Spatial Tone Reproduction

07

In this and the following chapter we discuss specific algorithms that prepare HDR images for display on LDR display devices. These algorithms are called tone-reproduction or tone-mapping operators (we do not distinguish between these two terms). For each operator we describe how dynamic range reduction is achieved, which user parameters need to be specified, and how these user parameters affect the displayed material. These chapters are intended as a reference for those who want to understand specific operators with a view toward implementing them.

Tone-reproduction operators may be classified in several ways. The classification followed here is to distinguish operators loosely based on how light reflects from a diffuse surface (as discussed in the following chapter) from operators working directly on pixels (i.e., operating in the spatial domain).

A common classification of spatial tone-reproduction operators distinguishes between "local" and "global" operators, as discussed in Chapter 6. In summary, a local operator would compute a local adaptation level for each pixel based on the pixel value itself, as well as a neighborhood of pixels surrounding the pixel of interest. This local adaptation level then drives the compression curve for this pixel. Because the neighborhood of a pixel helps determine how this pixel is compressed, a bright pixel in a dark neighborhood will be treated differently than a bright pixel in a bright neighborhood. A similar argument can be made for dark pixels with bright and dark neighborhoods.

If an operator uses the entire image as the neighborhood for each pixel, such operators are called global. Within an image, each pixel is compressed through a compression curve that is the same for all pixels. As a result, global operators are frequently less expensive to compute than local operators.

Alternatively, tone reproduction may be achieved by transforming the image into a different representation, such as with use of the Fourier domain or by differentiation. These operators form different classes, and are discussed in the following chapter. Thus, four different approaches to dynamic range reduction are distinguished in this book, and each tone-reproduction operator may be classified as one of the following four broad categories.

- *Global operators:* Compress images using an identical (nonlinear) curve for each pixel.
- *Local operators:* Achieve dynamic range reduction by modulating a nonlinear curve by an adaptation level derived for each pixel independently by considering a local neighborhood around each pixel.
- *Frequency domain operators:* Reduce the dynamic range of image components selectively, based on their spatial frequency (Chapter 8).
- *Gradient domain operators:* Modify the derivative of an image to achieve dynamic range reduction (Chapter 8).

Factors common to most tone-reproduction operators are discussed first, including treatment of color, homomorphic filtering, and Fourier domain decompositions. Then, global operators are cataloged in Section 7.2, followed by local operators in Section 7.3.

7.1 PRELIMINARIES

Because all tone-reproduction operators are aimed at more or less the same problem (namely, the appropriate reduction of dynamic range for the purpose of display), there are several ideas and concepts that are shared by many of them. In particular, the input data are often expected to be calibrated in real-world values. In addition, color is treated similarly by many operators. At the same time, several operators apply compression in logarithmic space, whereas others compress in linear space. Finally, most local operators make use of suitably blurred versions of the input image. Each of these issues is discussed in the following sections.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 3

7.1.1 CALIBRATION

Several tone-reproduction operators are inspired by aspects of human vision. The human visual response to light at different levels is nonlinear, and photopic and scotopic lighting conditions in particular lead to very different visual sensations (as discussed in the preceding chapter). For those tone-reproduction operators, it is important that the values to be tone mapped are specified in real-world units (i.e., in cd/m^2). This allows operators to differentiate between a bright daylit scene and a dim night scene. This is not generally possible if the image is given in arbitrary units (see, for example, Figure 7.1).

However, unless image acquisition is carefully calibrated images in practice may be given in arbitrary units. For several tone-reproduction operators, this implies (for instance) that an uncalibrated night image may be tone mapped as if it were a representation of a daylit scene. Displaying such an image would give a wrong impression.

Images may be calibrated by applying a suitably chosen scale factor. Without any further information, the value of such a scale factor can realistically only be approximated, either by trial and error or by making further assumptions on the nature of the scene. In this chapter and the next we show a progression of images for each operator requiring calibrated data. These images are generated with different scale factors such that the operator's behavior on uncalibrated data becomes clear. This should facilitate the choice of scale factors for other images.

Alternatively, it is possible to use heuristics to infer the lighting conditions for scenes depicted by uncalibrated images. In particular, the histogram of an image may reveal if an image is overall light or dark, irrespective of the actual values in the image. Figure 7.2 shows histograms of dark, medium, and light scenes. For many natural scenes, a dark image will have pixels with values located predominantly toward the left of the histogram. A light image will often display a peak toward the right of the histogram, with images between having a peak somewhere in the middle of the histogram.

An important observation is that the shape of the histogram is determined both by the scene being captured and the capture technique employed. In that our main tool for capturing HDR images uses a limited set of differently exposed LDR im-

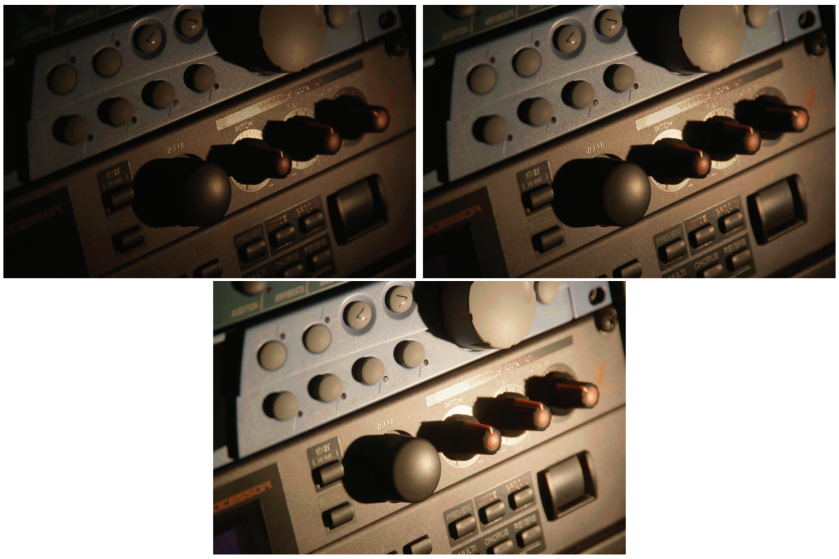


FIGURE 7.1 *This image is given in arbitrary units, and is tone mapped three times with different parameters using the photoreceptor-based technique discussed in Section 7.2.7. Without knowing the actual scene, it is difficult to assess which of these three renditions is most faithful to the actual scene. If the data were properly calibrated, the absolute values in the image would allow the overall brightness to be determined.*

ages (discussed in Chapter 4), images with the sun directly visible will still contain burned-out regions. Similarly, low-level details in nighttime scenes may also not be represented well in an HDR image captured with this method. These limitations affect the shape of the histogram, and therefore the estimation of the key of the scene. A number that correlates to the peak found in a histogram, but is not equal to the location of the peak, is the log average luminance found in the image, calculated as

reinhard v.2005/03/22 Prn:14/06/2005; 8:09   F:reinhard07.tex; VTEX/JOL p. 5

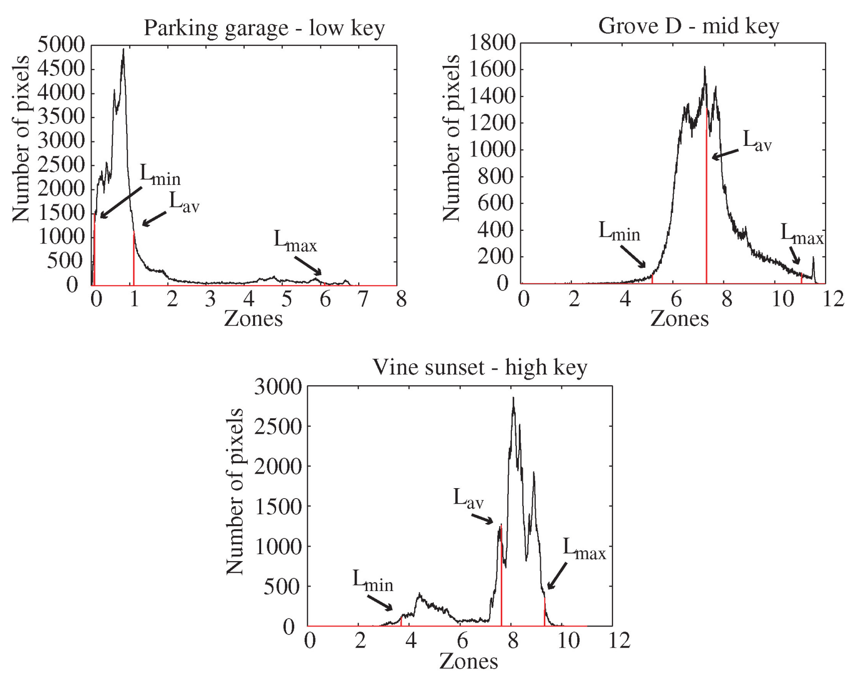


FIGURE 7.2 *Histograms for scenes that are overall dark (left), medium (middle), and light (right).*

$$L_{\mathrm{av}} = \exp\left(\frac{1}{N}\sum_{x,y}\log\left(L_{\mathrm{W}}(x,y)\right)\right), \tag{7.1}$$

where the summation is only over non-zero pixels.

The key of a scene, a unitless number that relates to the overall light level, may be inferred from a histogram. It is thus possible to empirically relate the log average luminance to the minimum and maximum luminance in the histogram (all three are shown in the histograms of Figure 7.2). The key α may be estimated using the following [106].

$$f = \frac{2\log_2 L_{\mathrm{av}} - \log_2 L_{\min} - \log_2 L_{\max}}{\log_2 L_{\max} - \log_2 L_{\min}} \tag{7.2}$$

$$\alpha = 0.18 \times 4^f \tag{7.3}$$

Here, the exponent f computes the distance of the log average luminance to the minimum luminance in the image relative to the difference between the minimum and maximum luminance in the image. To make this heuristic less dependent on outliers, the computation of the minimum and maximum luminance should exclude about 1% of the lightest and darkest pixels. For the photographic tone-reproduction operator (discussed in Section 7.3.6), a sensible approach is to first scale the input data such that the log average luminance is mapped to the estimated key of the scene, as follows.

$$L'_{\mathrm{W}}(x, y) = \frac{\alpha}{L_{\mathrm{av}}} L_{\mathrm{W}}(x, y) \tag{7.4}$$

Although unproven, this heuristic may also be applicable to other tone-reproduction techniques that require calibrated data. However, in any case the best approach would be to always use calibrated images.

7.1.2 COLOR IMAGES

The human visual system is a complex mechanism with several idiosyncrasies that need to be accounted for when preparing an image for display. Most tone-reproduction operators attempt to reduce an image in dynamic range while keeping the human visual system's response to the reduced set of intensities constant. This has led to various approaches that aim at preserving brightness, contrast, appearance, and visibility.

However, it is common practice among many tone-reproduction operators to exclude a comprehensive treatment of color. With few exceptions, it is generally accepted that dynamic range compression should be executed on a single-luminance

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 7

channel. Although this is the current state of affairs, this may change in the near future, as the fields of color-appearance modeling and tone reproduction are growing closer together. This is seen in Pattanaik's multiscale observer model [94] and in more recent developments, such as Johnson and Fairchild's iCAM model [29,30] and Reinhard and Devlin's photoreceptor-based operator [108].

Most other operators derive a luminance channel from the input RGB values (as discussed in Section 2.4) and then compress the luminance channel. The luminance values computed from the input image are called world luminance (L_{w}). The tone-reproduction operator of choice will take these luminance values and produce a new set of luminance values L_{d}. The subscript d indicates "display" luminance. After compression, the luminance channel needs to be recombined with the uncompressed color values to form the final tone-mapped color image.

To recombine luminance values into a color image, color shifts will be kept to a minimum if the ratio between the color channels before and after compression are kept constant [47,113,119]. This may be achieved if the compressed image $R_{\mathrm{d}}G_{\mathrm{d}}B_{\mathrm{d}}$ is computed as follows.

$$\begin{bmatrix} R_{\mathrm{D}} \\ G_{\mathrm{D}} \\ B_{\mathrm{D}} \end{bmatrix} = \begin{bmatrix} L_{\mathrm{D}} \dfrac{R_{\mathrm{W}}}{L_{\mathrm{W}}} \\ L_{\mathrm{D}} \dfrac{G_{\mathrm{W}}}{L_{\mathrm{W}}} \\ L_{\mathrm{D}} \dfrac{B_{\mathrm{W}}}{L_{\mathrm{W}}} \end{bmatrix}$$

Should there be a need to exert control over the amount of saturation in the image, the fraction in the previous equations may be fitted with an exponent s, resulting in a per-channel gamma correction as follows.

$$\begin{bmatrix} R_{\mathrm{D}} \\ G_{\mathrm{D}} \\ B_{\mathrm{D}} \end{bmatrix} = \begin{bmatrix} L_{\mathrm{D}} \left(\dfrac{R_{\mathrm{W}}}{L_{\mathrm{W}}}\right)^{s} \\ L_{\mathrm{D}} \left(\dfrac{G_{\mathrm{W}}}{L_{\mathrm{W}}}\right)^{s} \\ L_{\mathrm{D}} \left(\dfrac{B_{\mathrm{W}}}{L_{\mathrm{W}}}\right)^{s} \end{bmatrix}$$

The exponent s is then given as a user parameter that takes values between 0 and 1. For a value of 1, this method defaults to the standard method of keeping color ratios

FIGURE 7.3 *The saturation parameter s is set to 0.6, 0.8, and 1.0 (in reading order).*

constant. For smaller values, the image will appear more desaturated. Figure 7.3 demonstrates the effect of varying the saturation control parameter s. Full saturation is achieved for a value of $s = 1$. Progressively more desaturated images may be obtained by reducing this value.

An alternative and equivalent way of keeping the ratios between color channels constant is to convert the image to a color space that has a luminance channel and two chromatic channels, such as the Yxy color space. If the image is converted to Yxy space first, the tone-reproduction operator will compress the luminance channel Y and the result will be converted to RGB values for display. This approach is functionally equivalent to preserving color ratios.

7.1.3 HOMOMORPHIC FILTERING

The "output" of a conventional 35-mm camera is a roll of film that needs to be developed, which may then be printed. The following examines the representation of an image as a negative toward defining terminology used in the remainder of this and the following chapter. Under certain conditions, it may be assumed that the image recorded by a negative is formed by the product of illuminance E_{v} and the surface reflectance r, as follows.

$$L_{\mathrm{v}} = E_{\mathrm{v}} r$$

This is a much simplified version of the rendering equation, which ignores specular reflection, directly visible light sources, and caustics and implicitly assumes that surfaces are diffuse. This simplification does not hold in general, but is useful for developing the idea of homomorphic filtering.

If luminance is given, it may be impossible to retrieve either of its constituent components, E_{v} or r. However, for certain applications (including tone reproduction) it may be desirable to separate surface reflectance from the signal. Although this is generally an underconstrained problem, it is possible to transform the previous equation to the log domain, where the multiplication of E_{v} and r becomes an addition. Then, under specific conditions the two components may be separated. Horn's lightness computation, discussed in the following chapter, relies on this observation.

In general, processing applied in the logarithmic domain is called homomorphic filtering. We call an image represented in the logarithmic domain a density image for the following reason. A developed photographic negative may be viewed by shining a light through it and observing the transmitted pattern of light, which depends on the volume concentrations of amorphous silver suspended in a gelatinous emulsion. The image is thus stored as volume concentrations $C(z)$, where z denotes depth, given that a transparency has a certain thickness. Transmission of light through media is governed by Beer's law, and therefore the attenuation of luminance as a function of depth may be expressed in terms of the previous volume concentrations as

$$\frac{dL_v}{dz} = -kC(z)L_v,$$

with k the attenuation constant. In the following, the luminance at a point on the surface is denoted with $L_v(0)$. This equation may be solved by integration, yielding the following solution.

$$\begin{aligned}
\int_{L_v(0)}^{L_v} \frac{di}{i} &= -k \int_0^{z_t} C(z)\,dz \\
\ln\left(\frac{L_v}{L_v(0)}\right) &= -kd \\
L_v &= L_v(0)\exp(-kd)
\end{aligned}$$

Thus, if we integrate along a path from a point on the surface of the transparency to the corresponding point on the other side of the transparency we obtain a luminance value L_v attenuated by a factor derived from the volume concentrations of silver suspended in the transparency along this path. For a photographic transparency, the image is represented by the quantities d, which have a different value for each point of the transparency. The values of d have a logarithmic relationship with luminance L_v. This relationship is well known in photography, although it is usually represented in terms of density D, as follows.

$$D = \log_{10}\left(\frac{L_v(0)}{L_v}\right)$$

The density D is proportional to d and is related to the common logarithm of L_v in a manner similar to the definition of the decibel [123]. Because all such representations are similar (barring the choice of two constant parameters), logarithmic representations are also called density representations. The general transformation between luminance and a density representation may be written as follows.

$$\begin{aligned}
D &= \ln(L_v) \\
L_v &= \exp(D)
\end{aligned}$$

Although a luminance representation of an image necessarily contains values that are nonnegative, in a density representation the range of values is not bound, as in the following.

$$
\begin{array}{rcl}
-\infty < & D = \ln(E_{\mathrm{v}}) + \ln(r) & < \infty \\[2ex]
-\infty < & \ln(E_{\mathrm{v}}) & < \infty \\[2ex]
r_{\min} < & \ln(r) & < 0
\end{array}
$$

In addition, reflectance and illuminance are now added rather than multiplied, which is a direct result of operating in the log domain. Filtering operations such as tone reproduction may be carried out in this domain, which is then called homomorphic filtering. The advantage of this representation is that under circumstances in which light behavior may be modeled as a product of illuminance and reflectance homomorphic filtering allows this product to be represented as an addition, which makes separation of the two components simpler.

7.1.4 GAUSSIAN BLUR

Several operators require the computation of local averages for each pixel. A local average may be viewed as a weighted average of the pixel and some of its neighbors. In most cases the weights are chosen according to a Gaussian distribution. Images filtered by a Gaussian filter kernel may be computed directly in the image domain, where the computation is a convolution, as follows.

$$
L^{\mathrm{blur}}(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L(x, y) \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) dx\, dy
$$

For discrete images, the integrals are replaced by summations. In this chapter we will use the shorthand notation

$$
L^{\mathrm{blur}} = L \otimes R
$$

to indicate that image L is convolved with filter kernel R.

The Gaussian filter kernel is sampled at discrete points, normally at positions corresponding to the midpoints of each pixel. For very small filter kernels, point sampling a Gaussian function with very few samples leads to a large error. To account for the spacing between sample points, a fast way of integrating a Gaussian function over an area may be achieved by expressing the Gaussian in terms of the error function, which is given by[1]

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_{-\infty}^{x} \exp(-x^2)\,dx.$$

We may therefore build an image $R(x, y)$ the size of the input image, which represents the Gaussian filter (with the peak at pixel $(0, 0)$), as follows.

$$\begin{aligned} R(x, y) = {} & \frac{1}{4}\left(\operatorname{erf}\left(\frac{x - 0.5}{\sigma}\right) - \operatorname{erf}\left(\frac{x + 0.5}{\sigma}\right)\right) \\ & \times \left(\operatorname{erf}\left(\frac{y - 0.5}{\sigma}\right) - \operatorname{erf}\left(\frac{y + 0.5}{\sigma}\right)\right) \end{aligned}$$

The computational cost of the error function is not higher than evaluating the exponential function. In this scheme, four error functions are executed per pixel, and therefore the accuracy obtained by integration over each pixel's area comes at a slight computational cost. This extra expense is acceptable because for certain applications (such as the photographic tone-reproduction operator discussed in Section 7.3.6) the extra accuracy is nonnegligible. For all results shown in this and the following chapter (involving Gaussian blurred images) we have used this scheme.

