Edge-Preserving Decompositions for Multi-Scale Tone and Detail Manipulation

Zeev Farbman
The Hebrew University

Raanan Fattal
The Hebrew University

Dani Lischinski
The Hebrew University

Richard Szeliski
Microsoft Research

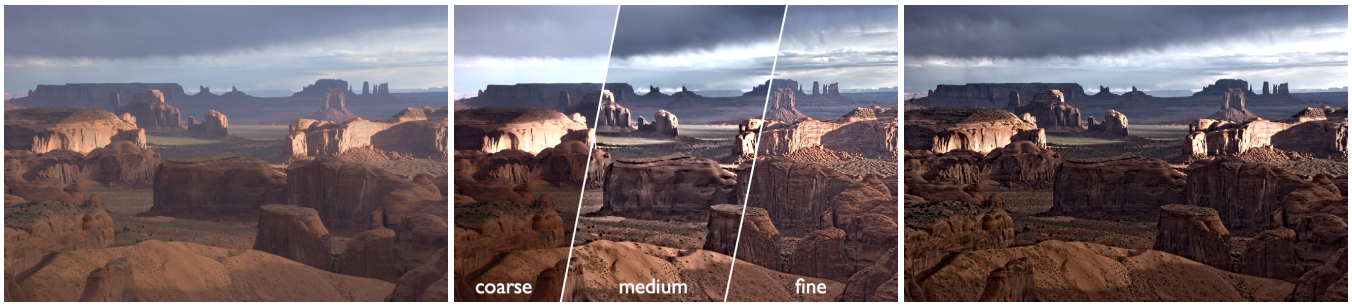


Figure 1: *Multi-scale tone manipulation. Left: input image (courtesy of Norman Koren, www.normankoren.com). Middle: results of (exaggerated) detail boosting at three different spatial scales. Right: final result, combining a somewhat milder detail enhancement at all three scales.* ***Note:*** *all of the images in this paper are much better appreciated when viewed full size on a computer monitor.*

Abstract

Many recent computational photography techniques decompose an image into a piecewise smooth base layer, containing large scale variations in intensity, and a residual detail layer capturing the smaller scale details in the image. In many of these applications, it is important to control the spatial scale of the extracted details, and it is often desirable to manipulate details at multiple scales, while avoiding visual artifacts.

In this paper we introduce a new way to construct edge-preserving multi-scale image decompositions. We show that current base-detail decomposition techniques, based on the bilateral filter, are limited in their ability to extract detail at arbitrary scales. Instead, we advocate the use of an alternative edge-preserving smoothing operator, based on the weighted least squares optimization framework, which is particularly well suited for progressive coarsening of images and for multi-scale detail extraction. After describing this operator, we show how to use it to construct edge-preserving multi-scale decompositions, and compare it to the bilateral filter, as well as to other schemes. Finally, we demonstrate the effectiveness of our edge-preserving decompositions in the context of LDR and HDR tone mapping, detail enhancement, and other applications.

CR Categories: I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms; I.4.3 [Image Processing and Computer Vision]: Enhancement—Grayscale Manipulation; I.4.9 [Image Processing and Computer Vision]: Applications;

Keywords: bilateral filter, edge-preserving smoothing, detail enhancement, digital darkroom, high dynamic range, image abstraction, multi-scale image decomposition, tone mapping

1 Introduction

Edge-preserving image smoothing has recently emerged as a valuable tool for a variety of applications in computer graphics and image processing. In particular, in computational photography it is often used to decompose an image into a piecewise smooth *base layer* and a *detail layer*. Such a decomposition may then be used for HDR tone mapping [Tumblin and Turk 1999; Durand and Dorsey 2002], flash/no-flash image fusion [Petschnigg et al. 2004; Eisemann and Durand 2004], transfer of photographic look [Bae et al. 2006], image editing [Oh et al. 2001; Khan et al. 2006], and for other tasks.

In many of these applications it is paramount to have control over the spatial scale of the details captured by the detail layer. Furthermore, it is often desirable to operate on details at a variety of scales, rather than just at a single scale. For example, in the digital darkroom, a photographer might want to separately manipulate the tone at several different scales, in order to add depth and increase detail clarity, and the final result might combine together several such manipulations, as demonstrated in Figure 1.

Traditionally, operating on images at multiple scales is done using multi-scale decompositions, such as the Laplacian pyramid [Burt and Adelson 1983]. Multi-scale decompositions were also used for tone mapping (e.g., [Jobson et al. 1997; Pattanaik et al. 1998]). However, such pyramids are constructed using linear filters, which are known to produce halo artifacts near edges. These artifacts may be reduced by using non-linear edge-preserving smoothing filters, such as anisotropic diffusion [Perona and Malik 1990], robust smoothing [Black et al. 1998], weighted least squares [Lagendijk et al. 1988], and the bilateral filter [Tomasi and Manduchi 1998]. Among these, the bilateral filter (BLF) has recently emerged as the *de facto* tool of choice for computational photography applications. Indeed, the BLF has several appealing characteristics [Paris 2007]. However, while this filter is well suited for noise removal and extraction of detail at a fine spatial scale, we show in Section 2 that it is less appropriate for extraction of detail at arbitrary scales, which is necessary for construction of effective multi-scale decompositions.

In this paper, we advocate an alternative edge-preserving operator, based on the weighted least squares (WLS) framework. This framework was originally used to reduce ringing while deblurring images in the presence of noise [Lagendijk et al. 1988], and it is closely related to biased anisotropic diffusion [Nordstrom 1989]. Recently,

this framework was employed for piecewise smooth propagation of sparse constraints [Levin et al. 2004; Lischinski et al. 2006]. We show that the WLS-based operator is robust and versatile, and may be used in many of the applications that have so far been based on the BLF, at the cost of longer computation times. We found this operator to be particularly well-suited for progressive coarsening of images, and for extraction of detail at various spatial scales. Thus, we use it to construct a new kind of an edge-preserving multi-scale image decomposition, which provides an excellent foundation for multi-scale HDR and LDR tone mapping, detail enhancement, and contrast manipulation.

The remainder of this paper is organized as follows. In the next section we discuss various applications of base-detail decompositions, explain the causes of common artifacts, and examine the characteristics of the bilateral filter in this context. Next, in Section 3 we show how the WLS framework may be used to perform edge-preserving smoothing, and describe the multi-scale decomposition construction process. Section 4 presents a detailed comparison between our WLS-based multi-scale decompositions with several previous schemes, while Section 5 discusses its connections with spatially-variant filtering and with anisotropic diffusion. Finally, we demonstrate the utility of our decompositions in the context of several applications.

2 Background

In computational photography, images are often decomposed into a piecewise smooth *base layer* and one or more *detail layers*. The base layer captures the larger scale variations in intensity, and is typically computed by applying an edge-preserving smoothing operator to the image (sometimes applied to the logarithm of the luminance or to the lightness channel of the CIELAB color space). The detail layer is then defined as the difference (or the quotient) between the original image and the base layer. Each of the resultant layers may be manipulated separately in various ways, depending on the application, and possibly recombined to yield the final result.

For example, to reduce the dynamic range of an HDR image, the base layer is typically subjected to a non-linear compressive mapping, and then recombined with the (possibly attenuated or boosted) detail layers [Pattanaik et al. 1998; Tumblin and Turk 1999; Durand and Dorsey 2002]. Fattal *et al.* [2007] employ a similar process for shape and detail enhancement, except that their emphasis is on boosting and/or combining together detail layers from multiple sources, rather than on compressing the overall dynamic range.

