the second image with the first differs from the image registration procedure commonly used in much of machine vision (Horn, 1986; Horn and Weldon, 1988; Faugeras and Lustman, 1988; Laveau and Faugeras, 1994) and image resolution enhancement (Irani and Peleg, 1991; Szeliski, 1996; Mann and Picard, 1995c) because it operates on the *range*, $f(q(\mathbf{x}))$ (tonal range), of the image $f_i(\mathbf{x})$ as opposed to its *domain* (spatial coordinates) $\mathbf{x} = (x, y)$.

1.5.2 JOINT HISTOGRAMS AND COMPARAGRAMS

The joint histogram between two images is a square matrix of size $M \times N$, where M is the number of gray levels in the first image and N is the number of gray levels in the second image (Mann and Picard, 1995b; Mann, 1996a). When the two images have the same number of gray levels, $M = N$, the joint histogram is a square $N \times N$ array. When the two images are (possibly) differently exposed images of the same subject matter, the joint histogram of these images is called a *comparagram* (Mann, 2000). When the two images are identical, the comparagram is zero except along the diagonal, and the diagonal values are those of the histogram of the input image.

It can be seen from Eq. (1.30) that the general problem of solving Eq. (1.30) can be done directly on the comparagram instead of the original pair of images. This comparametric approach has the added advantage of breaking the problem down into two separate simpler steps:

1. *Comparametric regression*: Finding a smooth semimonotonic function, g , that passes through most of the highest bins in the comparagram.
2. *Solving the comparametric equation*: Unrolling this function, $g(f(q)) = f(kq)$ into $f(q/q_0)$ by regarding it as an *iterative map* onto itself (see Fig. 1.10.) The iterative map (*logistic map*) is most familiar in chaos theory (Woon, 1995; Berger, 1989), but here, because the map is monotonic, the result is a deterministic function.

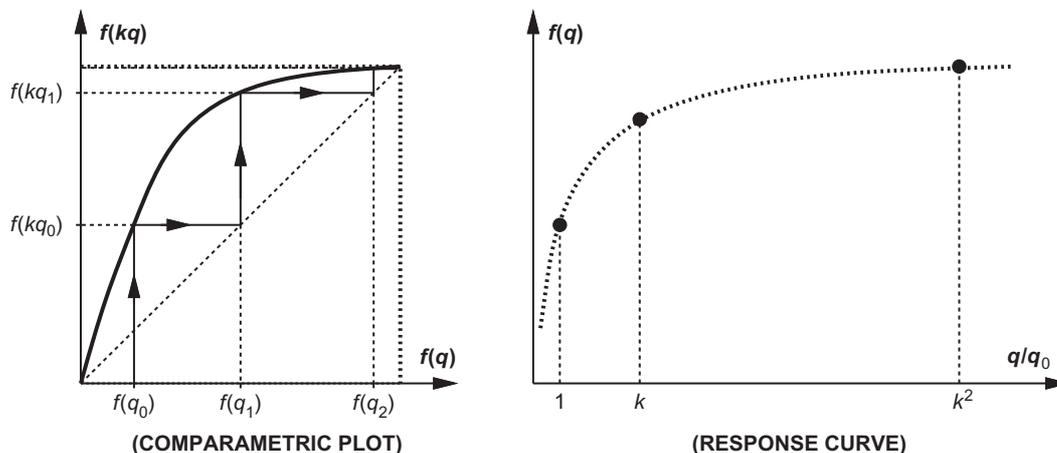


FIGURE 1.10

Comparametric procedure for finding the pointwise nonlinearity of an image sensor from two pictures differing only in their exposures. Left: Plot of pixel values in one image against corresponding pixel values in the other. Right: Points on the response curve, found from only the two pictures, without any knowledge of the characteristics of the image sensor. These discrete points are only for illustrative purposes. If a logarithmic exposure scale is used (as is done by most photographers), then the points fall uniformly on the $Q = \log(q/q_0)$ axis.

Separating this estimation process into two stages also allows us to have a more direct route to “registering” the image domains if, for example, we do not need to know f but require only g , which is the recipe for expressing the range of $f(kq)$ in the units of $f(q)$. In particular, we can lighten or darken images to match one another without ever having to solve for q . The first part of the earlier two-step process allows us to determine the relationship between two pictures that differ only in exposure, so that we can directly perform operations such as image exposure interpolation and extrapolation as in Fig. 1.9, but skip the intermediate step of computing q . Not all image-processing applications require q to be determined, so there is great value in simply understanding the relationship between differently exposed pictures of the same subject matter.

1.5.3 COMPARAMETRIC REGRESSION AND THE JOINT HISTOGRAM

In situations where the image data are extremely noisy, and/or where a closed-form solution for $f(q)$ is desired, a parameterized form of the comparametric function is used, in which a function $g(f)$ corresponding to a suitably parameterized response function $f(q)$ is selected. The method amounts to a *curve fitting* problem in which the parameters of g are selected so that g best fits one or more comparagrams (joint histograms constructed from two or more differently exposed images under analysis).

1.5.4 COMPARAMETRIC REGRESSION TO A STRAIGHT LINE

Eq. (1.24) suggests that we can determine $f(q)$ from two differently exposed images by applying linear regression to the joint histogram of the images, J , treating each entry as a data point and weighting it by the number of bin counts $J(m, n)$ at each point. One often does this by combining the input images, weighting them with $(J(m, n))^\lambda$. For example, $\lambda = 0$ (assuming empty bins are not counted) provides the classic linear regression problem in which all nonempty bins are weighted equally and the slope and intercept of the best-fit line through nonempty bins are found. Generally, λ is chosen somewhere between 0.25 and 2.

A simple example is presented, that of reverse engineering the standard Kodak Photo CD scanner issued to most major photographic processing and scanning houses. In most situations, a human operator runs the machine and decides, by visual inspection, what “brightness” level to scan the image at (there is also an automatic exposure feature which allows the operator to preview the scanned image and decide whether or not the chosen “brightness” level needs to be overridden). By scanning the same image at different “brightness” settings, one obtains a Wyckoff set. This allows the scanner to capture nearly the entire dynamic range of the film, which is of great utility because typical photographic negative film captures far greater dynamic range than is possible with the scanner as it is ordinarily used. A photographic negative taken from a scene of extremely high contrast (a sculpture on exhibit at the List Visual Arts Center, in a completely darkened room, illuminated with a bare flashlamp from one side only) was selected because of its great dynamic range that could not be captured in a single scan. A Wyckoff set was constructed by the scanning of the same negative at five different “brightness” settings (Fig. 1.11). The settings were controlled by a slider that was calibrated in arbitrary units from -99 to $+99$ while Kodak’s proprietary scanning software was being run. Kodak provides no information about what these units mean. Accordingly, the goal of the experiment was to find a closed-form mathematical equation describing the effects of the “brightness” slider on the scans, and to recover the unknown nonlinearity of the scanner. To make the problem a little more challenging and, more

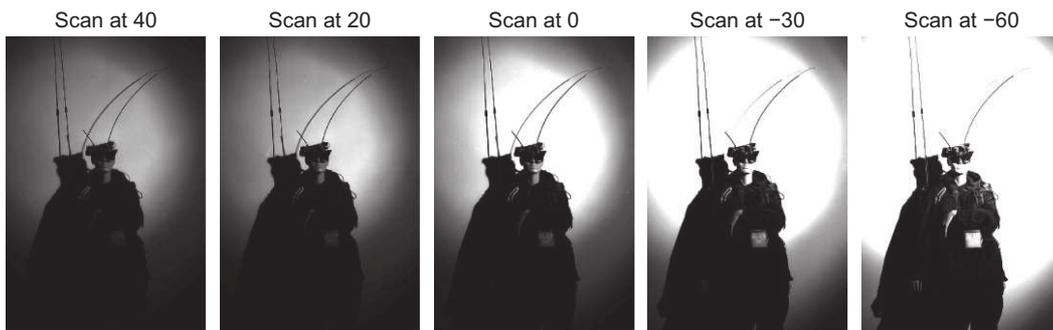


FIGURE 1.11

These scans from a photographic negative differ only in the choice of the “brightness” setting selected with the slider provided on the X-Windows screen by the proprietary Kodak Photo CD scanning software. The slider is calibrated in arbitrary units from -99 to $+99$. Five scans were done, and the setting of the slider is noted above each scan.

importantly, to better illustrate the principles of comparometric image processing, the *dmin* procedure of scanning a blank film at the beginning of the roll was overridden.

Joint histograms J_{01} , J_{12} , J_{23} , and J_{34} were computed from the five images (v_0 through v_4) in Fig. 1.11, and are displayed as density plots (eg, they are treated as images of dimension 256×256 pixels, where the darkness of the image is proportional to the number of counts — darkness rather than lightness to make it easier to see the pattern) in Fig. 1.12. Linear regression was applied to the data, and the best-fit straight line is shown passing through the data points. Because the *dmin* procedure was overridden, notice that the plots do not pass through the origin. The two leftmost plots had nearly identical slopes and intercepts, and likewise for the two rightmost plots, which indicates that the arbitrary Kodak units of “brightness” are self-consistent (eg, J_{01} , which describes the relationship between a scan at a “brightness” of 40 units and a scan at a “brightness” of 20 units, is essentially the same as J_{12} , which describes the relationship between a scan at a “brightness” of 20 units and a scan at a “brightness” of 0 units). Because there are three parameters in Eq. (1.22), k , α , and γ , which describe only two degrees of freedom (slope and intercept), γ may be chosen so that $k = \sqrt[\gamma]{a}$ is linearly proportional to arbitrary Kodak units. Thus, setting $(\sqrt[\gamma]{a_{\text{left}}})/(\sqrt[\gamma]{a_{\text{right}}}) = 20/30$ (where a_{left} is the average slope of the two leftmost plots in Fig. 1.12 and a_{right} is the average slope of the two rightmost plots in Fig. 1.12) results in $\gamma = 0.2254$. From this we obtain $\alpha = b/(1 - a) = 23.88$. Thus, we have

$$f(k_i q) = 23.88 + (k_i q)^{0.2254}, \tag{1.31}$$

where k_i is in arbitrary Kodak units (eg, $k_0 = 40$ for the leftmost image, $k_1 = 20$ for the next image, $k_2 = 0$ for the next image, $k_3 = -30$ for the next image, and $k_4 = -60$ for the rightmost image in Fig. 1.11). Thus, Eq. (1.31) gives us a closed-form solution that describes the response curve associated with each of the five exposures $f(k_i q)$, $i \in \mathbb{Z}, 0 \leq i \leq 4$. The curves $f(k_i q)$ may be differentiated, and if these derivatives are evaluated at $q_i = \frac{1}{k_i} \sqrt[\alpha]{f_i(x, y) - \alpha}$, the so-called *certainty images*, shown in Fig. 1.13, are obtained.

In the next example, the use of the certainty functions to construct an optimal estimate, $\hat{q}(x, y)$ will be demonstrated.

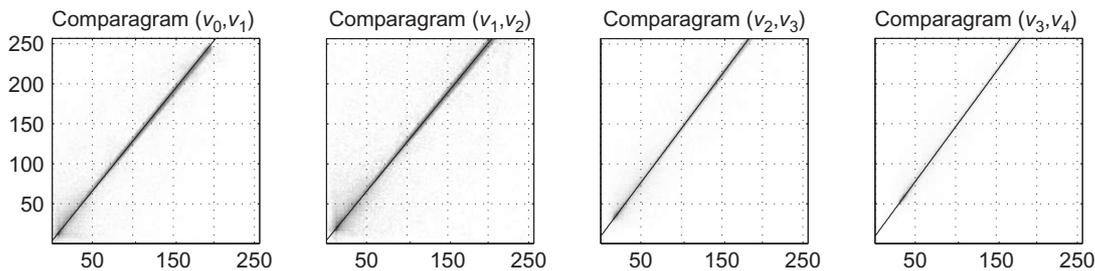


FIGURE 1.12 Pairwise joint histograms of the images in Fig. 1.12. It is evident that the data are well fitted by a straight line, which suggests that Kodak must have used the standard nonlinear response function $f(q) = \alpha + \beta q^\gamma$ in the design of its Photo CD scanner.