The cost of blurring an image lies in the convolution operator. Because for every pixel every other pixel needs to be considered, direct convolution takes $O(N^2)$ time in the number of pixels. For convolution kernels larger than 3 by 3 pixels (for example), this is too costly in practice. In such cases we may transform both filter kernel and image to the Fourier domain by means of a fast Fourier transform (FFT). The convolution then becomes a pointwise multiplication that takes $O(N)$ time. The FFT and inverse FFT each take $O(N \log(N))$ time, and thus the time complexity of blurring an image with a Gaussian filter kernel takes $O(N \log(N))$ time in total.

1 This idea was developed by Mike Stark and became part of the photographic tone-reproduction operator [109], whereby the robustness of the scale-selection mechanism improved as a result (see Section 7.3.6 for further details on scale selection).

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 13

Before the FFT of the filter kernel can be computed, the Gaussian needs to be mirrored in the center of the kernel image. The center of the Gaussian is then replicated in each of the four corners of the image. The process of blurring an image is shown in Figure 7.4. Note that the FFT of the Gaussian filter kernel is again a Gaussian function, albeit now as a function of frequency. It would therefore be possible to construct the Gaussian filter kernel directly in the Fourier domain, thereby saving one FFT transform.

Alternatively, it may be possible to truncate the Gaussian filter kernel to reduce the computational cost, or resort to fast approaches that are not based on Fourier decomposition. Examples are the elliptically weighted average approach [46] and Burt and Adelson's approximation [8]. The 2D Gaussian filter is separable (i.e., it may be expressed as the multiplication of two 1D Gaussian filters), as follows.

$$R(x, y) = R_{\mathrm{x}}(x) R_{\mathrm{y}}(y)$$

$$\frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{x^2}{2\sigma^2}\right)\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{y^2}{2\sigma^2}\right)$$

This means that the Gaussian convolution may be computed in x and y directions separately, providing a further opportunity to reduce the computational cost in that two 1D FFTs are quicker to execute than one 2D transform.

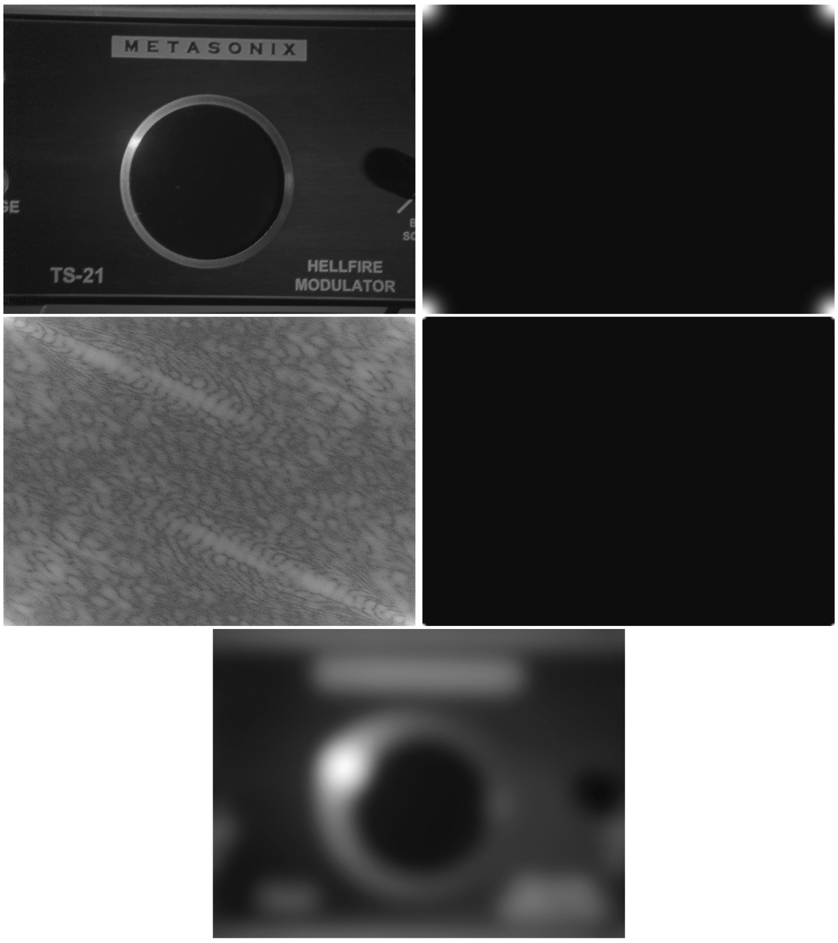
7.1.5 VALIDATION

To allow more informed decisions as to which operator is suitable for which task, there is a need for validation studies. At the time of writing, only two such studies exist [20,31,68], with more beginning to emerge [73]. In addition, the CIE has formed a technical committee (TC8-08) to study the issue of how tone-reproduction operators might be validated [60].[2]

Currently, visual comparison remains one of the most practical ways of assessing tone-reproduction operators, but this approach is not without pitfalls. In particular, the choice of parameter settings for each of the operators will have a large impact on the outcome of such visual comparisons. To avoid subconscious comparisons, we

2 See also the web site for CIE Division 8: Image Technology at *www.colour.org*.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 14



reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 15

FIGURE 7.4 *Top row: input image and Gaussian filter kernel. Middle row: Fourier representation of the image and the filter kernel. After pointwise multiplication of these images followed by the inverse Fourier transform, the Gaussian blurred image shown at the bottom is obtained.*

have purposely chosen to use different images for each of the operators discussed in this and the following chapter. Until proper validation studies start to show a pattern and reach agreement on which operators perform well, visual comparison, taste, and other secondary considerations will dominate the decision-making process.

7.2 GLOBAL OPERATORS

The simplest functions that reduce an image's dynamic range treat each pixel independently. Such functions usually take for each pixel its value and a globally derived quantity, usually an average of some type (see Section 6.4).

Global operators share one distinct advantage: they are computationally efficient. Many of them may be executed in real time. Because global operators are generally much faster than any of the other classes of operators, applications that require this level of performance should consider global operators over all others.

On the other hand, if the dynamic range of an image is extremely high the global tone-reproduction operators may not always preserve visibility as well as other operators. However, there are also differences among global operators. Some operators are able to handle a larger class of HDR images than others. This issue is discussed further in the following sections.

7.2.1 MILLER BRIGHTNESS-RATIO-PRESERVING OPERATOR

The first global tone-reproduction operator we know of was documented in 1984 by Miller and colleagues [186]. They aimed to introduce the field of computer graphics to the lighting engineering community. For rendering algorithms to be useful in lighting design, they should output radiometric or photometric quantities, rather than arbitrarily scaled pixel intensities. Physically based rendering algorithms therefore produce imagery that is typically not directly displayable on LDR display devices, thus requiring tone reproduction for display (see Chapter 4).

As a result, Miller et al. developed a tone-reproduction operator that aims at preserving the sensation of brightness of the image before and after dynamic range reduction. Brightness is a complex function of both luminance and spatial configuration that may be simplified for the purposes of this work. Here, brightness Q is approximated as a power function of luminance L_v, as follows.

$$Q = kL_\mathrm{v}^b$$

Miller et al. assert that the visual equivalence of an image before and after dynamic range reduction may be modeled by keeping brightness ratios constant. Thus, for two elements Q_1 and Q_2 to be visually equivalent to their compressed counterparts Q_1' and Q_2', their ratios should be constant. That is,

$$\frac{Q_1}{Q_2} = \frac{Q_1'}{Q_2'}.$$

It should be noted that visual equivalence between pairs of brightness values is not the same as being equal (i.e., in general, Q_1 will be different from Q_1', and Q_2 will not be equal to Q_2').

The procedure for preparing an image for display starts by converting an image to brightness values $Q_\mathrm{w}(x, y)$. Then the maximum brightness of the image $Q_\mathrm{w,max}$ is determined. The image's brightness values are then normalized by dividing each pixel's brightness representation by the image's maximum brightness.

The display device's maximum brightness $Q_\mathrm{d,max}$ is then determined from its maximum luminance value using the same luminance brightness relationship. Display brightnesses $Q_\mathrm{d}(x, y)$ are determined using the following.

$$Q_\mathrm{d}(x, y) = \frac{Q_\mathrm{w}(x, y)}{Q_\mathrm{w,max}} Q_\mathrm{d,max}$$

These brightnesses are then converted to luminances by applying the inverse of the brightness function. There exist different formulas for determining brightness

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 17

values from luminances. Miller et al. experimented with three different formulations, and determined that the one proposed by Stevens [121] produced the most plausible results. Fitting functions to Stevens' psychophysical data, Miller created functional forms for k and b, as follows.

$$b = 0.338 L_{\mathrm{v}}^{0.034}$$
$$k = -1.5 \log_{10}(L_{\mathrm{v}}) + 6.1$$

The relationship between luminance and brightness then becomes

$$Q = (-1.5 \log_{10}(L_{\mathrm{v}}) + 6.1) L_{\mathrm{v}}^{0.338 L_{\mathrm{v}}^{0.034}}.$$

A plot of this function is shown in Figure 7.5. The function monotonically increases until about 2,000 cd/m^2, and then steeply declines. This is a result of fitting the previous function to psychophysical data that are only valid for a limited range (up to about 1,000 cd/m^2).

As Miller et al.'s work is aimed at lighting design, their operator is suitable for compressing luminance ranges that are typically found in indoor situations. They assert that actual room luminances range between 100 and 1,000 cd/m^2, whereas display devices are typically limited to the range of 1 to 33 cd/m^2. Current display devices can be brighter, though. Most tone-reproduction operators requiring an estimate of the maximum display luminance use values in the range of 30 to 100 cd/m^2.

A second implication of the sharp decline of the previous function is that this brightness equation is not analytically invertible, which is necessary for Miller's operator to be useful. However, the inverse of this function may be approximated with a lookup table of sufficiently high resolution, allowing us to experiment with this tone-reproduction operator.

For practical purposes, we normalize each image within the range between 0 and 1,000 cd/m^2. This places an assumption on the input image, which is that it depicts an indoor scene. This is not unreasonable, in that the operator is not suitable for images with a higher dynamic range.

The maximum display luminance depends on the display device, and therefore the maximum brightness of the display device for which the operator should compress will also vary. To simulate tone reproduction for different display devices, we

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 18

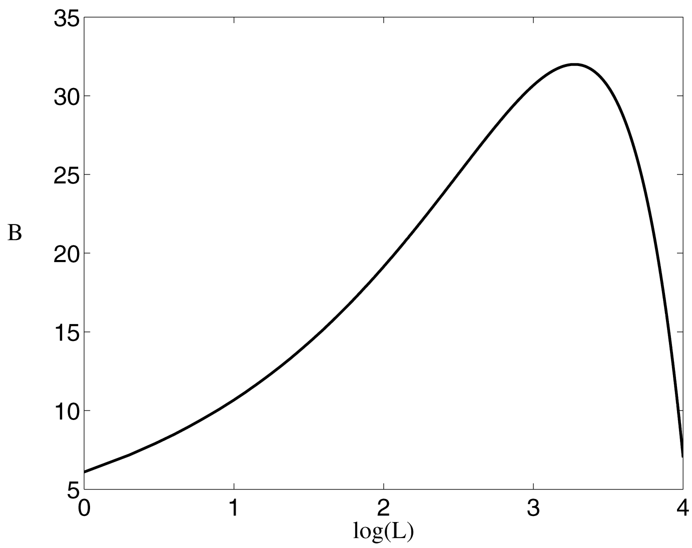


FIGURE 7.5 *Miller's mapping of log luminance to brightness. This mapping is valid over its monotonically increasing domain (i.e., up to a luminance of about 1,000 cd/m^2).*

make $Q_{\mathrm{d,max}}$, the maximum display brightness, a user parameter. Varying this parameter for an image with a dynamic range comparable to an indoor scene, the set of images in Figure 7.6 was produced. This parameter behaves as expected: higher maximum monitor brightness values result in lighter images.

Other than the invertibility of the brightness function, which is solved by employing a lookup table, this operator is both simple to implement and fast. However, the previously cited limitations make this operator mainly of interest for historical reasons.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 19

FIGURE 7.6 Using Miller's operator, the maximum monitor brightness $Q_{\max}$ was set from 10 (top left) and incremented by 10 in subsequent images.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 20

7.2.2 TUMBLIN–RUSHMEIER BRIGHTNESS-PRESERVING OPERATOR

Where Miller et al. were the first to introduce computer graphics to the field of lighting design, focusing on tone reproduction to accomplish this goal, it was Tumblin and Rushmeier who introduced the problem of tone reproduction to the field of computer graphics in 1993 [131]. Tumblin and Rushmeier also based their work on Stevens' psychophysical data, realizing that the human visual system is already solving the dynamic range reduction problem.

The Tumblin–Rushmeier operator exists in two different forms: the original operator [131] and a revised version [132] (which corrects a couple of shortcomings including the fact that it was calibrated in sieverts, a unit that is not in wide use). For this reason, we limit our discussion to the revised Tumblin–Rushmeier operator and will refer to it simply as the Tumblin–Rushmeier operator.

Although the Tumblin–Rushmeier operator is based on the same psychophysical data as Miller's operator, the brightness function is stated slightly differently, as follows.

$$Q(x, y) = C_0\left(\frac{L(x, y)}{L_\mathrm{a}}\right)^{\gamma}$$

Here, Q is brightness (or perceived luminance), measured in brils. L is luminance in cd/m^2 and L_a is the adaptation luminance, also measured in cd/m^2. The constant $C_0 = 0.3698$ is introduced to allow the formula to be stated in SI units. Finally, γ is a measure of contrast sensitivity and is itself a function of the adaptation luminance L_a.

This function may be evaluated for an HDR image as well as for the intended display device. This leads to two sets of brightness values as a function of input luminances (or world luminances) and display luminances. In the following, the subscripts w and d indicate world quantities (measured or derived from the HDR image) and display quantities. Whereas Miller et al. conjecture that image and display brightness ratios should be matched, Tumblin and Rushmeier simply equate the image and display brightness values, as follows.

$$Q_\mathrm{w}(x, y) = C_0\left(\frac{L_\mathrm{w}(x, y)}{L_\mathrm{wa}}\right)^{\gamma(L_\mathrm{wa})}$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 21

$$Q_{\mathrm{d}}(x,y) = C_0 \left(\frac{L_{\mathrm{d}}(x,y)}{L_{\mathrm{da}}} \right)^{\gamma(L_{\mathrm{da}})}$$

$$Q_{\mathrm{w}}(x,y) = Q_{\mathrm{d}}(x,y)$$

The gamma function $\gamma(L)$ models Stevens' human contrast sensitivity for the image and the display by plugging in L_{wa} and L_{da}, respectively, given by

$$\gamma(L) = \begin{cases} 2.655 & \text{for } L > 100\ \mathrm{cd/m^2} \\ 1.855 + 0.4 \log_{10}(L + 2.3 \cdot 10^{-5}) & \text{otherwise.} \end{cases}$$

These equations may be solved for $L_{\mathrm{d}}(x,y)$, the display luminance that is the quantity we wish to display. The result is

$$L_{\mathrm{d}}(x,y) = L_{\mathrm{da}} \left(\frac{L_{\mathrm{w}}(x,y)}{L_{\mathrm{wa}}} \right)^{\gamma(L_{\mathrm{wa}})/\gamma(L_{\mathrm{da}})}.$$

The adaptation luminances are L_{da} for the display and L_{wa} for the image. The display adaptation luminance is typically between 30 and 100 $\mathrm{cd/m^2}$, although this number will be higher when HDR display devices are used. The image adaptation luminance is given as the log average luminance L_{wa} (Equation 7.1). The mid-range scene luminances now map to mid-range display luminances close to L_{da}, which for dim scenes results in a uniform gray appearance in the display. This may be remedied by introducing a scale factor $m(L_{\mathrm{wa}})$, which depends on the world adaptation level L_{da}, as follows.

$$m(L_{\mathrm{wa}}) = \left(\sqrt{C_{\max}}\right)^{\gamma_{\mathrm{wd}}-1}$$

$$\gamma_{\mathrm{wd}} = \frac{\gamma_{\mathrm{w}}}{1.855 + 0.4 \log(L_{\mathrm{da}})}$$

Here, $C_{\max}$ is the maximum displayable contrast, which is typically between 30 and 100 for an LDR display device. The full operator is then given by

$$L_{\mathrm{d}}(x,y) = m(L_{\mathrm{wa}}) L_{\mathrm{da}} \left(\frac{L_{\mathrm{w}}(x,y)}{L_{\mathrm{wa}}} \right)^{\dfrac{\gamma(L_{\mathrm{wa}})}{\gamma(L_{\mathrm{da}})}}.$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 22

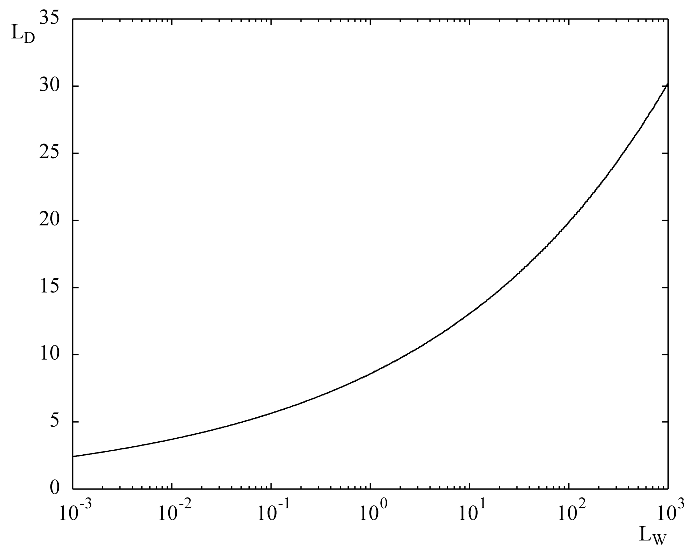


FIGURE 7.7 *Mapping of world luminances to display luminances for the Tumblin–Rushmeier operator.*

For suitably chosen input parameters, a plot of this function is given in Figure 7.7. As this operator is calibrated in SI units, the image to be tone mapped also needs to be specified in SI units. For an image in unknown units, we experimented with different scale factors prior to tone reproduction, the results of which are shown in Figure 7.8. This image was scaled by factors of 0.1, 1, 10, 100, and 1,000, with the scaling resulting in progressively lighter images. For this particular image, a scale factor of close to 1,000 would be optimal. Our common practice of normalizing the image, applying gamma correction, and then multiplying by 255 was abandoned for this image sequence, because this operator already includes a display gamma correction step.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 23

FIGURE 7.8 In this sequence of images, the input luminances were scaled by factors of 0.1, 1, 10, 100, and 1,000 prior to applying Tumblin and Rushmeier's revised tone-reproduction operator.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 24

In summary, the revised Tumblin–Rushmeier tone-reproduction operator is based on the same psychophysical data as Miller's operator, but the crucial difference is that Miller et al. aim to preserve brightness ratios before and after compression whereas Tumblin–Rushmeier attempt to preserve the brightness values themselves. In our opinion, the latter leads to a useful operator that produces plausible results, provided the input image is specified in cd/m^2. If the image is not specified in cd/m^2, it should be converted to SI units. In that case, the image may be pre-scaled by a factor that may be determined by trial and error, as shown in Figure 7.8.

