Compressing and Companding High Dynamic Range Images with Subband Architectures

Yuanzhen Li   Lavanya Sharan   Edward H. Adelson

Dept. of Brain and Cognitive Sciences, and Computer Science and Artificial Intelligence Lab

Massachusetts Institute of Technology, Cambridge, MA

Abstract

High dynamic range (HDR) imaging is an area of increasing importance, but most display devices still have limited dynamic range (LDR). Various techniques have been proposed for compressing the dynamic range while retaining important visual information. Multiscale image processing techniques, which are widely used for many image processing tasks, have a reputation of causing halo artifacts when used for range compression. However, we demonstrate that they can work when properly implemented. We use a symmetrical analysis-synthesis filter bank, and apply local gain control to the subbands. We also show that the technique can be adapted for the related problem of "companding", in which an HDR image is converted to an LDR image, and later expanded back to high dynamic range.

CR Categories: I.4.3 [Image Processing and Computer Vision]: Enhancement; I.3.3 [Computer Graphics]: Picture/Image Generation

Keywords: high dynamic range, tone mapping, range compression, wavelets, subbands, multiscale, multiresolution, companding

1 Introduction

In recent years there has been an explosion of interest in high dynamic range (HDR) imagery. HDR image data is increasingly available from sources such as digital photography, computer graphics, and medical imaging [Ward 1994; Debevec and Malik 1997; Mitsunaga and Nayar 2000; Mann and Picard 1995]. Although new HDR display systems are being developed [Seetzen et al. 2004], the dominant display technologies, such as printed paper, CRTs, and LCDs, have limited dynamic ranges. Therefore various techniques have been developed for compressing the dynamic range of the signal so the information can be displayed effectively. Ideally, these techniques should be easy to implement, and should work automatically, with minimal human intervention. They should also avoid introducing unpleasant artifacts.

It would also be desirable to retrieve an HDR image from an LDR image with minimal degradation. In accord with audio terminology, we refer to the compression/expansion process as "companding". We will describe a technique that can, for example, turn a 12 bit/channel image into an 8 bit/channel TIFF, and later convert it back to a good approximation of the original 12-bit image. Since a great deal of hardware and software is designed around 8 bit imagery, this could have many uses. It is possible to do further data compression with JPEG, and still retrieve a 12 bit image with only modest degradations.

2 Previous Work

The recent literature on HDR range compression has been extensively reviewed by others [Tumblin 1999; Dicarlo and Wandell 2001; Devlin et al. 2002] and we refer the reader to these sources. The most straightforward techniques, sometimes called "global" tone-mapping methods, use compressive point nonlinearities. The image, $I(x,y)$, is simply mapped to a modified image, $I'(x,y) = p(I(x,y))$, where $p()$ is a compressive function such as a power function, or a function that is adapted to the image histogram [Tumblin and Rushmeier 1993; Ward 1994; Ferwerda et al. 1996; Larson et al. 1997]. The dynamic range is reduced, but the contrast of details is compromised and the images can look washed out. To compress the range while maintaining or enhancing the visibility of details, it is necessary to use more complex techniques.An early technique was described by Stockham [1972], who observed that the image $L(x,y)$ is a product of two images: an illumination image $I(x,y)$, and a reflectance image, $R(x,y)$. The illumination can vary greatly from region to region, which causes the dynamic range problems. Stockham estimated the local illumination as a geometric mean over a patch, and divided it out. This is equivalent to subtracting a blurred version of the image in the log luminance domain. The method unfortunately introduces artifacts known as "banding" or "halos" when there is an abrupt change of luminance, i.e., at large step edges. The size of the halo depends on the size of the blur. Multiscale techniques [Jobson et al. 1997; Pattanaik et al. 1998; Tumblin and Turk 1999], including some designed to capture properties of the human visual system, have reduced the visibility of the halos but have not removed them, and the computer graphics community has therefore explored other approaches. One popular approach is to estimate the illumination level, and a corresponding gain map, with an edge-preserving blur. The notion is that the gain map should have sharp edges at the same points that the original image does, thereby preventing halos [Nakazawa et al. 1995; Dicarlo and Wandell 2001]. Durand and Dorsey [2002] achieved particularly good results by computing a gain map with the bilateral filter described by Tomasi and Manduchi [1998]. They also developed methods for fast computation. An alternate approach is to work in the gradient domain, as is done in Retinex algorithms [Land and McCann 1971]. Fattal et al. [2002] computed a gain map for the gradient of the image, reducing large gradients relative to small ones, and then solved Poisson's equation to retrieve an image with compressed range. Solving Poisson's equation after manipulating the gradient field can be problematic, but Fattal et al. developed approximations that gave visually satisfying results with reasonable computation times.

Although multiscale representations have lost favor in the computer graphics community, there is some patent literature that suggests their utility. Labaere and Vuylsteke [1998] adapted Mallat and Zhong's wavelet method [1992], which represents signal in terms of positions of and magnitudes of maxima of the outputs of edge-sensitive filters. By reducing the size of the high magnitude edges, the dynamic range can be controlled. Lee [2001] described a method that combines multiscale processing with traditional tone mapping. First, an image is run through a point non-linearity to reduce its dynamic range. The resulting image suffers from the usual reduced visibility of edges and other details. Lee then computes a

subband decomposition of the original image, and adds portions of the subbands back to the the tone-mapped image in order to augment the visibility of detail at various scales. Gain maps are used to control the amount of augmentation from the subbands. Vuylsteke and Schoeters [1998] describe the use of several subband decompositions, including Laplacian pyramids, wavelets, and Gabor transforms, along with sigmoidal nonlinearities to limit the amplitude of the subband signals. This approach is effective, but can introduce distortions including haloes. We have explored a set of methods with a similar structure, in an effort to achieve good range compression with minimal artifacts.

3   Subbands and Nonlinear Distortion

There are many ways of building subband systems for decomposing and reconstructing images. Each has its advantages and disadvantages. Here we discuss how this choice interacts with the problem of dynamic range compression.

For simplicity, we start by considering continuous signals. A simple multiscale decomposition is shown in Fig 1(a). A signal, $s(x)$, is split into a set of bandpass signals, $b_1(x), b_2(x), \ldots$ with filters $f_1, f_2, \ldots$ chosen so that the original signal can be reconstructed by directly summing these bandpass signals:

$$s(x) = \sum_n b_n(x) \tag{1}$$

A nonlinearity, labelled "NL", can be imposed on the bandpass signals before summation.

Suppose that the filters consist of difference-of-Gaussians, with scales increasing by factors of two. Fig 2(a) shows a step edge, along with four of the subbands (Fig 2(b)) decomposed using this filter bank. The full set of subband signals can be summed to retrieve the original input signal.

To limit the amplitude of strong edges, we can limit the amplitudes of the strong subband responses to these signals. If a particular subband signal is $b(x)$, then a soft limit can be imposed with a sigmoid, e.g., $b'(x) = b(x)[b(x) + \varepsilon]^{-2/3}$ (ε is a constant, if equal to 0 then $b'(x)$ is the cube root of $b(x)$). Fig 2(d) shows a picture of the nonlinearity, and Fig 2(e) shows the result of imposing it on one of the subbands. The peaks are flattened, and the low values are expanded. This prevents $b'(x)$ from being too large, but it also leads to a distortion in its shape. When the subbands are summed, they produce a distorted signal. The multiscale scheme described by Dicarlo and Wandell [2001], which uses ratios of blurred images, is quite similar to this one when computed in the log intensity domain.