Another example is image and video stylization and abstraction [DeCarlo and Santella 2002; Winnemöller et al. 2006]. Here the details are discarded, while the base layer is further processed to achieve a stylized look. Discarding details at different spatial scales makes it possible to retain more detail inside intended regions of interest, while achieving more abstraction in the background.

Computing the base layer is an image *coarsening* process. The coarsening must be done carefully in order to avoid artifacts that might arise once the base and the detail layers are manipulated separately and recombined. As discussed by several researchers [Schlick 1994; Tumblin and Turk 1999; Li et al. 2005] these artifacts are caused by distortions introduced when the signal is factored into its base and detail components. Figure 2 (after [Tumblin and Turk 1999]) demonstrates that *both blurring and sharpening* of edges in the coarsened image cause ringing in the detail layer, which may later manifest itself as halos and gradient reversals. Consequently, neither linear filtering nor hard segmentation is well-suited for computing base-detail decompositions, and researchers have turned to edge-preserving filters, which provide a compromise between these two extremes.

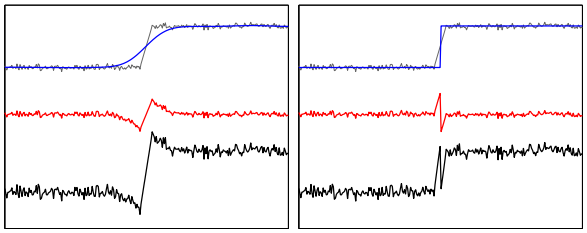


Figure 2: *Artifacts resulting from edge blurring (left) and edge sharpening (right). The original signal is shown in gray, and the coarsened signal in blue. Boosting the details (red) and recombining with the base results in halos and gradient reversals (black).*

The ideal edge-preserving filter must neither blur nor sharpen the edges that separate coarse scale image features, while smoothing the regions between such edges. Unfortunately, such an operator does not exist, because in general it is impossible to unambiguously determine which edges should be preserved. Furthermore, in order to produce multi-scale base-detail decompositions, the operator must allow increasingly larger image features to migrate from the base layer to the detail layer. In other words, it must allow increasingly larger regions to become increasingly smoother.

Several edge-preserving smoothing operators have been used in computational photography over the last decade. Tumblin and Turk [1999] pioneered their use in tone mapping, by introducing LCIS, a variant of anisotropic diffusion [Perona and Malik 1990]. Anisotropic diffusion is capable of drastic smoothing while preserving crisp edges between the smooth regions, and it has been used for multiscale image segmentation and edge detection. However, the original Perona and Malik scheme tends to oversharpen edges. Furthermore, it is a slowly converging non-linear iterative process, and the piecewise smooth images are obtained by stopping the diffusion process partway through, since it eventually converges to a constant image. These limitations of anisotropic diffusion have been addressed by more recent work (e.g., [Alvarez et al. 1992; Scherzer and Weickert 2000]). We refer the reader to [Aubert and Kornprobst 2006] for a more comprehensive exposition of state-of-the-art PDE-based methods, and a formal analysis of their ability to preserve edges.

Despite the recent improvements to anisotropic diffusion in the image processing literature, most recent applications in computer graphics and computational photography use the bilateral filter, popularized by Tomasi and Manduchi [1998]. This is a non-linear filter, where each pixel in the filtered result is a weighted mean of its neighbors, with the weights decreasing both with spatial distance and with difference in value. Formally, we have:

$$BLF(g)_p = \frac{1}{k_p}\sum_q G_{\sigma_s}(\|p-q\|)\, G_{\sigma_r}(\|g_p-g_q\|)\, g_q \quad (1)$$

$$k_p = \sum_q G_{\sigma_s}(\|p-q\|)\, G_{\sigma_r}(\|g_p-g_q\|) \quad (2)$$

where g is an image, and the subscripts p and q indicate spatial locations of pixels. The kernel functions G_{σ_s} and G_{σ_r} are typically Gaussians, where σ_s determines the spatial support, while σ_r controls the sensitivity to edges. Although a naïve implementation of the filter is costly, several fast approximations have been proposed [Durand and Dorsey 2002; Paris and Durand 2006; Weiss 2006; Chen et al. 2007].

While the BLF is quite effective at smoothing small changes in intensity while preserving strong edges, its ability to achieve progressive coarsening is rather limited. Below, we explain these limitations, using the simple synthetic example in Figure 3. The input

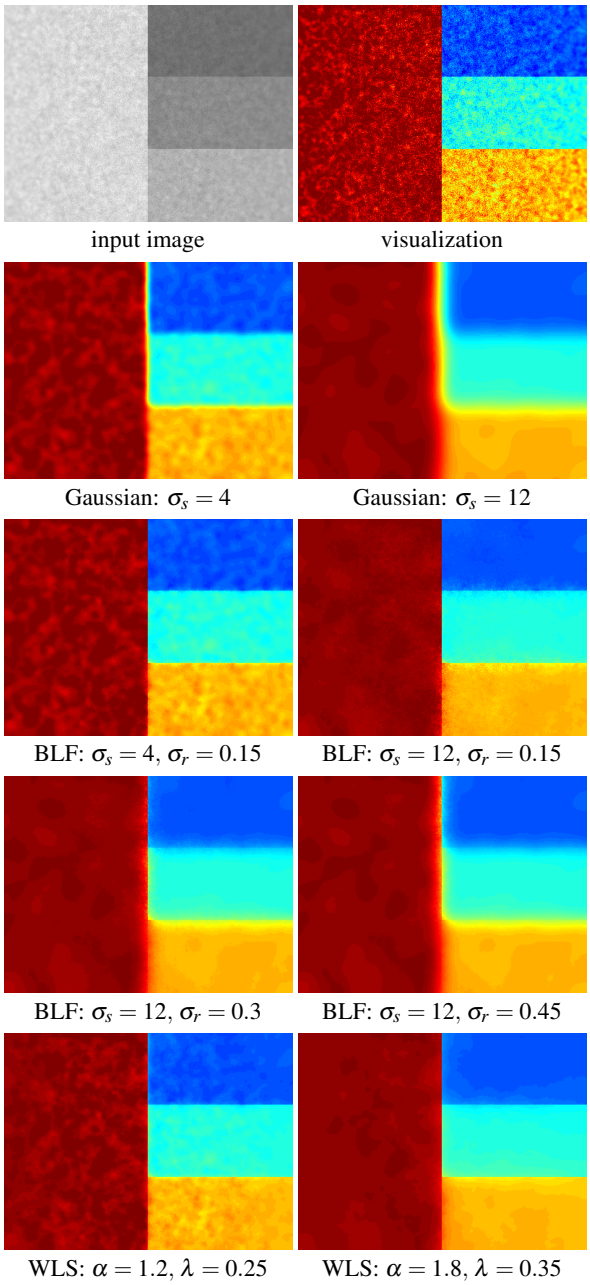


Figure 3: *Filtering a set of noisy step edges (constant regions) with a variety of coarsening filters.*

image (top left) is roughly piecewise constant, with several step edges of different magnitudes, and it contains noise at two different scales. For clarity, we visualize the image intensities using a color map (top right); we use the same color map throughout the figure.