7.2.3 WARD CONTRAST-BASED SCALE FACTOR

Whereas both Miller's and Tumblin and Rushmeier's operators aim at preserving the sensation of brightness, other operators focus less on brightness perception and attempt to preserve contrasts instead. An early example is Ward's contrast-based scale factor [139]. The model matches JNDs one might discern in the image with JNDs an observer of an LDR display device may distinguish. Thus, differences are preserved without spending the limited number of display steps on differences undetectable by the human visual system. The operator maps image or world luminances to display luminances linearly, as follows.

$$L_{\mathrm{d}}(x, y) = m L_{\mathrm{w}}(x, y)$$

The scale factor m is chosen to match threshold visibility in image and display. This requires a threshold-versus-intensity function (TVI) $t(L_{\mathrm{a}})$, which maps a threshold luminance that is just visible for adaptation luminance L_{a}. We also need to estimate the adaptation level for an observer of the image (L_{wa}), as well as for an observer viewing the display (L_{da}). The scale factor m may then be chosen such that

$$t(L_{\mathrm{da}}) = m t(L_{\mathrm{wa}}).$$

Solving for m yields

$$m = t(L_{\mathrm{da}})/t(L_{\mathrm{wa}}).$$

Based on Blackwell's studies [11], this yields the following scale factor.

$$m = \frac{1}{L_{\mathrm{d,max}}} \left(\frac{1.219 + (\frac{L_{\mathrm{d,max}}}{2})^{0.4}}{1.219 + L_{\mathrm{wa}}^{0.4}} \right)^{2.5}$$

In this equation, the display adaptation level is estimated to be half the maximum display luminance, which is specified as $L_{\mathrm{d,max}}$. The maximum display luminance should be specified by the user, and is typically in the range of 30 to 100 cd/m^2. The world adaptation level may be estimated as the log average of the image's luminance values, as follows.

$$L_{\mathrm{wa}} = \exp\left(\frac{1}{N} \sum_{x,y} \log\left(10^{-8} + L_{x,y}\right) \right)$$

In this equation, we sum the log luminance values of all pixels and add a small offset to avoid the singularity that occurs for black pixels. This log average computation is slightly different from the one used in the preceding section. The small offset could be omitted, but then the summation should only include non-zero pixels. Because the offset is small, the difference between the two log average computations should also be small. The division is by N, the number of pixels in the image.

As with tone-reproduction operators discussed earlier, the input image needs to be specified in SI units. In Figure 7.9 we show the effect of pre-scaling an uncalibrated image with various values. Scaling the image to larger values produces a brighter result, which should not be surprising.

As this operator scales the input linearly, choices of pre-scaling and values of maximum display luminance amount to choosing which luminances in the image are mapped to middle gray on the display device. As such, the images shown in Figure 7.9 are effectively brighter or darker versions of each other.

7.2.4 FERWERDA MODEL OF VISUAL ADAPTATION

The concept of matching JNDs as explored by Ward was also used by Ferwerda et al. in their operator. They based their operator on different psychophysical data,

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 26

FIGURE 7.9 *Scaling of an image given in uncalibrated units prior to application of Ward's contrast-based scale factor. From left to right we scaled the image by factors of 0.01, 0.1, and 1.*

with a somewhat different functional shape as a result. The operator, however, is still intrinsically a linear mapping between world and display luminances. Whereas Ward's contrast-based scale factor incorporates only photopic lighting conditions, Ferwerda et al. added a scotopic component. They also modeled the loss of visual acuity under scotopic lighting, as well as the process of light and dark adaptation, which takes place over time.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 27

In Ferwerda's operator, display intensities are computed from world intensities by multiplying the latter with a scale factor m and adding an offset b, which allows contrast and overall brightness to be controlled separately. This is calculated as follows.

$$\begin{aligned}
R_\mathrm{d}(x,y) &= d\big(mR_\mathrm{w}(x,y) + bL_\mathrm{w}(x,y)\big)\\
G_\mathrm{d}(x,y) &= d\big(mG_\mathrm{w}(x,y) + bL_\mathrm{w}(x,y)\big)\\
B_\mathrm{d}(x,y) &= d\big(mB_\mathrm{w}(x,y) + bL_\mathrm{w}(x,y)\big)\\
m &= \frac{t_\mathrm{p}(L_\mathrm{da})}{t_\mathrm{p}(L_\mathrm{wa})}\\
b &= \frac{t_\mathrm{s}(L_\mathrm{da})}{t_\mathrm{s}(L_\mathrm{da})}\\
d &= \frac{L_\mathrm{max}}{L_\mathrm{d,max}}
\end{aligned}$$

This operator thus scales each of the three red, green, and blue channels by a factor m, but adds an achromatic term that depends on the pixel's luminance. The scale factor m governs photopic conditions, whereas the b term handles scotopic conditions. Both depend on TVI functions.

Modeling cones, which are active under photopic lighting conditions, the TVI function $t_\mathrm{p}(L_\mathrm{a})$ is approximated by the following.

$$\log_{10} t_\mathrm{p}(L_\mathrm{a}) = \begin{cases} -0.72 & \text{if } \log_{10}(L_\mathrm{a}) \le -2.6\\ \log_{10}(L_\mathrm{a}) - 1.255 & \text{if } \log_{10}(L_\mathrm{a}) \ge 1.9\\ (0.249\log_{10}(L_\mathrm{a}) + 0.65)^{2.7} - 0.72 & \text{otherwise} \end{cases}$$

For the rods, active under scotopic lighting conditions, the TVI function $t_\mathrm{s}(L_\mathrm{a})$ is approximated by the following.

$$\log_{10} t_\mathrm{s}(L_\mathrm{a}) = \begin{cases} -2.86 & \text{if } \log_{10}(L_\mathrm{a}) \le -3.94\\ \log_{10}(L_\mathrm{a}) - 0.395 & \text{if } \log_{10}(L_\mathrm{a}) \ge -1.44\\ (0.405\log_{10}(L_\mathrm{a}) + 1.6)^{2.18} - 2.86 & \text{otherwise} \end{cases}$$

For both the photopic and scotopic range, a separate scale factor (m_p and m_s, respectively) may be computed using the previous TVI curves, as follows.

$$m_p = \frac{t_p(L_{da})}{t_p(L_{wa})}$$

$$m_s = \frac{t_s(L_{da})}{t_s(L_{wa})}$$

These scale factors depend on the display adaptation luminance L_{da} and the world (or image adaptation) luminance L_{wa}. The display adaptation luminance may be estimated to be half the maximum display luminance. For typical LDR displays, the maximum display luminance is about 100 cd/m^2, and thus the display adaptation luminance is estimated as 50 cd/m^2. For this operator, the world adaptation luminance is approximated by half the maximum world luminance L_{max}.

In addition to mapping luminances to a displayable range, tone reproduction may attempt to preserve other aspects of human vision across viewing conditions. One of these is visual acuity. Under scotopic lighting conditions, the human visual system may not resolve as much detail as under photopic lighting conditions. Ferwerda et al. outline a solution that may be applied in addition to the previously cited mapping. This involves removing from the displayable image frequencies that would not have been resolvable by the world observer. This may be accomplished in the Fourier domain by removing frequencies higher than the threshold frequency for the world observer, as follows.

$$f^*\big(w_c(L_{wa})\big) = \frac{t(L_{wa})}{L_{wa}}$$

As with Ward's contrast-based scale factor and Tumblin and Rushmeier's operator, Ferwerda's operator is based on psychophysical measurements, and is calibrated in SI units. An example of different pre-scaling factors for an uncalibrated image is shown in Figure 7.10, which reveals that for images that are scaled to small values the range of input values covers the scotopic range (in which vision is achromatic). For larger scale factors, the HDR data covers the mesopic and photopic ranges, where color is retained. Note that in this figure the change in visual acuity as a function of world adaptation level was not modeled.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 29

FIGURE 7.10 Pre-scaling factors of 0.1, 1, 10, 33, 66, and 100 applied prior to tone mapping with Ferwerda's operator.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 30

Ferwerda's operator was later adapted by Durand and Dorsey [22] for the purpose of interactive tone reproduction. They also proposed various computationally efficient extensions that allow modeling the blue shift (associated with scotopic lighting conditions), light adaptation, and chromatic adaptation.

Although Ferwerda's operator is a linear scale factor (like Ward's contrast-based scale factor), it models visual acuity and includes an achromatic component that models scotopic vision. It is therefore a more complete model than Ward's. However, it is still a linear model, which means that the maximum dynamic range that may be successfully tone mapped for display on an LDR device is limited. For very high dynamic range images, a nonlinear mapping may be a better approach.

7.2.5 LOGARITHMIC AND EXPONENTIAL MAPPINGS

Of all nonlinear mappings, logarithmic and exponential mappings are among the most straightforward. Their main use is in providing a baseline result against which all other operators may be compared. After all, any other operator is likely to be more complex and we may expect other operators to provide improved visual performance compared with logarithms and exponential mappings (although we would like to keep the notion of visual performance deliberately vague).

For medium-dynamic-range images (i.e., images with a dynamic range somewhat higher than can be accommodated by current LDR display devices), these very simple solutions may in fact be competitive with more complex operators. The logarithm is a compressive function for values larger than 1, and therefore range compression may be achieved by mapping luminances as follows.

$$L_\mathrm{d}(x, y) = \frac{\log_{10}(1 + L_\mathrm{w}(x, y))}{\log_{10}(1 + L_\mathrm{max})} \tag{7.5}$$

A second mapping converts world luminances to display luminances by means of the exponential function [33], as follows.

$$L_\mathrm{d}(x, y) = 1 - \exp\left(-\frac{L_\mathrm{w}(x, y)}{L_\mathrm{av}}\right) \tag{7.6}$$

This function is bound between 0 for black pixels and 1 for infinitely bright pixels. Because world luminances are always smaller than infinity, the resulting display

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 31

FIGURE 7.11 *Left: logarithmic mapping using* L_{max}. *Right: exponential mapping using* L_{av}. *Compare with Figure 7.12, where the roles of maximum and average luminance are reversed.*

luminances $L_{\mathrm{d}}(x, y)$ will in practice never quite reach 1. Subsequent normalization would therefore somewhat expand the range of display values.

The division by the average luminance L_{av} in the previous exponential function causes pixels with this value to be mapped to $1 - 1/e \approx 0.63$. Because this value is slightly above 0.5, the arithmetic average is employed rather than the more commonly used log average luminance.

Figure 7.11 shows example results of the logarithmic and exponential mappings. Both images successfully map world luminances to display luminances. However, the logarithmic mapping produces an image that is somewhat dull. The exponential mapping, on the other hand, is overall much lighter. The original scene appeared to sit somewhere between these two renditions, and thus neither algorithm produced a displayable image that was faithful to the original scene.

The differences between the images produced with logarithmic and exponential mappings could be due to the shape of the compression curve, but we also note that the logarithmic mapping is anchored to the maximum luminance value in

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 32

FIGURE 7.12 *Left: logarithmic mapping using L_{av}. Right: exponential mapping using L_{max}. Compare with Figure 7.11.*

the image, whereas the exponential mapping uses the average luminance value. Figure 7.12 shows the same two algorithms, but now we have swapped L_{av} and L_{max} in both operators. This small change to these operators has also caused their appearance to be reversed.

Plots of logarithmic and exponential mappings with either L_{av} or L_{max} used to anchor the mapping are shown in Figure 7.13. The functional form, as well as the choice of anchor value, has a significant impact on the shape of the compressive function. Although more experimentation would be required to draw definite conclusions, it appears that the value chosen to anchor the mapping — L_{max} or L_{av} — has the more profound effect on the result.

In summary, logarithmic and exponential mappings are among the most straightforward nonlinear mappings. For images with a dynamic range that only just exceeds the capabilities of the chosen display device, these approaches may well suffice. For images with a higher dynamic range, however, other approaches may be more suitable.

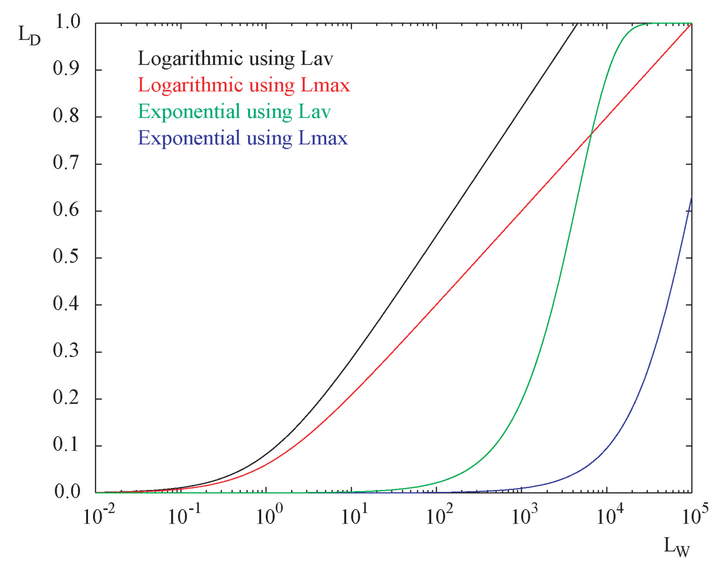


FIGURE 7.13 Plots of Equations 7.5 and 7.6 using L_{av} and L_{max}.

7.2.6 DRAGO LOGARITHMIC MAPPING

Building upon the observation that the human visual system to a first approximation uses a logarithmic response to intensities, Drago et al. show how logarithmic response curves may be extended to handle a wider dynamic range than the simple operators discussed in the preceding section [21].

The operator effectively applies a logarithmic compression to the input luminances, but the base of the logarithm is adjusted according to each pixel's value. The base is varied between 2 and 10, allowing contrast and detail preservation in dark and medium luminance regions while still compressing light regions by larger amounts. A logarithmic function of arbitrary base b may be constructed from log-

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 34

arithmic functions with a given base (for instance, base 10), as follows.

$$\log_b(x) = \frac{\log_{10}(x)}{\log_{10}(b)}$$

To smoothly interpolate between different bases, use is made of Perlin and Hoffert's bias function. In this function, the amount of bias is controlled by user parameter p [99], as follows.

$$\mathrm{bias}_p(x) = x^{\log(p)/\log(0.5)}$$

The basic tone-reproduction curve is the same as the logarithmic mapping presented in the preceding section, but with a base b (which is a function of each pixel's luminance), as follows:

$$L_\mathrm{d}(x, y) = \frac{\log_b(1 + L_\mathrm{w}(x, y))}{\log_b(1 + L_\mathrm{w,max})}$$

To smoothly interpolate between different bases, the preceding three equations are combined as follows.

$$L_\mathrm{d}(x, y) = \frac{L_\mathrm{d,max}/100}{\log_{10}(1 + L_\mathrm{w,max})} \cdot \frac{\log_{10}(1 + L_\mathrm{w}(x, y))}{\log_{10}[2 + 8\{(\frac{L_\mathrm{w}(x,y)}{L_\mathrm{w,max}})^{\log_{10}(p)/\log_{10}(0.5)}\}]}$$

The constants 2 and 8 bound the chosen base between 2 and 10. The maximum display luminance $L_\mathrm{d,max}$ is display dependent and should be specified by the user. In most cases, a value of 100 cd/m^2 would be appropriate.

This leaves the bias parameter p to be specified. For many practical applications, a value between 0.7 and 0.9 produces plausible results, with a value of $p = 0.85$ being a good initial value. Figure 7.14 shows an image created with different bias values. The bias parameter steers the amount of contrast available in the tone-mapped

FIGURE 7.14 *For Drago's logarithmic mapping, increasing values for the bias parameter p result in reduced contrast. In reading order, the bias parameter varied between 0.6 and 1.0 in increments of 0.1.*

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 35

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 36

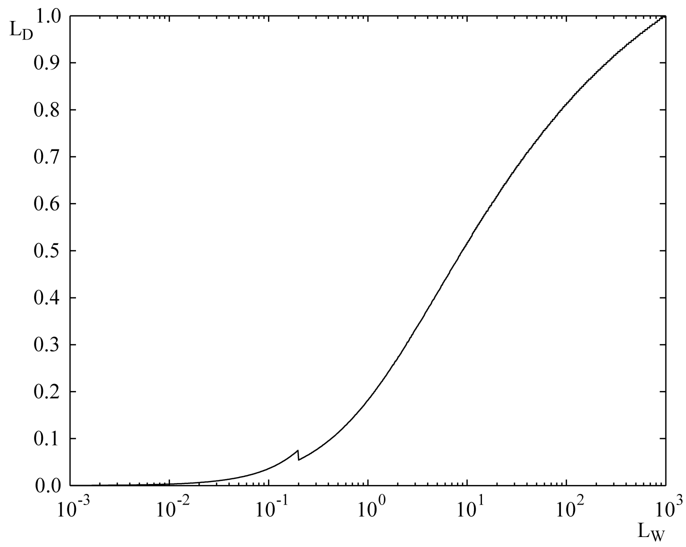


FIGURE 7.15 *Drago's logarithmic mapping. The discontinuity at* $L_{\mathrm{W}} = 0.2$ *is most likely an artifact of our implementation.*

image in a well-controlled manner. Higher values result in less contrast and better compression, whereas smaller values increase the available contrast.

A curve of this operator is plotted in Figure 7.15. The small discontinuity near $L_{\mathrm{W}} = 0.2$ is in all likelihood an artifact of our implementation.

7.2.7 REINHARD AND DEVLIN PHOTORECEPTOR MODEL

Logarithmic compression may be viewed as effectively computing a density image. The output therefore resembles the information stored in a negative. Although this metaphor holds, logarithmic responses are also sometimes attributed to parts of

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 37

the human visual system. This is intrinsically incorrect, although the human visual system responds approximately logarithmically over *some* of its operating range (see Section 6.2). Cells in the human visual system communicate with impulse trains, wherein the frequency of these impulse trains carries the information. Notable exceptions are the first few layers of cells in the retina, which communicate by generating graded potentials. In any case, this physiological substrate does not enable communication of negative numbers. The impulse frequency may become zero, but there is no such thing as negative frequencies. There is also an upper bound to realizable impulse frequencies.

Logarithms, on the other hand, may produce negative numbers. For large input values, the output may become arbitrarily large. At the same time, over a range of values the human visual system may produce signals that appear to be logarithmic. Outside this range, responses are no longer logarithmic but tail off instead. A class of functions that approximates this behavior reasonably well are sigmoids, or S-shaped functions, as discussed in Chapter 6. When plotted on a log-linear graph, the middle portion of such sigmoids is nearly linear and thus resembles logarithmic behavior. Moreover, sigmoidal functions have two asymptotes: one for very small values and one for large values.

This gives sigmoidal functions the right mathematical properties to be a possible candidate for modeling aspects of the human visual system. Evidence from electrophysiology confirms that photoreceptors of various species produce output voltages as a function of light intensity received that may be accurately modeled by sigmoids.

Naka and Rushton were the first to measure photoreceptor responses, and managed to fit a sigmoidal function to their data [87]. For the purpose of tone reproduction, the following formulation by Hood et al. is practical [52].

$$V(x, y) = \frac{I(x, y)}{I(x, y) + \sigma(I_a(x, y))}$$

Here, I is the photoreceptor input, V is the photoreceptor response, and σ is the semisaturation constant (which is a function of the receptor's adaptation level I_a). The semisaturation constant thus determines to which value of V the adaptation level is mapped, and therefore provides the flexibility needed to tailor the curve to the image being tone mapped. For practical purposes, the semisaturation constant may be computed from the adaptation value I_a as follows.

$$\sigma\big(I_a(x, y)\big) = \big(f\, I_a(x, y)\big)^m$$

In this equation, f and m are user parameters that need to be specified on a per-image basis. The scale factor f may be used to steer the overall luminance of the tone-mapped image and can initially be estimated as 1. Images created with different values of f are shown in Figure 7.16.