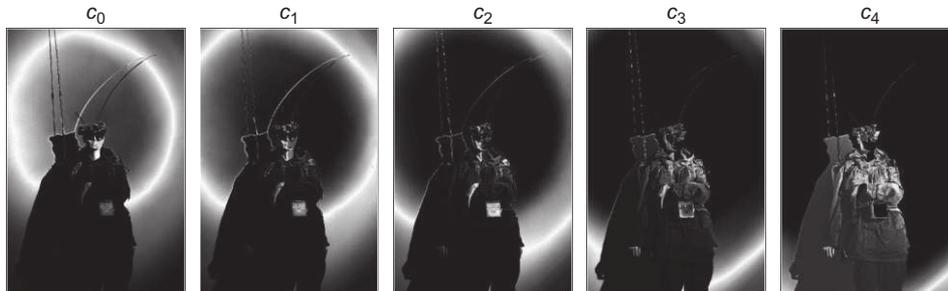


FIGURE 1.13

The *certainty functions* express the rate of change of $f(q(x, y))$ with $Q(x, y)$. The certainty functions may be used to compute the *certainty images*, $f(c_i)$. White areas in one of the certainty images indicate that pixel values $f(q)$ change fastest with a corresponding change in the photoquantity, Q . When the camera is used as a light meter (eg, a quantigraphic instrument to estimate q), it will be most sensitive where the certainty images are white. White areas of these certainty images correspond to mid-gray values (*midtones*) of the corresponding original images in Fig. 1.11, while dark areas correspond to extreme pixel values (either highlights or shadows) of the original images in Fig. 1.11. Black areas of the certainty image indicate that Q changes drastically with small changes in pixel value, and thus an estimate of Q in these areas will be overcome by image noise n_{f_i} .

1.5.5 COMPARAMETRIC REGRESSION TO THE PREFERRED MODEL

For this second example, the comparametric model proposed in Eq. (1.28) will be used.

In many practical situations, real-world images are very noisy. Accordingly, an example of noisy images that constitute a Wyckoff set (Fig. 1.14), in which an extremely poor scan was deliberately used to scan images from a publication (Mann and Picard, 1995b), is now considered.

That the images in Fig. 1.14 are of very poor quality is evidenced by their comparagrams (Fig. 1.15A). With use of regression of Eq. (1.28) to the comparagram combined with the knowledge (from the publication from which the images were obtained Mann and Picard, 1995b) that $K = 2$, it was found that $a = 0.0017$ and $c = -3.01$. These data provide a closed-form solution for the response function. The two effective response functions, which are shifted versions of this single response function, where the relative shift is K , are plotted in Fig. 1.16, together with their derivatives. (Recall that the derivatives of the response functions are the certainty functions.) Because a closed-form solution has been obtained, it may be easily differentiated without the further increase in noise that usually accompanies differentiation. Otherwise, when one is determining the certainty functions from poor estimates of f , the certainty functions would be even noisier than the poor estimate of f itself. The resulting certainty images, denoted by $c(f_i)$, are shown in Fig. 1.17. Each of the images, $f_i(x, y)$, gives rise to an actual estimate of the quantity of light arriving at the image sensor (1.9). These estimates were combined by way of Eq. (1.10), resulting in the composite image shown in Fig. 1.18.

The resulting image \hat{I}_1 looks very similar to f_1 , except that it is a floating-point image array, of much greater tonal range and image quality.

Furthermore, given a Wyckoff set, a composite image may be rendered at any in-between exposure from the set (exposure interpolation), as well as somewhat beyond the exposures given (exposure

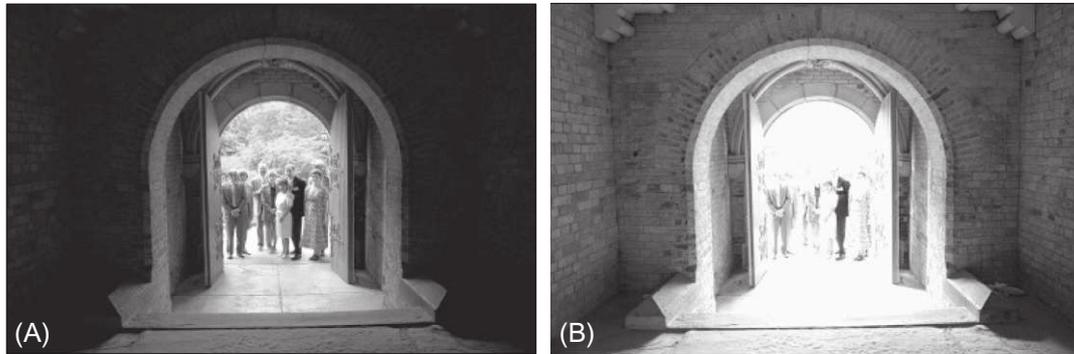


FIGURE 1.14

Noisy images badly scanned from a publication. These images are identical except for exposure and a good deal of quantization noise, additive noise, scanning noise, etc. Left: The darker image shows clearly the eight people standing outside the doorway, but shows little of the architectural details of the dimly lit interior. Right: The lighter image shows the architecture of the interior, but it is not possible to determine how many people are standing outside, let alone to recognize any of them.

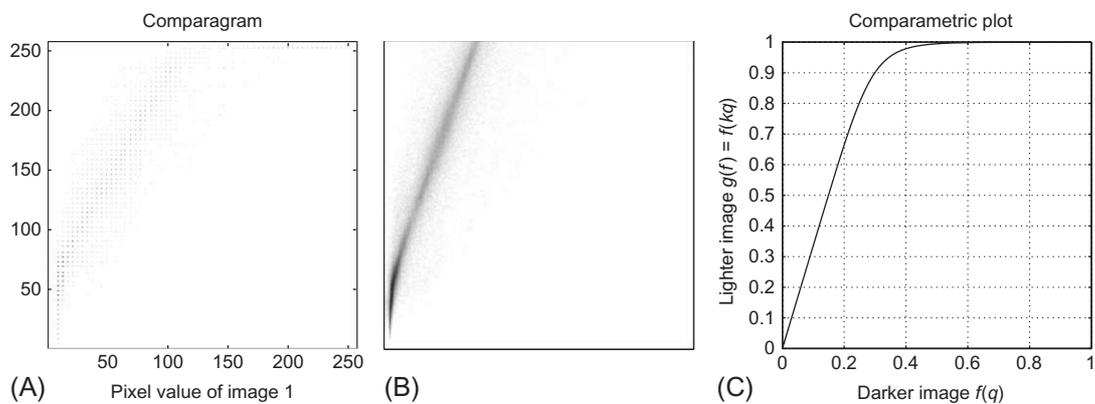


FIGURE 1.15

Comparagrammatic regression. (A) Comparagrams. Because the images were extremely noisy, the comparagram is spread out over a fat ridge. Note also the gaps in the comparagram owing to the poor quality of the scanning process. (B) Even the comparagram of the images before the deliberately poor scan of them is itself quite spread out, indicating the images were quite noisy to begin with. (C) Comparagrammatic regression is used to solve for the parameters of the comparagrammatic function. The resulting comparagrammatic plot is a noise-removed version of the comparagram (eg, provides a smoothly constrained comparagrammatic relationship between the two differently exposed images).

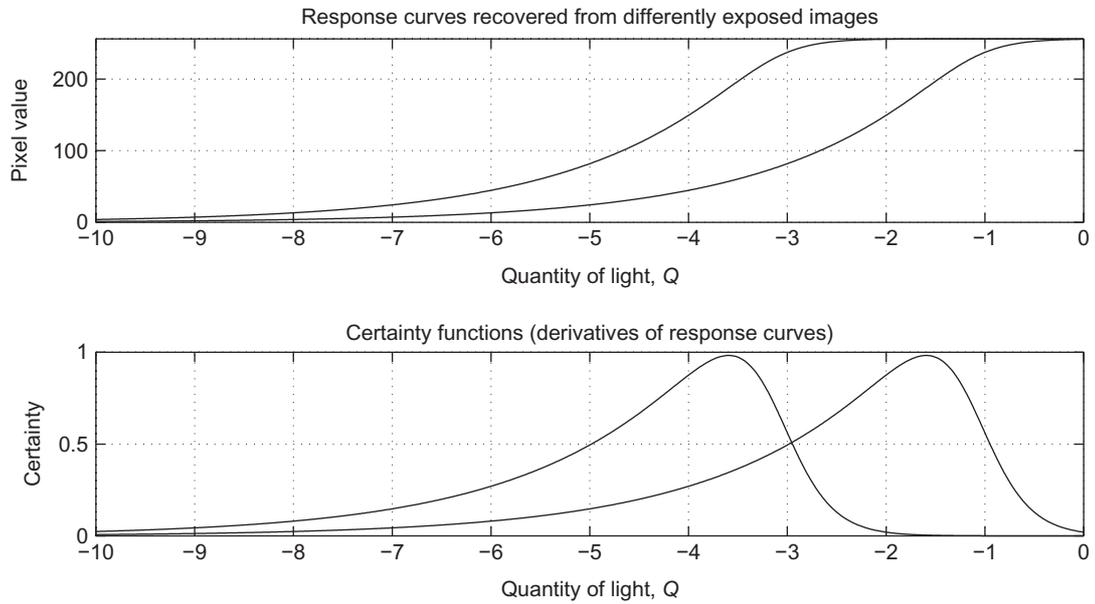


FIGURE 1.16

Relative response functions $F^{-1}(K_i; Q)$ recovered from the images in Fig. 1.14, plotted together with their derivatives. The derivatives of these response functions suggest a degree of confidence in the estimate $\hat{Q}_i = F(f_i) - K_i$ derived from each input image.

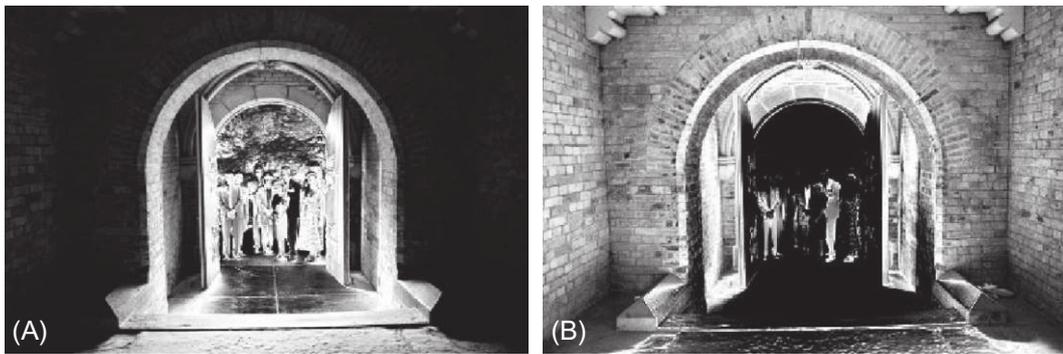


FIGURE 1.17

Certainty images which will be used as weights when the weighted sum of estimates of the actual quantity of light is computed. Bright areas correspond to large degrees of certainty.



FIGURE 1.18

Composite image made by simultaneous estimation of the unknown nonlinearity of the camera as well as the true quantity of light incident on the camera’s sensor array, given the two input images from Fig. 1.14. The combined optimal estimate of \hat{q} is expressed here, in the coordinates of the lighter (rightmost) image in Fig. 1.14. Although nothing has been done to appreciably enhance this image (eg, the procedure of estimating q and then just converting it back into a picture again may seem pointless), while the image appears much like the rightmost input image, the clipping of the highlight details has been softened somewhat.

From © Steve Mann, 1993.

extrapolation). This result suggests the “virtual camera” (Mann, 1999), which allows images to be rendered at any desired exposure, once q has been computed.

This capability is somewhat similar to QuickTime VR and other image-based rendering systems, except that it operates in the range of the images \hat{f}_i rather than their domain.

1.6 TONE MAPPING IN HDR SYSTEMS

Ordinarily, most print and display media have limited dynamic range. Thus, one might be tempted to argue against the utility of the Wyckoff principle based on this fact — for example, one might ask why, because televisions and print media cannot display more than a very limited dynamic range, we

should bother building a Wyckoff camera that can capture such dynamic ranges. Why should we bother capturing the photoquantity q with more accuracy than is needed for display?