To get better results we need to reduce the distortion of the subband signals. There are various ways to do this, either by modifying the way that signal strength is controlled (gain control), or by modifying the filter bank architecture. We will discuss both.

3.1   Smooth Gain Control

It is useful to think of the sigmoid as controlling the gain at each location. The gain is low for high values and high for low values. In the case considered above, the effective gain, $G_1(x)$ is shown in Fig 2(f). It dips twice, at the two extrema of the signal. The compressed subband signal, can be expressed as $b'(x) = b(x)G_1(x)$. The rapid variation of $G_1(x)$ is the cause of the distortion of the compressed signal $b'(x)$.

To prevent the rapid variation in gain, we can simply compute a new gain signal (gain map) and force it to be smooth. If the gain varies more slowly than the subband signal itself, then there will be reduced distortion. In Fig 2(i), we have constructed a smooth gain signal, $G_2(x)$, by taking the absolute value of the subband signal, blurring it (Fig2(g)), and imposing the sigmoid on this smoothed

(a) A subband system without synthesis filtering.

(b) An analysis-synthesis subband system.

(c) A two-band system with subsampling.

(d) A cascaded two-band system with nonliearity.

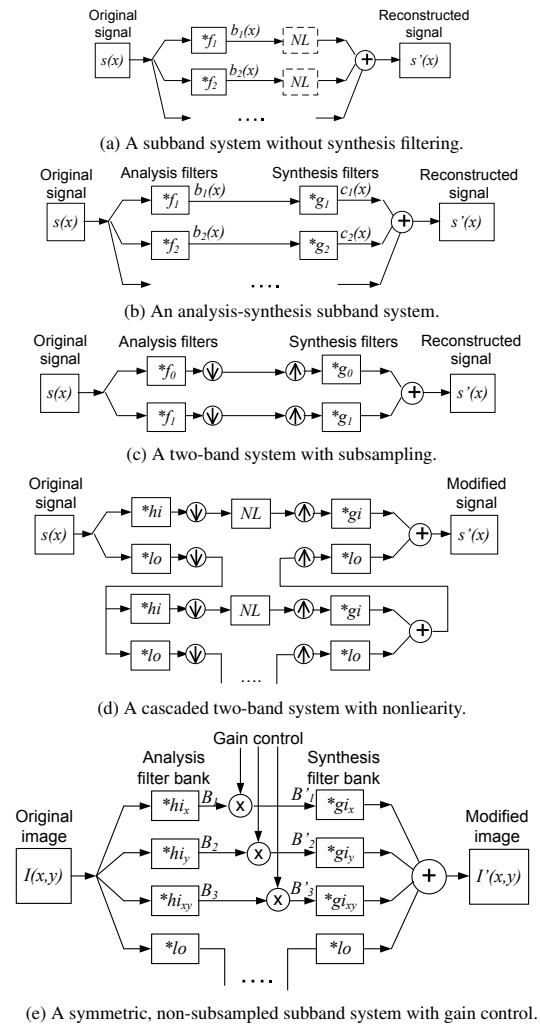(e) A symmetric, non-subsampled subband system with gain control.
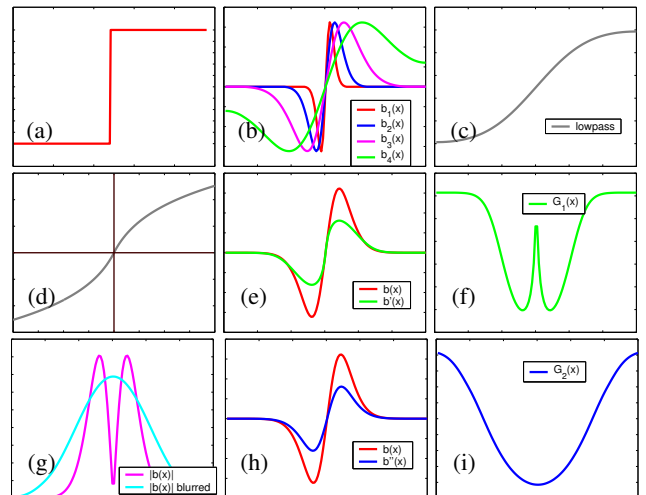
Figure 1: Subband Architectures.

Figure 2: Subbands and nonlinear distortions. (a) A step edge $s(x)$. (b) Subbands of $s(x)$. (c) Lowpass residue of $s(x)$. (d) A sigmoid. (e) Subband $b(x)$ modified by the sigmoid in (d). Note the shape distortions. (f) Effective gain $G_1(x)$ of the sigmoid. (g) Rectified and blurred subband to derive a smooth gain control signal $G_2(x)$. (h) Subband modified by $G_2(x)$. Distortions are reduced. (i) $G_2(x)$.

version of subband activity. The compressed subband signal $b''(x)$ is shown in Fig2(h). It is almost the same shape as $b(x)$, but attenuated in amplitude.

The use of smooth gain maps leads to a major reduction in artifacts, and is one of the most important improvements one can make in a subband scheme. The details of computing gain maps for range compression are discussed in section *3.3* and *3.4*.

3.2 Analysis-Synthesis Filter Banks

The filter bank above is conceptually simple, but in many applications a different architecture is preferred. Fig 1(b) shows an analysis-synthesis filter bank, in which one set of filters, $f_1, f_2, \ldots$, called the analysis filter bank, is used to split the signal $s(x)$ into subbands $b_1(x), b_2(x), \ldots$ and then another set of filters, $g_1, g_2, \ldots$, called the synthesis filter bank, is applied to those subband signals $b_1(x), b_2(x), \ldots$ to produce signals $c_1(x), c_2(x), \ldots$. These post-filtered subband signals $c_1(x), c_2(x), \ldots$ are summed to reconstruct the original signal $s(x)$. It is common for the filter banks to be constructed symmetrically, so that the synthesis filters are essentially the same as the analysis filters. Nonlinear distortions generally produce frequencies outside the original subband, and these will tend to be removed by the corresponding synthesis filter. The signal is forced into its proper frequency band before summation, which reduces distortion.

Analysis-synthesis filter banks are often implemented with hierarchical subsampling, leading to a pyramid. Wavelets and quadrature mirror filters (QMFs) are often used this way, in which case they yield orthogonal transforms [Adelson et al. 1987; Mallat 1998]. This is most easily explained by starting in 1-D and using the Haar wavelet pair, which consists of a lowpass filter $f_0 = [1, 1]$ and a highpass filter $f_1 = [-1, 1]$. In Fig 1(c), an input signal $s(x)$ is split into a low band and a high band by convolution with f_0 and f_1. The filter outputs are subsampled by a factor of two, meaning that every other sample is dropped. If the input has N samples, each subband will have $N/2$ samples (sometimes called subband coefficients). The subbands are now upsampled by a factor of two by inserting a zero between each sample. Each of these zero-padded subband signals is convolved with a second filter, which is $g_0 = [1, 1]$ for the low band and $g_1 = [1, -1]$ for the high band. These signals are summed, and the original is reconstructed exactly.

If the same bandsplitting and subsampling procedure is applied to the lowpass signal, as shown in Fig 1(d), and the process is iterated, we have a Haar pyramid. The number of samples falls by 1/2 at each stage. The effective spatial scale of the corresponding highpass filter doubles, and the effective peak spatial frequency halves.