It should be noted that in electrophysiological studies the exponent m also features and tends to lie between 0.2 and 0.9 [52]. A reasonable initial estimate for m may be derived from image measures such as the minimum, maximum, and average luminance, as follows.

$$m = 0.3 + 0.7k^{1.4}$$
$$k = \frac{L_{\max} - L_{\mathrm{av}}}{L_{\max} - L_{\min}}$$

The parameter k may be interpreted as the key of the image (i.e., a measure of how light or dark the image is on average). The nonlinear mapping from k to exponent m is determined empirically. The exponent m is used to steer the overall impression of contrast, as shown in Figure 7.17.

A tone-reproduction operator may be created by equating display values to the photoreceptor output V, as demonstrated by Reinhard and Devlin [108]. Note that this operator is applied to each of the red, green, and blue color channels separately. This is similar to photoreceptor behavior, in which each of the three different cone types is thought to operate largely independently. Also note that sigmoidal functions that are part of several color appearance models — such as the Hunt model [55], CIECAM97 [54], and CIECAM02 [84] (see Section 2.8) — are executed independently to the red, green, and blue channels. This approach may account for the Hunt effect, which predicts desaturation of colors for both light and dark pixels, but not for pixels with intermediate luminances [55].

The adaptation level I_a may be computed in traditional fashion, for instance, as the (log) average luminance of the image. However, additional interesting features, such as light adaptation and chromatic adaptation, may be modeled by a slightly more elaborate computation of I_a.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 39

FIGURE 7.16 *Luminance control with user parameter f in Reinhard and Devlin's photoreceptor-based operator. User parameter f, set here to $\exp(-8)$, $\exp(0) = 1$ and $\exp(8)$. The top right-hand image shows the default value.*

Strong color casts may be removed by interpolating between the luminance value $L(x, y)$ of the pixel and the red, green, and blue values of each pixel $I_{r|g|b}(x, y)$. This produces a different adaptation level for each pixel individually, which is controlled by a user-specified interpolation weight c, as follows.

$$I_a(x, y) = cI_{r|g|b}(x, y) + (1 - c)L(x, y)$$

FIGURE 7.17 *The exponent m in Reinhard and Devlin's operator. Images are shown with exponent m set to 0.6, 0.7, and 0.8 (in reading order).*

This approach achieves a von Kries style of color correction by setting c equal to 1, whereas no color correction is applied if c equals 0. We also call this color adjustment "chromatic adaptation." Its effect is shown in Figure 7.18 for three values of c.

Similarly, the adaptation level (see Figure 7.19) may be thought of as determined by the current light level to which a receptor is exposed, as well as levels to which the receptor was exposed in the recent past. Because the eye makes saccadic eye

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 41

FIGURE 7.18 Simulation of chromatic adaptation in Reinhard and Devlin's photoreceptor-based operator. The level of chromatic adaptation may be approximated by setting user parameter c (shown here with values of 0.0, 0.5, and 1.0).

movements, and because there is the possibility of lateral connectivity within the retina, we may assume that the current adaptation level is a function of the pixel value itself and all other pixels in the image. This has given rise to all manner of spatially varying tone-reproduction models (see Section 7.3), but here a much faster and simpler solution is used (namely, interpolation between pixel values and global averages), as follows.

$$I_a(x, y) = a I_{r|g|b}(x, y) + (1 - a) I^{av}_{r|g|b}$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 42

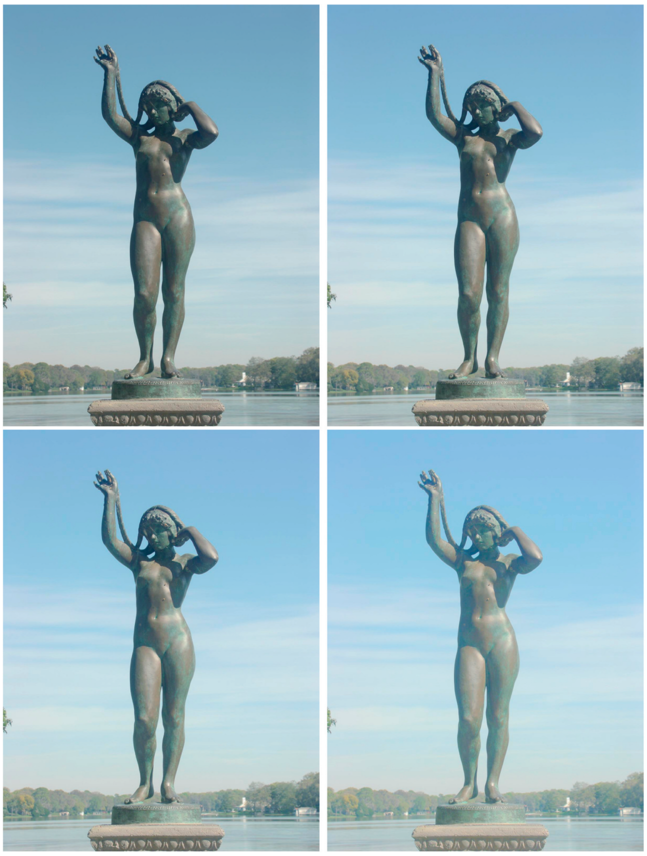


FIGURE 7.19 Simulation of light adaptation in Reinhard and Devlin's operator. The level of light adaptation may be approximated by setting user parameter a to 0, 1/3, 2/3, and 1. (Image courtesy of the Albin Polasek Museum, Winter Park, Florida.)

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 43

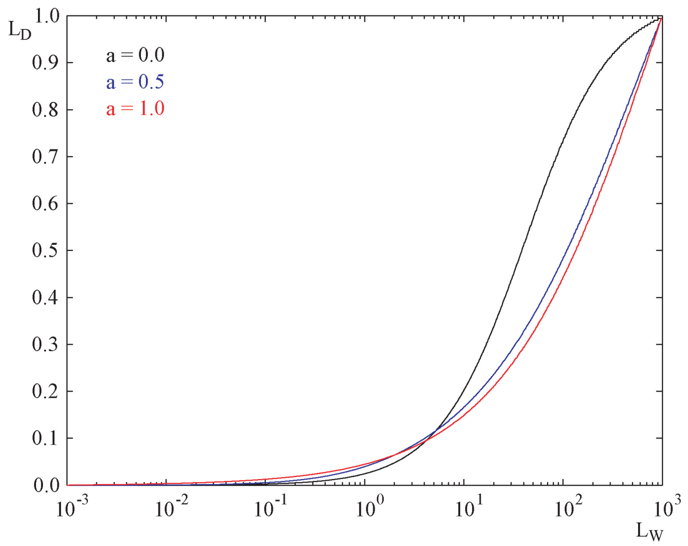


FIGURE 7.20 Luminance mapping by the photoreceptor-based operator for different values of user parameter a.

The interpolation weight a is user specified and controls image appearance, which to some extent correlates with light adaptation. Plots of the operator for different values of a are presented in Figure 7.20. Light adaptation and chromatic adaptation may be combined by bilinear interpolation, as follows.

$$\begin{aligned}
I_a^{\text{local}}(x,y) &= cI_{r|g|b}(x,y) + (1-c)L(x,y) \\
I_a^{\text{global}} &= cI_{r|g|b}^{\text{av}} + (1-c)L^{\text{av}} \\
I_a(x,y) &= aI_a^{\text{local}}(x,y) + (1-a)I_a^{\text{global}}
\end{aligned}$$

This operator is directly inspired by photoreceptor physiology. Using default parameters, it provides plausible results for a large class of images. Most other results may be optimized by adjusting the four user parameters. The value of c determines to what extent any color casts are removed, a and m affect the amount of contrast in the tone-mapped image, and f' make the overall appearance lighter or darker. Because each of these parameters has an intuitive effect on the final result, manual adjustment is fast and straightforward.

7.2.8 WARD HISTOGRAM ADJUSTMENT

Most global operators define a parametric curve with a few parameters that are estimated from the input image or that need to be specified by the user. Histogram enhancement techniques provide a mechanism for adjusting the mapping in a more fine-grained, albeit automatic, manner. Image enhancement techniques manipulate images that are already LDR to maximize visibility or contrast. On the other hand, Ward et al. borrow key ideas from histogram enhancement techniques to reproduce HDR images on LDR displays, simulating both visibility and contrast [142]. Their technique is termed *histogram adjustment*.

The simulation of visibility and contrast serves two purposes. First, the subjective correspondence between the real scene and its displayed image should be preserved so that features are only visible in the tone-mapped image if they were also visible in the original scene. Second, the subjective impression of contrast, brightness, and color should be preserved.

The histogram adjustment operator computes a histogram of a density image (i.e., the log of all pixels taken first) to assess the distribution of pixels over all possible luminance values. The shape of its associated cumulative histogram may be directly used to map luminance values to display values. However, further restrictions are imposed on this mapping to preserve contrast based on the luminance values found in the scene and on how the human visual system would perceive those values. As a postprocessing step, models of glare, color sensitivity, and visual acuity may further simulate aspects of human vision.

The histogram is calculated by first downsampling the image to a resolution that corresponds roughly to 1 degree of visual angle. Then the logarithm of the downsampled image is taken and its histogram is computed. The minimum and maximum log luminance values are taken to define the range of the histogram, with the exception that if the minimum log luminance value is smaller than -4 this value is used as the lower bound of the histogram. This exception models the lower threshold of human vision. The number of bins in the histogram is 100, which in practice provides a sufficiently accurate result. If $f(b_i)$ counts the number of pixels that lie in bin b_i, the cumulative histogram $P(b)$, normalized by the total number of pixels T, is defined as

$$P(b) = \sum_{b_i < b} f(b_i)/T$$
$$T = \sum_{b_i} f(b_i).$$

A naïve contrast equalization formula may be constructed from the cumulative histogram and the minimum and maximum display luminances, as follows.

$$\log\bigl(L_\mathrm{d}(x, y)\bigr) = \log(L_{\mathrm{d,min}}) + \bigl(\log(L_{\mathrm{d,max}}) - \log(L_{\mathrm{d,min}})\bigr) P\bigl(\log L_\mathrm{w}(x, y)\bigr)$$

This approach has a major flaw in that wherever there is a peak in the histogram, contrasts may be expanded rather than compressed. Exaggeration of contrast is highly undesirable and is avoidable through the following refinement. Based on the observation that linear tone mapping produces reasonable results for images with a limited dynamic range, contrasts due to histogram adjustment should not exceed those generated by linear scaling. That is,

$$\frac{dL_\mathrm{d}}{dL_\mathrm{w}} \le \frac{L_\mathrm{d}}{L_\mathrm{w}}.$$

Because the cumulative histogram is the numerical integration of the histogram, we may view the histogram itself as the derivative of the cumulative histogram—provided it is normalized by T and the size of a bin δb is small, and thus

$$\frac{dP(b)}{db} = \frac{f(b)}{T\delta b}$$
$$\delta b = \frac{1}{N} \log(L_{\max}) - \log(L_{\min}).$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 46

This naïve histogram equalization gives an expression for the display luminance L_{d} as a function of world luminance L_{w}. Its derivative may therefore be plugged into the previous inequality to yield a ceiling on $f(b)$, as follows.

$$L_{\mathrm{d}} \frac{f(\log(L_{\mathrm{w}}))}{T\delta b} \frac{\log(L_{\mathrm{d,max}}) - \log(L_{\mathrm{d,min}})}{L_{\mathrm{w}}} = \frac{L_{\mathrm{d}}}{L_{\mathrm{w}}}$$

$$\frac{T\delta b}{\log(L_{\mathrm{d,max}}) - \log(L_{\mathrm{d,min}})} \geq f(b)$$

This means that as long as $f(b)$ does not exceed this ceiling, contrast will not be exaggerated. For bins with a higher pixel count, the simplest solution is to truncate $f(b)$ to the ceiling. Unfortunately, this changes the total pixel count T in the histogram, which by itself will affect the ceiling. This may be solved by an iterative scheme that stops if a certain tolerance is reached. Details of this approach are given in [142].

A second refinement is to limit the contrast according to human vision. The linear ceiling described previously assumes that humans detect contrast equally well over the full range of visible luminances. This assumption is not correct, prompting a solution that limits the contrast ceiling according to a just-noticeable difference function δL_{t}. This function takes an adaptation value L_{a} as a parameter, as follows. (This is the same as the function used by Ferwerda's model of visual adaptation. See Section 7.2.4.)

$$\delta L_{\mathrm{t}}(L_{\mathrm{a}}) = \begin{cases} -2.86 \text{ for } \log_{10}(L_{\mathrm{a}}) < -3.94 \\ (0.405 \log_{10}(L_{\mathrm{a}}) + 1.6)^{2.18} - 2.86 \text{ for } -3.94 \leq \log_{10}(L_{\mathrm{a}}) < -1.44 \\ \log_{10}(L_{\mathrm{a}}) - 0.395 \text{ for } -1.44 \leq \log_{10}(L_{\mathrm{a}}) < -0.0184 \\ (0.249 \log_{10}(L_{\mathrm{a}}) + 0.65)^{2.7} - 0.72 \text{ for } -0.0184 \leqslant \log_{10}(L_{\mathrm{a}}) < 1.9 \\ \log_{10}(L_{\mathrm{a}}) - 1.255 \text{ for } \log_{10}(L_{\mathrm{a}}) \geq 1.9 \end{cases}$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 47

FIGURE 7.21 *Example images tone mapped with the histogram adjustment operator. The mappings produced for these images are plotted in Figure 7.22.*

This yields the following inequality and ceiling on $f(b)$, which also requires an iterative scheme to solve.

$$\frac{dL_\mathrm{d}}{dL_\mathrm{w}} \le \frac{\delta L_\mathrm{t}(L_\mathrm{d})}{\delta L_\mathrm{t}(L_\mathrm{w})}$$

$$f(b) \le \frac{\delta L_\mathrm{t}(L_\mathrm{d})}{\delta L_\mathrm{t}(L_\mathrm{w})} \cdot \frac{T\,\delta b L_\mathrm{w}}{(\log_{10}(L_\mathrm{d,max}) - \log_{10}(L_\mathrm{d,min}))L_\mathrm{d}}$$

The result is a practical hands-off tone-reproduction operator that produces plausible results for a wide variety of HDR images. Because the operator adapts to each image individually, the mapping of world luminance values to display values will be different for each image. As an example, two images are shown in Figure 7.21. The mappings for these two images are shown in Figure 7.22.

Further enhancements model human visual limitations such as glare, color sensitivity, and visual acuity. Veiling glare is caused by bright light sources in the periphery of vision, which cause light scatter in the ocular media. Light scatter causes a reduction of contrast near the projection of the glare source on the retina.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 48

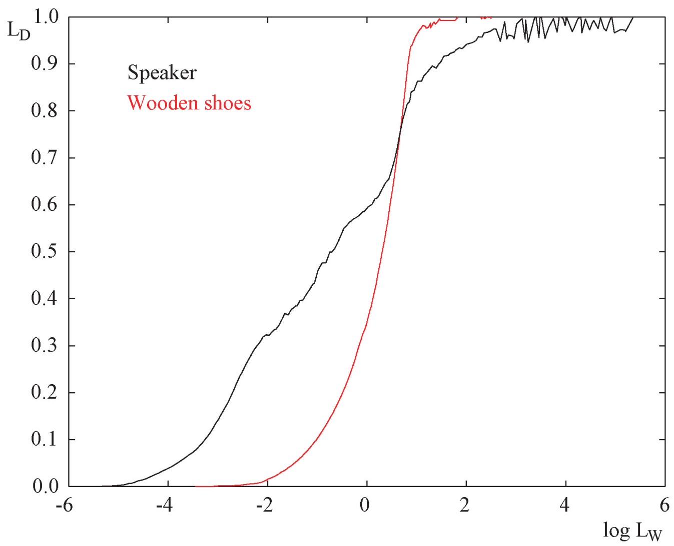


FIGURE 7.22 *Mapping of world luminances to display luminance for the images shown in Figure 7.21.*

In dark environments, color sensitivity is lost because only one type of receptor is active. In brighter environments, the three cone types are active and their relative activity is used by the human visual system to infer the spectral composition of the scene it is viewing. Finally, in dark environments visual acuity is lost because only very few rods are present in the fovea.

The histogram adjustment technique may accommodate each of these effects, and we refer to Ward's original paper for a full description [142]. Figure 7.23 shows a daytime image processed with the various options afforded by this operator, and Figure 7.24 shows the same applied to a nighttime image.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 49

FIGURE 7.23 *Histogram adjustment with its various simulations of human visual limitations for a daylight scene. In reading order: histogram adjustment, histogram adjustment with simulation of visual acuity loss, veiling glare, color sensitivity, and contrast sensitivity. The final image shows a combination of all of these. Compare with Figure 7.24, which shows the same techniques applied to a nighttime image.*

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 50

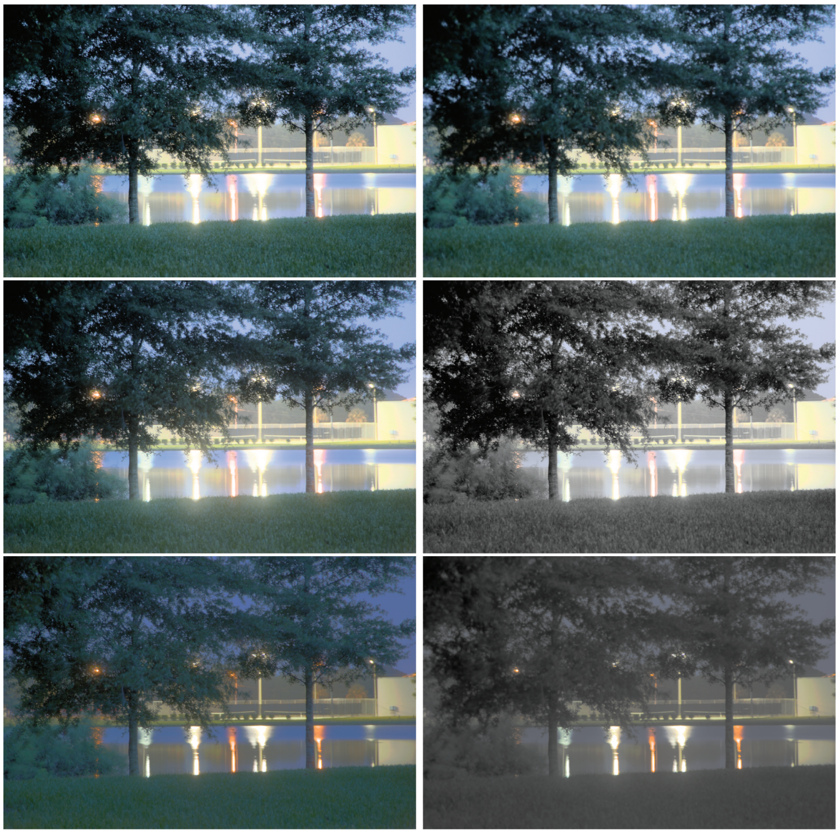


FIGURE 7.24 *Histogram adjustment with its various simulations of human visual limitations for a night scene. In reading order: histogram adjustment, histogram adjustment with simulation of visual acuity loss, veiling glare, color sensitivity, and contrast sensitivity. The final image shows a combination of all of these.*

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 51

7.2.9 SCHLICK UNIFORM RATIONAL QUANTIZATION

Uniform rational quantization is aimed at providing an improved tone-reproduction operator as compared with simple logarithmic mappings and ad hoc procedures (such as gamma correction) for the purpose of dynamic range reduction. It was not developed to be an alternative to more complete perceptually-based operators. However, this method does provide a simple scheme with only two user parameters.

