Seeing Mt. Rainier: Lucky Imaging for Multi-Image Denoising, Sharpening, and Haze Removal

Neel Joshi and Michael F. Cohen
Microsoft Research
[neel,mcohen]@microsoft.com

Abstract

Photographing distant objects is challenging for a number of reasons. Even on a clear day, atmospheric haze often represents the majority of light received by a camera. Unfortunately, dehazing alone cannot create a clean image. The combination of shot noise and quantization noise is exacerbated when the contrast is expanded after haze removal. Dust on the sensor that may be unnoticeable in the original images creates serious artifacts. Multiple images can be averaged to overcome the noise, but the combination of long lenses and small camera motion as well as time varying atmospheric refraction results in large global and local shifts of the images on the sensor.

An iconic example of a distant object is Mount Rainier, when viewed from Seattle, which is 90 kilometers away. This paper demonstrates a methodology to pull out a clean image of Mount Rainier from a series of images. Rigid and non-rigid alignment steps brings individual pixels into alignment. A novel local weighted averaging method based on ideas from "lucky imaging" minimizes blur, resampling and alignment errors, as well as effects of sensor dust, to maintain the sharpness of the original pixel grid. Finally, dehazing and contrast expansion results in a sharp clean image.

1. Introduction

Distant objects present difficulties to photograph well. Seeing detail obviously requires lenses with a very long focal length, thus even small motions of the camera during exposure cause significant blur. But the most vexing problem is atmospheric haze which often leads to the majority of photons arriving from scattering in the intervening media rather than from the object itself. Even if the haze is fully removed, there are only a few bits of signal remaining, thus quantization noise becomes a significant problem. Other noise characteristics of the sensor are also increased in the contrast expansion following haze removal. Variations in the density of air also cause refraction thus photons cannot be counted on to travel in straight lines. Finally, small dust particles on the sensor that cause invisible artifacts on the original images can become prominent after haze removal.

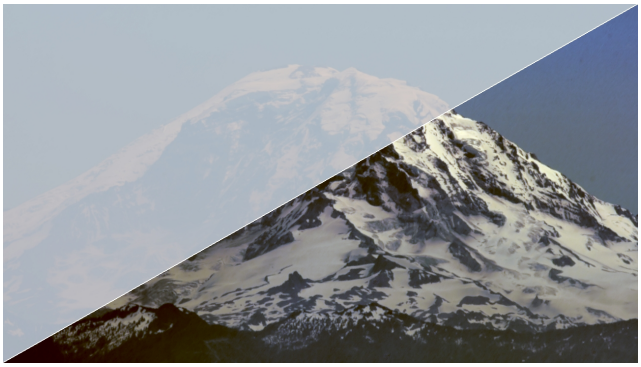


Figure 1. Multi-Image Dehazing of Mount Rainier: Given multiple input images, a sequence of rigid and non-rigid alignment and per-pixel weighted averaging, minimizes blur, resampling, and alignment errors. Dehazing and contrast expansion then results in a sharp clean image.

One such distant subject often photographed is Mount Rainier when viewed from Seattle, approximately 90 kilometers distant. For those who live in or visit Seattle, seeing *the mountain* on a clear day is an exhilarating experience. One can just make out the glaciers which pour down from its 14,411 foot peak rising from the sea. Unfortunately, in most amateur photographs the mountain seems to simply disappear. Even with a long lens and tripod on a clear day, the haze precludes creating a clean image of the mountain.

This paper demonstrates a methodology to create a clean shot of a distant scene from a temporal series of images. Care is taken to align the images due to global camera motion as well as considerable local time varying atmospheric refraction. Noise reduction is achieved through a novel weighted image averaging that avoids sacrificing sharpness. Our main technical contribution is in the weight determination. Significant loss of sharpness can occur due to the interaction of the pixel grid with strong edges in the scene as well as resampling due to sub-pixel alignment. We overcome this loss of sharpness through a novel weighted averaging scheme by extending ideas related to *lucky imaging* developed in the astronomy literature.