In 2-D, the process can be applied separably in the x and y directions. This leads to three highpass filters and one lowpass filter at each stage, with a subsampling by a factor of 2 in each dimension.

The subsampled pyramids are highly efficient in terms of computation and representation, because the number of samples falls by half in each dimension at each level. The subsampling can lead to problems with aliasing. In the absence of nonlinearities, the aliasing from one subband cancels that from the others, by construction. However, if nonlinearities are imposed, the aliasing cancellation no longer holds. Since range compression inherently involves nonlinearities, this is a concern.

A straightforward solution is to avoid the subsampling altogether. The doubling of spatial scale is achieved by spreading the filter taps and padding with zeros, so that $f_1 = [1, -1]$ becomes $[1, 0, -1]$ and then $[1, 0, 0, 0, -1]$ on succeeding stages. f_0 is padded in the same way, and by combining f_0 and f_1 separably in the x and y directions we get four 2D zero padded filters (hi_x, hi_y, hi_{xy}, lo in Fig 1(e)). The synthesis filters are basically the same, also combining f_0 and f_1 separably, except that f_1 is temporally reversed. This means that the transform is highly overcomplete, but the math still works out

so that the output is a replica of the input, if no operations are performed on the subband signals. This oversampling technique is commonly used in denoising.

The Haar filters that we have used in the above discussion are not very frequency selective, and so don't cleanly separate the information in the subbands. Vuylsteke and Schoeters [1998] specifically eschew the Haar filters due to their poor bandpass characteristics. However, they are the easiest filters to explain and to implement. We find that they can produce surprisingly good results when coupled with oversampling and smooth gain maps.

Since step edges are such important stimuli, one might assume that the best filters would be those that are specifically responsive to edges, i.e., odd-symmetric filters such as first derivatives. Retinex and other gradient domain methods have this attractive property, and both the Lee [2001] and the Labaere and Vuylsteke [1998] patents advocate the use of the Mallat and Zhong wavelets, which are discrete derivatives on the analysis side and more extended edge operators on the synthesis side. However, we have found that even-symmetric filters such as Adelson et al's 9-tap QMFs [1987] perform very well on this task when used with smooth gain maps, often giving more pleasing results than the Haars. Note that these QMFs have much better frequency tuning than the Haars.

It is interesting at this point to compare the Haar bandsplitting approach to the gradient domain approach used by Fattal et al. [2002], in the simple case of 1-D signals. In both cases the signal is convolved with the filter $f_1 = [-1, 1]$, which is a discrete derivative operator and emphasizes the high frequencies. In the case of the one stage Haar, there is a second filter path containing the low frequencies passed by the filter $f_0 = [1, 1]$. Reconstruction (the inverse transform) involves convolutions and summation using matching filters. By contrast, in the gradient (derivative) domain, although the gradients can be modified in a multi-scale fashion, there is no second signal devoted to the low frequencies. All the information (except DC) is carried in the highpass signal, and the inversion process implicitly involves amplification of the low frequencies.

The Laplacian pyramid is another example of a subsampled system with analysis and synthesis filters. Note, however, that it is not symmetrical. The analysis filters are bandpass, and the synthesis filters are lowpass. Thus the synthesis filters can remove high frequency artifacts introduced by nonlinear processing, but not low frequency artifacts. It is possible to use the Laplacian pyramid architecture without subsampling, which reduces aliasing effects, but the asymmetry remains. When nonlinearities introduce distortions that show up in low frequencies, the synthesis filters cannot remove them. In spite of these problems, we can get fairly good results with the Laplacian pyramid when we compute smooth gain maps.

In summary, there are many ways to build subband systems, and they will deal with nonlinear distortions differently. It is generally better to oversample, in order to avoid the introduction of aliasing artifacts. It is generally better to use an analysis-synthesis filter bank, with the nonlinear operations sandwiched in the middle. A symmetrical analysis-synthesis filter bank, in which the synthesis filters are tuned to the same frequency band as the analysis filters, will be especially effective in controlling the nonlinear distortions.

3.3 Automatic Gain Control

As noted above, it is advantageous to use a smooth gain map to control the strength of the subband signals. For ideas on creating this map, it is interesting to consider the use of gain control in the human visual system.

The human visual system confronts a serious challenge with dynamic range in everyday life. Neurons have a low dynamic range, and they are noisy, so it is important to keep them within an optimal operating range whenever possible. The first type of automatic gain control happens at the retina, where the photoreceptors rapidly

adapt to the ambient light level. For our purposes this process can be crudely modeled as taking the log of the input intensity. In the first stages of visual cortex, area V1, the important variable is contrast rather than intensity[Peli 1990], and the neurons have a gain control mechanism known as "contrast gain control" or "contrast normalization" [Heeger 1992]. Responses to moderately low contrasts are boosted, while responses to high contrasts are reduced. This makes good use of the information capacity of the neurons by keeping the responses away from the ceiling and floor. Area V1 has neurons tuned to different orientations and spatial frequencies, and can be thought of as doing a local subband decomposition using filters similar to wavelets. The gain of a given neuron is controlled by the activity level of many neurons in its immediate neighborhood. Additionally, the gain is not just controlled by neurons tuned for the same orientation and spatial scale; rather, the gain signal involves a pooling of multiple orientations and scales.

The gain control varies from point to point depending on the activity, so we can think of it as forming a gain map in register with the subband image. This is analogous to Fattal et al's gain map applied to the gradient image.

In building gain maps for range compression, we first construct an activity map from local filter responses. Since the responses can be positive or negative, we take the absolute value. We then pool over a neighborhood with a simple blur. The activity map is then converted to a gain map, which has lower gain in regions of high activity.

Here is a more detailed description of the construction of a gain map. In a standard separable n-level subband pyramid there are $3n+1$ subband images, and they are denoted as $B_i(x,y)$ ($i = 1,\ldots,3n+1$), where $B_{3n+1}(x,y)$ is the lowpass residue. We rectify each subband image B_i by taking the absolute value, and then blur it with a Gaussian kernel to get an activity map:

$$A_i(x,y) = g(\sigma) * |B_i(x,y)| \tag{2}$$

The size of the Gaussian kernel is proportional to the subband's scale. If the kernel used for the subbands at the finest scale has variance σ_1, then the kernel for the subbands at the next coarser level will be twice as big.

The nonlinear function $p()$ used to derive a gain map from an activity map, should be monotonic decreasing, turning the gain down where the activity is high and up where the activity is low. There are various choices as of the particular form of $p()$. One of them gives gamma-like mapping:

$$G_i(x,y) = p\{A_i(x,y)\} = \left(\frac{A_i(x,y)+\varepsilon}{\delta}\right)^{(\gamma-1)} \tag{3}$$

where γ is a compressive factor between 0 and 1, ε is a noise level related parameter which prevents the noise from being blown up, and also prevents singularities in the gain map, considering the power $(\gamma-1)$ is below zero. δ can be understood as a gain control stability level: the gain is turned up for places where activities are below δ and turned down for places where activities are above δ, in either case bringing the activities closer to δ.

Since we are modifying each subband separately, it is possible that gains at different scales will be mismatched at important features, leading to distortions of these features. Therefore we need a method that keeps the gains matched. Similar to the method proposed by Fattal et al. [2002], we set the parameter δ according to the activity statistics (with M and N being the width and height of the subband image):

$$\delta_i = \alpha_i \frac{\sum\limits_{(x,y)} \left(A_i(x,y)\right)}{M \times N} \tag{4}$$

where α_i is a constant related to spatial frequency. We have it linearly range from 0.1 at the lowest frequency to 1.0 at the highest fre-

quency. In natural images, the subband activity measures are highly correlated at different scales, and the separate gain maps with δ set this way, tend to line up. Other parameters like γ and ε are set to be the same for all the subbands.