A rational function is defined as the quotient of two polynomials. The specific mapping function proposed by Schlick is [113] as follows.

$$L_{\mathrm{d}}(x, y) = \frac{p L_{\mathrm{w}}(x, y)}{(p-1) L_{\mathrm{w}}(x, y) + L_{\max}} \quad \text{where } p \in [1, \infty)$$

This function bears some resemblance to sigmoidal functions (see also Section 6.3.1), although instead of a semisaturation constant the maximum world luminance $L_{\max}$ is used, and instead of an exponent to control the overall appearance a scale factor p is introduced.

The value of p may be estimated such that the smallest value that is not black remains just visible after tone mapping. This JND δL_0 in quantized display luminance steps should be specified by the user. A simple way of determining this value is to show an image with various patches on a black background. The patches will vary in gray level. The user then selects the darkest patch that is just visible. The parameter p may then be approximated by

$$p = \frac{\delta L_0}{N} \cdot \frac{L_{\max}}{L_{\min}},$$

where N is the number of different luminance levels that can be reproduced by the display device. For 8-bit display devices, its value will be 256. Figure 7.25 shows results created with different values of δL_0. A plot of this operator is shown in Figure 7.26. An empirically determined refinement to the previous uniform rational quantization scheme uses the following slightly different form, which also depends on the pixel's luminance value.

$$p = \frac{\delta L_0}{N} \cdot \frac{L_{\max}}{L_{\min}} \left(1 - k + k \frac{L_{\mathrm{w}}(x, y)}{\sqrt{L_{\min} L_{\max}}} \right)$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 52

FIGURE 7.25 Using Schlick's uniform rational quantization, the just-noticeable difference from black was set to 10^{-7}, 10^{-6}, 10^{-5}, and 10^{-4}.

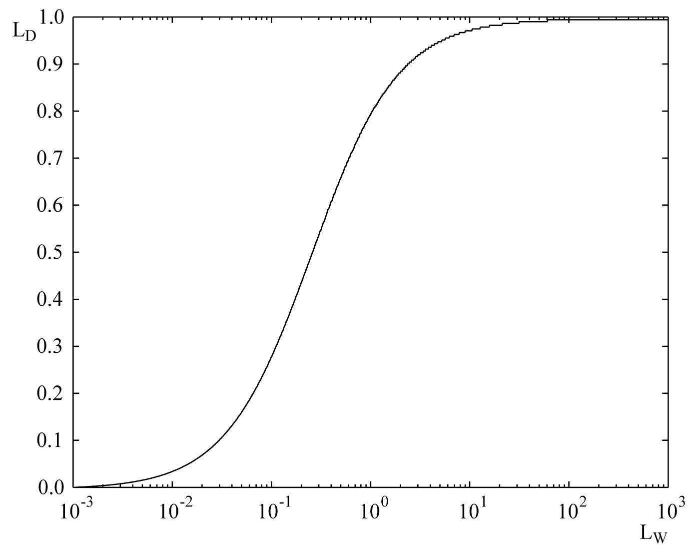


FIGURE 7.26 *Schlick's uniform rational quantization shown for a value of $L_0 = 1$.*

Schlick's original intent was to extend the uniform rational quantization function to be spatially varying. The pixel's luminance value in this formulation would then be replaced by a weighted average of the pixel's luminance and its neighbors. However, he found that the best results were obtained by making the local neighborhood no larger than the pixel itself. This yields the previous formulation, which is no longer spatially varying.

The user parameter k should be specified in the range $[0,1]$. Its effect on a tone-mapped image is shown in Figure 7.27.

Schlick's operator produces plausible results and is computationally efficient. However, it may be somewhat difficult to find values for the two user parameters without some experimentation.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 54

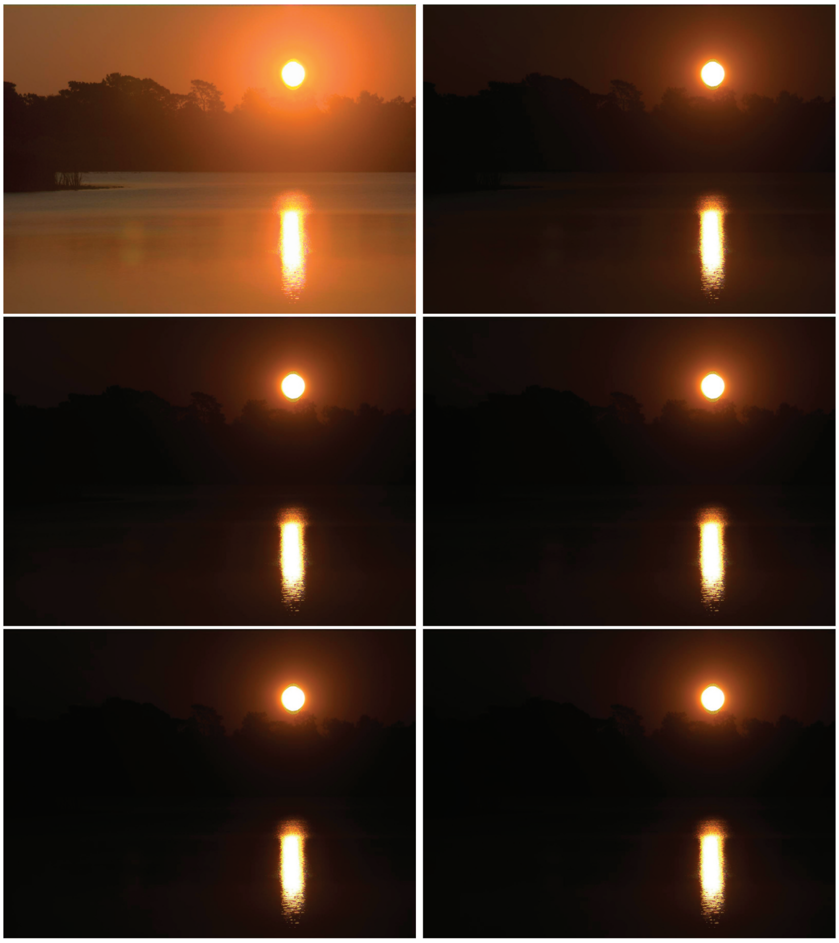


FIGURE 7.27 User parameter k of Schlick's uniform rational quantization varied between 0.0 and 1.0 in steps of 0.2.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 55

7.3 LOCAL OPERATORS

Global operators are characterized by a mapping of world luminances to display luminances that is identical for all pixels (i.e., a single tone-mapping curve is used throughout the image). This makes them computationally efficient, but there is a limit to the dynamic range of the input image beyond which successful compression becomes difficult. Global operators are of necessity monotonically increasing operators. Otherwise, visually unpleasant artifacts will be introduced. Because display devices can usually not accommodate more than 256 levels, all world luminances must be mapped to that range and quantized to unit increments. The higher the dynamic range of an image, the more values must be mapped to 256 different numbers by a monotonically increasing function. For extreme HDR images this will almost inevitably lead to loss of visibility or contrast, or both.

Thus, global operators are limited in their capacity to compress HDR images. To some extent this limit may be lifted by local operators by compressing each pixel according to its luminance value, as well as to the luminance values of a set of neighboring pixels. Thus, instead of anchoring the computation to a globally derived quantity (such as the image's log average value) for each pixel the computation is adjusted according to an average over a local neighborhood of pixels.

Local operators more often than not mimic features of the human visual system. For instance, a reasonable assumption is that a viewer does not adapt to the scene as a whole, but to smaller regions instead. An active observer's eyes tend to wander across the scene, focusing on different regions. For each focus point, there is a surrounding region that helps determine the state of adaptation of the viewer.

For tone-reproduction operators this has the implication that we may be able to compute an adaptation level individually for each pixel by considering the pixel itself and a set of neighboring pixels. Classic problems to be solved by local tone-reproduction operators are to determine how many neighboring pixels need to be included in the computation, how to weight each neighboring pixel's contribution to the local adaptation level, and how to use this adaptation level within a compressive function. These issues are solved differently by the operators described in this section.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 56

7.3.1 CHIU SPATIALLY VARIANT OPERATOR

The first to observe that a spatially varying operator may be useful for tone reproduction were Chiu et al. [9]. They noted that artists frequently make use of spatially varying techniques to fool the eye into thinking that a much larger dynamic range is present in artwork than actually exists. In particular, the areas around bright features may be dimmed somewhat to accentuate them. The basic formulation of their operator, as follows, multiplies each pixel's luminance by a scaling factor $s(x, y)$, which depends on the pixel itself and its neighbors.

$$L_{\mathrm{d}}(x, y) = s(x, y) L_{\mathrm{w}}(x, y)$$

For $s(x, y)$ to represent a local average, we may produce a low-pass filtered version of the input image. Chiu et al. note that most low-pass filters produce similar results. For demonstration purposes, we show the technique with a Gaussian filter with a width controlled by a user parameter (see Section 7.1.4).

In a blurred image, each pixel represents a weighted local average of the pixel in the corresponding position in the input image. The reciprocal of these blurred pixels may be used to compress HDR images, as follows.

$$L_{\mathrm{d}}(x, y) = \frac{1}{k L_{\mathrm{w}}^{\mathrm{blur}}(x, y)} L_{\mathrm{w}}(x, y)$$

Here, k is a constant of proportionality (a user parameter) that controls the weight given to the blurred image relative to the unblurred input. This approach immediately highlights one of the main problems faced by all local operators: halos arise around bright features. These halos, or contrast reversals, are more often disturbing than helpful. However, at the same time we have argued that artists use such dimming of areas around bright objects with great success. We conclude that some halos are good and some are bad. Finding a spatially variant tone-reproduction operator that does not produce obtrusive halos is a challenge. In our opinion, some operators succeed better than others.

To illustrate the haloing problem, we created a series of images (using the previous formulation) with different values of k, shown in Figure 7.28. This places more or less weight on the Gaussian blurred image, which was chosen with a kernel size of 128 pixels in each case. In that k controls the relative contribution of the

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 57

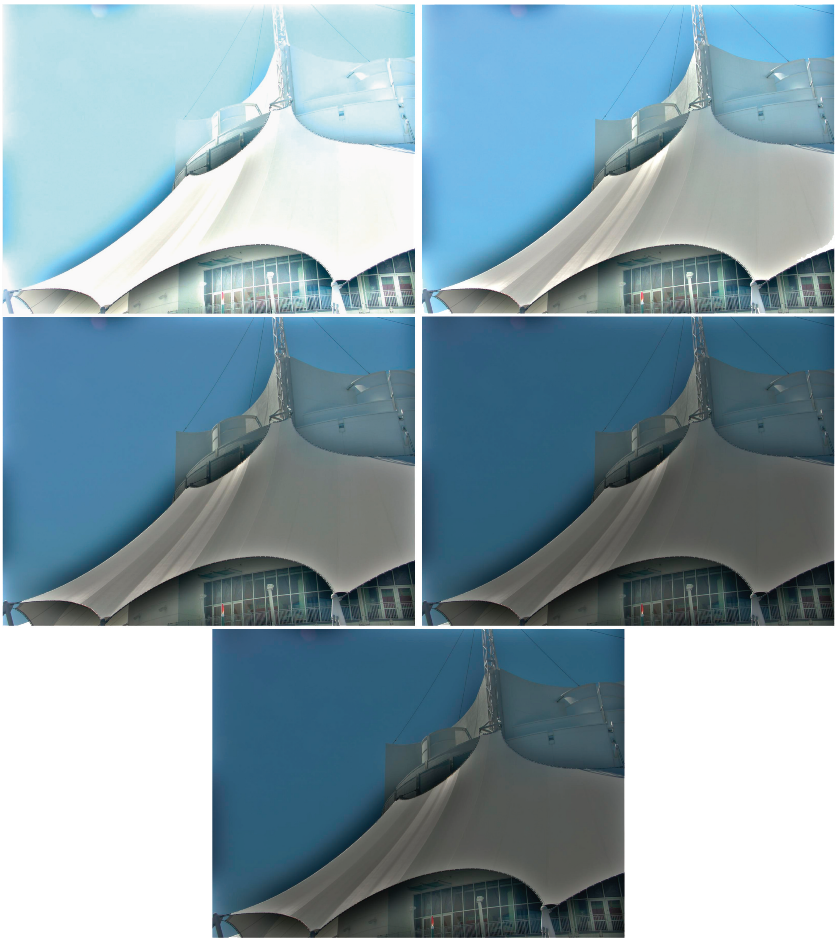


FIGURE 7.28 *The relative weight of the Gaussian-blurred image was controlled by specifying user parameter* k*, which was given values of* 1*,* 2*,* 4*,* 8*, and* 16*. This parameter thus varies the strength of the halo.*

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 58

FIGURE 7.29 *Effect of kernel size on image quality in Chiu's operator. The width of the Gaussian kernel was varied from 4 pixels to 128 pixels, doubling its width with each consecutive image.*

Gaussian blurred image to the final result, it affects the strength of the halos and the amount of achievable compression.

Whereas k controls the depth of the Gaussian, the kernel size may also be varied by a user parameter. Its effect is shown in Figure 7.29. The extreme haloing effect seen for small Gaussians starts to disappear for larger Gaussians. The image created with a kernel size of 128 pixels looks plausible. Although the halos in the bottom right-hand image of Figure 7.29 are not absent, we believe that here the transition between bad halos and good halos can be seen. This figure is in agreement with the observation made by Chiu et al. that wide filter kernels need to be used for local operators of this form to produce plausible results.

Chiu's original implementation included a smoothing stage that would iterate at least 1,000 times over the image with a small filter kernel. This would somewhat reduce the effect of contrast reversals. This approach is too expensive to be practical, however, and we therefore did not include this stage in our experimentation.

Chiu's work is intended to be exploratory and is of interest because it highlights the issues faced by other local tone-reproduction operators. Dependent on the application, halos may be desirable or completely undesirable. In any case, contrast reversals are a feature of most spatially varying operators. The extent to which they are visible depends on the method chosen and the amount of parameter tuning applied.

7.3.2 RAHMAN RETINEX

Whereas Chiu's work is exploratory and is not advertised as a viable tone-reproduction operator, Rahman and Jobson developed their interpretation of the retinex theory for use in various applications, including tone reproduction [59, 102,103]. However, the differences between their approach and Chiu's are relatively minor. They too divide the image by a Gaussian-blurred version with a wide filter kernel.

Their operator comes in two different forms: single-scale and multiscale. In the single-scale version, Chiu's model is followed closely, although the algorithm operates in the log domain. However, the placements of the logarithms are somewhat peculiar (namely, after the image is convolved with a Gaussian filter kernel), as follows.

$$I_{\mathrm{d}}(x,y) = \exp\bigl(\log\bigl(I_{\mathrm{w}}(x,y)\bigr) - k\log\bigl(I_{\mathrm{w}}^{\mathrm{blur}}(x,y)\bigr)\bigr)$$

This placement of logarithms is empirically determined to produce visually improved results. We add exponentiation to the results to return to a linear image.

Note that this operator works independently on the red, green, and blue channels, rather than on a single luminance channel. This means that the convolution that produces a Gaussian-blurred image needs to be repeated three times per image.

In the multiscale retinex version, this equation is repeated several times for Gaussians with different kernel sizes. This results in a stack of images, each image blurred by increasing amounts. In the following, an image at level n will be denoted $I_{\mathrm{w},n}^{\mathrm{blur}}$. In the examples we show in this section, we used a stack of six levels and made the smallest Gaussian filter kernel two pixels wide. Each successive image is convolved with a Gaussian twice as large as that of the previous image in the stack.

The multiscale retinex version is then simply the weighted sum of a set of single-scale retinexed images. The weight given to each scale is determined by the user. We have found that for experimentation it is convenient to weight each level by a power function, which gives straightforward control over the weights. For an image stack with N levels, the normalized weights are then computed by

$$w_n = \frac{(N-n-1)^f}{\sum_{m=0}^{N}(N-m-1)^f}.$$

A family of curves of this function is plotted in Figure 7.30. The user parameter f determines the relative weighting of each of the scales. For equal weighting, f should be set to 0. To give smaller scales more weight, f should be given a positive value (such as 0.1 or 0.2). If the larger scales should be emphasized, f should be given negative values. The multiscale retinex takes the following form.

$$I_{\mathrm{d}}(x,y) = \exp\left(\sum_{n=0}^{N} w_n\bigl(\log\bigl(I_{\mathrm{w}}(x,y)\bigr) - k\log\bigl(I_{\mathrm{w},n}^{\mathrm{blur}}(x,y)\bigr)\bigr)\right)$$

The two user parameters are k and f, which are in many ways equivalent to the user parameters required to control Chiu's operator. The value of k specifies the relative weight of the blurred image. Larger values of k will cause the compression to be

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 61

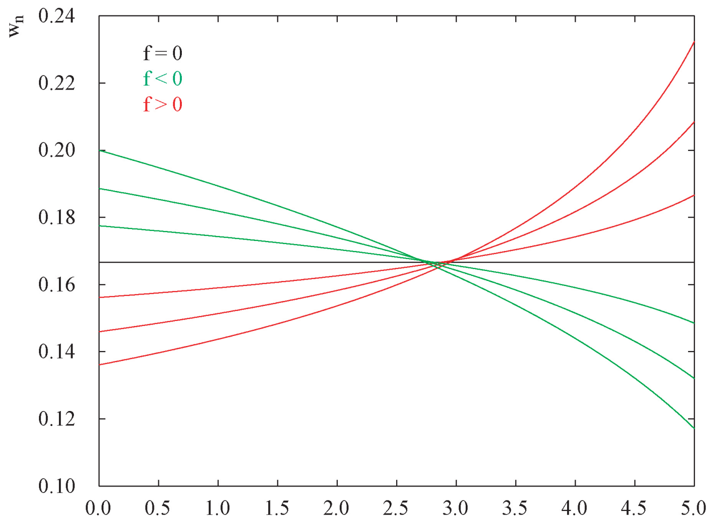


FIGURE 7.30 *Weight factors w_n as function of scale n for different user parameters f. The values for f used a range from -0.3 to 0.3 in steps of 0.1.*

more dramatic, but also create bigger halos. Parameter f, which controls the relative weight of each of the scales, determines which of the Gaussian-blurred images carries the most importance. This is more or less equivalent to setting the spatial extent of the Gaussian in Chiu's method. With these two parameters we therefore expect to be able to control the operator, balancing amount of compression against severity of the artifacts. This is indeed the case, as Figures 7.31 and 7.32 show.