Some possible answers to this question are as follows:

1. Estimates of q are still useful for machine vision and for other applications that do not involve direct viewing of a final picture. An example is the wearable face-recognizer (Mann, 1996b) which determines the identity of an individual from a plurality of differently exposed pictures of that person, and then presents the identity in the form of a text label (virtual name tag) on the retina of an eye of the wearer of the eyeglass-based apparatus. Because \hat{q} need not be displayed, the problem of output dynamic range, etc., of the display (eg, number of distinct intensity levels of the laser beam shining into the lens of the eye of the wearer) is of no consequence.
2. Even though the ordinary dynamic range and the range resolution (typically 8 bits) is sufficient for print media (given the deliberately introduced nonlinearities that best use the limited range resolution), when operations such as deblurring are performed, noise artifacts become more evident. In general, sharpening involves high-pass filtering, and thus sharpening will often tend to noise artifacts being uncovered that would normally exist below the perceptual threshold when viewed through ordinary display media. In particular, sharpening often uncovers noise in the shadow areas, making dark areas of the image appear noisy in the final print or display. Thus, in addition to the benefits of performing sharpening quantigraphically by applying an antihomomorphic filter as in Fig. 1.7 to undo the blur of Eq. (1.5), there is also further benefit from doing the generalized antihomomorphic filtering operation at the point \hat{q} in Fig. 1.8, rather than just that depicted in Fig. 1.7.
3. A third benefit of capturing more information than can be displayed is that it defers the choice of which information to get rid of. For example, a camera could be constructed in such a way that it had no exposure adjustments: neither automatic nor manual settings. Instead, the camera would be more like an array of light meters that would capture an array of light measurements. Decisions as to what subject matter is of importance could then be made at the time of viewing or the time of printing. Such a camera has been incorporated into eyeglasses (Mann, 1998), allowing the wearer to completely forget about the camera, with no need to worry about settings or adjustments. In this way the wearer can capture once-in-a-lifetime moments such as a baby's first steps, and worry about adjusting the exposure settings later. Exposure can then be adjusted in the peaceful quiet of the living room, long after the picture has been captured and the confusing excitement of the moment has passed. In this way exposure can be adjusted carefully in a quiet setting away from the busy and distracting action of everyday life. Because these decisions are made later, they can also be changed, as there is no need to commit to one particular exposure setting. Moreover, deferring exposure decisions may have forensic value. For example, ordinary everyday subject matter and scenes may later become crime scenes, such that pictures previously taken in those spaces may help solve a crime. A family photographing their child's first trip to a grocery store may inadvertently capture an image of a fire exit illegally chained shut in the background. A fatal fire at some time in the future might call for evidence against the owner of the shop, where deferred choices of exposure may assist in the production of a picture exposed optimally for the fire exit in the background rather than the child in the foreground. Because the wearable apparatus transmits images to the World Wide Web, various viewers can each adjust the image interactively to suit their own display and perceptual capabilities, as well as their own preferences. These capabilities are a major reason that digital SLR cameras provide RAW data formats suitable for postcapture processing.

4. A fourth benefit from capturing a true and accurate measurement of the photoquantity, even if all that is desired is a nice looking picture (eg, even if what is desired is not necessarily a true or accurate depiction of reality), is that additional processing may be done to produce a picture in which the limited dynamic range of the display or print medium shows a much greater dynamic range of input signal, through the use of further image processing on the photoquantity before display or printing.

It is this fourth benefit that will be further described, as well as illustrated through a very compelling example, in this section. Pursuit of this benefit, of spatially varying tonal adjustment, has created the field commonly referred to as “tone mapping,” which is a substantial field in the HDR area. Also see [Part II](#) of this book, where tone mapping and simulation of perceptual phenomena are discussed.

Ordinarily, humans cannot directly perceive the “signal” we process numerically, but, rather, we perceive the effects of the “signal” on perceptible media such as television screens or the like. In particular, to display $\hat{q}(x, y)$, it is typically converted into an image $f(\hat{q}(x, y))$ and displayed, for example, on a television screen.

[Fig. 1.18](#) is an attempt to display, on the printed page, a signal which contains much greater dynamic range than can be directly represented on the page. To do this, the estimate \hat{q} was converted into an image by evaluation of $\hat{f}(\hat{k}_2\hat{q})$. Even though we see some slight benefit over f_2 (one of the input images), the benefit has not been made fully visible in this print.

1.6.1 AN EXTREME EXAMPLE WITH SPATIOTONAL PROCESSING OF PHOTOQUANTITIES

To fully appreciate the benefits of quantigraphic image processing, let us consider a seemingly impossible scene to photograph reasonably (in a natural way without bringing in lighting equipment of any kind).

[Fig. 1.19](#) depicts a scene in which there is a dynamic range in excess of a $10^6:1$. In this case, two pictures were captured with several orders of magnitude difference between the two exposures. Thus, the quantigraphic estimate \hat{q} has far greater dynamic range than can be directly viewed on a television or on the printed page. Display of $\hat{f}(\hat{k}_1q)$ would fail to show the shadow details, while display of $\hat{f}(\hat{k}_2q)$ would fail to show the highlight details.

In this case, even if we use the virtual camera architecture depicted in [Fig. 1.9](#), there is no single value of display exposure k_d for which a display image $f_d = \hat{f}(k_d \hat{q})$ will capture both the inside of the abandoned fortress and the details looking outside through the open doorway.

Therefore, a strong high-pass (sharpening) filter, S , is applied to \hat{q} to sharpen the photoquantity \hat{q} , as well as to provide lateral inhibition similarly to the way in which the human eye functions. Then the filtered result, $\hat{f}\left(k_d S \hat{q} \left(\frac{\hat{A}_2 x + \hat{b}_2}{\hat{c}_2 x + \hat{d}_2}\right)\right)$, is displayed on the printed page ([Fig. 1.19C](#)) in the projective coordinates of the second ([Fig. 1.19B](#)) image, $i = 2$. Note the introduction of spatial coordinates \mathbf{A} , \mathbf{b} , \mathbf{c} , and d . These compensate for projection (eg, if the camera moves slightly between pictures), as described in [Mann \(1996a, 1998\)](#). In particular, the parameters of a projective coordinate transformation are typically estimated together with the nonlinear camera response function and the exposure ratio between pictures ([Mann, 1996a, 1998](#)); see also [Chapter 3](#).

As a result of the filtering operation, notice that there is no longer a monotonic relationship between input photoquantity q and output level on the printed page. Notice, for example, that the sail is as dark

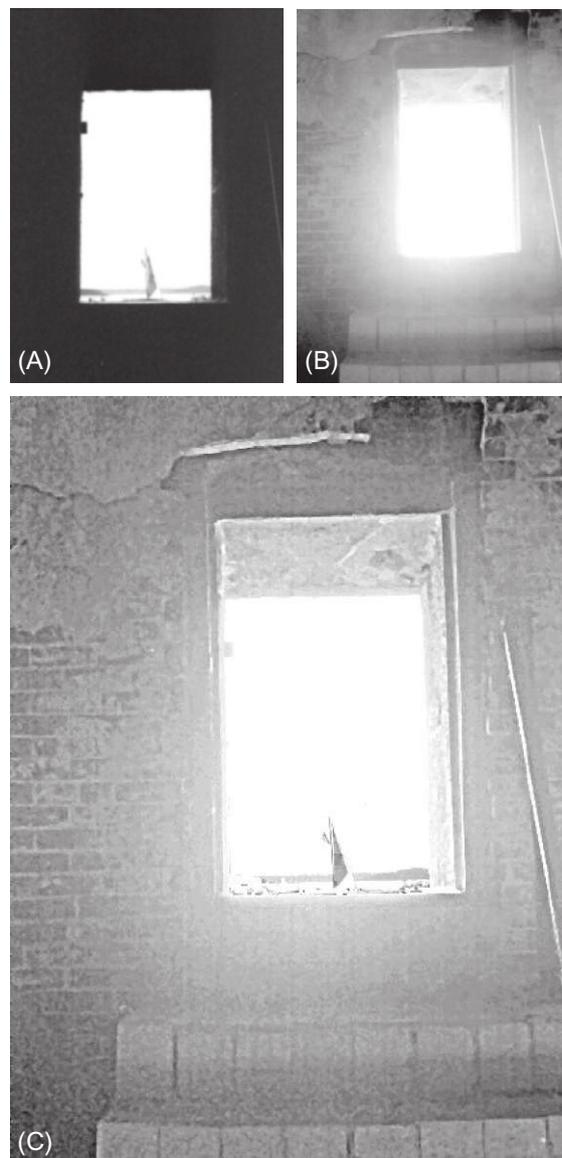


FIGURE 1.19

Extreme example to illustrate nonmonotonic processing. (A) An underexposed picture shows details such as the horizon and the sail of a boat, as seen through an open doorway, even though the sail is backlit with extremely bright light. (B) The picture is taken from inside an abandoned fortress with no interior lights. Light coming in from the open door is largely lost in the vastness of the dark interior, so a much longer exposure is needed to show any detail of the inside of the fortress. (C) Sharpened (spatiototally filtered) quantigraphic estimate $\hat{f}\left(k_d S \hat{q}\left(\frac{\hat{A}_2 \mathbf{x} + \hat{\mathbf{b}}_2}{\hat{c}_2 \mathbf{x}} + \hat{d}_2\right)\right)$ expressed in the projective coordinates of the second image in the image sequence (B). A dynamic range in excess of $10^6:1$ was captured in \hat{q} , and the estimate was then quantigraphically sharpened, with filter S , resulting in a lateral inhibition effect so that the output is no longer monotonically related to the input. Notice, for example, that the sail is as dark as some shadow areas inside the fortress. Because of this filtering, a tremendous dynamic range has been captured and reduced to that of printed media.

From © Steve Mann, 1991.

as some shadow areas inside the fortress. Because of this filtering, the dynamic range of the image may be reduced to that of printed media, while still revealing details of the scene. This example answers the question of why we should capture more dynamic range than we can display.

1.7 ANALYTICAL SOLUTION OF COMPARAMETRIC EQUATIONS

1.7.1 OVERVIEW

In this section, we demonstrate the connection between comparametric equations and the scaling operator arising in quantum field theory, and provide a general method of solution to the comparametric class of functional equations.

The development of the scaling operator here follows the standard formulation for operators based on generators of infinitesimal transformations. This theory can be found in any standard graduate-level introductory modern quantum mechanics text covering quantum field theory, such as Chapter 1 of [Sakurai \(1994\)](#) or Chapter 3 of [Ballentine \(1998\)](#).

1.7.2 FORMAL SOLUTION BY SCALING OPERATOR

In quantum field theory we have a scaling operator S_k (which is also referred to as the dilation operator D_k). We will derive an explicit expression for this operator. First consider an infinitesimally small scaling of the function f . Let ϵ be a very small number. We have, up to first order in ϵ ,

$$f((1 + \epsilon)q) \approx f(q) + \epsilon q \frac{\partial}{\partial q} f(q). \quad (1.32)$$

By repeated application of the infinitesimal scaling of q , we can scale q by any amount e^Λ

$$e^\Lambda q = \lim_{N \rightarrow \infty} \left(1 + \frac{\Lambda}{N}\right)^N q. \quad (1.33)$$

Application of Eq. (1.32) N times with $\epsilon = \frac{\Lambda}{N}$ for N large gives

$$f(e^\Lambda q) = \lim_{N \rightarrow \infty} \left(1 + \frac{\Lambda}{N} q \frac{\partial}{\partial q}\right)^N f(q), \quad (1.34)$$

$$= \exp\left(\Lambda q \frac{\partial}{\partial q}\right) f(q). \quad (1.35)$$

Choosing $\Lambda = \log(k)$, we scale q by k and thus define the scaling operator S_k as

$$S_k = \exp\left(\log k \times q \frac{\partial}{\partial q}\right). \quad (1.36)$$

The exponential here is defined as the formal series

$$\exp\left(\log k \times q \frac{\partial}{\partial q}\right) = \sum_{n=0}^{\infty} \frac{(\log k)^n}{n!} \left(q \frac{\partial}{\partial q}\right)^n. \quad (1.37)$$

Each differential operator in this series acts on every term to the right of it. The inverse of the scaling operator is then

$$(S_k)^{-1} = S_{1/k} = \exp\left(-\log k \times q \frac{\partial}{\partial q}\right). \quad (1.38)$$

Now, given $g(k, f)$, we can write $f(q)$ formally as

$$f(q) = S_{1/k} g(k, f(q)). \quad (1.39)$$

Although this is not a convenient formulation for explicit computation of $f(q)$, it opens the possibility for further analysis of the general comparametric problem using the machinery of the well-known operators arising in quantum field theory. Because Eq. (1.39) holds for any value of k , we can take k close to 1 (ie, $k = 1 + \epsilon$ for $\epsilon \approx 0$) and we see that

$$f(q) = \exp\left(-\log(1 + \epsilon) q \frac{\partial}{\partial q}\right) g(1 + \epsilon, f). \quad (1.40)$$

If we expand this in ϵ up to first order, noting that higher-order terms vanish, we find

$$f(q) = \left(1 - \epsilon q \frac{\partial}{\partial q}\right) \left(g(1, f) + \epsilon \frac{\partial g(k, f)}{\partial k}\right) \Big|_{k=1}. \quad (1.41)$$

Using the identity $f = g(1, f)$, we end up with an ordinary differential equation,

$$\frac{df}{dq} = \frac{1}{q} \frac{\partial g(k, f)}{\partial k} \Big|_{k=1}, \quad (1.42)$$

from which we can obtain $f(q)$. This equation is always separable, and the solution family always contains at least one valid solution when $g(f)$ is monotonic and smooth. This analytical form provides the benefit that any arbitrary camera response function can be solved exactly, and that the behavior of noise terms can be modeled as shown in [Section 1.8](#).