After the gain maps are computed they are used to modify the subbands:

$$B'_i(x,y) = G_i(x,y) \times B_i(x,y) \tag{5}$$

The modified subbands are then convolved with the synthesis filters and summed to reconstruct the range compression result.

3.4 Aggregated Gain Map

To some extent, the matching of local subband gains depends on accidents of image statistics: it is usually the case that high activity in one band is spatially correlated with high activity in adjacent bands. To avoid depending on this assumption, we can create a single gain map that will be used to modify all the subbands. This is straightforward to apply when all of the subbands are represented at full resolution. To compute the gain map, we first compute an aggregated activity map by pooling activity maps over scales and orientations:

$$A_{ag}(x,y) = \sum_{i=1,\ldots,3n+1} A_i(x,y) \tag{6}$$

A single gain map can then be derived from this aggregated activity map $G_{ag}(x,y) = p(A_{ag}(x,y))$, where $p()$ is of the same form as in Eq.(3). δ is set to one tenth the average of A_{ag}.

(a)   (b)   (c)
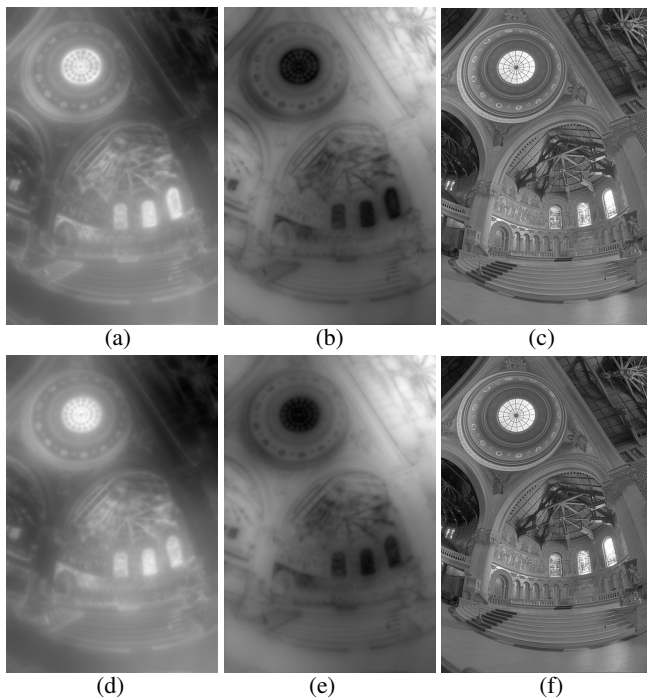
(d)   (e)   (f)

Figure 3: Activity and gain maps, with Haars (a-c) and QMFs (d-f), respectively. (a,d) Aggregated activity map A_{ag} pooled from all orientations and scales. (b,e) Gain map G_{ag} computed from A_{ag}. (c,f) The resulting range-compressed monochrome image. HDR image source: Paul Debevec.

This gain map is then used to modify all the subbands, and a scale-related constant m_i is used to control to what extent different frequencies are modified:

$$B'_i(x,y) = m_i G_{ag}(x,y) \times B_i(x,y) \tag{7}$$

Such a gain map G_{ag} with a Haar pyramid is shown in Fig 3(b), along with the corresponding aggregated activity map A_{ag} shown in Fig 3(a), from which G_{ag} is derived. Fig 3(c) shows the gray-scale range compression result after G_{ag} is applied to the subbands. Fig 3(d,e,f) show G_{ag}, A_{ag}, and the range compression result using QMFs. As A_{ag} is pooled from all frequencies, G_{ag} contains energy in all frequencies. At first it may seem strange to modify the low frequency subbands with a gain map that contains a lot of high frequency detail, or vice versa, but due to the symmetric analysis-synthesis subband architecture, modified subbands are post-filtered by the synthesis filter bank, and therefore all modifications are confined within the subbands themselves.

4 Experimental Results on Range Compression

Handling color and clipping. For color images we first convert RGB to the HSV space. Then we perform range compression on the V (value) channel, keep the hue (H) and the saturation (S) unchanged, and then convert it back to RGB to get the result. Sometimes the range compressed images look over-saturated, in which cases they can be desaturated, by reducing the saturation (S) by a factor of r_s ($S' = S/r_s$) before converting HSV back to RGB. r_s can be set between 1.0 and 2.0.

As a final step the extreme percentiles of the intensities are clipped, and values in between are linearly scaled, so as to eliminate the sparse regions on the ends of the final histogram, and to maximize the use of the display range. This can cause some minor clipping in the very brightest and the very darkest pixels, but in practice does not cause visible problems.
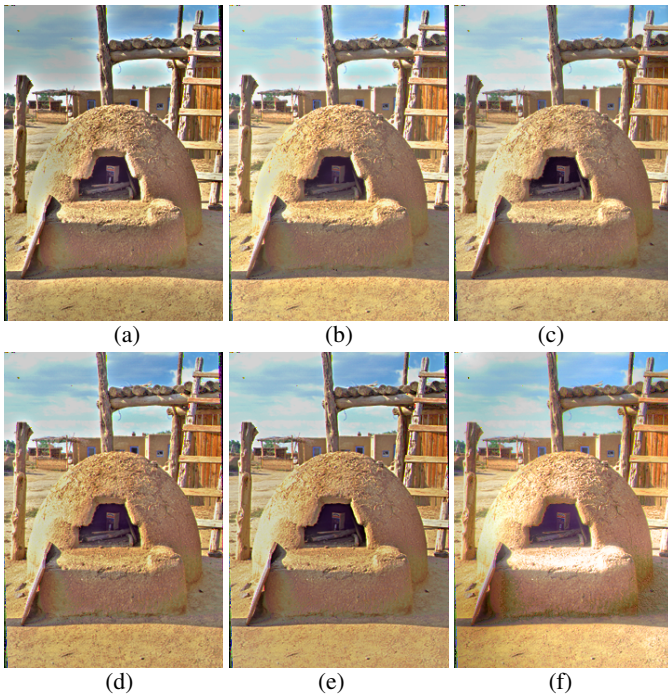

Figure 4: Igloo. (a)Laplacian pyramid with sigmoid. (b) Oversampled Haars with sigmoid. (c)Laplacian pyramid with smooth gain control. (d) Oversampled Haars with multiple gain maps. (e) Oversampled Haars with an aggregated gain map. (f) Result by Fattal et al. [2002] (color is modified). HDR image source: Shree Nayar.