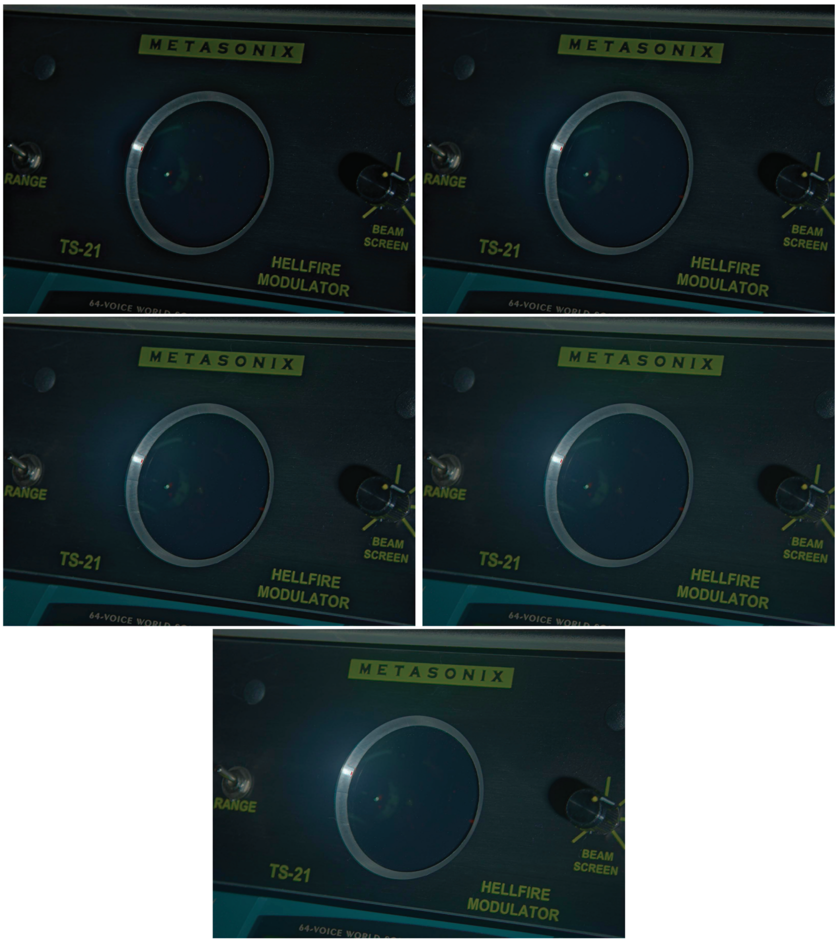
In summary, Rahman and Jobson's interpretation of Land's retinex theory is similar to the exploratory work by Chiu. There are three main differences. The algorithm works in the log domain, which causes contrasts at large image values to lie closer together. This generally results in fewer issues with haloing. Second, the algorithm operates on the three color channels independently. This approach is routinely fol-

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 62

FIGURE 7.31 In Rahman's retinex implementation, parameter k controls the relative weight of the Gaussian-blurred image stack. Here, k is varied from 0.0 to 0.5, and 1.0 (in reading order).

FIGURE 7.32 In Rahman's retinex implementation, the Gaussian-blurred images may be weighted according to scale. The most important scales are selected with user parameter f, which is varied between -2 to 2. Equal weight is given for a value of $f = 0$, shown on the left of the middle row.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 63



reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 64

lowed by various color appearance models (see, for instance, the CIECAM02 model discussed in Section 2.8, and the iCAM model discussed in material following). Finally, this work operates on multiple scales that are weighted relative to one another by a user-specified parameter. Multiscale techniques are well known in the literature, including the tone-reproduction literature. Other examples of multiscale techniques are the multiscale observer model, Ashikhmin's operator, and photographic tone reproduction, described respectively in Sections 7.3.4, 7.3.5, and 7.3.6.

7.3.3 FAIRCHILD iCAM

Although most operators discussed in this chapter are aimed at dynamic range reduction, Pattanaik's multiscale observer model [94] (discussed in the following section) and Fairchild's iCAM model [30] are both color appearance models.

Most color appearance models — such as CIECAM97, CIECAM02 and the Hunt model — are intended for use in simplified environments. It is normally assumed that a uniform patch of color is viewed on a larger uniform background with a different color. The perception of this patch of color may then be predicted by these models with the XYZ tristimulus values of the patch and a characterization of its surround as input, as described in Section 2.8.

Images tend to be more complex than just a patch on a uniform background. The interplay between neighboring pixels may require a more complex spatially variant model that can account for the local adaptation of regions around each pixel. This argument in favor of spatially variant color appearance models is effectively the same as the reasoning behind spatially variant tone-reproduction operators. The parallels between the iCAM model described here and operators such as Chiu's and Rahman's are therefore unmistakable. However, there are also sufficient differences to make a description of the model worthwhile.

The iCAM image appearance model is a direct refinement and simplification of the CIECAM02 color appearance model [30,61]. It omits the sigmoidal compression found in CIECAM02 but adds spatially variant processing in the form of two separate Gaussian-blurred images that may be viewed as adaptation levels. Like most color appearance models, the model needs to be applied in the forward direction and in the reverse direction.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 65

The input to the model is expected to be specified in XYZ device-independent coordinates. Like CIECAM02, the model uses various color spaces to execute the stages of the algorithm. The first stage is a chromatic adaptation transform, for which sharpened cone responses are used. Sharpened cone responses are obtained with the M_{CAT02} transform, given in Section 2.4.

The chromatic adaptation transform pushes the colors in the image toward the D_{65} white point. The amount of adaption in this von Kries transform is determined by a user parameter D, which specifies the degree of adaptation. In addition, for each pixel a white point $W(x, y)$ is derived from the XYZ image by applying a low-pass filter with a kernel a quarter the size of the image. This may be applied to each color channel independently for chromatic adaptation, or on the Y channel only for achromatic adaptation. This low-pass filtered image is then also converted with the M_{CAT02} matrix. Finally, the D_{65} white point — given by the $Y_{\mathrm{w}} = 95.05$, 100.0, 108.88 triplet — is also converted to sharpened cone responses. The subsequent von Kries adaptation transform is given by the following.

$$R_{\mathrm{c}}(x, y) = R'(x, y)\left(Y_{\mathrm{w}}\frac{D}{W_{R'}(x, y)} + 1 - D\right)$$

$$G_{\mathrm{c}}(x, y) = G'(x, y)\left(Y_{\mathrm{w}}\frac{D}{W_{G'}(x, y)} + 1 - D\right)$$

$$B_{\mathrm{c}}(x, y) = B'(x, y)\left(Y_{\mathrm{w}}\frac{D}{W_{B'}(x, y)} + 1 - D\right)$$

This transform effectively divides the image by a filtered version of the image. This step of the iCAM model is therefore similar to Chiu's and Rahman's operators. In those operators, the trade-off between amount of available compression and presence of halos is controlled by a scaling factor k. Here, D plays the role of the scaling factor. We may therefore expect this parameter to have the same effect as k in Chiu's and Rahman's operators. However, in the previous equation D also determines the amount of chromatic adaptation. It serves the same role as the degree of adaptation parameter found in other color appearance models (compare, for instance, with CIECAM02, described in Section 2.8).

For larger values of D, the color of each pixel is pushed closer to the D_{65} white point. Hence, in the iCAM model the separate issues of chromatic adaptation, haloing, and amount of compression are directly interrelated.

Figure 7.33 shows the effect of parameter D, which was given values of 0.0, 0.5, and 1.0. This figure also shows the effect of computing a single white point shared between the three values of each pixel and computing a separate white point for each color channel independently. For demonstration purposes, we have chosen an image with a higher dynamic range than usual. The halo visible around the light source is therefore more pronounced than for images with a medium dynamic range. Like Chiu's and Rahman's operators, the iCAM model appears most suited for medium-dynamic-range images.

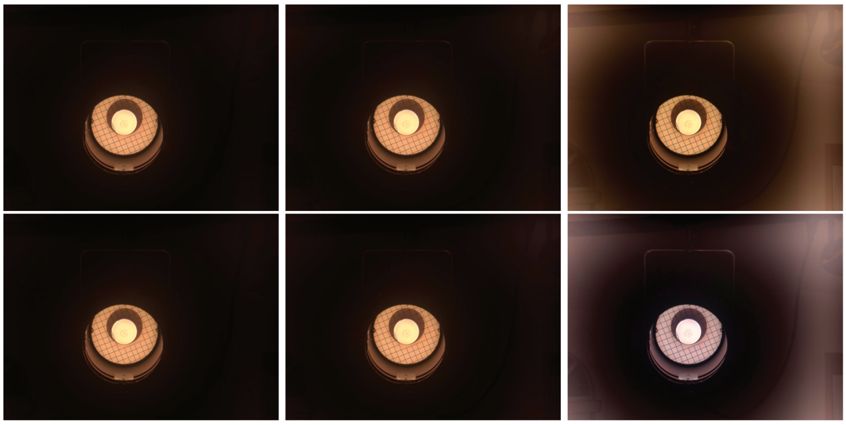


FIGURE 7.33 *The iCAM image appearance model. Top row: luminance channel used as adaptation level for all three channels. Bottom row: channels are processed independently. From left to right the adaptation parameter D was varied from 0.0 to 0.5 and 1.0.*

After the chromatic adaptation transform, further compression is achieved by an exponential function executed in LMS cone space (see Section 2.4). The exponential function that compresses the range of luminances is given by the following.

$$L'(x, y) = \left|L(x, y)\right|^{0.43 F_L(x,y)}$$
$$M'(x, y) = \left|M(x, y)\right|^{0.43 F_L(x,y)}$$
$$S'(x, y) = \left|S(x, y)\right|^{0.43 F_L(x,y)}$$

The exponent is modified on a per-pixel basis by F_{L}, which is a function of a spatially varying surround map derived from the luminance channel (Y channel) of the input image. The surround map $S(x, y)$ is a low-pass filtered version of this channel with a Gaussian filter kernel size of one-third the size of the image. The function F_{L} is then given by the following.

$$F_{\mathrm{L}}(x, y) = \frac{1}{1.7}\left(0.2\left(\frac{1}{5S(x, y) + 1}\right)^4 \left(5S(x, y)\right) + 0.1\left(1 - \left(\frac{1}{5S(x, y)}\right)^4\right)^2 \sqrt[3]{5S(x, y)}\right)$$

Thus, this computation of F_{L} may be seen as the spatially variant extension of CIECAM02's factor for partial adaptation, given in Equation 2.1.

This step completes the forward application of the iCAM model. To prepare the result for display, the inverse model should be applied. The model requires the same color spaces to be used as in the forward model in each of the steps. The first step is to invert the previous exponentiation, as follows.

$$L'(x, y) = \left|L(x, y)\right|^{1/0.43}$$
$$M'(x, y) = \left|M(x, y)\right|^{1/0.43}$$
$$S'(x, y) = \left|S(x, y)\right|^{1/0.43}$$

The inverse chromatic adaptation transform does not require a spatially variant white point, but converts from a global D_{65} white point $Y_{\mathrm{w}} = 95.05, 100.0, 108.88$

to an equiluminant white point $Y_\mathrm{e} = 100{,}100{,}100$. Because full adaptation is assumed, D is set to 1 and this transform simplifies to the following scaling, which is applied in sharpened cone response space.

$$R' = R\frac{Y_\mathrm{e}}{Y_\mathrm{w}}$$

$$G' = G\frac{Y_\mathrm{e}}{Y_\mathrm{w}}$$

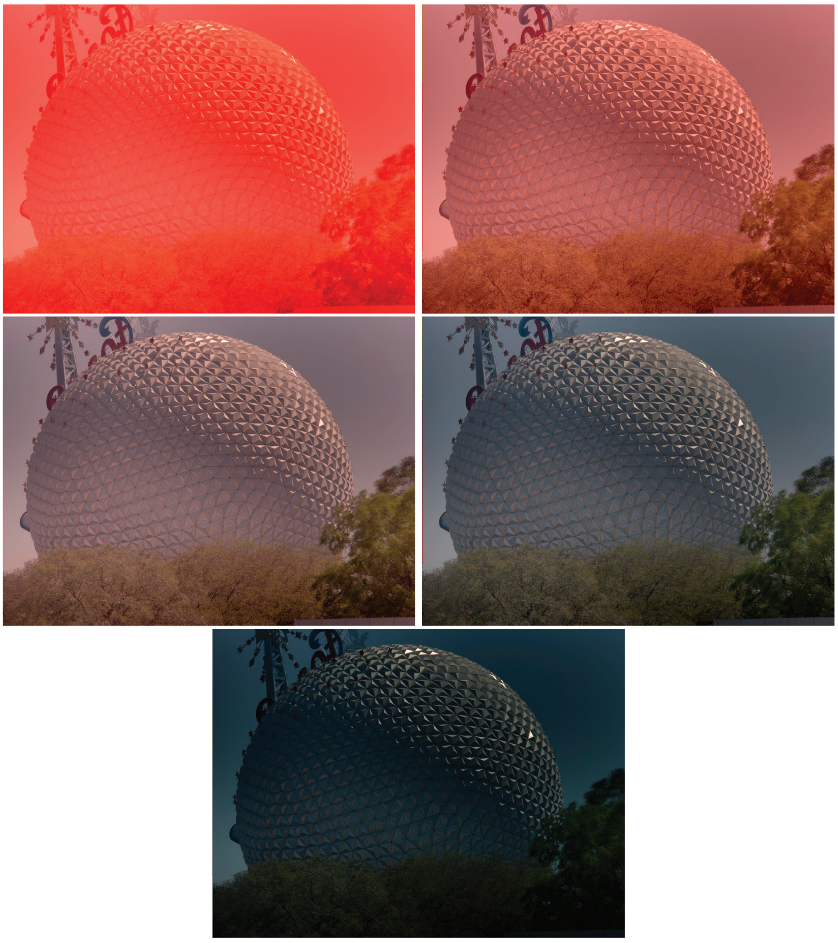
$$B' = B\frac{Y_\mathrm{e}}{Y_\mathrm{w}}$$

After these two steps are executed in their appropriate color spaces, the final steps consist of clipping the top 99% of all pixels, normalization, and gamma correction. The user parameters for this model are D, as discussed previously, and a prescaling of the input image. This prescaling may be necessary because the iCAM model requires the input to be specified in $\mathrm{cd/m}^2$. For arbitrary images, this requires the user to scale the image to its appropriate range prior to tone mapping. The effect of pre-scaling is shown in Figure 7.34. For images that contain values that are too small, a red shift is apparent. If the values in the image are too large, the overall appearance of the image becomes too dark.[3]

Further parameters for consideration are the kernel sizes of the two Gaussian filters. For the images shown in this section, we used the recommended kernel sizes of 1/4 and 1/3 the size of the image, but other sizes are possible. As with Chiu's and Rahman's operators, the precise kernel size is unimportant, as long as the filter width is chosen to be large.

FIGURE 7.34 *Effect of pre-scaling on the iCAM model. The factor used for the top left-hand image was 0.01 and each subsequent image was scaled with a factor 10 times larger than the previous image.*

3 The images in this figure, as with similar image sequences for other operators, were scaled beyond a reasonable range — too small and too large — to show the effect of the parameter. It should be noted that in practice a reasonable parameter setting should be chosen to avoid such extremes.



In summary, the iCAM model consists of two steps: a chromatic adaptation step followed by an exponential function. The achromatic adaptation step strongly resembles Chiu's and Rahman's operators because the image is divided by a blurred version of the image. The second step may be viewed as an advanced form of gamma correction, whereby the gamma factor is modulated on a per-pixel basis. The forward model needs to be followed by the inverse application of the model to prepare the image for display. A final clipping and normalization step brightens the overall appearance. The model is best suited for images with a medium dynamic range, in that the trade-off between compression and presence of halos is less critical for this class of images than for extreme HDR images.

7.3.4 PATTANAIK MULTISCALE OBSERVER MODEL

Pattanaik's multiscale observer model ranks among the more complete color appearance models and consists of several steps executed in succession [94]. The output of this model (and all other color appearance models) are color appearance correlates, as discussed in Section 2.8. A tone-reproduction operator may be derived from these correlates by executing the inverse model and substituting characteristics of the display device into the equations in the appropriate place.

For simplicity, we present a version of the model that is reduced in complexity. For the purpose of tone reproduction, some of the forward and backward steps of the model cancel out and may therefore be omitted. In addition, compared to the original model we make small changes to minimize visual artifacts, for instance by choosing the filter kernel sizes smaller than in the original model. We first give a brief overview of the full model and then detail a simplified version.

The first step in the forward model is to account for light scatter in the ocular media, followed by spectral sampling to model the photoreceptor output. This yields four images representing the rods and the L, M, and S cones. These four images are then each spatially decomposed into seven-level Gaussian pyramids and subsequently converted into four six-level difference-of-Gaussian (DoG) stacks that represent bandpass behavior as seen in the human visual system. DoGs are computed by subtracting adjacent images in the pyramid.

The next step consists of a gain control system applied to each of the DoGs in each of the four channels. The shape of the gain control function resembles TVI curves such that the results of this step may be viewed as adapted contrast pyramidal images. The cone signals are then converted into a color opponent scheme that contains separate luminance, red-green, and yellow-blue color channels. The rod image is retained separately.

Contrast transducer functions that model human contrast sensitivity are then applied. The rod and cone signals are recombined into an achromatic channel, as well as red-green and yellow-blue color channels. A color appearance map is formed next, which is the basis for the computation of the aforementioned appearance correlates. This step cancels in the inverse model, and we therefore omit a detailed description. We also omit computing the rod signals because we are predominantly interested in photopic lighting conditions.

The model calls for low-pass filtered copies with spatial frequencies of 0.5, 1, 2, 4, 8, and 16 cycles per degree (cpd). Specifying spatial frequencies in this manner is common practice when modeling the human visual system. However, for a practical tone-reproduction operator this would require knowledge of the distance of the observer to the display device and the spatial resolution of the display device. Because viewer distance is difficult to control, let alone anticipate, we restate spatial frequencies in terms of cycles per pixel (cpp).

Further, we omit the initial modeling of light scatter in the ocular media. Modeling light scatter would have the effect of introducing a small amount of blur in the image, particularly near areas of high luminance. On occasion, modeling of glare may be important and desirable and should be included in a complete implementation of the multiscale observer model. However, for simplicity we omit this initial processing. This set of simplifications allows us to focus on the part of the multiscale observer model that achieves dynamic range reduction.

The model expects input to be specified in LMS cone space, discussed in Section 2.4. The compressive function applied in all stages of the multiscale observer model is given by the following gain control.

$$G(L) = \frac{1}{0.555(L+1)^{0.85}}$$

Multiplying either a low-pass or bandpass image by this gain control amounts to applying a sigmoid. Using the techniques presented in Section 7.1.4, a stack of seven increasingly blurred images is created next. The amount of blur is doubled at each level, and for the smallest scale we use a filter kernel the size of which is determined by a user parameter (discussed later in this section). An image at level s is represented by the following triplet.

$$\left(L_s^{\text{blur}}(x, y), M_s^{\text{blur}}(x, y), S_s^{\text{blur}}(x, y)\right)$$

From this stack of seven Gaussian-blurred images we may compute a stack of six DoG images that represent adapted contrast at six spatial scales, as follows.

$$L_s^{\text{DoG}}(x, y) = \left(L_s^{\text{blur}}(x, y) - L_{s+1}^{\text{blur}}(x, y)\right) G\left(L_{s+1}^{\text{blur}}(x, y)\right)$$

$$M_s^{\text{DoG}}(x, y) = \left(M_s^{\text{blur}}(x, y) - M_{s+1}^{\text{blur}}(x, y)\right) G\left(M_{s+1}^{\text{blur}}(x, y)\right)$$

$$S_s^{\text{DoG}}(x, y) = \left(S_s^{\text{blur}}(x, y) - S_{s+1}^{\text{blur}}(x, y)\right) G\left(S_{s+1}^{\text{blur}}(x, y)\right)$$

The DoG scheme involves a division by a low-pass filtered image (through the gain control function), which may be viewed as a normalization step. This approach was followed in both Ashikhmin's operator (see following section) and in the photographic tone-reproduction operator (Section 7.3.6). DoGs are reasonable approximations of some of the receptive fields found in the human visual system.[4] They are also known as center-surround mechanisms.