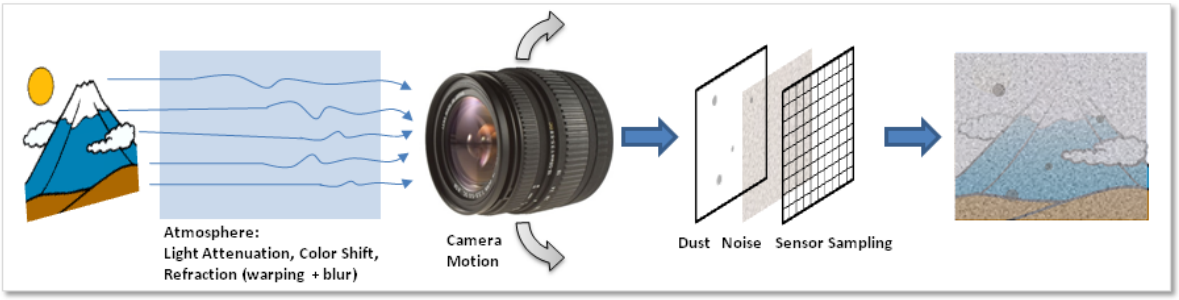


Figure 2. Imaging Mount Rainier: Several processes occur that introduce errors in the captured images. The atmosphere absorbs, scatters, refracts, and blurs light rays, while the camera adds artifacts due to motion, defocus blur, dust, noise, and discrete sensor sampling. Our method compensates for these multiple sources of error.

2. Related Work

There are three bodies of previous work that have the most influence on our current problem. Those are the literatures on denoising, image alignment and optical flow, and dehazing. We'll discuss the most relevant work.

Denoising: Image denoising methods have been reported in a very wide and deep body of literature [11, 18, 12, 13]. Most methods address the problem of denoising a single image. In general, for each pixel, a weighted averaging is performed over a local neighborhood. The weights can be as simple as a radially symmetric Gaussian (simple smoothing), may be determined by their similarity to the pixel being smoothed as in Bilateral Filters [18], or are based on higher order local statistics [16]. If one has multiple exact copies of an image, with each pixel corrupted independently by Gaussian noise, the temporal *stack* of corresponding pixels from each image can simply be averaged to remove the noise. Video denoising operates in a similar manner. Typically, an alignment phase is first performed to align the spatial neighborhoods in each frame. Then a weight is determined for pixels in the aligned spatiotemporal neighborhood. The weights may be based on the confidence in the alignment [3], temporal similarity, not unlike spatial bilateral filtering, to avoid averaging over moving objects for example [1], and/or other local statistics. In our case, we perform a weighted averaging of the pixel stacks, where the weights are determined from the local (spatial and temporal) statistics as well as a model to avoid spatial resampling of pixel values due to sub-pixel alignment. Since we have a deep stack to choose from, we can highly weight only a small percentage of the pixels and still achieve a good denoising. We extend ideas from *lucky imaging* [9, 6] in the astronomy domain for this purpose.

Alignment and Flow: Our task involves both performing a rigid alignment of images caused by small rotations of the camera as well as local alignment of pixels due to time-varying air turbulence. Szeliski [17] gives a nice tutorial of alignment and stitching methodologies. Similarly, there is a very rich literature on optical flow [5] for tracking pixels that move small amounts from frame to frame. Our case is relatively simple compared to finding general flow since the motion is spatially smooth, with no occlusions, and small enough to use a simple patch based SSD search after the global alignment.

Dehazing: There has been considerable work on removing haze from photographs. Haze removal is challenging because the haze is dependent on the scene depth which is, in general, unknown. Many methods use multiple images, such as a pair with and without a polarizing filter [14] or taken under different weather conditions [10]. The differences between the images are then used to estimate depth and the airlight color for dehazing. In some cases, depth can be derived from external sources by geo-registering the image to known 3D models[8]. Recently, single image haze removal [4, 7] has made progress by using a strong prior. Fattal [4] assumes the transmission and surface shading are locally uncorrelated to derive the effects of haze. He *et al*. [7] propose an interesting dark channel prior. They observe that for outdoor scenes, in any local region of a haze free image, there is at least one channel of one pixel that is dark. The presence and quantity of haze is therefore derived from the darkest pixel channel in some local region. We will use a variation of this work in our processing.

None of the above methods address the issue of noise when dehazing very distant objects obscured by a lot of haze. Most show results where they visual quality of distant regions is improved by "adding back a bit of haze".