Experimental Results. Fig 4 shows the effects of smooth gain control and different subband architectures on the "igloo" picture.
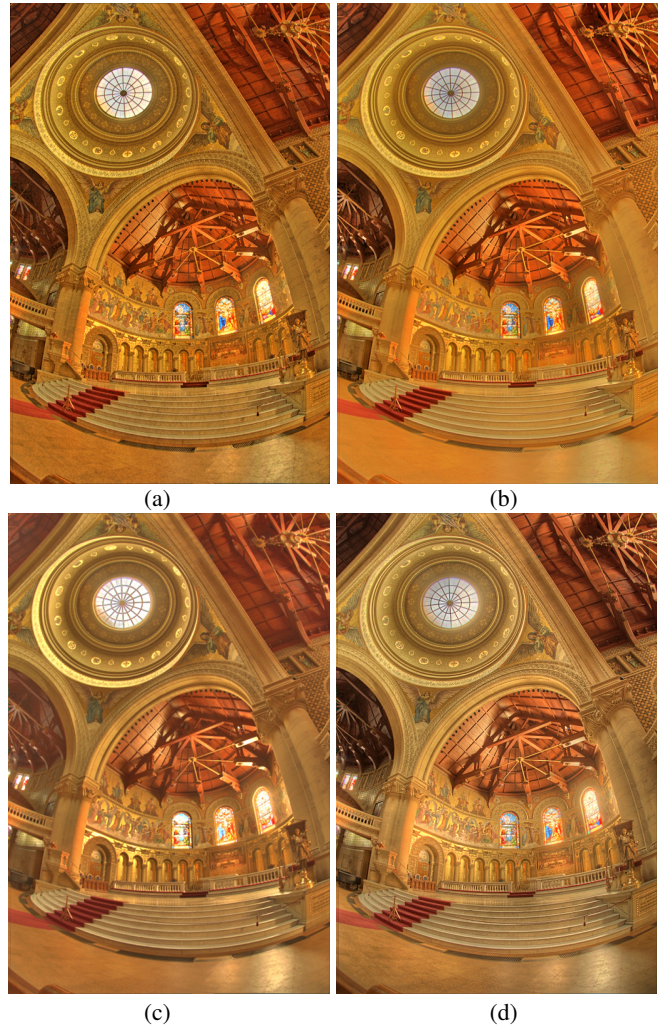

Figure 5: Memorial Church. (a) Our result using multiple gain maps; (b) Our result using one aggregated gain map. (c) Result by Durand and Dorsey [2002]. (d) Result by Fattal et al. [2002]. HDR image source: Paul Debevec.

We get Fig 4(a) using Laplacian pyramid and a point-wise sigmoid on the coefficients, Fig 4(b) using oversampled Haars and a point-wise sigmoid, Fig 4(c) using Laplacian pyramid and smooth gain control, Fig 4(d) using oversampled Haars, where each subband is modified by its own gain map (3.3), Fig 4(e) using oversampled Haars, where all the subbands are modified by one single gain map computed from an aggregated activity map (3.4). Note the halo artifacts around the pole, in (a) and (b). The worst halos are seen with the Laplacian pyramid and a sigmoid (Fig 4(a)); however, the Laplacian pyramid performs fairly well when smooth gain maps are used (Fig 4(c)). Pattanaik et al. [1998] also used Laplacian pyramids with gain control, but got halo artifacts. The difference between their method and the one giving Fig 4(c) lies in how the gains are computed. Pattanaik et al. [1998] control the bandpass gains using the lowpass signals, whereas for Fig 4(c) we control the gain of each bandpass signal using a rectified and blurred version of the bandpass signal itself. We also compare these results with that published by Fattal et al. [2002], shown in (f). The colors of (f) are adjusted so that they match those of (a-e). We can see Fig 4(c-f) all did well in both preserving details and compressing the range. The difference in appearance between (f) and (c,d,e) depends mainly on how aggressive the compression is, and should

not be over-interpreted.

Shown in Fig 5 are range compression results on the memorial HDR image. For Fig 5(a) each subband is modified by its own gain map (*3.3*), while for Fig 5(b) all the subbands are modified by the aggregated gain map (*3.4*). The one using a single gain map achieves a cleaner look. We compare our results with the ones published by Durand and Dorsey [2002] (Fig 5(c)), and by Fattal et al. [2002] (Fig 5(d)). All of the methods give visually pleasing results, and are successful in making detail visible in both the bright and dark regions. Again, there are some differences between the results, including overall difference in color and sharpness, but these should not be over-interpreted since they may change depending on the details of the implementation.

More results with a single gain map are shown in Fig 6. For all the results shown here, gamma nonlinearity (Eq.(3)) is used, and γ is set to 0.6. m_i in Eq.(7) is set to 1.0 for the three subbands at the finest scale, 0.8 for the three subbands at the second finest scale, and 0.6 for all the others including the lowpass.

Figure 6: More range compression results. HDR image sources: Shree Nayar, Cornell CS.

5 Companding of HDR Images

Given that we can compress the range of an HDR image into an LDR image, it is interesting to ask whether the process can be inverted. Suppose, for instance, that we have squeezed a 12-bit image into an 8-bit image. Can we retrieve a good 12-bit image? Clearly we can't do it perfectly, but perhaps we can get a good approximation. We will refer to this process as "HDR image companding". This problem appears to have received little attention.

There are various ways of representing 12 bit images, including various lossless and lossy standards. There are also some hybrid techniques that combine an 8-bit format like JPEG with auxiliary information (a second image) to increase the dynamic range [Ward and Simmons 2004]. However, the question we ask is this: Can we retrieve a high quality 12 bit image from an 8 bit image without sending another image in a side channel? And further, can we do this so that the 8 bit image is one that we would want to view directly on an 8 bit display?

The default method for converting a 12 bit image to 8 bits is simply to divide by 16 and quantize the 4096 levels to 256 levels. To retrieve a 12 bit image, the 256 levels are stretched back to the original 4096. It is better to do this with non-linear quantization, in which the original linear intensity values are compressed with, for example, a log or a power function, followed by quantization. The 12 bit image is retrieved by applying the inverse function. This method will lead to visible quantization steps in the 12 bit image, since there are only 8 bits worth of intensity levels.

Suppose, however, that we convert the 12 bit image to an 8 bit image through subband range compression, and then invert the process to retrieve a 12 bit image. The compression process amplifies

low amplitudes and high frequencies, and the expansion process reduces them (relative to the other components). Since quantization artifacts tend to be dominated by low amplitudes and high frequencies, this means that the artifacts will have less visibility in the expanded image than they would with ordinary quantization. One application would be in driving HDR displays. Most software applications today only handle 8 bit images, and most video cards can only put out 8 bit images. It would be very useful if our laptop could output an 8 bit image and have it magically converted into a clean 12 bit image by a specialized display. Of course, we cannot hope to make this conversion without any loss of information, but we can distort our image space so that the accessible set of images more closely matches the ones that we wish to display.

Another application is HDR image storage and transmission. After we turn a 12 bit image into an 8 bit one, the image can be stored in a standard lossless 8 bit format, or can be further compressed with a lossy format such as JPEG. The JPEG will not have the same quality as the original raw 12 bit image, but it will require much less storage space and will be in a standard format. A digital camera that stores HDR JPEGs rather than standard JPEGs will give its user much more flexibility when manipulating the captured image data.

Suppose we ran our range compression algorithm and generated an 8 bit image. If we knew the gain map that was used for each subband, the inversion process would be simple. Unfortunately, we don't know the gain maps, since they were not stored; all we have is the range compressed image itself. We can estimate the gain maps from this image, but these estimates will be imperfect so we will not get the original image back.

To solve this dilemma it is useful to begin at the end. Let us establish a standard method for doing range expansion; i.e., given an 8 bit image, we have an algorithm for expanding it to a 12-bit image. This can be thought of as a decoding process. Our problem now is to create an "encoded" image that will yield the desired image when it is decoded. We do not have a method for finding this image directly, but we can search for it using an iterative technique. In the next section, we describe our range-expansion method, and then an iterative range-compression method that can be coupled with it.