The second row of Figure 3 shows the results of filtering the image with a linear Gaussian filter. Using a small spatial kernel (σ_s) removes the fine scale noise, while a larger kernel smoothes out the larger scale noise as well. As expected, the step edges are blurred accordingly. With edge-preserving smoothing, one hopes to preserve the edges, while achieving similar degrees of smoothing in the regions bounded by them. Indeed, applying the BLF with a small σ_s (3rd row, left) manages to smooth out most of the fine scale noise, while keeping the edges mostly sharp. Next, we attempt to smooth out the larger scale noise by increasing only the spatial support (3rd row, right), but with only partial success. Note how some of the fine scale noise is reintroduced into the filtered image, particularly near some of the edges. To understand this counter-intuitive behavior, consider a particular pixel p, whose unfiltered value is g_p. As σ_s is increased, more and more distant pixels q, whose value g_q is close to g_p are being averaged together, and as a result the filtered value does not stray too far from g_p. In the limit ($\sigma_s \to \infty$), the bilateral filter becomes a *range filter*. This behavior has been pointed out already by Tomasi and Manduchi [1998].

Thus, more aggressive smoothing cannot be achieved only by increasing σ_s, and the range support σ_r must be increased as well. However, increasing σ_r reduces the ability of the bilateral filter to preserve edges, and some of them become blurred, as evident from the fourth row of Figure 3. In the limit ($\sigma_r \to \infty$) the bilateral filter becomes a linear Gaussian filter.

In summary, this example demonstrates that the BLF trades off its edge-preservation abilities with its smoothing abilities. As the scale of the extracted details increases, the BLF tends to blur over more edges, which may produce halo artifacts. In contrast, by applying WLS smoothing to the same signal, it is possible to achieve both finer and coarser smoothing, while preserving the step edges, and without introducing obvious artifacts (Figure 3, bottom row). In Section 4 we demonstrate these differences using real images.

It is possible to achieve stronger smoothing with the BLF by applying it iteratively [Tomasi and Manduchi 1998; Winnemöller et al. 2006]. Fattal *et al.*[2007] did this while reducing σ_r at each iteration. However, this process produces results resembling those that might be obtained with mean-shift filtering [Comaniciu and Meer 2002]; it tends to oversharpen strong edges and causes small features bounded by strong edges to persist in the coarsened images.

A number of researchers attempted to address certain shortcomings of the BLF. For example, [Elad 2002; Choudhury and Tumblin 2003; Buades et al. 2006] extended the BLF to better handle piecewise linear functions. While the trilateral filter of Choudhury and Tumblin is somewhat better at progressive coarsening, we show in Section 4 that it tends to exhibit artifacts next to sharp features. Durand and Dorsey [2002] describe a variant designed specifically to avoid halos along thin, high contrast features, while Bae *et al.* [2006] detect and fix reversed gradients when manipulating the detail layer. However, the former approach does not eliminate wider halos, while the latter requires solving a Poisson equation as a postprocess.

Finally, gradient domain tone mapping methods [Fattal et al. 2002; Mantiuk et al. 2006] attempt to avoid halos by manipulating gradients. In particular, Mantiuk's multi-resolution gradient domain framework prevents halo artifacts by imposing constraints on contrast at multiple spatial frequencies, but the manipulations they demonstrate are less extreme than the ones in this work.

3 Edge-Preserving Smoothing via WLS

In this section, we first describe an alternative edge-preserving smoothing approach based on the WLS optimization framework, and then show how to construct multi-scale edge-preserving decompositions that capture details at a variety of scales.

Edge-preserving smoothing may be viewed as a compromise between two possibly contradictory goals. Given an input image g, we seek a new image u, which, on the one hand, is as close as possible to g, and, at the same time, is as smooth as possible everywhere, except across significant gradients in g. Formally, this may

be expressed as seeking the minimum of

$$\sum_p \left(\left(u_p - g_p\right)^2 + \lambda \left(a_{x,p}(g) \left(\frac{\partial u}{\partial x} \right)_p^2 + a_{y,p}(g) \left(\frac{\partial u}{\partial y} \right)_p^2 \right) \right), \quad (3)$$

where the subscript p denotes the spatial location of a pixel. The goal of the *data term* $(u_p - g_p)^2$ is to minimize the distance between u and g, while the second (*regularization*) term strives to achieve smoothness by minimizing the partial derivatives of u. The smoothness requirement is enforced in a spatially varying manner via the *smoothness weights* a_x and a_y, which depend on g. Finally, λ is responsible for the balance between the two terms; increasing the value of λ results in progressively smoother images u.

Using matrix notation we may rewrite eq. (following quadratic form:

$$(u-g)^{\mathrm{T}}(u-g) + \lambda \left(u^{\mathrm{T}} D_x^{\mathrm{T}} A_x D_x u + u^{\mathrm{T}} D_y^{\mathrm{T}} A_y D_y u \right). \quad (4)$$

Here A_x and A_y are diagonal matrices containing the smoothness weights $a_x(g)$ and $a_y(g)$, respectively, and the matrices D_x and D_y are discrete differentiation operators.

The vector u that minimizes eq. (4) is uniquely defined as the solution of the linear system

$$\left(I + \lambda L_g\right) u = g, \quad (5)$$

where $L_g = D_x^{\mathrm{T}} A_x D_x + D_y^{\mathrm{T}} A_y D_y$. Modulo the differences in notation, this is exactly the linear system used in [Lischinski et al. 2006], where it was primarily used to derive piecewise smooth adjustment maps from a sparse set of constraints.

In our implementation, D_x and D_y are forward difference operators, and hence D_x^{T} and D_y^{T} are backward difference operators, which means that L_g is a five-point spatially inhomogeneous Laplacian matrix. As for the smoothness weights, we define them in the same manner as in [Lischinski et al. 2006]:

$$a_{x,p}(g) = \left(\left| \frac{\partial \ell}{\partial x}(p) \right|^{\alpha} + \varepsilon \right)^{-1} \quad a_{y,p}(g) = \left(\left| \frac{\partial \ell}{\partial y}(p) \right|^{\alpha} + \varepsilon \right)^{-1}, \quad (6)$$

where ℓ is the log-luminance channel of the input image g, the exponent α (typically between 1.2 and 2.0) determines the sensitivity to the gradients of g, while ε is a small constant (typically 0.0001) that prevents division by zero in areas where g is constant.

To complete the exposition of the WLS-based operator, let us examine the relationship between the value of the parameter λ and the degree of smoothing. When using a linear spatially invariant smoothing filter (e.g., a Gaussian filter), doubling the spatial support of the kernel makes the filter roughly twice narrower in the frequency domain. We would like to understand how to achieve the same effect by changing the value of λ.

Equation (5) tells us that u is obtained from g by applying a nonlinear operator F_λ, which depends on g:

$$u = F_\lambda(g) = (I + \lambda L_g)^{-1} g \quad (7)$$

Since this operator is spatially variant, it is hard to analyze its frequency response. Therefore, as in [Fattal et al. 2007], we restrict our analysis to regions of the image that do not contain significant edges. Specifically, in regions where g is roughly constant, the smoothness weights a_x and a_y are roughly equal, i.e., $a_x \approx a_y \approx a$, and hence

$$F_\lambda(g) \approx (I + \lambda a L)^{-1} g \quad (8)$$

where $L = D_x^{\mathrm{T}} D_x + D_y^{\mathrm{T}} D_y$ is the ordinary (homogeneous) Laplacian matrix. The frequency response of F_λ is then given by [Oppenheim and Schafer 1989]

$$\mathscr{F}_\lambda(\omega) = 1/(1 + a\lambda\omega^2). \quad (9)$$

Thus, scaling by a factor of c in the frequency domain is equivalent to multiplying λ by a factor of c^2:

$$\mathscr{F}_\lambda(c\omega) = 1/(1 + ac^2\lambda\omega^2) = \mathscr{F}_{c^2\lambda}(\omega). \quad (10)$$

The same conclusion may be reached for image regions of roughly constant slope, where each of a_x and a_y is constant (but not necessarily equal to each other).