1.7.3 SOLUTION BY ORDINARY DIFFERENTIAL EQUATION

The machinery of the previous section allows us to proceed directly, merely using the result as a recipe to solve any given analytical comparametric equations. To begin, we examine the comparametric equation given by

$$f(kq) = g(k, f(q)). \quad (1.43)$$

Consider two cases. In the first case, the function f is known. Then g is easily found. For example, consider the classical model

$$f(q) = \alpha + \beta q^\gamma. \quad (1.44)$$

Then,

$$f(kq) = \alpha + \beta k^\gamma q^\gamma. \quad (1.45)$$

From Eq. (1.44) it follows that

$$q = \left(\frac{f - \alpha}{\beta} \right)^{1/\gamma}. \quad (1.46)$$

Substituting this in Eq. (1.45), we find

$$g(k, f) = \alpha + \beta k^\gamma \left(\frac{f - \alpha}{\beta} \right). \quad (1.47)$$

In the second, more difficult case, $g(k, f)$ is given and we have to find $f(q)$. This is actually solving the comparometric equation. The way to do this is as follows. By partial differentiation of Eq. (1.47) by k , and substituting $k = 1$, we find

$$\begin{aligned} \left. \frac{\partial f(kq)}{\partial k} \right|_{k=1} &= qf'(kq)|_{k=1} = qf'(q) \\ &= \left. \frac{\partial g(k, f)}{\partial k} \right|_{k=1}, \end{aligned} \quad (1.48)$$

where f' is the derivative of f . So f satisfies the following ordinary differential equation:

$$\frac{df}{dq} = \frac{1}{q} \left. \frac{\partial g(k, f)}{\partial k} \right|_{k=1}, \quad (1.49)$$

which we derived in the previous section using the scaling operator. This differential equation is easily solved because it is always separable. For example, take

$$g(k, f) = \alpha (1 - k^\gamma) + k^\gamma f. \quad (1.50)$$

We have

$$\begin{aligned} \left. \frac{\partial g(k, f)}{\partial k} \right|_{k=1} &= ((f - \alpha) \gamma k^{\gamma-1})|_{k=1} \\ &= (f - \alpha) \gamma. \end{aligned} \quad (1.51)$$

Now f satisfies

$$\frac{df(q)}{dq} = \frac{1}{q} (f - \alpha) \gamma. \quad (1.52)$$

By separating the variables, integrating, and taking the exponential of both sides, we obtain

$$f(q) = \beta q^\gamma + \alpha, \quad (1.53)$$

where β appears as a constant of integration.

1.8 COMPOSITING AS BAYESIAN JOINT ESTIMATION

Up to this point in the discussion, we have not characterized the noise terms inherent in the imaging process, nor have we used them to improve the estimation of the light quantity present in the original scene. In this section we develop a technique to incorporate a noise model into the estimation process.

Our approach for creating an HDR image from N input LDR images begins with the construction of a notional N -dimensional inverse camera response function that incorporates the different exposure and weighting values between the input images. Then we could use this to estimate the photoquantity \hat{q} at each point by writing $\hat{q}(\mathbf{x}) = f^{-1}(f_1, f_2, \dots, f_N)/k_1$. In this case f^{-1} is a joint estimator that could potentially be implemented as an N -dimensional lookup table (LUT). Recognizing the impracticality of this for large N , we consider pairwise recursive estimation for larger N values in the next section. The joint estimator $f^{-1}(f_1, f_2, \dots, f_N)$ may be referred to more precisely as a *comparametric inverse camera response function* because it always has the domain of a comparagram and the range of the inverse of the response function of the camera under consideration.

Pairwise estimation

Assume we have N LDR images that are a constant change in exposure value apart, so that $\Delta_{EV} = \log_2 k_{i+1} - \log_2 k_i$ is a positive constant $\forall i \in \{1, \dots, N - 1\}$, where k_i is the exposure of the i th image. Now consider specializing to the case $N = 2$ so we have two exposures, one at $k_1 = 1$ (without loss of generality, because exposures have meaning only in proportion to one another) and the other at $k_2 = k$. Our estimate of the photoquantity may then be written as $\hat{q}(\mathbf{x}) = f_{\Delta_{EV}}^{-1}(f_1, f_2)$, where $\Delta_{EV} = \log_2 k$.

To apply this pairwise estimator to three input LDR images, each with a constant difference in exposure between them, we can proceed by writing

$$f(\hat{q}) = f(f_{\Delta_{EV}}^{-1}(f(f_{\Delta_{EV}}^{-1}(f_1, f_2)), f(f_{\Delta_{EV}}^{-1}(f_2, f_3)))). \tag{1.54}$$

In this expression, we first estimate the photoquantity based on images 1 and 2, and then the photoquantity based on images 2 and 3, then we combine these estimates using the same joint estimator, by first putting each of the earlier round (or “level”) of estimates through a virtual camera f , which is the camera response function.

This process may be expanded to any number N of input LDR images, by use of the recursive relation

$$f_i^{(j+1)} = f(f_{\Delta_{EV}}^{-1}(f_i^{(j)}, f_{i+1}^{(j)})),$$

where $j = 1, \dots, N - 1$, $i = 1, \dots, N - j$, and $f_1^{(N)}$ is the final output image, and in the base case, $f_i^{(1)}$ is the i th input image. This recursive process may be understood graphically as in Fig. 1.20. This process forms a graph with estimates of photoquantities as the nodes, and comparametric mappings between the nodes as the edges. A single estimation step using a CCRF is illustrated in Fig. 1.21.

For efficient implementation, rather than computing at runtime or storing values of $f^{-1}(f_1, f_2)$, we can store $f(f^{-1}(f_1, f_2))$. We call this the *comparametric camera response function* (CCRF). It is the comparametric inverse camera response function evaluated via (or “imaged” through, because we are in effect using a virtual camera) the camera response function f . This means at runtime we require $N(N - 1)/2$ recursive lookups, and we can perform all pairwise comparisons at each level in parallel, where a level is a row in Fig. 1.20.

The reason we can use the same CCRF throughout is because each virtual comparametric camera $f \circ f^{-1}$ returns an exposure that is at the same exposure point as the less exposed of the two input images (recall that we set $k_1 = 1$), so Δ_{EV} between images remains constant at each subsequent level.

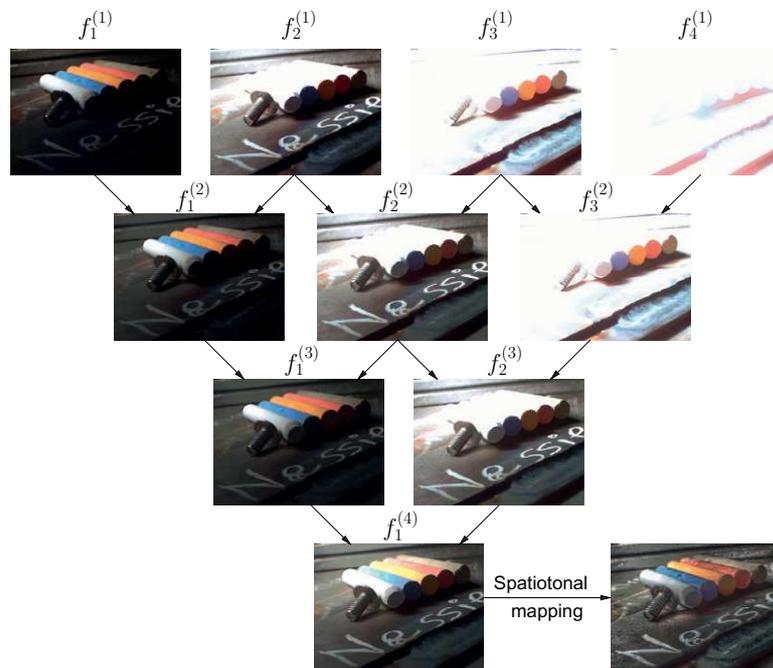


FIGURE 1.20

Graph structure of pairwise comparametric image compositing. The HDR image $f_1^{(4)}$ is composited from the LDR source camera images $f_{1...4}^{(1)}$. Nodes $f_i^{(j>1)}$ are rendered here by the scaling and rounding of the output from the *comparametric camera response function* (CCRF). To illustrate the details captured in the highlights and lowlights in the LDR medium of this paper, we include a spatiotonally mapped LDR rendering of $f_1^{(4)}$.

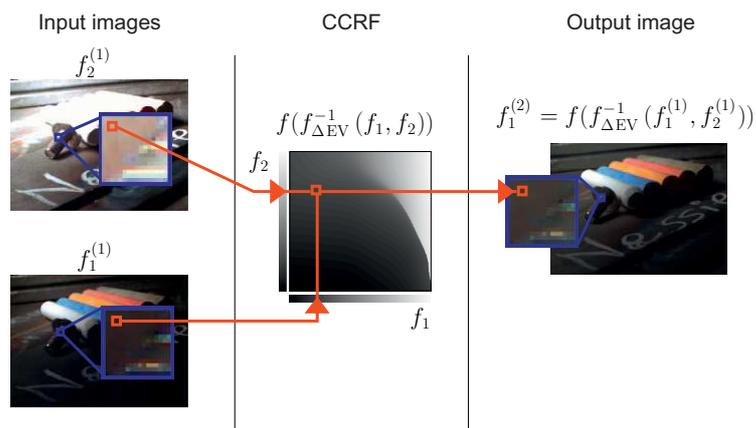


FIGURE 1.21

CCRF-based compositing of a single pixel. The floating-point tonal values f_1 and f_2 are the arguments to the CCRF $f \circ f_{\Delta EV}^{-1}$, which returns a refined estimate of an ideal camera response to the scene being photographed. This virtual camera's exposure setting is equal to the exposure of the lower-exposure image f_1 .

In comparison with Fig. 3 in Mann and Mann (2001), wherein the objective is to recover the camera response function and its inverse, in this case we are using a similar hierarchical structure to instead combine information from multiple source images to create a single composite image.

The memory required to store the entire pyramid, including the source images, is $N(N + 1)$ times the amount of memory needed to store a single uncompressed source image with floating-point pixels. Multichannel estimation (eg, for color images) can be done by use of separate response functions for each channel, at a cost in compute operations and memory storage that is proportional to the number of channels.