5.1 Range Expansion

The range expansion follows almost exactly the same scheme as the range compression does, except that instead of multiplying the subband coefficients with their gains we divide them by their gains. The gain maps are computed in the same way as described in section 3.3. An LDR image I_l is first decomposed into subbands $B_{l,i}$, which are then rectified and blurred to give the activity maps. Gain maps $G_{l,i}$ are then computed from the activity maps using Eq.(3), and they are used to modify the subbands:

$$B'_{l,i}(x,y) = \frac{B_{l,i}(x,y)}{G_{l,i}(x,y)} \qquad (8)$$

A range expanded image I_e is reconstructed from the modified subbands.

Next, given this range expansion method, we want to find an LDR image I_l that, when expanded, well approximates a target HDR image I_h. A first thought would be to get I_l directly by compressing the range of I_h, using subband decomposition and automatic gain control as described in Section 3.3. Gain maps $G_{h,i}$ are computed from the subbands $B_{h,i}$ of I_h, and are multiplied with the subbands: $B'_{h,i}(x,y) = G_{h,i}(x,y) \times B_{h,i}(x,y)$. If the transforms are orthogonal, and somehow magically $G_{h,i}(x,y)$ is equal to $G_{l,i}(x,y)$, then by doing the expansion in Eq.(8) we can get I_e equal to I_h. This will not occur because $G_{h,i}$ and $G_{l,i}$ cannot be the same, since one is estimated from the subbands of I_l and the other from the subbands of I_h. But these will be close, as the subbands of I_l and those of I_h are
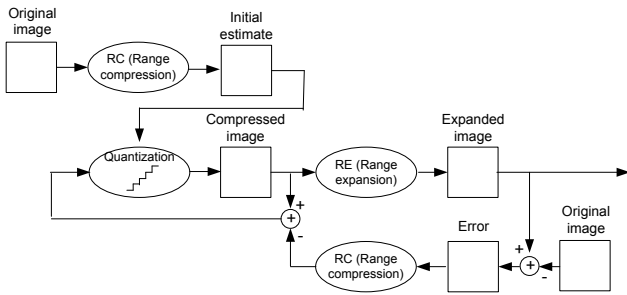
Figure 7: The companding flowchart.

highly correlated, which makes $G_{l,i}$ and $G_{h,i}$ highly correlated. We can look at how much I_e and I_h differ, and add a signal E_l to I_l in order to reduce the error between I_e and I_h. We do this iteratively until we find a satisfactory result.

5.2 Error Feedback Search

The search procedure is illustrated in Fig 7. We start the search by computing the initial estimate as the range-compressed version of the original image. This initial estimate is then quantized and passed through the RE (range expansion) box. We feed the reconstruction error back into the loop and improve our estimate. We compute the difference between the expanded image and the original image, run this error image through RC (range compression), and add this compressed error back to the previous quantized estimate. The resulting image is then quantized to get the updated estimate. This process is repeated. In our experience we reach satisfactory results after 8-12 runs. The choice of parameters (γ in Eq.(3) and α's in Eq.(4)) in our experience doesn't affect the process much, but RE and RC will have to use the same set of parameters, which means the parameters should be sent as header information with the LDR image.

We find the 8 bit image iteratively, but the procedure for expanding it to 12 bits is a one-shot multiscale procedure.

Note that the RC and RE boxes in the above iterations don't include taking the log of the image intensities. For high dynamic range images the companding is assumed to be applied in the log domain, i.e., the original image has gone through a log transformation before going into the loop.

5.3 Experimental Results on Companding

For companding color images we first convert RGB to the HSV space. The value (V) is then run through the companding loop and a compressed V is obtained when the iterations stop. This compressed V is combined with the original hue (H) and the original saturation divided by a factor of r_s (r_s is set between 1.0 and 2.0), and converted back to RGB to get the compressed color image. This is the same as what we did for color HDR image compression. Similarly when we're going to expand a compressed color image up to 12 bits, the one-step range expansion is done on its V channel. The saturation is multiplied by the same r_s, the hue is kept the same, and they are combined with the expanded V to get the HDR color image back.

Since it is impossible to display a true HDR image in this paper, we will demonstrate an example in which the "HDR image" is 8 bits, and the "LDR image" is 3 bits. That is, we will compress an 8 bit image to a 3 bit image - dropping its bit depth by 5 bits – and then expand it back to 8 bits.

Fig 8(a) shows an ordinary picture of a baby at 8 bits (256 levels). The dynamic range of the displayed image is appropriate for an 8 bit image. Fig 8(b) shows the same image after it was scaled down
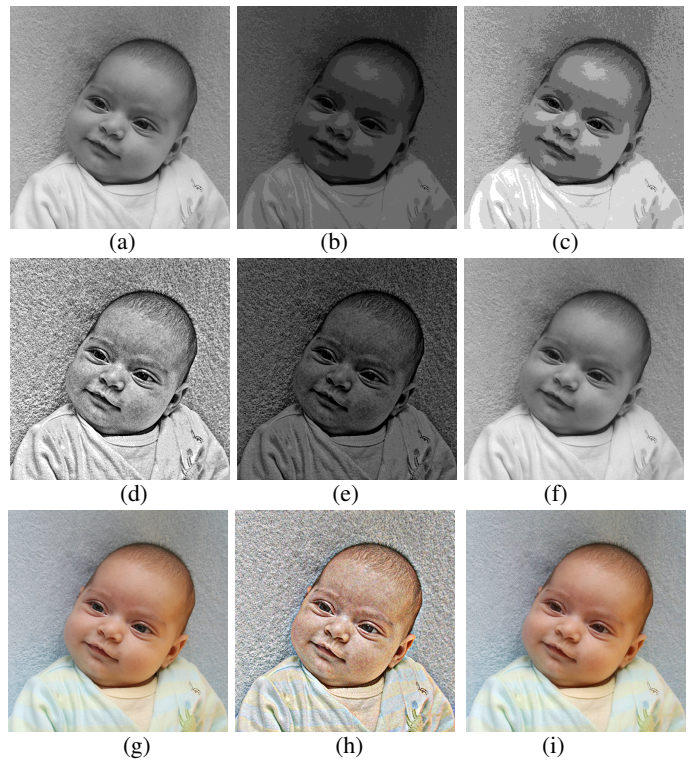

Figure 8: Baby companding. (a-f) In monochrome: (a) the original. (b) The original quantized to 3 bits, viewed at low contrast. (c) The 3-bit image in (b) scaled up to fill the range. (d) Compressed image at 3 bits. (e) The 3-bit compressed image in (d) viewed at low contrast. (f) The 8-bit image reconstructed from the 3-bit image (d) using our expansion technique. (g-i) In color: (g) the original, 8 bits/channel. (h) Compressed, 3 bits/channel. (i) Expanded from (h).

to a smaller range and linearly quantized to 3 bits (8 levels). This image is shown with lower contrast and brightness, to suggest a low dynamic range device. (Since the image has 5 fewer bits, we might in principle show it at 1/32 the dynamic range of the original image, but here we show it at about 1/3.) Fig 8(c) shows the same 3-bit image with the brightness scaled up to fill the full range of the display. The quantization artifacts are quite visible as contouring. It is possible to improve this result using nonlinear quantization, but only slightly.

Fig 8(d) shows an image that has been compressed and quantized to 3 bits. Fig 8(e) shows this image as it would appear on our hypothetical LDR display. Fig 8(f) shows the same image after expansion using our subband technique. This picture appears nearly identical to the original picture and it has no visible contouring artifacts.