It should be noted that since the smoothness coefficients (6) separate between gradients in the x and y directions, the resulting operator is not rotationally invariant, with a slight tendency to preserve axis aligned edges more than diagonal ones. However, this did not result in any visible artifacts in our experiments; one must also keep in mind that the discrete representation of images on a regular grid is rotationally variant in itself.

3.1 Multi-scale edge-preserving decompositions

Using the edge-preserving operator described above, it is easy to construct a multi-scale edge-preserving decomposition, fashioned after the well known Laplacian pyramid [Burt and Adelson 1983]. The decomposition consists of a coarse, piecewise smooth, version of the image, along with a sequence of difference images, capturing detail at progressively finer scales.

More specifically, let g denote the input image for which we would like to construct a $(k+1)$-level decomposition. Let $u^1, \ldots, u^k$ denote progressively coarser versions of g. The coarsest of these versions, u^k will serve as the *base layer* b, with the k *detail layers* defined as

$$d^i = u^{i-1} - u^i, \quad \text{where } i = 1, \ldots, k \text{ and } u^0 = g. \quad (11)$$

The original image g is easily recovered from this decomposition by simply adding up the base and the detail layers:

$$g = b + \sum_{i=1}^{k} d^i. \quad (12)$$

Note that we do not perform any downsampling of the smoothed images u^i, since they are obtained via edge-preserving smoothing and are not band-limited in the usual sense. Thus, our multi-scale decomposition is an over-complete description of the input image.

We have experimented with two methods of computing the progressive coarsening sequence $u^1, \ldots, u^k$. The first one is to solve the linear system (5) k times, each time increasing the value of the parameter λ. In other words,

$$u^{i+1} = F_{c^i\lambda}(g), \quad (13)$$

for some initial value of λ and some factor c. A coarsening sequence generated in this manner and the corresponding detail layers is shown in the left column of Figure 4. We found the resulting decompositions to be well suited for HDR compression and multi-scale detail enhancement (using $\alpha = 1.2$ to 1.4).

The second method is to apply the operator iteratively:

$$u^{i+1} = F_{c^i\lambda}(u^i). \quad (14)$$

In this method the image is repeatedly smoothed, similarly to mean-shift filtering [Comaniciu and Meer 2002] and to the multi-scale

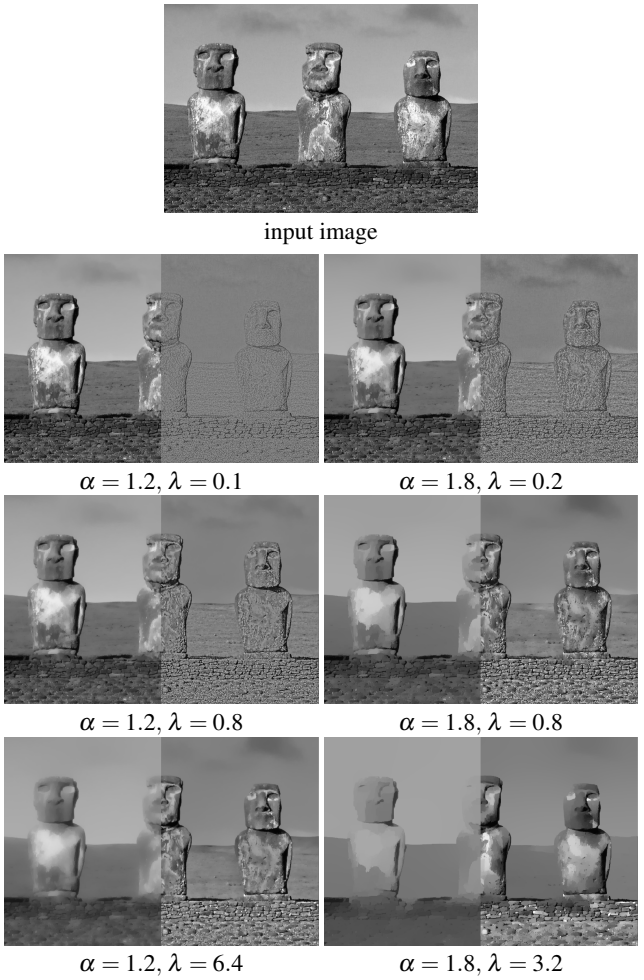


Figure 4: *WLS-based multi-scale decompositions. Left column: three levels computed using eq. (13). Right column: levels computed using eq. (14). The left half of each image shows the coarsening, while the right half visualizes the corresponding detail layer. The spatial scale of the details increases from one level to the next.*

bilateral transform [Fattal et al. 2007], and the resulting coarsened images tend more strongly towards piecewise constant regions separated by strong edges (right column of Figure 4). We still increase λ by a factor of c at each iteration, as this results in a more significant increase in smoothness in each iteration. We found the iterative scheme to be better suited for applications that discard or attenuate some of the details, such as image abstraction (with $\alpha = 1.8$ or 2.0).

4 Comparison

At this point, it is interesting to compare our WLS-based multi-scale decompositions to several previous schemes. Recall that our goal is to be able to extract and manipulate (e.g., enhance or suppress) details at a variety of spatial scales, so we must keep this context in mind, when analyzing and comparing the alternatives. One should also keep in mind that, in real images, it is impossible to consistently classify each edge into one of two classes (those that should be preserved and those that should not). Thus, neither previous schemes, nor WLS are able to produce a "perfect" edge-preserving smoothing. Still, it is instructive to compare the manner in which each method resolves this ambiguity.

As was shown in Section 2, in order to achieve progressive coarsening with the BLF it is necessary to increase both σ_s and σ_r. Indeed, this strategy was used by Chen *et al.* [2007] to compute their *bilateral pyramid* for progressive video abstraction. However, while this strategy produces smoother regions, it may also cause significant blurring of strong edges. Figure 5 demonstrates this using a 1D section of an image containing a large feature in the left half of the image, and a smaller (narrower) feature in the right half. The blue plot curves show the result of progressively coarsening the input signal. With the BLF (left plot), the edges around both the large and the small features gradually become smoother, which generates ringing in the corresponding detail signals (shown below in red). In contrast, with WLS, the same edges are gradually eroded, but without significantly distorting the shape of the signal. Furthermore, the edges surrounding the smaller feature are eroded *faster* than those around larger features. This causes small features to *migrate* to the detail layers, without noticeable ringing, even when the surrounding edges are strong.

Another progressive coarsening strategy [Fattal et al. 2007], is to obtain each image in the sequence by applying BLF to the previous image. The range parameter σ_r is reduced at each iteration to ensure that edges which were preserved in the previous level would remain preserved in the next one. This strategy is demonstrated in the rightmost plot of Figure 5. While it avoids smoothing strong edges, some small regions (surrounded by edges that were preserved in the finer levels) might never migrate to any of the detail layers. Consequently, detail layer manipulation is unable to remove, suppress, or emphasize such detail. Another problem is the oversharpening of edges, which might lead to thin gradient reversal artifacts (note the thin spikes in Figure 5 and the pixel-wide halos in Figure 10).