3. Imaging *The Mountain*

To create a clean image of Mount Rainier we will work from a temporal series of images. For each of these images, I_t, we observe at each pixel, p, the following:

$$I_t(p) = D(p)[B(p + \Delta_t(p)) \otimes [J(p + \Delta_t(p))\alpha(p + \Delta_t(p)) + A(1 - \alpha(p + \Delta_t(p))]] + N_t(p) \quad (1)$$

where $J(p)$ will represent a measure (after tone-mapping) of the true radiance reflecting from the mountain in a given direction. $\Delta_t(p)$ expresses the pixel's offset due to the shifts of the camera's orientation and the air turbulence that may

have refracted the light to arrive from different direction. $\alpha(p+\Delta_t(p))$ expresses the attenuation of light due to atmospheric scattering, and A is the airlight. The total radiance recorded at a pixel due to airlight goes up just as the true radiance from the mountain is attenuated. $B(p + \Delta_t(p))$ captures any blurring that may occur due to atmospheric scattering and in-camera defocus resulting in a point spread on the image. $D(p)$ is another attenuation factor due to dust on the sensor. Finally $N_t(p)$ is a zero mean additive noise as a result of both quantization and shot noise. An example of one observation is shown in the upper half of Figure 1 and in Figure 4(a).

Our goal is to extract an image which is as close as possible to $J(p)$ using a temporal series of such observations. Thus we must attempt to undo the spatial shifts $\Delta_t p$, as well as remove the airlight and minimize the corruption due to blur, noise, and sensor dust. An example result is shown in the bottom half of Figure 1 and in Figure 4(f).

3.1. Input and System Overview

We create a final image of Mount Rainier from a sequence of 124 individual images shot at approximately 1 per second on a Canon 1Ds Mark III camera at ISO 100 with a 400mm lens. The aperture and exposure were at f/14 and $1/200^{th}$ second, respectively. The mountain only occupied about one quarter of the frame, so we cropped out a 2560 by 1440 portion of the center of the frame for further processing. We also down-sampled the image to half-resolution, as we ran into memory limitations when processing at the original image resolution.

The camera was mounted on a tripod but the shutter release was operated manually. The images were recorded with as JPEGs. Although the camera's automated sensor cleaning was activated, as will be seen, small dust particles become apparent.

We create our final image of Mount Rainier with the following steps:

- Perform a global translational alignment of each image to a single image and average over the resulting images.
- Compute pixel-wise optical flow to the globally aligned average image, initialized by the global alignment result for each image.
- For each pixel location, determine a pixel-wise weight for each corresponding pixel in each image. Created a weighted average image from the set of normalized weights.
- Dehaze the result.

We will describe each of these steps in more detail below and provide intermediate results.

3.2. Image Alignment

The images of Mount Rainier are misaligned due to camera motion and temporally varying warping due to atmospheric refraction. Fortunately, while the misalignments are quite large, several aspects of our setup simplify the alignment process significantly: 1) images taken from 90 km away with a long focal length are well modeled by an orthographic camera model, 2) the scene is mostly static, thus all misalignment is due to the camera and atmosphere, 3) the lighting on the mountain is effectively static over the relatively short time the images were taken, and finally 4) sensor noise is reasonably low during the daylight shooting conditions.

Given these properties, a straightforward combination of a global translation and local block-based flow allows us to create very well aligned images. In fact, we found more sophisticated methods, such as Black and Anandan's well-known method [2], to perform worse, as the regularization intended to handle the complexities of general scenes, (such as occlusions and parallax, scene motion, noise, lighting changes, etc.), led to overly smooth flow that did not align the small local features in our images.

Our alignment process proceeds in four steps. First, we perform a global translational alignment of each image to a single reference image using a full-frame alignment [15]. Both the camera x, y translation and yaw and pitch rotation are modeled by translation, due to the orthographic projection model. The remaining z translation is irrelevant also due to the orthographic projection. Any camera roll is handled in the next step.

Next, we average these globally aligned frames to produce a reference frame for the local alignment process. For each pixel in each image, we compute the sum-of-squared-differences (SSD) between the 5×5 neighborhood around the pixel and a corresponding translated window on the averaged image. The per pixel flow is chosen as the minimum SSD over a $1/2$ pixel discrete sampling within $[-5, 5]$ pixels translation in x and y. This flow vector captures both the camera roll and atmospheric warping. Lastly, the global and local translations are added to determine the offset, $\Delta_t(p)$, for each pixel. These offsets are used to warp each input image, I_t using bilinear interpolation to produce a warped result, I'_t, such that $I'_t(p) = I_t(p + \Delta_t(p))$.