This companding scheme provides us with an image that can be displayed directly on a low dynamic range device, or can be displayed after range expansion on a high dynamic range device. Fig 8(g) shows the baby image in color, at 8 bits per color channel (i.e. a normal RGB image). Fig 8(h) shows the image having been compressed to a 3-bit/channel image. Fig 8(i) shows the 8 bit image that is reconstructed by the expansion technique. The expanded picture is not identical to the original, but the errors are almost invisible.

Turning now to the more pertinent problem of coding an HDR image consider the example in Fig 9. The 8 bit range compressed version of the HDR lamp is shown in Fig 9(a). Fig 9(b)-(d) are a few "slices" of the reconstructed HDR image simulating increasing

(a)   (b)

(c)   (d)

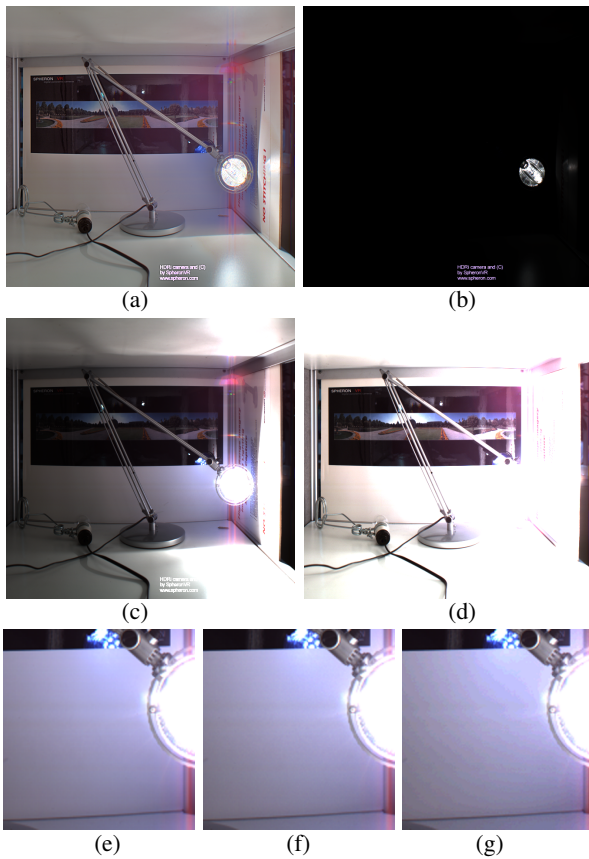(e)   (f)   (g)

Figure 9: Lamp companding. (a) The range compressed image, 8 bits/channel. (b)-(d) Three exposure slices of the HDR image reconstructed from (a). (e) A closeup of the original. (f) A closeup of our reconstructed HDR image. (g) A closeup of the HDR image reconstructed with log quantization. HDR image source: Spheron.

exposures. Fig 9(e) shows a closeup of part of an exposure slice of the original HDR lamp image. Fig 9(f) shows the reconstruction of this slice achieved by expanding our 8 bit compressed image. It replicates the visual impression of the original. Fig 9(g) shows the result of compressing and expanding with 8 bits in the log domain. This image shows visible contouring due to quantization. In our experience the reconstruction PSNR (Peak Signal to Noise Ratio) on a typical image (measured in the log intensity domain) is 60-75 dB. From the standpoint of squared error, the proposed companding method doesn't perform quite as well as ordinary LUT (Look-Up Table) companding, but it is much better visually. The artifacts do not take the form of visible contours; instead, they are small errors in local contrast within subbands, and these are not visually disturbing. Even when there is a visible difference between the original and the companded image, it is difficult to guess which is which.

A question is whether we can get the best of both worlds, and full backward compatibility. Is the 8 bit image that is best for expansion to 12 bits also the image that looks best when displayed directly on a standard LDR display? We cannot guarantee it is, due to the emphasis the high frequencies. But in our experience the images look similar to those produced by our direct range compression technique.

5.4 Combining JPEG with Companding

It would be useful to take one more step, and encode the 8 bit image with JPEG. JPEG compression is lossy and introduces its own arti-

facts. The question is how bad these artifacts will become after the expansion step. We find it is possible to get good results if the JPEG encoding is done correctly. Not surprisingly, it is necessary to code the JPEG at a fairly high bit rate, such as 1.5 to 4 bits per pixel. This still represents a substantial savings: When a 12 bit/channel image is converted to a 4 bit/pixel JPEG, the compression is from 36 bits to 4 bits, for a factor of 9.

The most troublesome artifacts, for our technique, arise when the chrominance channels (Cr, Cb) are subsampled, as is done in most off-the-shelf JPEG encoders. We used the IJG (Independent JPEG Group [IJG]) encoder with chrominance subsampling turned off. Fig 10 shows results at a bit rate of 1.8 bpp and 3.8 bpp.

(a)   (b)

(c)   (d)

(e)   (f)

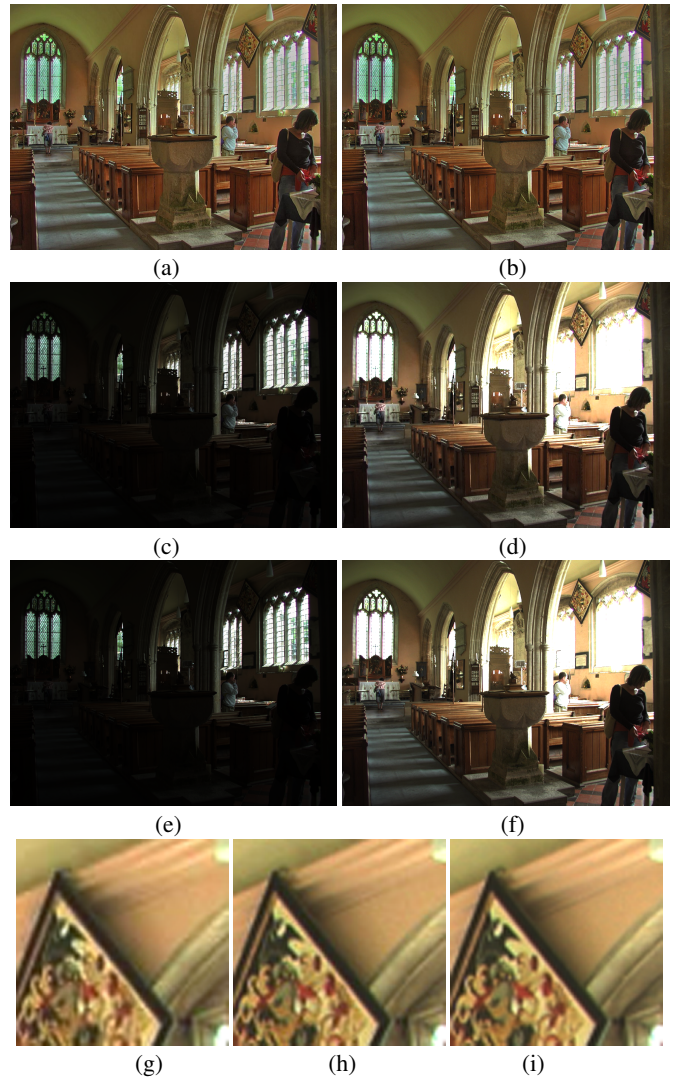(g)   (h)   (i)

Figure 10: Dyrham Church companding with JPEGs. (a,b) The range compressed images, saved as: (a) a 1.8 bpp JPEG, and (b) a 3.8 bpp JPEG. (c,d) Two exposure slices of the HDR image reconstructed from the 1.8 bpp JPEG shown in (a). (e,f) Two exposure slices of the HDR image reconstructed from the 3.8 bpp JPEG shown in (b). (g) A closeup of (d). (h) A closeup of (f). (i) A closeup of the original. HDR image source: Greg Ward.