Figure 6 demonstrates the behavior of both strategies described above using a real image. In the bilateral pyramid (left column) some strong edges "collapse" in the coarsening process, introducing visible ringing in the detail layer. Compare this with the base-detail pairs in the bottom row of Figure 4, where strong edges are more consistently preserved. As pointed out earlier, ringing in the detail layer may cause halo artifacts in applications that manipulate the layers and recombine them together.

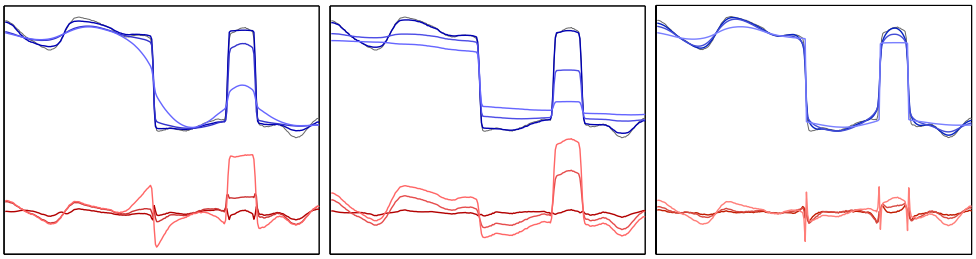
For example, the tone mapping algorithm of Durand and Dorsey [2002] sometimes produces subtle, yet visible, halos. Applying the same algorithm using our WLS-based decomposition yields a result without any visible halos, while preserving the fine local contrasts just as well (Figure 7). The halo artifacts stand out much more if the detail layer undergoes a substantial boost. An example is shown in Figure 8.

It is also instructive to examine the decompositions produced by the LCIS method [Tumblin and Turk 1999] and the more recent trilateral filter [Choudhury and Tumblin 2003], which addresses some of the shortcomings of both LCIS and the BLF. These decompositions are also shown in Figure 6. In both of these approaches strong edges collapse as the coarsening progresses, while around the remaining strong edges there is significant ringing in the detail layer.

5 Connections with Other Operators

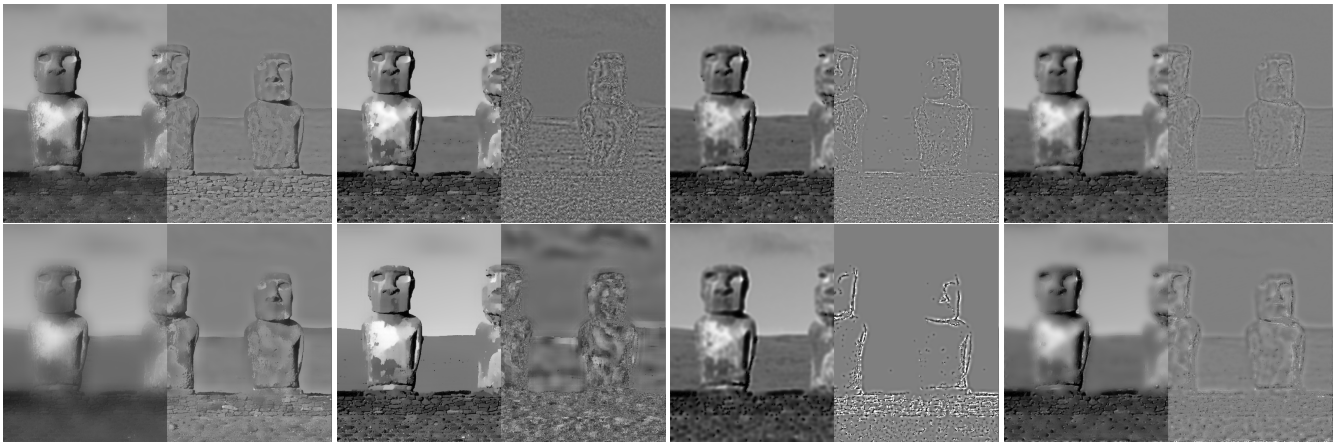
A number of researchers have already analyzed the mathematical relationships between various edge-preserving operators, e.g., [Black et al. 1998; Barash 2002; Elad 2002; Durand and Dorsey 2002; Mrázek et al. 2006]. In this section, we also discuss some of these relationships, but focus specifically on the WLS-based operator. Our discussion is a pragmatic one, with the goal of comparing and evaluating the alternatives in the context of the typical applications of base-detail decompositions in computational photography.

As was pointed out earlier, our edge-preserving smoothing may be expressed as applying the operator $F_\lambda = (I + \lambda L_g)^{-1}$ to the input



[Chen et al. 2007] WLS, eq. (13) [Fattal et al. 2007]

Figure 5: *Progressively coarsening a signal using different edge-preserving schemes. The coarsened versions are shown superimposed on the signal (using different shades of blue: lighter is coarser). The corresponding detail signals are plotted in shades of red below.*



[Chen et al. 2007] [Fattal et al. 2007] [Tumblin and Turk 1999] [Choudhury and Tumblin 2003]

Figure 6: *Coarsened images and their corresponding detail layers for several different edge-preserving filtering schemes. Coarsening progresses from top to bottom. The bilateral filter, LCIS, and the trilateral filter exhibit ringing in the detail layer (easiest to see in the bottom row). [Fattal et al. 2007] retains many small features even in the coarsest image, which never make their way into the detail layer.*

image vector g (eq. 7). Each row of F_λ may be thought of as a kernel (with possibly very large support) that determines the value of the corresponding pixel in the image as a weighted combination of other pixels in g. Similarly to the BLF, each kernel's weights are affected by the proximity to edges in g, and thus we can think of our smoothing process as spatially-variant filtering.

However, applying the spatially-variant filter directly is not a viable option. Explicitly forming F_λ involves matrix inversion, which costs $O(N^3)$ operations, where N is the number of pixels. Even if we set aside this cost, just applying F_λ might require $O(N^2)$ operations. Fortunately, since F_λ is the inverse of a sparse matrix, computing the filtered image amounts to solving the sparse linear system (5), requiring only $O(N)$ operations with an appropriate preconditioner or a multi-grid solver. In return for this efficient solution, we give up the ability to explicitly control the precise shape of the spatially varying kernels in F_λ, and instead control them implicitly via the regularization term in eq. (3).

Solving eq. (5) using Jacobi iterations [Saad 2003] can be thought of as anisotropic diffusion, where the stopping function between neighboring pixels is determined once and for all at the beginning of the process. While this may seem like a minor difference, it has two important implications. One implication is that the iterative process converges to a unique non-trivial solution, rather than to a constant image; the other implication is that we can obtain this solution efficiently, by solving a sparse linear system.

There is also a particular variant of anisotropic diffusion, known as *biased anisotropic diffusion* [Nordstrom 1989], where a non-trivial steady-state solution is ensured by "biasing" the solution towards the input image by introducing a term $g-u$ into the diffusion equation:

$$\frac{\partial u}{\partial t} = \nabla \cdot (a(\|\nabla u\|) \cdot \nabla u) + (g-u), \tag{15}$$

where a is some flux-stopping function. Again, if one uses this scheme with $a(\|\nabla g\|)$, rather than $a(\|\nabla u\|)$, eq. (15) becomes linear and its steady-state solution is exactly the result obtained by our operator.