Alternative graph topology

Other connection topologies are possible — for example, we can trade memory usage for speed by compositing using the following form for the case $N = 4$:

$$f(\hat{q}) = f(f_{2\Delta EV}^{-1}(f(f_{\Delta EV}^{-1}(f_1, f_2)), f(f_{\Delta EV}^{-1}(f_3, f_4))))),$$

in which case we perform only three lookups at runtime, instead of six with the previous structure. However, we must store twice as much lookup information in memory: for $f \circ f_{\Delta EV}^{-1}$ as before, and for $f \circ f_{2\Delta EV}^{-1}$, because the results of the inner expressions are no longer ΔEV apart, but instead are twice as far apart in exposure value, $2\Delta EV$, as shown in Fig. 1.22. As a recursive relation for $N = 2^n, n \in \mathbb{N}$ we have

$$f_i^{(j+1)} = f(f_{j\Delta EV}^{-1}(f_{2i-1}^{(j)}, f_{2i}^{(j)})),$$

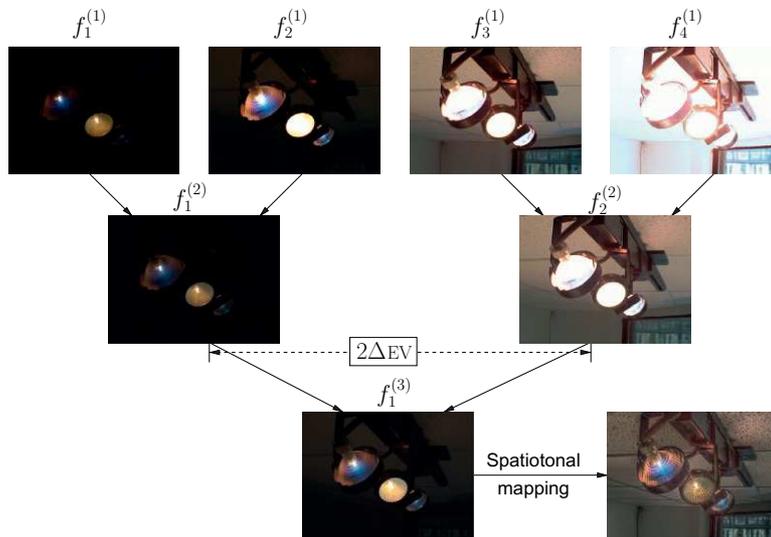


FIGURE 1.22

Example of an alternative graph structure for pairwise comparametric image compositing.

where $j = 1, \dots, \log_2 N$ and $i = 1, \dots, N/2^{j-1}$. The final output image is $f_1^{(\log_2 N + 1)}$, and $f_i^{(1)}$ is the i th input image. This form requires $N - 1$ lookups. In general, by combining this approach with the previous graph structure, we can see that comparometric image composition can always be done in $O(N)$ lookups $\forall N \in \mathbb{N}$.

The alternative form described so far is only a single option of many possible configurations. For example, use of multiple LUTs provides less locality of reference, causing cache misses in the memory hierarchy, and uses more memory. In a memory-constrained environment, or one in which memory access is slow, we could use a single LUT while still using this alternate topology, as in

$$\hat{q}(\mathbf{x}) = f_{\Delta\text{EV}}^{-1}(f(f_{\Delta\text{EV}}^{-1}(f_1, f_2)/2), f(f_{\Delta\text{EV}}^{-1}(f_3, f_4)/2)). \quad (1.55)$$

This comes at the expense of our performing more arithmetic operations per comparometric lookup.

Constructing the CCRF

To create a CCRF $f \circ f^{-1}(f_1, f_2, \dots, f_N)$, the ingredients required are a camera response function $f(q)$ and an algorithm for creating an estimate \hat{q} of photoquantity by combining multiple measurements. Once these have been selected, $f \circ f^{-1}$ is the camera response evaluated at the output of the joint estimator, and is a function of two or more tonal inputs f_i .

To create a LUT means sampling through the possible tonal values, so, for example, to create a 1024×1024 LUT we could execute our \hat{q} estimation algorithm for all combinations of $f_1, f_2 \in \{0, \frac{1}{1023}, \frac{2}{1023}, \dots, 1\}$ and store the result of $f(\hat{q})$ in a matrix indexed by $[1023f_1, 1023f_2]$, assuming zero-based array indexing. Intermediate values may be estimated by linear or other interpolation.

Incremental updates

In the common situation that there is a single camera capturing images in sequence, we can easily perform updates of the final composited image incrementally, using partial updates, by only updating the buffers dependent on the new input.

1.8.1 EXAMPLE JOINT ESTIMATOR

In this section we describe a simple joint photoquantity estimator, using nonlinear optimization to compute a CCRF. This method executes in real time for HDR video, using pairwise comparometric image compositing. Examples of the results of this estimator can be seen in [Figs. 1.20](#) and [1.22](#).

1.8.1.1 Bayesian probabilistic model for the CCRF

In this section we propose a simple method for estimating a CCRF. First, we select a comparometric model, which determines the analytical form of the camera response function. As an example, we illustrate our compositing approach using the “preferred saturation” camera model ([Mann, 2001](#)), for which an analytical is known and can be verified by use of the approach in the previous section.

The next step is to determine the model parameters, as in [Fig. 1.23](#); however, any camera model with good empirical fit may be used with this method.

Let scalars f_1 and f_2 form a Wyckoff set from a camera with zero-mean Gaussian noise, and let random variables $X_i = f_i - f(k_i q)$, $i \in \{1, 2\}$ be the difference between the observation and the model, with $k_1 = 1$ and $k_2 = k$.

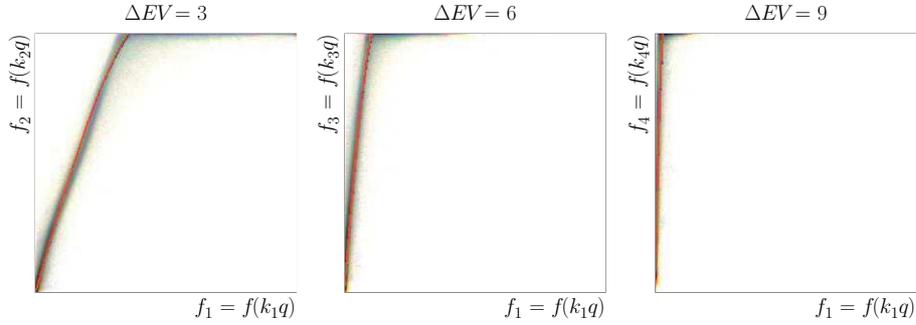


FIGURE 1.23

Comparametric model fitting. Preferred saturation model parameters were found via nonlinear optimization, by the method of least squares with the Levenberg-Marquardt algorithm. The optimal comparametric model function, determined per color channel, is plotted directly on empirical comparasums to verify a good fit. Comparasums are sums of comparagrams from the same sensor with the same difference in exposure value ΔEV . They are shown range compressed with the log function, and color inverted, to show finer variation. The best results for comparametric compositing are found when the camera response function model parameters are optimized against a range of k values. Here $k_1 = 1$ and $k_2 = 8$, $k_3 = 64$, and $k_4 = 512$, which implies that for comparametric image compositing we would use $\Delta EV = 3$.

We can estimate the variances of X_i can be estimated between exposures by calculating the *interquartile range* along each column (for X_1) and row (for X_2) of the comparagram with the ΔEV of interest (ie, using the “fatness” of the comparagram). A robust statistical formula, based on the quartiles of the normal distribution, gives $\hat{\sigma} \approx \text{interquartile range}/1.349$, which can be stored in two one-dimensional vectors.

Discontinuities in $\hat{\sigma}_{X_i}$ with respect to f_i can be mitigated by Gaussian blurring of the sample statistics, as shown in Fig. 1.24. Using interpolation between samples of the standard deviation, and extrapolation beyond the first and last samples, we can estimate for any value of f_1 or f_2 the corresponding constant σ_{X_1} or σ_{X_2} .

The probability of \hat{q} , given f_1 and f_2 , is

$$\begin{aligned}
 P(q = \hat{q}|f_1, f_2) &= \frac{P(q)P(f_1|q, f_2)P(f_2|q)}{P(f_1, f_2)} \\
 &= \frac{P(q)P(f_1|q)P(f_2|q)}{P(f_1|f_2)} \\
 &= \frac{P(q)P(f_1|q)P(f_2|q)}{\int_0^\infty P(f_1|q)P(f_2|q) dq} \\
 &\propto P(q = \hat{q})P(f_1|q)P(f_2|q).
 \end{aligned}$$

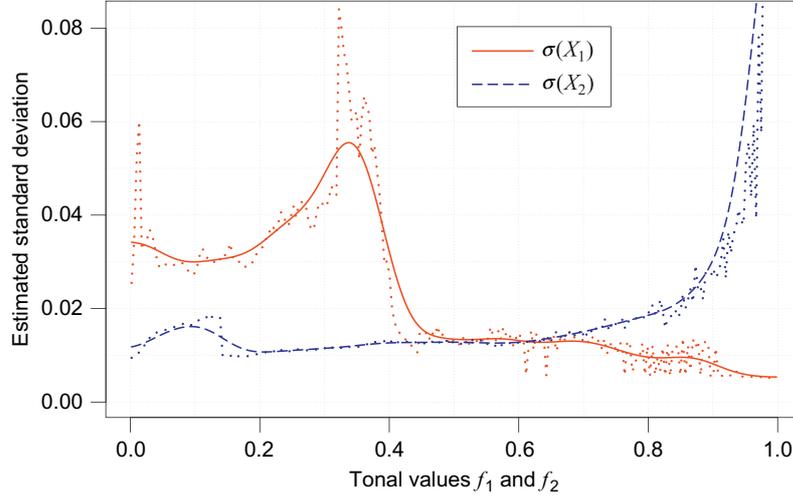


FIGURE 1.24

Trace plot of estimated standard deviations from a comparagram. Each estimate is proportional to the IQR, calculated from each column f_1 and row f_2 of a comparagram; here with $\Delta EV = 3$ as given in Fig. 1.23. Gaussian smoothing is applied to reduce discontinuities due to edge effects, quantization, and other noise.

For simplicity, we choose a uniform prior, which gives us $P_{\text{prior}}(q = \hat{q}) = \text{constant}$. Using X_i , we have

$$\begin{aligned} P_{\text{model}}(f_i|q) &= \text{normal}(\mu_{X_i} = 0, \sigma_{X_i}^2) \\ &= \frac{1}{\sqrt{2\pi}\sigma_{X_i}} \exp\left(-\frac{(f_i - f(k_iq))^2}{2\sigma_{X_i}^2}\right). \end{aligned}$$

To maximize $P(q = \hat{q}|f_1, f_2)$ with respect to q , we remove constant factors and equivalently minimize $-\log(P)$. Then the optimal value of q , given f_1 and f_2 , is

$$q = \underset{q}{\text{argmin}} \left(\frac{(f_1 - f(q))^2}{\sigma_{X_1}^2} + \frac{(f_2 - f(kq))^2}{\sigma_{X_2}^2} \right).$$

In practice, good estimates of optimal q values can be found with use of, for example, the Levenberg-Marquardt algorithm.

1.8.2 DISCUSSION REGARDING COMPOSITING VIA THE CCRF

Use of direct computation for nonlinear iterative methods as in Pal et al. (2004) is not feasible for real-time HDR video, because the time required to converge to a solution on a per-pixel basis is too long.

Table 1.3 Performance of Pairwise Composition Versus Direct Calculation of a Composite HDR Image on Four Input LDR Images

Platform	Method			Speedup
	Direct Calculation	CCRF, Full Update	CCRF, Incremental	
	Speed (output frames per second)			
CPU (serial)	0.0154	51	78	5065 times
CPU (threaded)	0.103	191	265	2573 times
GPU	–	272	398	–

For our simplistic probabilistic model given in [Section 1.8.1](#), it takes more than 1 min (approximately 65 s) to compute each output frame with a single processor. With the method proposed in [Section 1.8](#), the multicore speedup is more than 2500 times for CPU-based computation, and 3800 times for graphics processing unit (GPU)-based computation (versus CPU), as shown in [Table 1.3](#).

The selection of the size of the LUT depends on the range of exposures for which it is used. It was found empirically that 1024×1024 samples of a CCRF is enough for the practical dynamic range of our typical setups. Further increases in the size of the LUT resulted in no noticeable improvement in output video quality.

Because GPUs implement floating-point texture lookup with linear interpolation in hardware, and can execute highly parallelized code, the GPU execution would seem to be a natural application of general-purpose GPU computation. However, for this application much of the time is spent waiting for data transfer between the host and the GPU — the pairwise partial update is useful in this context because we can reuse partial results from the previous estimate, and transfer only the new data.

1.8.3 REVIEW OF ANALYTICAL COMPARAMETRIC EQUATIONS

In this section we have developed the general solution to any comparametric problem. This solution converts the comparametric system from a functional equation to a separable ordinary differential equation. The solution to this differential equation can then be used to perform image compositing based on photoquantity estimation.