It should be noted that all computations are done in floating point to avoid further quantization errors. Figures 4 and 5 illustrate the affect of the image alignment process.

3.3. Determining Weights for Averaging

Once the images are aligned, they can be temporally averaged, (i.e., across a stack of pixels), to reduce both sensor and quantization noise. Unfortunately, a simple averaging of these pixels (Figure 4(c) and 5(g)) does not produce a result with very high visual quality, due to the errors intro-

duced into the capture process as discussed in Section 3. Residual mis-alignments after flow warping, interpolation during bilinear resampling, dust on the sensor, and varying atmospheric blur all lead to artifacts when using only a simple average. To overcome these issues we developed a novel per-pixel weighting scheme that is a function of local sharpness.

There are two main properties we believe to be ideal for overcoming errors due to the atmosphere and alignment process. Specifically, our weighting scheme is designed with these two goals in mind:

1. To maximally suppress noise it is best to average over as many samples as possible, and
2. to maximize image sharpness it is best to only average over a few well-aligned, sharp pixels.

It may seem that these goals are contradictory, and they are in some sense – as the number of samples in the average increase, if any of those samples are mis-aligned or blurred, the sharpness of the resulting image will decrease. Our approach to merging these goals is to break-down the per-pixel weight as a combination of sharpness weight and a "selectivity" parameter that governs how many samples are averaged. For both of these aspects we drew partly on ideas from from "lucky imaging".

Lucky imaging is used in earth-based astronomic photography to overcome warping and blurring due to the atmosphere. There are many similarities between the approach's goals and ours. Mackay *et al*. [9] compensate for atmospheric shifts and blurs by first ranking each image by a sharpness measure which, in the domain of images of stars, is simply the maximum pixel value in the image. Then the top $N\%$ (often 1% to 10%) of the images, ranked by sharpness, are aligned by computing a global translation – this represents the "selectivity" of the averaging process. The resulting images are averaged. Harmeling *et al*. [6] propose an online method that extracts signal from each image by estimating the PSF to update a final result.

We will use a combination of three weights and a selectivity measure to determine the final weight given to each pixel. The weights measure local sharpness, resampling error, and the presence of dust. The selectivity is based on a measure of local variance to promote more noise reduction in smooth areas.

Sharpness Weight: In contrast with the astronomy domain, simple intensity is not a meaningful sharpness measure and, as we have densely textured images, a full frame metric is not appropriate. Instead, we compute a per-pixel weight that is a function of a local sharpness measure. We use the discrete Laplacian of the image as the local sharpness measure and set our sharpness weight proportional to the magnitude of the Laplacian.

Specifically, consider $\mathcal{L}'_t$ to be the convolution of an warped input image I'_t with a 3×3 discrete Laplacian filter, and $\mathcal{L}_\mu$ to be the Laplacian of the un-weighted mean image:

$$\mu(p) = \frac{1}{N} \sum_{t=1}^{N} I'_t(p), \tag{2}$$

where p is a pixel and there are $t = [1...N]$ images. The use of $\mathcal{L}_\mu$ is discussed later in this section. The sharpness weight for a pixel is then:

$$w^*_{tex}(p) = |\mathcal{L}'_t(p)|. \tag{3}$$

We create a normalized weight, $w_{tex}(p)$, by linearly re-mapping the output range of the absolute value of the Laplacian to the range of $[0, 1]$.

Resampling: In addition, we consider that smoothing can be introduced during global and local alignment, as the process requires pixels' values to be estimated by an interpolation of the original input pixel values. If an edge falls across integer pixel coordinates, it is known that the sub-pixel interpolation of that edge will be smoothed. To reduce this type of smoothing, we have also derived a "resampling" weight that down-weights pixels interpolated at fractional pixel locations as a quadratic function of distance of the fractional location from the nearest integer location. Specifically,

$$f_{samp}(p) = 1 - \sqrt{frac(\Delta_t(p)_x)^2 + frac(\Delta_t(p)_y)^2} \tag{4}$$

$$w^*_{samp}(p) = f_{samp}(p)^2. \tag{5}$$

$\Delta_t(p)$ is the total alignment translation of pixel p, and the "frac" function returns the fractional distance to the nearest integer, i.e., $frac(x) = min(mod(x, 1), 1 - mod(x, 1))$. We create a normalized resampling weight, $w_{samp}(p)$ by linearly re-mapping the output range of $w^*_{samp}(p)$ to the range of $[\epsilon, 1]$. We map to a minimum value of ϵ instead of 0 as we have observed qualitatively better results when allowing the interpolated pixels have some small non-zero weight. We have found $\epsilon = 0.1$ to work well.