Similarly to anisotropic diffusion, robust estimation techniques [Zervakis 1990; Black et al. 1998] employ an "M-function" which depends on the unknown solution u, and plays the same role as the regularization term in WLS. Again, the dependence on u yields a non-linear process.

Finally, note that applying our WLS-based operator iteratively, with the default $\alpha = 1.2$ exponent in the smoothing weights of eq. (6), is equivalent to optimizing eq. (3) with the 0.8-norm in the smoothness term, using iteratively reweighted least squares [Scales and Gersztenkorn 1988]. This norm has recently been used as a sparse prior for derivatives of natural images (e.g., [Levin et al. 2007]).

6 Applications and Results

We have implemented a number of simple tools that use our multi-scale edge-preserving decompositions for photographic tone manipulation, HDR tone mapping, detail enhancement, and image abstraction. Below, we briefly describe these tools and show some

Figure 7: *Top: a tone-mapped image, taken directly from [Durand and Dorsey 2002], with some halos visible around the picture frames and the light fixture. Bottom: a halo-free result with a similar amount of local contrast may be produced using the same tone mapping algorithm, simply by replacing BLF with WLS-based smoothing ($\alpha = 1.2, \lambda = 2$).*

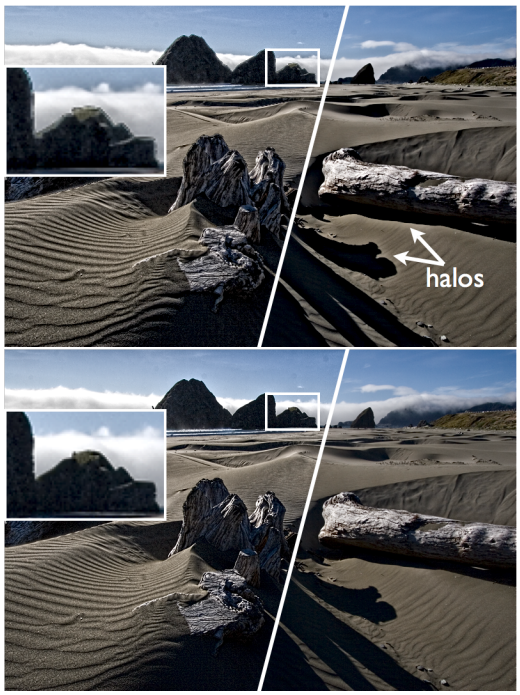


Figure 8: *Boosting BLF-based detail layers (top) results in artifacts along the high-contrast edges, which are absent when the decomposition is WLS-based (bottom). In the right part of each image medium scale details have been boosted, also resulting in halos when done using BLF. (Input image courtesy of Norman Koren.)*

results. Note the purpose of these tools is to demonstrate in the simplest possible way the robustness and versatility of our decompositions. Any of these tools could have been made much more sophisticated, but this is outside the scope of this paper.

To solve the linear system (5) we use the multiresolution preconditioned conjugate gradient solver described by Lischinski *et al.* [2006]. The performance of this solver is linear in the number of pixels, requiring roughly 3.5 seconds per megapixel on a 2.2 GHz Intel Core 2 Duo. Although we currently do not have a GPU implementation of the solver, such an implementation should be possible; Buatois *et al.* [2007] describe a GPU implementation of a more general preconditioned conjugate gradient sparse matrix solver, and report speedups by a factor of 3.2 over a highly optimized CPU implementation. It is hoped that a GPU implementation tuned to our specific linear system should achieve even more significant speedups.

6.1 Multi-scale tone manipulation

We have implemented a simple interactive tool for manipulating the tone and contrast of details at different scales. Given an image, we first construct a three-level decomposition (coarse base level b and two detail levels d^1, d^2) of the CIELAB lightness channel. This is done using the first (non-iterative) construction given by eq. (13). The user is then presented with a set of sliders for controlling the exposure η of the base layer, as well as the *boosting factors*, δ_0 for the base, and δ_1, δ_2 for the medium and fine detail layers. The result of the manipulation $\hat{g}$ at each pixel p is then given by

$$\hat{g}_p = \mu + S(\delta_0, \eta b_p - \mu) + S(\delta_1, d_p^1) + S(\delta_2, d_p^2), \qquad (16)$$

where μ is the mean of the lightness range, and S is a sigmoid curve, $S(a,x) = 1/(1+\exp(-ax))$ (appropriately shifted and normalized). The goal of this sigmoid is to avoid the hard clipping that would otherwise occur when the detail layers are significantly boosted. The term $S(\delta_0, \eta b_p - \mu)$ controls the exposure and contrast of the base layer, while the remaining terms control the boosting of the medium and fine scale details. Note that once the decomposition has been computed, eq. (16) is evaluated in real time.

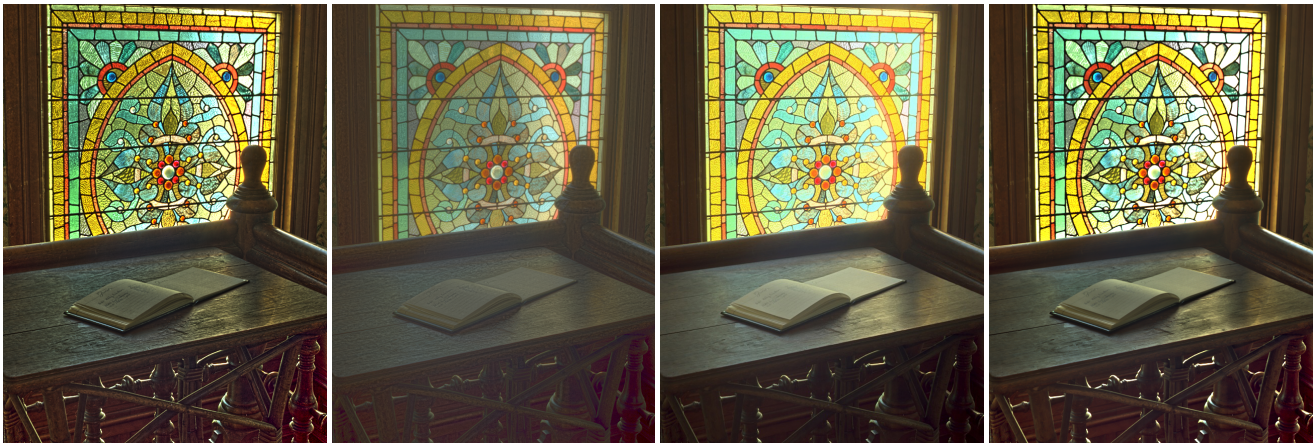
We found that this simple tool is already very effective for controlling the amount of local contrast at the different scales. The effective manipulation range is very wide: it typically takes a rather extreme manipulation to cause artifacts to appear. Example results are shown in Figures 1, 8, and 9. The decomposition for all these results was constructed with the parameters $\alpha = 1.2, \lambda = 0.1$ for the fine scale filtering and $\alpha = 1.4, \lambda = 0.4$ for the medium scale filtering. We found three pyramid levels to be sufficient for the images we experimented with. Using more levels would enable finer control, but would also require the user to manipulate more sliders.