We have further illustrated comparametric compositing as a novel computational method that uses multidimensional LUTs, recursively if necessary, to estimate HDR output from LDR inputs. The runtime cost is fixed irrespective of the algorithm implemented if it can be expressed as a comparametric lookup. Pairwise estimation decouples the specific compositing algorithm from runtime, enabling a flexible architecture for real-time applications, such as HDR video, that require fast computation. Our experiments show that we approach a data transfer barrier rather than a compute-time limit. We demonstrated a speedup of three orders of magnitude for nonlinear optimization-based photoquantity estimation.

1.9 EFFICIENT IMPLEMENTATION OF HDR RECONSTRUCTION VIA CCRF COMPRESSION

High-quality HDR typically requires large computational and memory requirements. We seek to provide a system for efficient real-time computation of HDR reconstruction from multiple LDR samples, using very limited memory. The result translates into hardware such as field-programmable gate array (FPGA) chips that can fit within eyeglass frames or other miniature devices.

The CCRF enables an HDR compositing method that performs a pairwise estimate by taking two pixel values p_i and p_j at an exposure difference of ΔEV and outputs \hat{q} , the photoquantigraphic quantity (Ali and Mann, 2012). The CCRF results can be stored in a LUT. The LUT contains $N \times N$ elements of precomputed CCRF results on pairs of discretized p_i and p_j values. The benefit of the use of a CCRF is that the final estimate of photoquantity can incorporate multiple samples at different exposures to determine a single estimate of the photoquantity at each pixel.

Quadtree representation

The CCRF LUT can be represented in a tree structure. For $N \times N$ elements, we can generate a quadtree (a parent node in a such tree contains four child nodes) to fully represent the CCRF LUT. Such a quadtree is a complete tree with $\log_4 N^2$ or $\log_2 N$ levels.

One method of generating such a tree is to recursively divide a unit square into four quadrants (four smaller but equally sized squares). We can visualize the center of a divided unit square as the parent node of the four quadrants. The center of each quadrant is considered a child node. Such a process is performed recursively in each quadrant until the root unit square is divided into $N \times N$ equally sized squares. The bottom nodes of the quadtree are the leaves of the tree, each of which stores the CCRF lookup value of the corresponding pixel pair (p_i, p_j) .

Reducing the quadtree

Storing the leaves of the complete quadtree representation of the CCRF costs as much space as the CCRF LUT itself. To reduce the number of elements needed for storage, we proceed to interpolate the CCRF value $\hat{p}_{i,j}$ of a specific pair (p_i, p_j) on the basis of its neighbor CCRF lookups. The value $\hat{p}_{i,j}$ interpolated on the basis of its neighbors is compared against the actual CCRF lookup value $p_{i,j}$, which gives $e_{i,j}$, the error per lookup entry:

$$e_{i,j} = |\hat{p}_{i,j} - p_{i,j}|. \quad (1.56)$$

We accept the approximated result if $e_{i,j}$ is within a fraction of $\frac{1}{2^D}$ as the error threshold e_{th} :

$$e_{th} = \alpha \cdot \frac{1}{2^D}, \quad (1.57)$$

where α is the fraction constant and D is the bit depth of the pixel value. We define the neighbor CCRF points as the four corner CCRF points of the square. We interpolate all CCRF (p_i, p_j) within the same square on the basis of these four corner values. We denote $I(e_{i,j}, e_{th})$ as the indicator function of unsatisfied error condition:

$$I(e_{i,j}, e_{th}) = \begin{cases} 1, & \text{if } e_{i,j} \geq e_{th}, \\ 0, & \text{otherwise,} \end{cases} \quad (1.58)$$

We divide the square into four quadrants if

$$\sum I(e_{i,j}, e_{\text{th}}) > 0, \forall p_{i,j} \in \text{square}. \quad (1.59)$$

The purpose of the division is to obtain new corner values that are closer to the point. The closer corner values to the point for interpolation may yield lower $e_{i,j}$ as the CCRF LUT generally varies smoothly over a continuous and wide range of p_i and p_j values. Therefore, we expect that the density of the divisions corresponds to the local gradient of the CCRF LUT: the higher the local gradient, the more recursive divisions are required to bring corners closer to the point, whereas the points formed of a large and smooth region of the CCRF with low local gradient share the same corners.

Error weighting and tree depth criteria

The error of the interpolation against the original lookup value of the input pair (p_i, p_j) should be within e_{th} . This error is affected by the interpolation method. Empirically, we find that bilinear interpolation works better than quartic interpolation in terms of minimizing the number of lookup points while satisfying the error constraint.

Statistically, we observe that the most frequently accessed CCRF values lie along the comparagram, as shown in Fig. 1.25. Therefore, higher precision on interpolation may not be necessary for CCRF lookup points that are distant from the comparagram. This suggests that the error constraint for the pair (p_i, p_j) should vary depending on its likelihood of occurrence. To further compress the CCRF LUT, we can scale $e_{i,j}$ by the number of observed occurrences of the pair (p_i, p_j) . This information is obtained through the construction of the comparagram. For each entry of the CCRF lookup, we weight the interpolation error by directly multiplying it by the occurrence count observed on the comparagram:

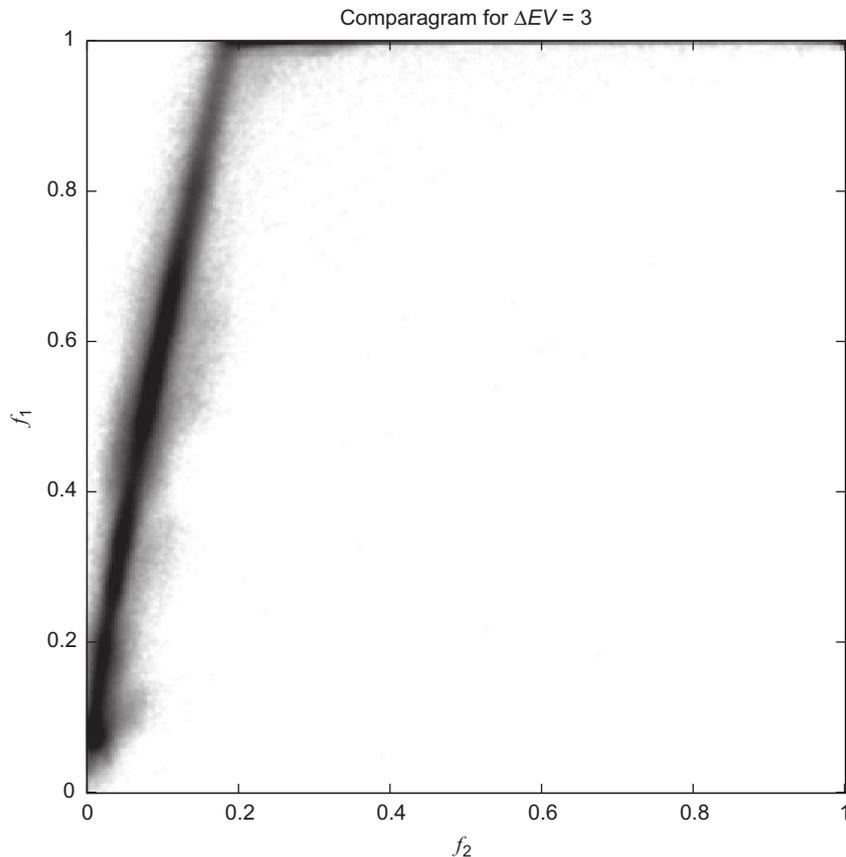
$$e_{i,j} \cdot (B_{i,j} + 1), \quad (1.60)$$

where $B_{i,j}$ is the count of the number of occurrences on the comparagram entry of (p_i, p_j) . The result with the weighted errors also favors bilinear interpolation over quartic interpolation in terms of minimizing the number of entries of the CCRF for storage, as shown in Fig. 1.26. Reconstruction of the original CCRF from this quadtree is shown in Fig. 1.27, with the corresponding error shown in Fig. 1.28.

The CCRF LUT we use has 1024 entries for p_i and 1024 entries for p_j . Therefore, there is no point in constructing a tree that has a depth of more than $\log_2 1024 = 10$. We may constrain the depth of the tree to fewer than 10 levels as long as the error constraint is met. This affects the resulting number of entries in the CCRF quadtree, as well as the number of iterations required to find a leaf node.

Corner value access

Each node of the quadtree is the center point of the square that contains it. To access the corner values of a leaf node, we can perform a recursive comparison of pair (p_i, p_j) with (p_x, p_y) of the nonleaf nodes in the tree until a leaf node has been reached, seen in Algorithm 1. The leaf nodes contain memory addresses of the corresponding corner values. The corner values of a leaf node are stored in the memory for retrieval.

**FIGURE 1.25**

A comparagram is a joint histogram of pixel values from images of the same scene taken with different exposures. For any given sensor, the comparagram is directly related to the camera response function. Areas that are dark in this plot correspond to joint values that are expected to occur in practice.

1.9.1 IMPLEMENTATION

The algorithm can be implemented efficiently on a medium-sized FPGA. Given a finalized quadtree data structure, a system can be generated with software. The four corner values are stored in ROMs implemented with on-chip block RAM, and then selected by multiplexer chains based on the inputs f_1 and f_2 . An arithmetic circuit that follows can then calculate the result on the basis of bilinear interpolation. Thus, as shown in Fig. 1.29, the system consists of two major parts: an addressing circuit and an interpolation circuit.

Because the size of the quadtree can grow to 10 levels, a C program is written to generate the implementation of the two circuits in Verilog hardware description language. Given a compressed

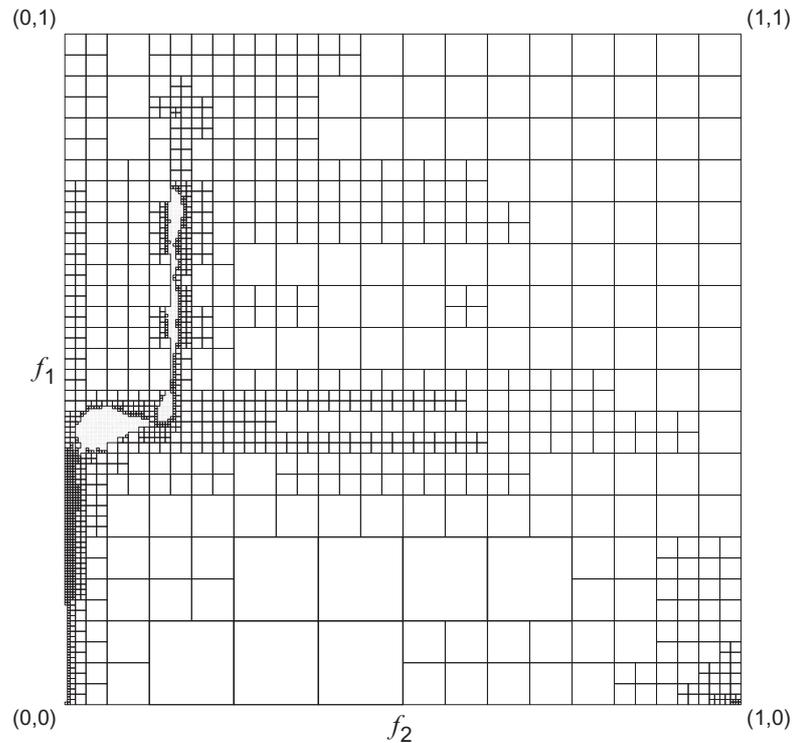


FIGURE 1.26

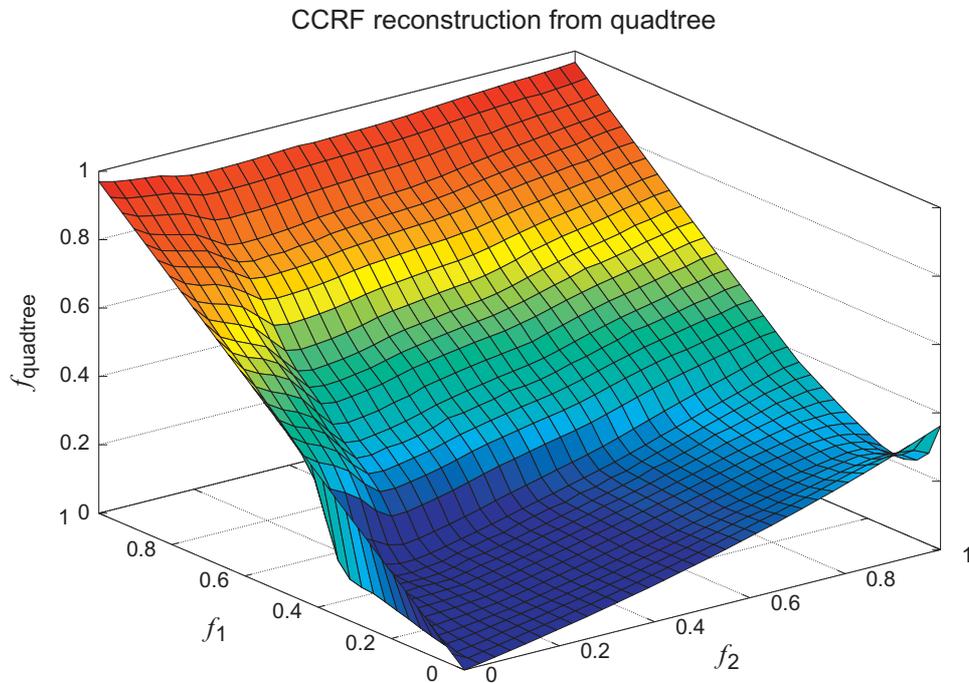
Quadtree-based representation of the CCRF based on weighting from the comparagram shown in Fig. 1.25. Areas of rapid change or high use are more finely subdivided for greater accuracy. Inside each square the CCRF value is approximated by bilinear interpolation based on the corner values.

quadtree data structure, this program generates four ROM initialization files and a circuit that retrieves corner values stored in the ROMs based on the inputs.