Selectivity: As discussed above, it is important to weigh pixels not only by sharpness, but to also have a selectivity parameter. The more selective, i.e., fewer pixels averaged, the sharper the result. One might think that being extremely selective is ideal, which is the approach lucky imaging takes. However, this has as a downside, as with fewer samples, noise is not well suppressed. When averaging a fixed number of images, an equal amount of denoising occurs across the entire image. In our image of Mount Rainier, this has an undesired affect: being less selective, (i.e., using many samples), denoised the sky well, but softens features on the mountain, while fewer samples resulted in the mountain being sharp, but the sky containing unacceptable noise.

Thus, just as we found a full frame sharpness measure to be unsuitable for our images, we found a fixed selectivity measure non-ideal. We developed a per-pixel selectively

measurement that is more selective in areas of high local texture, (i.e. the mountain), and averages over more samples in areas of low local texture, (i.e., the sky). Specifically, we implement this selectivity parameter as an exponent on our per-pixel weights which lie in $[0, \lambda]$ for some large value λ:

$$w_{sharp}(p) = (w_{samp}(p) * w_{tex}(p))^{\gamma(p)}. \quad (6)$$

The exponent is calculated by first computing:

$$\gamma^*(p) = |\mathcal{L}_\mu(p)|, \quad (7)$$

and then we compute the exponent values $\gamma(p)$ by linearly re-mapping the output range of $\gamma^*(p)$ to the range of $[0, \lambda]$ for some large value λ. We have found $\lambda = 10$ to work well in practice.

Dust Removal: Lastly, we also consider sensor dust. To minimize the effect of sensor dust on the final image we can leverage the fact that the alignment shifts the dust around from image to image. We hand-mark dust spots (a single image of the clear sky can be used to automate this step) on a single initial input frame to create a binary dust mask, where a value of 1 indicated the presence of dust. We then warp this mask using the computed global alignment. The dust weight is then: $w_{dust}(p) = 1 - dust(p)$.

Only the global alignment is performed on the dust mask and the corresponding pixels in the input image, since the dust texture itself is not part of the true texture. The global alignment shifts the dust over a big enough range so that for any output pixel there will be choices in the pixel stack that do not have dust covering them. This effectively removes the dust spots from the final result.

Putting it all together: The final per-pixel weight includes the dust mask simply as an additional multiplier to down-weight dust spots:

$$w(p) = w_{dust} * (w_{samp}(p) * w_{tex}(p))^{\gamma(p)}. \quad (8)$$

Finally, we recover a denoised image as the weighted sum of warped images:

$$J(p) = \frac{\sum_{t=1}^{N} w(p) I'_t(p)}{\sum_{t=1}^{N} w(p)} - A(1 - \alpha(p)). \quad (9)$$

3.4. Dehazing and Constrast Expansion

Once we have a denoised image from the per-pixel weighted sum of aligned images, the final step is to dehaze the image. We adapt the dark channel method of He *et al*. [7] to model the haze and airlight color by surmising that in any local region of a haze free image, there is at least one channel of one pixel that is dark. The presence and magnitude of the haze is derived from the darkest pixel channel in some local region.

The local region model of He *et al*. is appropriate for many natural scenes as they often have local dark regions due to shadows or high-frequency textures, e.g., as images of trees, urban environments, etc. However, in our image

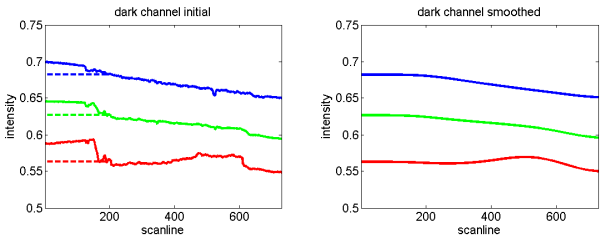


Figure 3. Computing the Airlight Component using the Dark Channel Prior: (left) The initial estimate of the dark channel for each pixel is the darkest value per horizontal scanline. The dashed line shows where we set the airlight contribution equal to that for the top of the mountain, which dehazes the sky region up to the depth of the top of the mountain. (right) Finally, as the dark-channel values are noisy across scanlines, we smooth the values.

of Mount Rainier there are many large local areas with no dark values, such as the large white glaciers. Thus the local model is not appropriate.