The low-pass image at level $s = 7$ is retained and will form the basis for image reconstruction. In the final step of the forward model, pixels in this low-pass image are adapted to a linear combination of themselves and the mean value $(\bar{L}_7^{\text{blur}}, \bar{M}_7^{\text{blur}}, \bar{S}_7^{\text{blur}})$ of the low-pass image, as follows.

$$L_7^{\text{blur}}(x, y) = L_7^{\text{blur}}(x, y) G\left((1 - A)\bar{L}_7^{\text{blur}} + A L_7^{\text{blur}}(x, y)\right)$$

$$M_7^{\text{blur}}(x, y) = M_7^{\text{blur}}(x, y) G\left((1 - A)\bar{M}_7^{\text{blur}} + A M_7^{\text{blur}}(x, y)\right)$$

$$S_7^{\text{blur}}(x, y) = S_7^{\text{blur}}(x, y) G\left((1 - A)\bar{S}_7^{\text{blur}} + A S_7^{\text{blur}}(x, y)\right)$$

The amount of dynamic range reduction is determined by user parameter A in these equations, which takes a value between 0 and 1. The effect of this parameter on the appearance of tone-mapped images is shown in Figure 7.35.

4 A receptive field may be seen as the pattern of light that needs to be present to optimally stimulate a cell in the visual pathway.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 73

FIGURE 7.35 Using the multiscale observer model, the interpolation parameter A was set to 0, 0.25, 0.50, 0.75, and 1.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 74

The forward version of the multiscale observer model is based on the human visual system. Although we could display the result of the forward model, the viewer's visual system would then also apply a similar forward model (to the extent that this model is a correct reflection of the human visual system). To avoid applying the model twice, the computational model should be reversed before an image is displayed. During the reversal process, parameters pertaining to the display device are inserted in the model so that the result is ready for display.

In the first step of the inverse model, the mean luminance $L_{\mathrm{d,mean}}$ of the target display device needs to be determined. For a typical display device, this value may be set to about 50 cd/m^2. A gain control factor for the mean display luminance is determined, and the low-pass image is adapted once more, but now for the mean display luminance, as follows.

$$L_7^{\mathrm{blur}}(x,y) = \frac{L_7^{\mathrm{blur}}(x,y)}{G(L_{\mathrm{d,mean}})}$$

$$M_7^{\mathrm{blur}}(x,y) = \frac{M_7^{\mathrm{blur}}(x,y)}{G(M_{\mathrm{d,mean}})}$$

$$S_7^{\mathrm{blur}}(x,y) = \frac{S_7^{\mathrm{blur}}(x,y)}{G(S_{\mathrm{d,mean}})}$$

The stack of DoGs is then added to the adapted low-pass image one scale at a time, starting with $s = 6$ and followed by $s = 5, 4, \ldots, 0$, as follows.

$$L_7^{\mathrm{blur}}(x,y) = \max\left(L_7^{\mathrm{blur}}(x,y) + \frac{L_s^{\mathrm{DoG}}(x,y)}{G(L_7^{\mathrm{blur}}(x,y))}, 0\right)$$

$$M_7^{\mathrm{blur}}(x,y) = \max\left(M_7^{\mathrm{blur}}(x,y) + \frac{M_s^{\mathrm{DoG}}(x,y)}{G(M_7^{\mathrm{blur}}(x,y))}, 0\right)$$

$$S_7^{\mathrm{blur}}(x,y) = \max\left(S_7^{\mathrm{blur}}(x,y) + \frac{S_s^{\mathrm{DoG}}(x,y)}{G(S_7^{\mathrm{blur}}(x,y))}, 0\right)$$

Finally, the result is converted to XYZ and then to RGB, where gamma correction is applied. The original formulation of this model shows haloing artifacts similar to those of other local operators discussed in this chapter. One of the reasons for this is that the model is calibrated in degrees of visual angle rather than in pixels. The transformation between degrees of visual angle to pixels requires assumptions on the size of the display and its resolution, as well as the distance between the observer and the display. The size of the filter kernel used to create the low-pass images is directly affected by these assumptions. For the purpose of demonstration, Figure 7.36 shows a sequence of images produced with different kernel sizes. Note that we only adjust the size of the smallest Gaussian. By specifying the kernel size for the smallest Gaussian, the size of all other Gaussians is determined. The figure shows that smaller Gaussians produce smaller halos, which are less obtrusive than the larger halos of the original model.

The reconstruction of a displayable image proceeds by successively adding bandpass images back to the low-pass image. These bandpass images by default receive equal weight. It may be beneficial to weight bandpass images such that higher spatial frequencies contribute more to the final result. Although the original multiscale observer model does not feature such a weighting scheme, we have found that contrast in the final result may be improved if higher frequencies are given a larger weight. This is shown in Figure 7.37, where each successive image places more emphasis on higher frequencies. The scale factor k used for these images relates to the index number s of the bandpass pyramid in the following manner.

$$k = (6 - s)g$$

The constant g is a user parameter, which we vary between 1 and 5 in Figure 7.37. A larger value for g produces more contrast in the tone-mapped image, but if this value is chosen too large the residual halos present in the image are emphasized (which is generally undesirable). For uncalibrated images tone mapped with the multiscale observer model, different prescale factors cause the overall image appearance to be lighter or darker, as shown in Figure 7.38.

The computational complexity of this operator remains high, and we would only recommend this model for images with an extreme dynamic range. If the amount of compression required for a particular image is less, simpler models likely suffice. The Fourier transforms used to compute the low-pass images are the main factor determining running time. There are seven levels in the Gaussian pyramid, and four color channels in the original model, resulting in 28 low-pass filtered images.

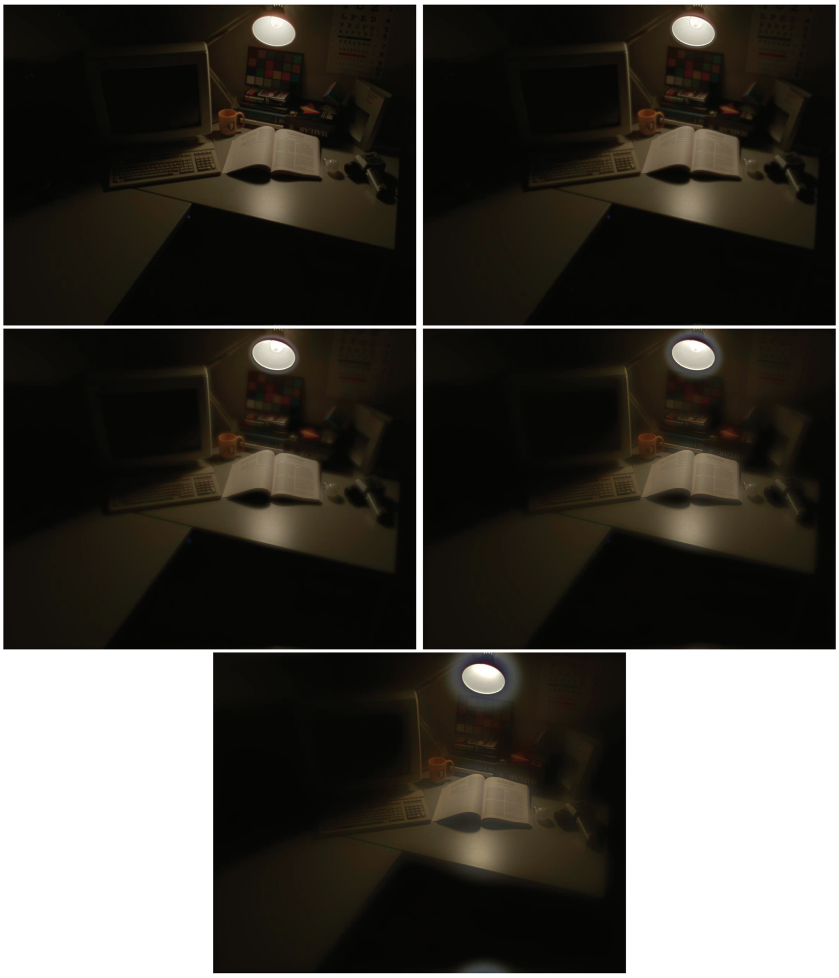


FIGURE 7.36 Using the multiscale observer model, the filter kernel size is set to 0.03, 0.06, 0.12, 0.25, and 0.5 in this sequence of images.

FIGURE 7.37 Relative scaling in the multiscale observer model. For a filter kernel size of 0.03, the relative scaling parameter was set to 1, 2, 3, 4, and 5.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 78

FIGURE 7.38 Effect of pre-scaling on the multiscale observer model. Images are pre-scaled by factors of 0.001, 0.01, 0.1, 1, and 10.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 79

In our simplified model, we only compute three color channels, resulting in a total of 21 low-pass images.

The multiscale observer model is the first to introduce center-surround processing to the field of tone reproduction, which is also successfully employed in Ashikhmin's operator (see following section) and in Reinhard et al.'s photographic tone-reproduction operator (see Section 7.3.6). The halos present in the original model may be minimized by carefully choosing an appropriate filter kernel size.

7.3.5 ASHIKHMIN SPATIALLY VARIANT OPERATOR

The multiscale observer model aims at completeness in the sense that all steps of human visual processing that are currently understood well enough to be modeled are present in this model. It may therefore account for a wide variety of appearance effects. One may argue that such completeness is not strictly necessary for the more limited task of dynamic-range reduction.

Ashikhmin's operator attempts to model only those aspects of human visual perception that are relevant to dynamic-range compression [6]. This results in a significantly simpler computational model consisting of three steps. First, for each point in the image a local adaptation value $L_{\mathrm{wa}}(x, y)$ is established. Next, a compressive function is applied to reduce the image's dynamic range. As this step may cause some detail to be lost, a final pass reintroduces detail. Ashikhmin's operator is aimed at preserving local contrast, which is defined as

$$c_{\mathrm{w}}(x, y) = \frac{L_{\mathrm{w}}(x, y)}{L_{\mathrm{wa}}(x, y)} - 1.$$

In this definition, L_{wa} is the world adaptation level for pixel (x, y). The consequence of local contrast preservation is that visible display contrast $c_{\mathrm{d}}(x, y)$, which is a function of display luminance $L_{\mathrm{d}}(x, y)$ and its derived local display adaptation level $L_{\mathrm{da}}(x, y)$, equals $c_{\mathrm{w}}(x, y)$. This equality may be used to derive a function for computing display luminances, as follows.

$$\begin{aligned} c_{\mathrm{d}}(x, y) &= c_{\mathrm{w}}(x, y) \\ \frac{L_{\mathrm{d}}(x, y)}{L_{\mathrm{da}}(x, y)} - 1 &= \frac{L_{\mathrm{w}}(x, y)}{L_{\mathrm{wa}}(x, y)} - 1 \end{aligned}$$

$$L_{\mathrm{d}}(x,y) = L_{\mathrm{da}}(x,y)\frac{L_{\mathrm{w}}(x,y)}{L_{\mathrm{wa}}(x,y)}$$

The unknown in these equations is the local display adaptation value $L_{\mathrm{da}}(x,y)$. Ashikhmin proposes to compute this value for each pixel from the world adaptation values. Thus, the display adaptation luminances are tone-mapped versions of the world adaptation luminances, as follows.

$$L_{\mathrm{da}}(x,y) = F\big(L_{\mathrm{wa}}(x,y)\big)$$

The complete tone-reproduction operator is then given by

$$L_{\mathrm{d}}(x,y) = F(L_{\mathrm{wa}}(x,y))\frac{L_{\mathrm{w}}(x,y)}{L_{\mathrm{wa}}(x,y)}.$$

There are now two subproblems to be solved. The functional form of the tone-mapping function $F(\,)$ needs to be given, and an appropriate local world adaptation level $L_{\mathrm{wa}}(x,y)$ needs to be computed.

To derive the compressive function $F()$, Ashikhmin introduces the notion of perceptual capacity of a range of luminance values. Human sensitivity to luminance changes is given by TVI functions (see also Chapter 6). This may be used as a scaling factor for a small range of luminance values ΔL. The intuition behind this approach is that the perceptual importance of a JND is independent of the absolute luminance value for which it is computed. For a range of world luminances between 0 and L, perceptual capacity $C(L)$ may therefore be defined as follows.

$$C(L) = \int_0^L \frac{dx}{T(x)}$$

Here, $T(x)$ is the threshold versus intensity function. The perceptual capacity for an arbitrary luminance range from L_1 to L_2 is then $C(L_2) - C(L_1)$. Following others, the TVI function is approximated by four linear segments (in log-log space), and

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 81

thus the perceptual capacity function becomes

$$C(L) = \begin{cases} L/0.0014 & \text{for } L < 0.0034 \\ 2.4483 + \log_{10}(L/0.0034)/0.4027 & \text{for } 0.0034 \le L < 1 \\ 16.5630 + (L-1)/0.4027 & \text{for } 1 \le L < 7.2444 \\ 32.0693 + \log_{10}(L/7.2444)/0.0556 & \text{otherwise.} \end{cases}$$

World adaptation luminances may now be mapped to display adaptation luminances such that perceptual world capacity is linearly mapped to a displayable range. Assuming the maximum displayable luminance is given by $L_{\mathrm{d,max}}$, the compressive function $F(L_{\mathrm{wa}}(x, y))$ is given by

$$F\big(L_{\mathrm{wa}}(x, y)\big) = L_{\mathrm{d,max}} \frac{C(L_{\mathrm{wa}}(x, y)) - C(L_{\mathrm{w,min}})}{C(L_{\mathrm{w,max}}) - C(L_{\mathrm{w,min}})}.$$

In this equation, $L_{\mathrm{w,min}}$ and $L_{\mathrm{w,max}}$ are the minimum and maximum world adaptation luminances. Finally, the spatially variant world adaptation luminances are computed in a manner akin to Reinhard's dodge-and-burn operator, discussed in the following section. The world adaptation luminance of a pixel is a Gaussian weighted average of pixel values taken over some neighborhood. The success of this method lies in the fact that the neighborhood should be chosen such that the spatial extent of the Gaussian filter does not cross any major luminance steps. As such, for each pixel its neighborhood should be chosen as large as possible without crossing sharp luminance gradients.

To compute if a pixel neighborhood contains any large gradients, consider a pixel of a Gaussian-filtered image with a filter kernel R of size s, as well as the same pixel position of a Gaussian-filtered image with a kernel of size $2s$. Because Gaussian filtering amounts to computing a weighted local average, the two blurred pixels represent local averages of two differently sized neighborhoods. If these two averages are similar, no sharp gradients occurred in the pixel's neighborhood. In other words, if the difference of these two Gaussian-filtered pixels is close to 0 the pixel's neighborhood of size $2s$ is LDR. The difference of Gaussians is normalized by one of the Gaussian-filtered images, yielding a measure of band-limited local contrast V_s, as follows.

$$V_s = \frac{L_{\mathrm{w}} \otimes R_s - L_{\mathrm{w}} \otimes R_{2s}}{L_{\mathrm{w}} \otimes R_s}$$

These arguments are valid for any scale s. We may therefore compute a stack of band-limited local contrasts for different scales s. The smallest scale is $s = 1$, and each successive scale in Ashikhmin's operator is 1 pixel larger than the previous. The largest scale is 10 pixels wide.

Each successive larger scale difference of Gaussians tests a larger pixel neighborhood. For each pixel, the smallest scale s_t for which $V_{s_t}(x, y)$ exceeds a user-specified threshold t is chosen. By default, the value of this threshold may be chosen to be $t = 0.5$. The choice of threshold has an impact on the visual quality of the operator. If a value of 0.0 is chosen, Ashikhmin's operator defaults to a global operator. If the threshold value is chosen too large, halo artifacts will result. The size of these halos is limited to 10 pixels around any bright features because this is the size of the largest center. To demonstrate the effect of this threshold, we have reduced an image in size prior to tone mapping and enlarged the tone-mapped result, which is shown in Figure 7.39.

The size of a locally uniform neighborhood is now given by s_t. The local world adaptation value $L_{\mathrm{wa}}(x, y)$ is a Gaussian-blurred pixel at scale s_t, as follows.

$$L_{\mathrm{wa}}(x, y) = (L_{\mathrm{w}} \otimes R_{s_t})(x, y)$$

Note that the scale s_t will be different for each pixel so that the size of the local neighborhood over which L_{wa} is computed varies according to image content. The

FIGURE 7.39 *Effect of thresholding on results obtained with Ashikhmin's operator. From left to right: threshold values are 0.0, 0.5, and 1.0.*

idea of using the largest possible filter kernel without crossing large contrast steps is in some sense equivalent to the output of edge-preserving smoothing operators such as the bilateral and trilateral filters discussed in Sections 8.1.2 and 8.1.3.

Other than the threshold value discussed previously, this operator does not have any user parameters, which is good if plausible results need to be obtained automatically. However, as with several other operators the input needs to be specified in appropriate SI units. If the image is in arbitrary units, it needs to be pre-scaled. The effect of pre-scaling an image is shown in Figure 7.40. We note that there appears to be a discontinuity in visual appearance between the image that was pre-scaled by a factor of 10 (top right) and 100 (bottom left in Figure 7.40). We suspect that this is due to the C_1 discontinuity in the TVI function used in the perceptual capacity function. The C_1 discontinuity in the TVI function is due to the different luminance levels at which rods and cones in the human visual system operate.

In summary, Ashikhmin's operator is based on sufficient knowledge of the human visual system to be effective without aiming for completeness. The operator is not developed to be predictive but to provide a reasonable hands-off approach to producing visually pleasing output in which local contrast is preserved.

7.3.6 REINHARD ET AL. PHOTOGRAPHIC TONE REPRODUCTION

The problem of mapping a range of world luminances to a smaller range of display luminances is not a new problem. Tone reproduction has existed in conventional photography since photography was invented. The goal of photographers is often to produce renderings of captured scenes that appear realistic. With photographic paper (like all paper) being inherently LDR, photographers have to find ways to work around the limitations of the medium.

Although many common photographic principles were developed in the last 150 years, and a host of media response characteristics were measured, a disconnect existed between the artistic and technical sides of photography. Ansel Adams' zone system, which is still in use today, attempts to bridge this gap. It allows the photographer to use field measurements to improve the chances of creating a good final print.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 84

FIGURE 7.40 *Pre-scaling applied to Ashikhmin's operator with factors ranging from 0.1 to 10,000. Each successive image was scaled by a factor 10 times the factor of the preceding image.*

The zone system may be used to make informed choices in the design of a tone-reproduction operator [109]. First, a linear scaling is applied to the image, which is analogous to setting exposure in a camera. Then, contrast may be locally adjusted using a computational model akin to photographic dodging and burning, which is a technique to selectively expose regions of a print for longer or shorter periods of time. This may bring up selected dark regions, or bring down selected light regions.

The key of a scene in photography is an indicator of how light or dark the overall impression of a scene is. Following other tone-reproduction operators, Reinhard et al. view the log average luminance $\bar{L}_{\mathrm{w}}$ (Equation 7.1) as a useful approximation of a scene's key. For average-key scenes, the log average luminance should be mapped to 18% of the display range, which is in line with common photographic practice (although see footnote 4 of Chapter 2). Higher-key scenes should be mapped to a higher value, and lower-key scenes should be mapped to a lower value. The value to which the log average is mapped is given as a user parameter a. The initial scaling of the photographic tone-reproduction operator is then given by the following.

$$L_{\mathrm{m}}(x, y) = \frac{a}{\bar{L}_{\mathrm{w}}} L_{\mathrm{w}}(x, y)$$

The subscript m denotes values obtained after the initial linear mapping. In that this scaling precedes any nonlinear compression, the operator does not necessarily expect the input to be specified in SI units. If the image is given in arbitrary units, the user parameter a could be adjusted accordingly. An example of this parameter's effect is shown in Figure 7.41. For applications that require hands-off operation, the value of this user parameter may be estimated from the histogram of the image [106]. This technique is detailed in Section 7.1.1.