1.9.1.1 Addressing circuit

Each of the quadtree leaves needs a unique address. This address is used to retrieve the corresponding corner values from ROMs. As shown in Fig. 1.30, the circuit outputs the address by comparing f_1 and f_2 with constant boundary values, in the same way as we traverse the quadtree. The main function of the boundary comparator is to send the controlling signal to traverse the multiplexer tree, on the basis of the given input pair. At each level, it compares the input values with the prestored center coordinate values, and determines which branch (if exists) it should take next. Otherwise, the current node is a quadtree leaf and a valid address will be selected.

The algorithm generates the circuit by traversing the quadtree. Because a new unique address is needed for every leaf being visited, a global counter is used to determine the addresses. The width of the circuit data path is then determined with the last address generated (ie, the maximum address).

**FIGURE 1.27**

Reconstruction of the CCRF LUT based on a compressed quadtree with an error constraint of within one pixel value (α set to 1.0). The error bound is weighted by the expected usage, so values used more often have a smaller error bound.

Algorithm 1 Recursive quadtree search

```

procedure GET CORNERS( $p_i, p_j, \text{Node}$ )
  if Node is not a leaf then
    if  $p_i < \text{Node} \rightarrow p_x$  then
      if  $p_j < \text{Node} \rightarrow p_y$  then
        get corners( $p_i, p_j, \text{Node} \rightarrow \text{Child}(\text{left}, \text{down})$ )
      else
        get corners( $p_i, p_j, \text{Node} \rightarrow \text{Child}(\text{left}, \text{up})$ )
    else
      if  $p_j < \text{Node} \rightarrow p_y$  then
        get corners( $p_i, p_j, \text{Node} \rightarrow \text{Child}(\text{right}, \text{down})$ )
      else
        get corners( $p_i, p_j, \text{Node} \rightarrow \text{Child}(\text{right}, \text{up})$ )
  else
    retrieve corner values
  end procedure

```

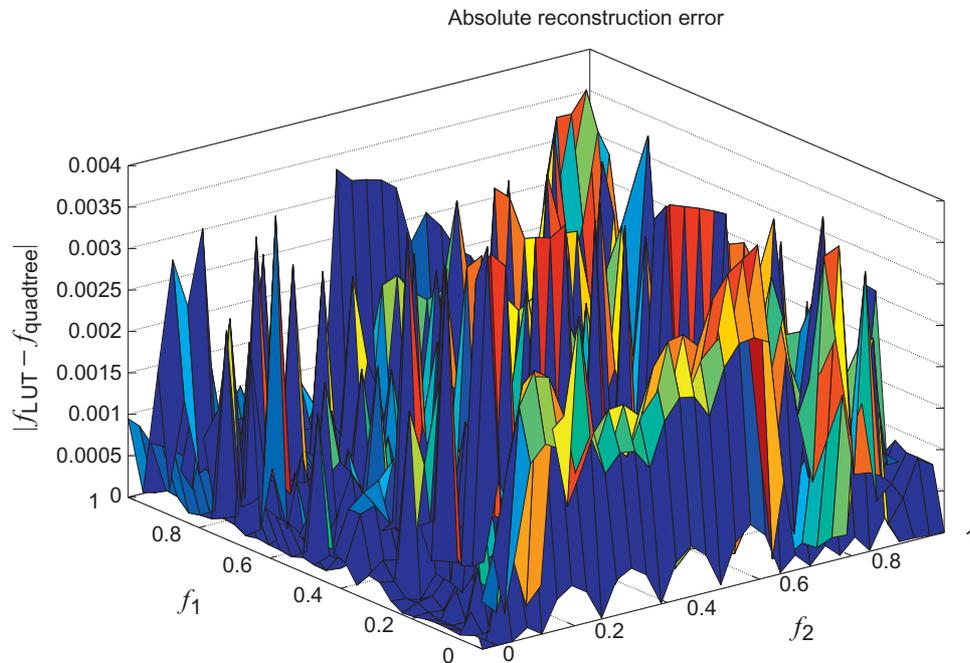


FIGURE 1.28

The absolute difference between the original and the reconstructed CCRF LUTs. The allowed error is greater in areas of the table that correspond to highly unlikely situations.

1.9.1.2 Interpolation circuit

The circuit takes the address and uses it to look up values that are prestored in the block RAM. These values can be used to perform an arithmetic operation (as shown in Fig. 1.31) for bilinear interpolation. To maintain high throughput, the intermediate stages are pipelined with use of registers.

1.9.2 COMPRESSION PERFORMANCE

The compressed CCRF LUT requires storage of all corner values per pair (p_i, p_j) for interpolation. Without any compression, this method would require as many as four times the storage space because of redundancies of identical CCRF values shared between adjacent lookups. However, the compression is able to reduce the number of lookup entries by a factor greater than four times. This compression factor depends on the selection of α .

We wrote the compression in the C programming language to output the compressed CCRF LUT. In Table 1.4, we list the minimum compression factor and the expected error over the entire CCRF range, for four selections of α .

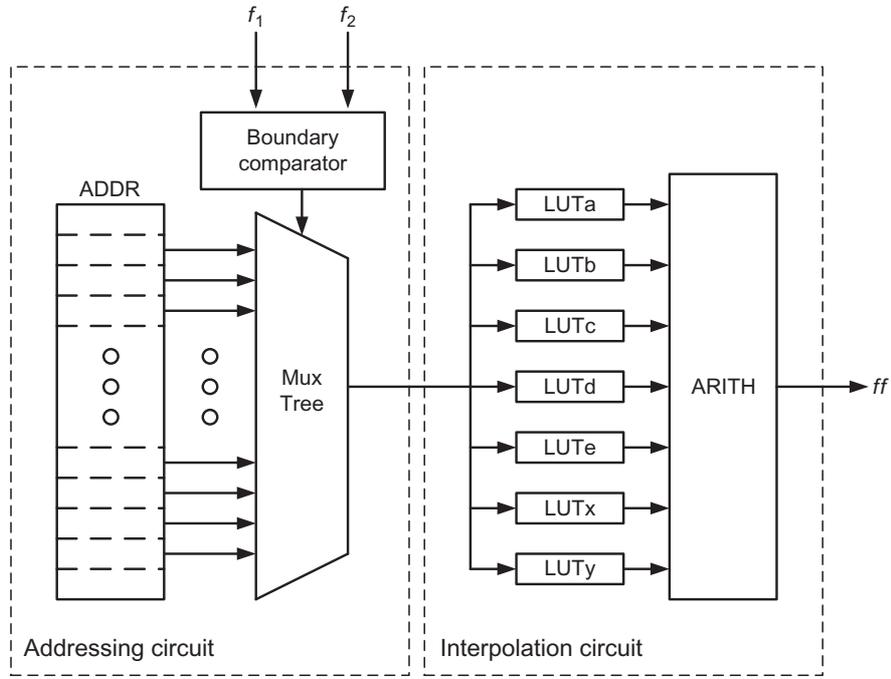


FIGURE 1.29

The top-level system consists of two main parts: the addressing circuit and the interpolation circuit. Input pixel values are normalized first to 16-bit fixed-point representation (f_1 and f_2) before entering the boundary block. According to the boundary conditions of the two values, the circuit generates controlling signals to the multiplexer tree, which selects the correct address that corresponds to the original quadtree node. The address is then used in the interpolation circuit to retrieve the stored values that are necessary for bilinear interpolation. After arithmetic circuit operation, the output ff can be further combined with ff from another instance of the same circuit.

The minimum compression factor summarizes the amount of compression achieved by taking the ratio of the total number of entries in the CCRF LUT to the number in the compressed one:

$$\text{Minimum compression factor} = \frac{N^2}{\text{number of leaf nodes} \times 4}. \quad (1.61)$$

The constant 4 is the upper bound of the maximum redundancy of the corner value storage that overlaps with adjacent CCRF lookup points.

1.9.2.1 Hardware resource usage

For resource estimation without consideration of optimization effort performed by the CAD tool, we have

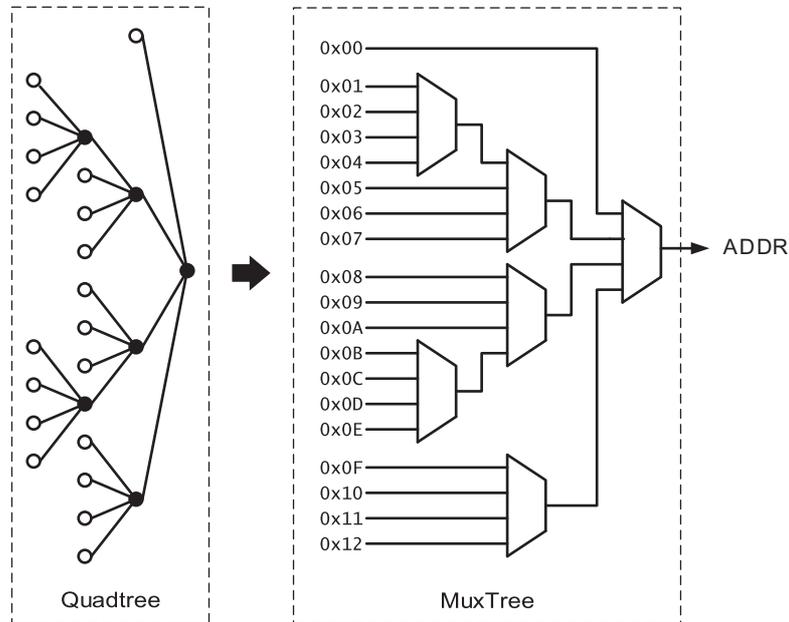


FIGURE 1.30

The relation between the original quadtree and its multiplexer implementation makes it very easy to generate the Verilog using the same quadtree data structure in software (ie, generate software that generates the Verilog code that describes the hardware design). Efficient use of four-to-one multiplexers in the six-LUT FPGA architecture can significantly reduce resource usage (ie, each multiplexer is mapped to one logic slice, instead of three slices if a two-to-one multiplexer is used) and code generation algorithm.

Table 1.4 Quadtree Properties Depending on the Choice of α

Maximum Depth = 8	$\alpha = 1$	$\alpha = 1/2$	$\alpha = 1/4$	$\alpha = 1/16$
Number of entries	3315	4828	6508	10,807
Compression factor	79.1	54.2	161.1	97.0
Mean depth	5.8	6.2	6.6	7.4
Expected depth	7.7	7.8	7.8	7.9
Error constraint	0.0039	0.0019	0.00098	0.00024
Expected error	0.00042	0.00038	0.00036	0.00035

Notes: The compression factor is calculated on the basis of the number of CCRF lookup entries required divided by the number of the entries after compression. The CCRF without compression contains 10^6 floating-point entries.