6 Discussion

There are a number of techniques for compressing high dynamic range images in such a way that they are viewable on ordinary displays. Multiscale techniques sometimes have the reputation of being difficult to use without introducing halo artifacts. However, the implementation we describe here, based on analysis-synthesis subband architectures and smooth gain control, gives good range compression without disturbing halos. We describe some simple implementations of subband range compression, and show that the results are competitive with the leading techniques such as Durand and Dorsey [2002], Reinhard et al. [2002], and Fattal et al. [2002].

We have not attempted to write optimized code, and cannot compare our speed with the other techniques. However, the filtering operations involved are simple to compute, and there is no need to use large or complex filters. In the future, it is likely that hardware wavelet processing will be common in image processing systems, and it will be straightforward to utilize this hardware for range compression.

This compression scheme can be inverted, so that a low dynamic range image, e.g., an 8 bit image, can be expanded into a high dynamic range image, e.g., a 12 bit image. Given an original 12 bit image, we can compute an 8 bit image that offers a good visual rendition of the HDR image, and which can be expanded to approximate the original 12 bit image with minimal degradation. This could be useful, for example, when using a standard video card to drive both LDR and HDR displays. The ability to represent 12 bit images in 8 bit file formats is also an advantage for backward compatibility in various systems, and when combined with JPEG compression can lead to further savings in storage.

7 Acknowledgments

References

ADELSON, E. H., SIMONCELLI, E., AND HINGORANI, R. 1987. Orthogonal pyramid transforms for image coding. In *Visual Communications and Image Processing II, Proc. SPIE*, vol. 845, 50–58.

DEBEVEC, P. E., AND MALIK, J. 1997. Recovering high dynamic range radiance maps from photographs. In *Proceedings of SIGGRAPH 97*, Computer Graphics Proceedings, Annual Conference Series, 369–378.

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

DICARLO, J. M., AND WANDELL, B. A. 2001. Rendering high dynamic range images. In *In Proceedings of the SPIE: Image Sensors*, vol. 3965, 392–401.

DURAND, F., AND DORSEY, J. 2002. Fast bilateral filtering for the display of high-dynamic-range images. *ACM Transactions on Graphics 21*, 3 (July), 257–266.

FATTAL, R., LISCHINSKI, D., AND WERMAN, M. 2002. Gradient domain high dynamic range compression. *ACM Transactions on Graphics 21*, 3 (July), 249–256.

FERWERDA, J. A., PATTANAIK, S., SHIRLEY, P. S., AND GREENBERG, D. P. 1996. A model of visual adaptation for realistic image synthesis. In *Proceedings of SIG-GRAPH 96*, Computer Graphics Proceedings, Annual Conference Series, 249–258.

HEEGER, D. J. 1992. Half-squaring in responses of cat simple cells. *Visual Neurosci. 9*, 427–443.

IJG. Independent jpeg group. *http://www.ijg.org/files/*.

JOBSON, D. J., RAHMAN, Z., AND WOODELL, G. A. 1997. A multi-scale retinex for bridging the gap between color images and the human observation of scenes. *IEEE Transactions on Image Processing 6*, 7 (July), 965–976.

LABAERE, F., AND VUYLSTEKE, P., 1998. Image contrast enhancing method. U.S. Patent no. 5,717,791.

LAND, E. H., AND MCCANN, J. J. 1971. Lightness and retinex theory. *Journal of the Optical Society of America 61*, 1 (Jan), 1–11.

LARSON, G. W., RUSHMEIER, H., AND PIATKO, C. 1997. A visibility matching tone reproduction operator for high dynamic range scenes. *IEEE Transactions on Visualization and Computer Graphics 3*, 4, 291–306.

LEE, H.-C., 2001. Automatic tone adjustment by contrast gain-control on edges. United States Patent 6,285,798, September.

MALLAT, S., AND ZHONG, S. 1992. Characterization of signals from multiscale edges. *IEEE Trans. on PAMI 14*, 7, 710–732.

MALLAT, S. 1998. *A Wavelet Tour of Signal Processing*. New York: Academic.

MANN, S., AND PICARD, R. 1995. Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. In *IS&T's 48th annual conference*, 422–428.

MITSUNAGA, T., AND NAYAR, S. K., 2000. High dynamic range imaging: Spatially varying pixel exposures. In IEEE CVPR, 472-479.

NAKAZAWA, MASAYUKI, TSUCHINO, AND HISANORI, 1995. Method of compressing a dynamic range for a radiation image. United States Patent 5,471,987.

PATTANAIK, S. N., FERWERDA, J. A., FAIRCHILD, M. D., AND GREENBERG, D. P. 1998. A multiscale model of adaptation and spatial vision for realistic image display. In *Proceedings of SIGGRAPH 98*, Computer Graphics Proceedings, Annual Conference Series, 287–298.

PELI, E. 1990. Contrast in complex images. *J. Opt. Soc. Am. A. 7*, 10, 2032–2040.

REINHARD, E., AND DEVLIN, K. 2005. Dynamic range reduction inspired by photoreceptor physiology. *IEEE Transactions on Visualization and Computer Graphics*.

REINHARD, E., STARK, M., SHIRLEY, P., AND FERWERDA, J. 2002. Photographic tone reproduction for digital images. *ACM Transactions on Graphics 21*, 3 (July), 267–276.

SEETZEN, H., HEIDRICH, W., STUERZLINGER, W., WARD, G., WHITEHEAD, L., TRENTACOSTE, M., GHOSH, A., AND VOROZCOVS, A. 2004. High dynamic range display systems. *ACM Transactions on Graphics 23*, 3 (Aug.), 760–768.

STOCKHAM, T. 1972. Image processing in the context of a visual model. *Proc. IEEE 60*, 828–842.

TOMASI, C., AND MANDUCHI, R. 1998. Bilateral filtering for gray and color images. In *In Proc. IEEE Int. Conf. on Computer Vision*, 836–846.

TUMBLIN, J., AND RUSHMEIER, H. E. 1993. Tone reproduction for realistic images. *IEEE Computer Graphics & Applications 13*, 6 (Nov.), 42–48.

TUMBLIN, J., AND TURK, G. 1999. Lcis: A boundary hierarchy for detail-preserving contrast reduction. In *Proceedings of SIGGRAPH 99*, Computer Graphics Proceedings, Annual Conference Series, 83–90.

TUMBLIN, J. 1999. *Three methods of detail-preserving contrast reduction for displayed images*. PhD thesis, College of Computing Georgia Inst. of Technology.

VUYLSTEKE, P., AND SCHOETERS, E., 1998. Method and apparatus for contrast enhancement. U.S. Patent no. 5,805,721.

WARD, G., AND SIMMONS, M. 2004. Subband encoding of high dynamic range imagery. In *APGV '04: Proceedings of the 1st Symposium on Applied perception in graphics and visualization*, ACM Press, 83–90.

WARD, G. J. 1994. The radiance lighting simulation and rendering system. In *Proceedings of SIGGRAPH 94*, Computer Graphics Proceedings, Annual Conference Series, 459–472.