6.2 Detail exaggeration

Our decomposition may be also used in place of the BLF-based multi-scale decomposition used by Fattal *et al.* [2007] for shape and detail enhancement from multi-light image collections. Our experiments show that our decomposition allows one to enhance and exaggerate details even further than with their approach, before objectionable artifacts appear. Figure 10 (top) shows one of the results from Fattal *et al.* next to a similar result produced with our approach. A close examination reveals that along many of the edges, their result exhibits one pixel wide gradient reversals, probably due to oversharpening of these edges in the decomposition. In contrast, the edges in our result appear much cleaner. Another comparison is shown at the bottom of Figure 10. Here, we demonstrate that we can generate highly exaggerated detail even from a single input image (rather than three multi-light images used by Fattal *et al.*).

input image | coarse scale boosting | medium scale boosting | fine scale boosting | combined result

Figure 9: *Multiscale tone manipulation with our tool. The boosting of the individual scales is intentionally exaggerated.*



WLS: increased detail visibility | LCIS | [Reinhard et al. 2002] | WLS: a photographic look

Figure 11: *HDR tone mapping with our tool. Saturation and exposure were manually adjusted in the WLS results in order to match the overall appearance of the other two images. (HDR image © Industrial Light & Magic. All rights reserved.)*

6.3 HDR tone mapping

Our decompositions are easily harnessed to perform detail preserving compression of HDR images. For example, we can simply replace the BLF in the tone mapping algorithm of Durand and Dorsey, with WLS-based smoothing and avoid the mild halo artifacts that are sometimes visible in their results (Figure 8).

Another option we experimented with is to use the tone mapping algorithm proposed by Tumblin and Turk [1999], but replace their LCIS-based multi-scale decomposition with our WLS-based decomposition. Specifically, we compute a 4-level decomposition (one coarse base layer and three detail layers) of the log-luminance channel, multiply each level by some scaling factor, and reconstruct a new log-luminance channel.

Figure 11 shows two tonemapped results that were produced in this manner. In the leftmost image, our goal was to achieve a rather flat image with exaggerated local contrasts (similar, but more extreme than the typical result produced with LCIS on this image). This was achieved by strongly compressing the base, and boosting the fine scale detail layer. In the rightmost image, we pursued a more photographic look but with more depth in the highlights than what is possible with Reinhard's operator [2002]. We achieved this result by using less compression of the base and moderate boosting of the coarser detail layers.

6.4 Progressive image abstraction

Our WLS-based operator can be used in many other applications that utilize an edge-preserving filter. For example, Winnemöler *et al.* [Winnemöller et al. 2006] and Chen *et al.* [2007] use the BLF for image and video abstraction. Using our multi-scale decomposition (the iterative version of eq. (14)) produces the results shown in Figure 12. In this application, the detail layers are attenuated, rather than boosted, to achieve a stylized abstract look. Using progressively coarser decomposition levels increases the degree of abstraction in the resulting image. These abstractions can also be combined together in a spatially varying manner to provide more detail in areas of interest. We do this with an interactive painting interface; a more automated mechanism is described in [Chen et al. 2007]. The images are overlaid with edges extracted from the appropriate detail layers.

7 Conclusions

Multi-scale contrast manipulation is a valuable digital darkroom technique. Currently it is possible to sharpen images (which may be viewed as increasing the local contrast of the finest scale details), as well as to adjust the global contrast. However, adjusting the local contrast for scales in between these two extremes is typically done with unsharp masking, which is prone to halos. The multi-scale edge-preserving decompositions we have introduced in this paper are intended to fill this gap.

Our decompositions are based on a weighted least squares formulation, which, as we have demonstrated, does not suffer from some of the drawbacks of bilateral filtering and other previous approaches. In particular, WLS allows small features to gracefully fade in magnitude, so that they do not persist into coarse levels, but without introducing significant blurring, which can result in halos when differences are magnified. We have also shown how the WLS formulation is related to other edge-preserving schemes, such as anisotropic diffusion and iterated BLF. Our results on a variety of applications, including tone mapping, contrast manipulation, and image abstraction, show that our approach is robust and versatile.

In future work we would like to investigate more sophisticated schemes for determining the smoothness coefficients for the WLS formulation in order to further improve the ability to preserve edges

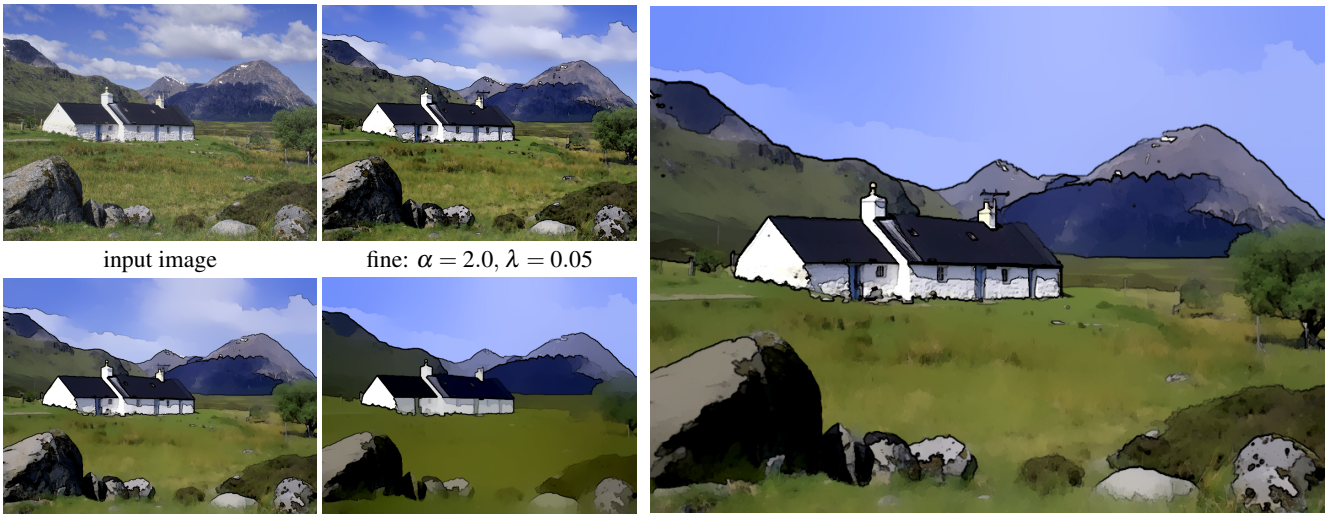


Figure 12: *Progressive image abstractions computed using our multi-scale decomposition.*

Figure 10: *Multi-scale detail enhancement of Fattal et al. [2007] (left) compared to results produced with our decomposition (right). We are able to achieve more aggressive detail enhancement and exaggeration, while avoiding artifacts.*

and extract details. Ultimately, we would like to use our decompositions as a basis for a more sophisticated and more automated tool — a single click (or single slider) solution for image enhancement.

Another important issue that must be tackled is better handling of color. Our multi-scale tone manipulation tool currently operates on the CIELAB lightness channel, and we have observed that strong manipulations result in significant changes in the perceived color. While manually adjusting the saturation alleviates the problem, a more principled solution is needed.

Acknowledgments: Much of this work was carried out while the first and third authors were visiting Microsoft Research. The authors would like to thank Norman Koren for letting us use his beautiful images. This work was supported in part by the Israel Science Foundation founded by the Israel Academy of Sciences and Humanities and by the German-Israeli Foundation for Scientific Research and Development (GIF).