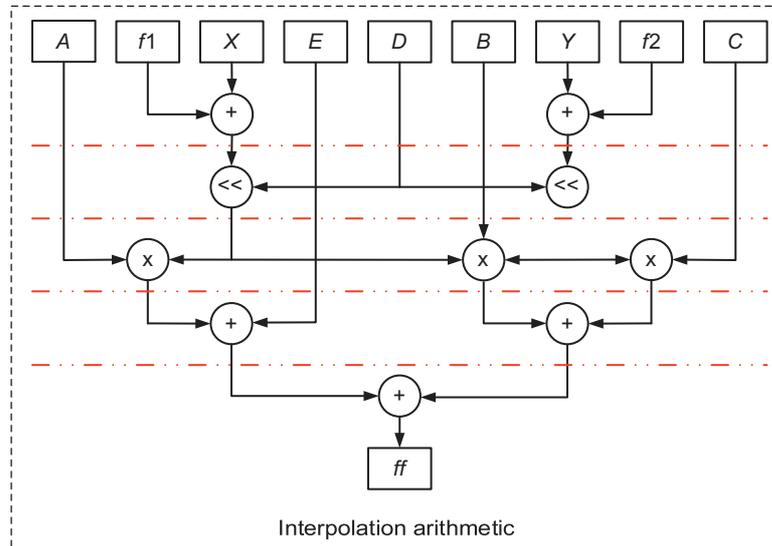


FIGURE 1.31

This circuit performs pipelined arithmetic that is necessary for bilinear interpolation. The inputs to this circuit are load from block RAM, which is initialized according to the compressed quadtree structure. The dotted red (dark gray in print versions) lines indicate the stage after which the data can be pipelined in order to have higher throughput.

$$\begin{aligned} \text{Number of multiplexers} &= \text{number of nonleaf nodes,} \\ \text{Number of comparators} &= \text{number nonleaf nodes} \times 2, \\ \text{Number of addresses} &= \text{number of leaves.} \end{aligned}$$

The actual number of resources needed to implement the system is much less (almost halved) than what we expected. The detailed data gathered from both expectation (using counters) and the CAD outputs are summarized in [Table 1.5](#).

1.9.3 CONCLUSION

This section presented an architecture for accurate HDR imaging that is amenable for implementation on highly memory constrained low-power FPGA platforms. Construction of the compositing function is performed by nonlinear optimization of a Bayesian formulation of the compositing problem, where the selected prior creates an accurate estimator that is smooth for robustness and enhanced compression. The estimator is solved over a regular grid in the unit square, forming a two-dimensional LUT. Implementation of this solution on an FPGA then relies on the compression of the LUT into a quadtree form that allows random access, and uses bilinear interpolation to approximate values for intermediate points. This form allows selective control over error bounds, depending on the expected use of the

Table 1.5 The Resource Usage of the Implementation on a Kintex 7 (Xilinx XC7K325T) FPGA

	Depth = 4	Depth = 6	Depth = 8	Depth = 10
Expected slice usage	63	633	3315	21,207
Portion used (%)	0.0309	0.31	1.6	10
Actual slice usage	2	119	737	11,101
Portion used (%)	9.8×10^{-4}	0.058	0.36	5.5

Notes: The total number of logic slices used in this FPGA is 203,800. The difference between the expected and actual usage is due to the optimization present in the synthesis process, enabling more efficient use of resources.

table, which is easily obtained for a particular sensor. This results in compression of more than 60 times relative to the original LUT, with visually indistinguishable results.

ACKNOWLEDGMENTS

The authors acknowledge assistance or contributions to this project from Antonin Kimla of York Radio and TV, Kopin, Kodak, Digital Equipment, Compaq, Xybernaut, WaveRider, CITO (Communications and Information Technology Ontario), and NSERC (Natural Sciences and Engineering Research Council of Canada).

REFERENCES

- Aczél, J., 1966. Mathematics in science and engineering. A series of monographs and textbooks edited by Richard Bellman, University of Southern California. In: Lectures on Functional Equations and Their Applications, vol. 19. Academic Press, New York/London, p. 510.
- Ali, M.A., Mann, S., 2012. Comparametric image compositing: computationally efficient high dynamic range imaging. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012, pp. 913–916.
- Ballentine, L., 1998. Quantum Mechanics — A Modern Development. World Scientific Publishing Co. Pte. Ltd., Singapore.
- Berger, M.A., 1989. Random affine iterated function systems: curve generation and wavelets. SIAM Rev. 31 (4), 614–627.
- Debevec, P.E., Malik, J., 1997. Recovering high dynamic range radiance maps from photographs. In: SIGGRAPH, pp. 369–378.
- Faugeras, O.D., Lustman, F., 1988. Motion and structure from motion in a piecewise planar environment. Int. J. Pattern Recognit. Artif. Intell. 2 (3), 485–508.
- Granados, M., Ajdin, B., Wand, M., Theobalt, C., Seidel, H.P., Lensch, H.P.A., 2010. Optimal HDR reconstruction with linear digital cameras. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 215–222.
- Horn, B.K.P., 1986. Robot vision. In: MIT Electrical Engineering and Computer Science Series. McGraw-Hill Book Company, Cambridge, MA.

- Horn, B., Schunk, B., 1981. Determining optical flow. *Artif. Intell.* 17, 185–203.
- Horn, B.K.P., Weldon Jr., E.W., 1988. Direct methods for recovering motion. *Int. J. Comput. Vis.* 2 (1).
- Inman, R., Smith, G., 2009. *Television Production Handbook*. <http://www.tv-handbook.com/Camera.html>.
- Irani, M., Peleg, S., 1991. Improving resolution by image registration. *CVGIP* 53, 231–239.
- Jack, K., Tsatsulin, V., 2002. *Dictionary of Video and Television Technology*. Gulf Professional Publishing, Houston.
- Janzen, R., Mann, S., 2014. An information-bearing extramissive formulation of sensing, to measure surveillance and sousveillance. In: *IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2014, pp. 1–10.
- Laveau, S., Faugeras, O., 1994. 3-D scene representation as a collection of images. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, Washington.
- Mann, S., 1992. Wavelets and Chirplets: time-frequency perspectives, with applications. In: Archibald, P. (Ed.), *Advances in Machine Vision, Strategies and Applications*. World Scientific, Singapore/New Jersey/London/Hong Kong, pp. 99–128.
- Mann, S., 1993. Compositing multiple pictures of the same scene. In: *Proc. Annual IS&T Conference*, Cambridge, MA, pp. 50–52.
- Mann, S., 1996a. Joint parameter estimation in both domain and range of functions in same orbit of the projective-Wyckoff group. In: *IEEE International Conference on Image Processing (ICIP-96)*, Lausanne, Switzerland, pp. III-193–III-196.
- Mann, S., 1996b. Wearable, tetherless computer-mediated reality: WearCam as a wearable face-recognizer, and other applications for the disabled. TR 361, M.I.T. Media Lab Perceptual Computing Section. MIT, Cambridge, MA. Also appears in *AAAI Fall Symposium on Developing Assistive Technology for People with Disabilities*, Nov. 9–11, 1996. <http://wearcam.org/vmp.htm>.
- Mann, S., 1997a. An historical account of the ‘WearComp’ and ‘WearCam’ projects developed for ‘personal imaging’. In: *International Symposium on Wearable Computing*. IEEE, Cambridge, MA, pp. 66–73.
- Mann, S., 1997b. Wearable computing: a first step toward personal imaging. *IEEE Comput.* 30 (2), 25–32.
- Mann, S., 1998. Humanistic intelligence/humanistic computing: ‘wearcomp’ as a new framework for intelligent signal processing. *Proc. IEEE* 86 (11), 2123–2151.
- Mann, S., 1999. Personal imaging and lookpainting as tools for personal documentary and investigative photojournalism. *ACM Mob. Netw.* 4 (1), 23–36.
- Mann, S., 2000. Comparometric equations with practical applications in quantigraphic image processing. *IEEE Trans. Image Proc.* 9 (8), 1389–1406. ISSN 1057-7149.
- Mann, S., 2001. *Intelligent Image Processing*. John Wiley and Sons, New York, p. 384.
- Mann, S., 2004. Sousveillance: inverse surveillance in multimedia imaging. In: *Proceedings of the 12th Annual ACM International Conference on Multimedia*. ACM, New York, pp. 620–627.
- Mann, S., 1985. Light painting. *Impulse* 12 (2).
- Mann, S., 2014. The Sightfield: visualizing computer vision, and seeing its capacity to “see”. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2014, pp. 618–623.
- Mann, S., Mann, R. 2001. Quantigraphic imaging: estimating the camera response and exposures from differently exposed images. In: *CVPR*, pp. 842–849.
- Mann, S., Picard, R., 1995a. Being ‘undigital’ with digital cameras: extending dynamic range by combining differently exposed pictures. In: *Proc. IS&T’s 48th Annual Conference*, Washington, DC, May 7–11, 1995. Also appears in M.I.T. M.L. T.R. 323, 1994, <http://wearcam.org/ist95.htm>.
- Mann, S., Picard, R., 1995b. Being ‘undigital’ with digital cameras: extending dynamic range by combining differently exposed pictures. Tech. Rep. 323, M.I.T. Media Lab Perceptual Computing Section. MIT, Cambridge, MA. Also appears in *Proc. IS&T’s 48th Annual Conference*, Washington, DC, May 7–11, 1995, pp. 422–428.

- Mann, S., Picard, R.W., 1994. Virtual bellows: constructing high-quality images from video. In: Proceedings of the IEEE First International Conference on Image Processing, Austin, Texas, Nov. 13–16, 1994, pp. 363–367.
- Mann, S., Picard, R.W., 1995c. Video orbits of the projective group; a simple approach to featureless estimation of parameters. TR 338. MIT, Cambridge, MA. <http://hi.eecg.toronto.edu/tip.html>. Also appears in IEEE Trans. Image Proc. 1997, 6 (9), 1281–1295.
- Mann, S., Janzen, R., Hobson, T., 2011. Multisensor broadband high dynamic range sensing. In: Proc. Tangible and Embedded Interaction (TEI 2011), pp. 21–24.
- Mann, S., Lo, R.C.H., Ovtcharov, K., Gu, S., Dai, D., Ngan, C., Ai, T., 2012. Realtime HDR (high dynamic range) video for eyetap wearable computers, FPGA-based seeing aids, and glasseyes (EyeTaps). In: 25th IEEE Canadian Conference on Electrical & Computer Engineering (CCECE), 2012, pp. 1–6.
- Pal, C., Szeliski, R., Uyttendaele, M., Jovic, N., 2004. Probability models for high dynamic range imaging. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004, vol. 2, pp. II-173–II-180.
- Poynton, C., 1996. A Technical Introduction to Digital Video. John Wiley & Sons, New York.
- Reinhard, E., Ward, G., Pattanaik, S., Debevec, P., 2005. High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting (The Morgan Kaufmann Series in Computer Graphics). Morgan Kaufmann Publishers Inc., San Francisco, CA.
- Robertson, M.A., Borman, S., Stevenson, R.L., 2003. Estimation-theoretic approach to dynamic range enhancement using multiple exposures. J. Electron. Imaging 12, 219.
- Ryals, C., 1995. Lightspace: a new language of imaging. PHOTO Electron. Imaging 38 (2), 14–16.
- Sakurai, J., 1994. Modern Quantum Mechanics, Revised Edition. Addison-Wesley Publishing Co. Inc., Reading, MA.
- Shakhakarmi, N., 2014. Next generation wearable devices: smart health monitoring device and smart sousveillance hat using D2D communications in LTE assisted networks. Int. J. Interdiscip. Telecommun. Netw. 6 (2), 25–51.
- Szeliski, R., 1996. Video mosaics for virtual environments. Comput. Graph. Appl. 16 (2), 22–30.
- Stockham Jr., T.G., 1972. Image processing in the context of a visual model. Proc. IEEE 60 (7), 828–842.
- Woon, S., 1995. Period-harmonic-tupling jumps to chaos and fractal-scaling in a class of series. Chaos, Solitons Fractals 5 (1), 125–130.
- Wyckoff, C.W., 1962. An experimental extended response film. S.P.I.E. Newsletter, June–July, 16–20.
- Wyckoff, C.W., 1961. An experimental extended response film. Tech. Rep. NO. B-321. Edgerton, Germeshausen & Grier, Inc., Boston, MA.