Instead, we note that as the haze amount and thus the dark channel value is proportional to depth; any neighborhood which captures a constant depth and has dark regions can be used to measure the dark channel value. As anyone who has flown into a metropolitan area has witnessed, the air quality and color often takes on a layered appearance. Due to the relatively conical shape of the volcano as well as the haze's relationship with altitude, we assume that the haze is effectively constant per scan-line. In contrast with previous work, we do not assume a single global airlight color [4, 7]. Instead the airlight color can vary per-scanline. We have found this necessary for our images where the airlight color appears quite different towards the bottom of the mountain (see Figure 3).

We estimate the dark channel value as the darkest value per horizontal scanline:

$$[A(1 - \alpha(p))] = min_{x=1}^{W} I(p), \quad (10)$$

where p is pixel and W is the image width.

We process the per-scanline minimum in two ways. The dark channel value is somewhat meaningless in the sky region, as this region is completely airlight. In previous work, pure sky regions were often simply ignored or masked out. We instead choose to set the airlight color for the sky above the mountain top to be equal to that at the top of the mountain. This effectively dehazes the sky region up to the depth of the mountain. Furthermore, as the dark-channel values can be somewhat noisy from scanline-to-scanline, we smooth the dark channel image in the vertical direction using a broad 1D Gaussian filter. Figure 3 shows plots of the per-scanline dark channel values.

The final dehazed image is computed as $I(p) - [A(1 - \alpha(p))]$, for an image I. This dehazing operation is not only valid for our final weighted mean. In the result section, we will show dehazing applied at various stages of our processing pipeline to illustrates the affect of each stage.

Finally, we stretch the contrast by a linear remapping of the luminance to the full image range of $[0, 1]$. We color balance the final image using the gray granite of the mountain and white glaciers as as a gray and white point.

4. Results

We demonstrate the results through a series of images and detail crops. The image in Figure 4(a) shows a single input image, $I_t(p)$. All further images demonstrate intermediate results of the pipeline after dehazing (Before dehazing the differences are almost imperceptible.) Figure 4(b) shows the same input image after haze removal, $I_t(p) - A(1 - \alpha(p))$. The effects of noise and dust becomes apparent. Before performing any processing on the images, we crop our full-frame 21 MP images to include on the relevant sections of the mountain and remove the gamma correction factor of 1.24, which we calibrated by imaging the gray panels on a Macbeth Color Checker, from the JPEG images. We re-apply a gamma correction of 1.45 for displaying our results.

We also show the effect of simply averaging the temporal samples. Figure 4(c) represents

$$\frac{\sum_{t=1}^{N} I_t(p)}{N} - A(1 - \alpha(p)) \tag{11}$$

after averaging and dehazing. The averaging removes the noise but also blurs considerably due to camera motion and air turbulence.

Removing the global motion of each image and averaging removes much of the blur as can be seen in Figure 4(d). Adding the local flow into the pixel motion further refines the image (Figure 4(e)):

$$\frac{\sum_{t=1}^{N} I'_t(p))}{N} - A(1 - \alpha(p)). \tag{12}$$

Finally, by weighting each sample as described in Section 3.3 we achieve our final result:

$$\frac{\sum_{t=1}^{N} w(p) I'_t(p)}{\sum_{t=1}^{N} w(p)} - A(1 - \alpha(p)), \tag{13}$$

which can be seen in Figure 4(f).

Figure 5 shows zoomed-in side-by-side comparisons of two regions on the mountain for each of the results presented above. Each results shows progressively increasing image quality, as a function of decreasing noise and increasing sharpness. Our final result, that uses full alignment and our novel per-pixel weights is significantly sharper than any of the other results.