References

ALVAREZ, L., LIONS, P.-L., AND MOREL, J.-M. 1992. Image selective smoothing and edge detection by nonlinear diffusion. ii. *SIAM Journal on Numerical Analysis 29*, 3 (June), 845–866.

AUBERT, G., AND KORNPROBST, P. 2006. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, vol. 147 of *Applied Mathematical Sciences*. Springer.

BAE, S., PARIS, S., AND DURAND, F. 2006. Two-scale tone management for photographic look. *ACM Trans. Graph. 25*, 3 (July), 654–662.

BARASH, D. 2002. A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 6, 844–847.

BLACK, M. J., SAPIRO, G., MARIMONT, D. H., AND HEEGER, D. 1998. Robust anisotropic diffusion. *IEEE Trans. Image Proc. 7*, 3 (Mar.), 421–432.

BUADES, A., COLL, B., AND MOREL, J. M. 2006. The staircasing effect in neighborhood filters and its solution. *IEEE Transactions on Image Processing 15*, 6, 1499–1505.

BUATOIS, L., CAUMON, G., AND LEVY, B. 2007. Concurrent number cruncher: An efficient sparse linear solver on the GPU. In *High Performance Computation Conference (HPCC)*, Springer.

BURT, P., AND ADELSON, E. H. 1983. The Laplacian pyramid as a compact image code. *IEEE Trans. Comm. 31*, 532–540.

CHEN, J., PARIS, S., AND DURAND, F. 2007. Real-time edge-aware image processing with the bilateral grid. *ACM Trans. Graph. 26*, 3 (July), Article 103.

CHOUDHURY, P., AND TUMBLIN, J. 2003. The trilateral filter for high contrast images and meshes. In *Proc. EGSR 2003*, Eurographics, 186–196.

COMANICIU, D., AND MEER, P. 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 5, 603–619.

DECARLO, D., AND SANTELLA, A. 2002. Stylization and abstraction of photographs. *ACM Trans. Graph. 21*, 3 (July), 769–776.

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

EISEMANN, E., AND DURAND, F. 2004. Flash photography enhancement via intrinsic relighting. *ACM Trans. Graph. 23*, 3 (August), 673–678.

ELAD, M. 2002. On the bilateral filter and ways to improve it. *IEEE Trans. Image Proc. 11*, 10, 1141–1151.

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

FATTAL, R., AGRAWALA, M., AND RUSINKIEWICZ, S. 2007. Multiscale shape and detail enhancement from multi-light image collections. *ACM Trans. Graph. 26*, 3 (July), Article 51.

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 Trans. Image Proc. 6*, 7 (July), 965–976.

KHAN, E. A., REINHARD, E., FLEMING, R. W., AND BÜLTHOFF, H. H. 2006. Image-based material editing. *ACM Trans. Graph. 25*, 3 (July), 654–663.

LAGENDIJK, R. L., BIEMOND, J., AND BOEKEE, D. E. 1988. Regularized iterative image restoration with ringing reduction. *IEEE Trans. Acoustics, Speech, and Signal Proc., Speech, Signal Proc. 36*, 12 (December), 1874–1888.

LEVIN, A., LISCHINSKI, D., AND WEISS, Y. 2004. Colorization using optimization. *ACM Trans. Graph. 23*, 3 (August), 689–694.

LEVIN, A., FERGUS, R., DURAND, F., AND FREEMAN, W. T. 2007. Image and depth from a conventional camera with a coded aperture. *ACM Trans. Graph. 26*, 3 (July), Article 70.

LI, Y., SHARAN, L., AND ADELSON, E. H. 2005. Compressing and companding high dynamic range images with subband architectures. *ACM Trans. Graph. 24*, 3 (July), 836–844.

LISCHINSKI, D., FARBMAN, Z., UYTTENDAELE, M., AND SZELISKI, R. 2006. Interactive local adjustment of tonal values. *ACM Trans. Graph. 25*, 3 (July), 646–653.

MANTIUK, R., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2006. A perceptual framework for contrast processing of high dynamic range images. *ACM Trans. Appl. Percept. 3*, 3, 286–308.

MRÁZEK, P., WEICKERT, J., AND BRUHN, A. 2006. On robust estimation and smoothing with spatial and tonal kernels. In *Geometric Properties from Incomplete Data*, R. Klette, R. Kozera, L. Noakes, and J. Weickert, Eds. Springer, Dordrecht, 335–352.

NORDSTROM, K. N. 1989. Biased anisotropic diffusion — a unified regularization and diffusion approach to edge detection. Tech. Rep. UCB/CSD-89-514, EECS Department, University of California, Berkeley, May.

OH, B. M., CHEN, M., DORSEY, J., AND DURAND, F. 2001. Image-based modeling and photo editing. In *Proc. ACM SIGGRAPH 2001*, ACM, E. Fiume, Ed., 433–442.

OPPENHEIM, A. V., AND SCHAFER, R. W. 1989. *Discrete-Time Signal Processing*. Prentice Hall.

PARIS, S., AND DURAND, F. 2006. A fast approximation of the bilateral filter using a signal processing approach. In *Proc. ECCV '06*, IV: 568–580.

PARIS, S. 2007. A gentle introduction to bilateral filtering and its applications. In *ACM SIGGRAPH 2007 courses, Course 13*.

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 *Proc. ACM SIGGRAPH 98*, M. Cohen, Ed., 287–298.

PERONA, P., AND MALIK, J. 1990. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Machine Intell. 12*, 7 (July), 629–639.

PETSCHNIGG, G., SZELISKI, R., AGRAWALA, M., COHEN, M., HOPPE, H., AND TOYAMA, K. 2004. Digital photography with flash and no-flash image pairs. *ACM Trans. Graph. 23*, 3 (August), 664–672.

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

SAAD, Y. 2003. *Iterative Methods for Sparse Linear Systems*, second ed. SIAM.

SCALES, J. A., AND GERSZTENKORN, A. 1988. Robust methods in inverse theory. *Inverse Problems 4*, 1071–1091.

SCHERZER, O., AND WEICKERT, J. 2000. Relations between regularization and diffusion filtering. *Journal of Mathematical Imaging and Vision 12*, 1 (February), 43–63.

SCHLICK, C. 1994. Quantization techniques for visualization of high dynamic range pictures. In *Photorealistic Rendering Techniques*, Springer-Verlag, P. Shirley, G. Sakas, and S. Müller, Eds., 7–20.

TOMASI, C., AND MANDUCHI, R. 1998. Bilateral filtering for gray and color images. In *Proc. ICCV '98*, IEEE Computer Society, 839–846.

TUMBLIN, J., AND TURK, G. 1999. LCIS: A boundary hierarchy for detail-preserving contrast reduction. In *Proc. ACM SIGGRAPH 99*, A. Rockwood, Ed., ACM, 83–90.

WEISS, B. 2006. Fast median and bilateral filtering. *ACM Trans. Graph. 25*, 3 (July), 519–526.

WINNEMÖLLER, H., OLSEN, S. C., AND GOOCH, B. 2006. Real-time video abstraction. *ACM Trans. Graph. 25*, 3 (July), 1221–1226.

ZERVAKIS, M. E. 1990. *Nonlinear image restoration techniques*. PhD thesis, Univ. Toronto, Toronto, ON, Canada.