Many scenes have a predominantly average dynamic range with a few high-luminance regions near highlights or in the sky. Traditional photography uses S-shaped transfer functions (sigmoids) to compress both high- and low-luminance values while emphasizing the midrange. However, modern photography uses transfer functions that predominantly compress high luminances. This may be modeled with the following compressive function.

$$L_{\mathrm{d}}(x, y) = \frac{L_{\mathrm{m}}(x, y)}{1 + L_{\mathrm{m}}(x, y)}$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 86

FIGURE 7.41 *Pre-scaling the image data is an integral part of the photographic tone-reproduction operator and may be automated. Here, user parameter a was set to 0.01, 0.04, 0.18 (default), and 0.72.*

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 87

This function scales small values linearly, whereas higher luminances are compressed by larger amounts. The function has an asymptote at 1, which means that all positive values will be mapped to a display range between 0 and 1. However, in practice the input image does not contain infinitely large luminance values, and therefore the largest display luminances do not quite reach 1. In addition, it may be artistically desirable to let bright areas burn out in a controlled fashion. This effect may be achieved by blending the previous transfer function with a linear mapping, yielding the following tone-reproduction operator:

$$L_{\mathrm{d}}(x, y) = \frac{L_{\mathrm{m}}(x, y)\left(1 + \dfrac{L_{\mathrm{m}}(x, y)}{L_{\mathrm{white}}^2}\right)}{1 + L_{\mathrm{m}}(x, y)}$$

This equation introduces a new user parameter, L_{white}, which denotes the smallest luminance value that will be mapped to white. By default, this parameter is set to the maximum world luminance (after the initial scaling). For lower-dynamic-range images, setting L_{max} to a smaller value yields a subtle contrast enhancement. Figure 7.42 shows various choices of L_{white} for an LDR image. Note that for hands-off operation this parameter may also be estimated from the histogram of the input image [106].

The previous equation is a reasonable global tone-reproduction operator. However, it may be modified to become a local tone-reproduction operator by applying an algorithm akin to photographic dodging and burning. In traditional dodging and burning, the area that receives a different exposure from the remainder of the print is bounded by sharp contrasts. This is a key observation that should be reproduced by any automatic dodge-and-burn algorithm.

For each pixel, we would therefore like to find the largest surrounding area that does not contain any sharp contrasts. A reasonable measure of contrast for this purpose is afforded by traditional center-surround computations. A Gaussian-weighted average is computed for a pixel (the center), and is compared with a Gaussian-weighted average over a larger region (the surround), both centered over the same pixel. If there are no significant contrasts in the pixel's neighborhood, the difference of these two Gaussians will be close to 0. However, if there is a contrast edge that overlaps the surround but not the center Gaussian the two averages will be significantly different.

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 88

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 89

FIGURE 7.42 *The L_{white} parameter in the Reinhard photographic tone-reproduction operator is effective in minimizing the loss of contrast when tone mapping a low dynamic range image. The value of L_{white} was set to 0.15 in the top left-hand image, and incremented by 0.10 for each subsequent image.*

If a Gaussian-blurred image at scale s is given by

$$L_s^{\text{blur}}(x, y) = L_{\text{m}}(x, y) \otimes R_s(x, y),$$

the center-surround mechanism at that scale is computed with

$$V_s(x, y) = \frac{L_s^{\text{blur}} - L_{s+1}^{\text{blur}}}{2^{\Phi} a/s^2 + L_s^{\text{blur}}}.$$

The normalization by $2^{\Phi} a/s^2 + L_s^{\text{blur}}$ allows this result to be thresholded by a common threshold that is shared by all scales, in that V_s is now independent of absolute luminance values. In addition, the $2^{\Phi} a/s^2$ term prevents the normalization from breaking for small values of L_s^{blur}. The user parameter Φ may be viewed as a sharpening parameter, the effect of which is shown in Figure 7.43. For small values of Φ, its effect is very subtle. If the value is chosen too large, haloing artifacts may occur. In practice, a setting of $\Phi = 8$ yields plausible results.

This process yields a set of differences of Gaussians, each providing information about how much contrast is available within increasingly large areas around the pixel of interest. To find the largest area that has relatively low contrast for a given pixel, we seek the largest scale $s_{\max}$ for which the difference of Gaussians remains below a threshold, as follows.

$$s_{\max} : \left| V_{s_{\max}}(x, y) \right| < \epsilon$$

For this scale, the corresponding center Gaussian may be taken as a local average. The local operator that implements a computational model of dodging and burning

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 90

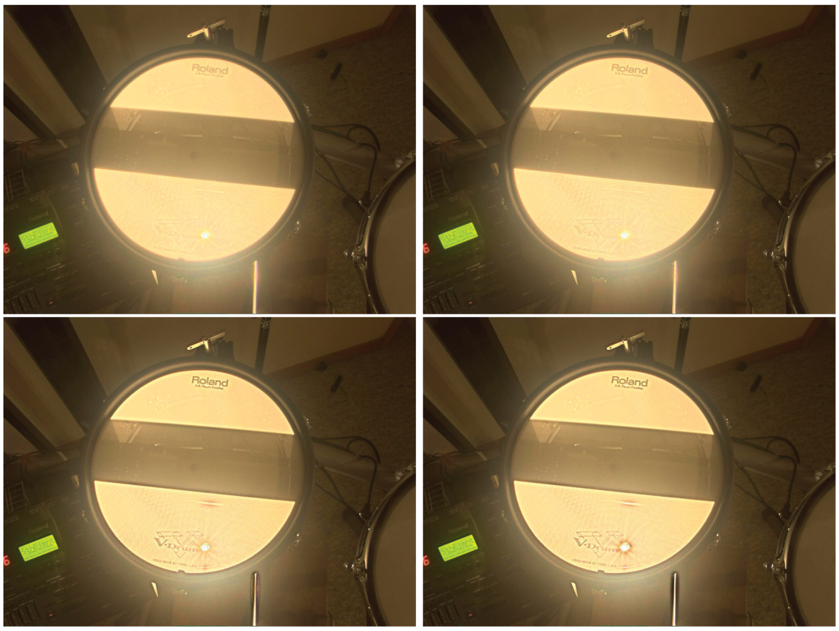


FIGURE 7.43 *The sharpening parameter Φ in the photographic tone-mapping operator is chosen to be 4 and 8 (top row) and 16 and 32 (bottom row).*

is then given by the following.

$$L_{\mathrm{d}}(x, y) = \frac{L_{\mathrm{m}}(x, y)}{1 + L^{\mathrm{blur}}_{s_{\max}}(x, y)}$$

The luminance of a dark pixel in a relatively bright region will satisfy $L < L^{\mathrm{blur}}_{s_{\max}}$, and thus this operator will decrease the display luminance L_{d}, thereby increasing the contrast at that pixel. This is akin to photographic "dodging." Similarly, a pixel in a relatively dark region will be compressed less, and thus "burned." In either case the pixel's contrast relative to the surrounding area is increased.

The memory efficiency of the dodge-and-burn version may be increased by realizing that the scale selection mechanism could be executed on the fly. The original implementation computes a Gaussian pyramid as a preprocess. Then, during tone mapping for each pixel the most appropriate scale is chosen. Goodnight et al. show that the preprocessing step may be merged with the actual tone-reproduction stage and thus avoid computing the low-pass images that will not be used [43]. Their work also shows how this operator may be implemented in graphics hardware.

In summary, the photographic tone-reproduction technique [109] exists in both global and local variants. For medium-dynamic-range images, the global operator is fast and provides sufficient compression. For very high-dynamic-range images, local contrast may be preserved better with the local version that implements dodging and burning. The local operator seeks for each pixel the largest area that does not contain significant contrast steps. This technique is therefore similar to edge-preserving smoothing filters such as the bilateral filter discussed in Section 8.1.2. We could therefore replace the scale selection mechanism with the more practical and efficient bilateral filter to produce a spatially localized average. This average would then serve the purpose of finding the average exposure level to which the pixel will be adjusted.

7.3.7 PATTANAIK ADAPTIVE GAIN CONTROL

Thus far, we have discussed several tone-reproduction operators that compute a local average. The photographic tone-reproduction operator uses a scale-space mechanism to select how large a local area should be and computes a weighted average for this local area. It is then used to adjust exposure level. Ashikhmin's operator does the same, but provides an alternative explanation in terms of human vision. Similarly, the bilateral filter is effectively an edge-preserving smoothing operator. Smoothing by itself can be viewed as computing an average over a local neighborhood. The edge-preserving properties of the bilateral filter are important, because it allows the space over which the average is computed to be maximized.

The defining characteristic of the bilateral filter is that pixels are averaged over local neighborhoods, provided their intensities are similar. The bilateral filter

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 92

is defined as

$$L^{\text{smooth}}(x, y) = \frac{1}{w(x, y)} \sum_u \sum_v b(x, y, u, v) L(x - u, y - v) \tag{7.7}$$

$$w(x, y) = \sum_u \sum_v b(x, y, u, v)$$

$$b(x, y, u, v) = f\left(\sqrt{(x - u)^2 + (y - v)^2}\right) g\big(L(x - u, y - v) - L(x, y)\big),$$

with $w()$ a weight factor normalizing the result and $b()$ the bilateral filter consisting of components $f()$ and $g()$. There is freedom to choose the shape of the spatial filter kernel $f()$, as well as the luminance-domain filter kernel $g()$. Different solutions were independently developed in the form of SUSAN [118] and the bilateral filter [128]. At the same time, independent and concurrent developments led to alternative tone-reproduction operators: one based on the bilateral filter [23] and one based on the SUSAN filter [96].

Whereas Durand and Dorsey experimented with Gaussian filters and Tukey's filter, Pattanaik and Yee employed a near box-shaped filter kernel in the luminance domain to steer the amount of compression in their tone-reproduction operator [96]. The latter used the output of their version of the bilateral filter as a local adapting luminance value, rather than as a mechanism to separate the image into a base layer and a detail layer as Durand and Dorsey did.

Taking their cue from photography, Pattanaik and Yee note that white tends to be five times as intense as medium gray and black is one-fifth the luminance of medium gray. Their local gain control is derived from a weighted local average in which each surrounding pixel is weighted according to its luminance in relation to the luminance of the pixel of interest. Pixels more than five times as intense as the center pixel, and pixels less than one-fifth its luminance, are excluded from consideration. For a circularly symmetric area around pixel (x, y), the local average is then computed for all pixels as follows.

$$\frac{1}{5} \le \frac{L_{\text{w}}(x - u, y - v)}{L_{\text{w}}(x, y)} \le 5$$

The circularly symmetric local area is determined by bounding the value of u and v by the radius r of the area under consideration, as follows.

$$\sqrt{u^2+v^2} \le r$$

An alternative notation for the same luminance-domain constraint may be formulated in the log domain, with the base of the log being 5, as follows.

$$\left|\log_5\bigl(L_{\mathrm{w}}(x-u, y-v)\bigr) - \log_5\bigl(L_{\mathrm{w}}(x, y)\bigr)\right| \le 1$$

This implies a box filter in the luminance domain and a "box filter" (albeit circularly symmetric) in the spatial domain. A box filter in the luminance domain suffices if the image consists solely of sharp edges. Smoother high-contrast edges are best filtered with a luminance-domain filter that has a somewhat less abrupt cutoff. This may be achieved with the following luminance-domain filter kernel $g()$.

$$g(x-u, y-v) = \exp\bigl(-|\log_5\bigl(L_{\mathrm{w}}(x-u, y-v)\bigr) - \log_5\bigl(L_{\mathrm{w}}(x, y)\bigr)|^{25}\bigr)$$

The spatial filter kernel $f()$ is circularly symmetric and unweighted, as follows.

$$f(x-u, y-v) = \begin{cases} 1 & \text{if } \sqrt{u^2+v^2} \le r \\ 0 & \text{otherwise} \end{cases}$$

The result of producing a filtered image with this filter is an image that is blurred, except in areas where large-contrast steps occur. This filter may therefore be viewed as an edge-preserving smoothing filter, as are the bilateral and trilateral filters. The output of this filter may therefore be used in a manner similar to tone-reproduction operators that split an image into a base layer and a detail layer. The base layer is then compressed and recombined with the detail layer under the assumption that the base layer is HDR and the detail layer is LDR.

Alternatively, the output of this filter may be viewed as a local adapting luminance. Any of the global operators that make use of a global average may thus be extended to become local operators. For instance, the output of any edge-preserving smoothing operator, as well as the scale selection mechanism of Reinhard et al.'s photographic operator, may serve as a local adaptation luminance. In each case, the typical trade-off between amount of achievable compression and visibility of

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 94

haloing artifacts will return. However, by using edge-preserving smoothing operators or the aforementioned scale selection mechanism, the local average is ensured to be relatively close to the pixel value itself. Although halos may not be avoided altogether, they are minimized with these approaches.

7.3.8 YEE SEGMENTATION-BASED APPROACH

Many HDR images contain large areas that are relatively dark and large areas that are bright. An often-quoted example of such a configuration is a room with a window. In such cases, it may be desirable to apply different compression functions for the bright and dark regions.

Any algorithm that uses a local adaptation level — such as the semisaturation constant in the Michaelis–Menten equations (6.1) — may be modified to explicitly use an adaptation level based on segmentation of the bright and dark areas into separate regions.

At least two operators are currently known that segment an image into separate regions for the purpose of tone reproduction [67,150]. In this section, we discuss Yee and Pattanaik's approach. They effectively segment the image into separate regions, and then determine a suitable adaptation level for each region [150]. Their approach consists of the four following steps.

1. *Segmentation:* Based on the histogram of a density representation, the image is segmented into regions. A histogram is created with a specific number of bins (as discussed in material following).
2. *Grouping:* Pixels in the segmented image are grouped, and each pixel within a group is assigned the average density of the group.
3. *Assimilation:* Small groups and groups with only one neighbor are merged. The result of the assimilation process is called a *layer*.
4. *Layer averaging:* The previous three steps are repeated several times for histograms with different bin sizes (and numbers of bins), and for each pixel the results are averaged.

After layer averaging is complete, the resulting image provides a local adaptation level (in the log domain) for each pixel. Several user parameters are introduced to steer the quality of the results. The layer-averaging step has the effect of smoothing

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 95

FIGURE 7.44 *This image is segmented into 10 bins, and then encoded with a separate gray level per bin.*

the adaptation level image. This image should not contain sharp discontinuities, in that such discontinuities would lead to artifacts in the final tone-mapped result. By choosing more layers, each created with a histogram with a different spacing of bins, a smoother result is obtained. Hence, the total number of layers is an important parameter, trading computation time for visual quality. The number of layers required to minimize artifacts depends on the composition of the image and on its dynamic range. The bin size B_n is determined by the total number of layers N, the current layer number n, and two further user parameters that limit the minimum and maximum bin size ($B_{\min}$ and $B_{\max}$), as follows.

$$B_n = B_{\min} + (B_{\max} - B_{\min}) \frac{n}{N-1}$$

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 96

FIGURE 7.45 *This image is segmented into 10 bins, grouped, and then encoded with a separate gray level per group.*

The minimum and maximum bin size are by default set to 0.5 and 1.0, respectively. Given the bin size B_n for the current layer, each pixel may be categorized as belonging to bin b, as follows.

$$b(x, y) = \frac{D(x, y) - D_{\min}}{B_n}$$

An example is shown in Figure 7.44, where each gray level indicates a separate bin. Once each pixel is labeled with its bin number b, pixels may be grouped. An image after grouping is shown in Figure 7.45. During the grouping process, the average density of the group is determined and stored. The grouping makes use of a recursive flood-fill algorithm. A potential problem with this approach is that if

FIGURE 7.46 *Layer averaging demonstrated with a total of 10 layers.*

large areas are filled recursively the number of recursive calls may cause the system to run out of stack space. The flood-fill algorithm also keeps track of how many pixels are added to a group. After each group is filled, the average density of the group is computed.

The assimilation process merges small groups with larger ones. For details, we refer to the original paper [150]. For the images shown in this section, we have omitted this step. It is possible that for certain image compositions the assimilation step produces an improved estimate of local adaptation levels, but we have found that for the test images used here (in combination with the sigmoidal compression function we used) the results without the assimilation step are very good.

In our implementation, the average density of a group is used in the layer-averaging process. Because for each pixel the group number is known, the average

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 98

FIGURE 7.47 *Final result of tone mapping with Yee and Pattanaik's method for deriving local adaptation levels. This image was created with layer averaging over 10 layers to allow comparison with Figure 7.46.*

density assigned to each pixel is found by computing the following.

$$D_{n,\mathrm{av}}(x, y) = \texttt{group\_list[group[y][x]].lum\_av}$$

The previous steps are repeated for all layer numbers $0, \ldots, N$, and the results are averaged, as follows.

$$L_\mathrm{a}(x, y) = \exp\left(\frac{1}{N}\sum_{n=0}^{N-1} D_{n,\mathrm{av}}(x, y)\right)$$

FIGURE 7.48 *Image tone mapped with a global tone-reproduction operator (left), and with a replacement of the global adaptation level for local adaptation levels based on segmentation (right).*

The resulting adaptation levels are shown in Figure 7.46. The final result for a total of 10 layers is shown in Figure 7.47. For this image, 10 layers are not quite sufficient for an artifact-free result. Rather, this choice allows a direct comparison between Figures 7.46 and 7.47.

To demonstrate the effect of this approach in deriving the local adaptation luminance for each pixel, we adapted Reinhard and Devlin's photoreceptor-based algorithm to accept the previously cited local averages. Figure 7.48 shows the result of this operator with a global adaptation level (left) and locally computed adaptation levels obtained with the previous segmentation procedure (right).

The effect of varying the number of layers on the quality of the results is shown in Figure 7.49, where the number of layers was varied between 5 and 30. It is clear that the smoothing effect of averaging multiple layers is important in avoiding visual artifacts. The number of layers required varies with the dynamic range of the image, as well as the composition of the image. For this particular example, 30 layers are sufficient.

In summary, Yee and Pattanaik propose to segment the image into regions and compute an adaptation level for each region. By smoothing the results—accomplished by repeating the segmentation for different histogram bin sizes—local

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 100

reinhard v.2005/03/22 Prn:14/06/2005; 8:09 F:reinhard07.tex; VTEX/JOL p. 101

FIGURE 7.49 *The number of layers computed with Yee and Pattanaik's segmentation approach is varied between 5 and 30 (with increments of five layers) in this image sequence. If this number is too small, artifacts occur. For this example, artifacts are removed completely when 30 layers are used.*

adaptation levels are computed suitable for steering local tone-reproduction operators. The usefulness of this approach is demonstrated in this section by augmenting the photoreceptor-based operator with these local adaptation levels.

7.4 SUMMARY

Tone-reproduction operators may reduce the dynamic range of images by applying the same function to all pixels, or they may compress pixels based on their value and the values of a local neighborhood of pixels. The former category is computationally efficient and generally suitable for medium-dynamic-range images.

Extra compression may be achieved by making the compressive function dependent on neighboring pixels. This may be achieved by dividing the input image by a blurred version of the image, in which case the amount of blur to apply should be large to avoid haloing artifacts. A blurred version of the image may also be seen as an adaptation level. Then it can be used as the semisaturation constant in a sigmoidal (or S-shaped) function. In this case, artifacts are minimized by choosing a small filter kernel — typically only a few pixels wide. Small filter kernels have the added advantage of low computational cost.

In either case, artifacts may be minimized by using filter kernels that do not cross sharp image contrasts. Edge-preserving smoothing operators such as the bilateral filter, as well as the scale selection mechanism employed by the photographic operator, are examples of techniques that avoid blurring across stark contrasts and therefore show fewer artifacts than operators that blur each pixel by the same amount.