5. Discussion

We have shown that with careful registration and selecting the most reliable pixels, multiple images can provide a sharp clean signal. The key contribution of this work is the concept that such an image can be captured through 90 kilometers of hazy air. The main technical contribution is in the choice of weights based on local sharpness measures and resampling. While we have used these weights for denoising images for input to a dehazing process, we believe our weighting methodology would improve general multi-image denoising algorithms.

A second contribution is the use of spatially-varying (in our case, per scanline) airlight color when performing dehazing. We have found this necessary for the scene we consider, and it is likely important for dehazing any large and very distant outdoor object.

One might consider which parts of the process can be achieved on the sensor. If the camera is static and there is no air turbulence, the main problem becomes one of removing the airlight before it saturates the pixels. A sensor could open an electron drain per pixel or small patch. The drain could be set equivalent to the electron gain from a large percentage of the minimum incoming radiance over the patch. This would minimize the quantization noise by allowing a longer exposure to spread the signal over more bits in the sensor range. The final image plus the "drain" image, which would approximate the airlight, would need to be recorded to recover the full image. The effect would be similar to that outlined in the Gradient Camera [19]. It is less clear how any blur due to camera motion and air turbulence could be minimized.

Our work demonstrates overcoming one specific difficult imaging scenario. We hope this paper inspires further work in capturing difficult to image scenes.

References

[1] E. P. Bennett and L. McMillan. Video enhancement using per-pixel virtual exposures. In *SIGGRAPH '05*, pages 845–852, New York, NY, USA, 2005. ACM.

[2] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Computer Vision, 1993*, pages 231–236, 1993.

[3] J. Chen and C.-K. Tang. Spatio-temporal markov random field for video denoising. In *CVPR '07*, pages 1–8, June 2007.

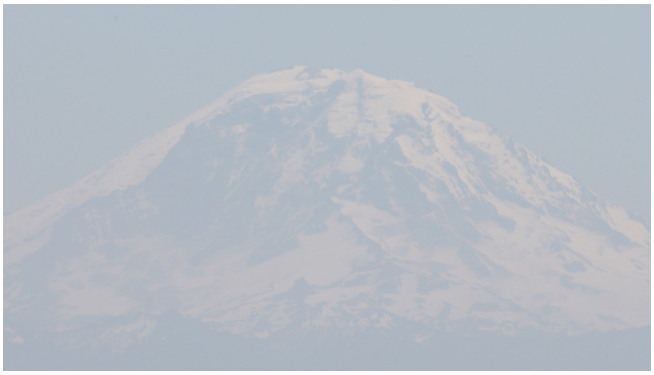
[4] R. Fattal. Single image dehazing. In *SIGGRAPH '08*, pages 1–9, New York, NY, USA, 2008. ACM.

[5] D. Flett and Y. Weiss. Optical flow estimation. In N. Paragios, Y. Chen, and O. Faugeras, editors, *Handbook of Mathematical Models in Computer Vision*, chapter 15, pages 239–258. Springer, 2005.

[6] S. Harmeling, M. Hirsch, S. Sra, and B. Schölkopf. Online blind deconvolution for astronomical imaging. In *ICCV '09*, May 2009.

[7] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. pages 1956–1963, 2009.

[8] J. Kopf, B. Neubert, B. Chen, M. Cohen, D. Cohen-Or, O. Deussen, M. Uyttendaele, and D. Lischinski. Deep photo:



(a) Original Single Input Image

(b) Dehazed Single Input Image

(c) Dehazed Mean of Input Images

(d) Dehazed Mean of Globally Aligned Images

(e) Dehazed Mean of Globally + Locally Aligned Images

(f) Dehazed Weighted Mean of Globally + Locally Aligned Images

Figure 4. Dehazing Results: (a) A single input image. (b) The dehazed single image is very noisy and does not show very much detail. (c) Due to camera movement and local shifts due to atmospheric refraction, taking the mean across the input images results in a very blurry result. (d) Global alignment improves the result, while (e) adding local alignment leads to an even sharper result. (e) In our final result, per-pixel weights lead to increased sharpness on the mountain, while smooth regions such as the sky are denoised successfully. Dust spots are also removed. Note: We have not shown the pre-dehazing images for results (b)–(e) as before dehazing there is almost no perceptual difference when compared to image (a). Only after dehazing and contrast expansion are the differences are very apparent.

Model-based photograph enhancement and viewing. *ACM Transactions on Graphics (SIGGRAPH Asia)*, 27(5):116:1–116:10, 2008.

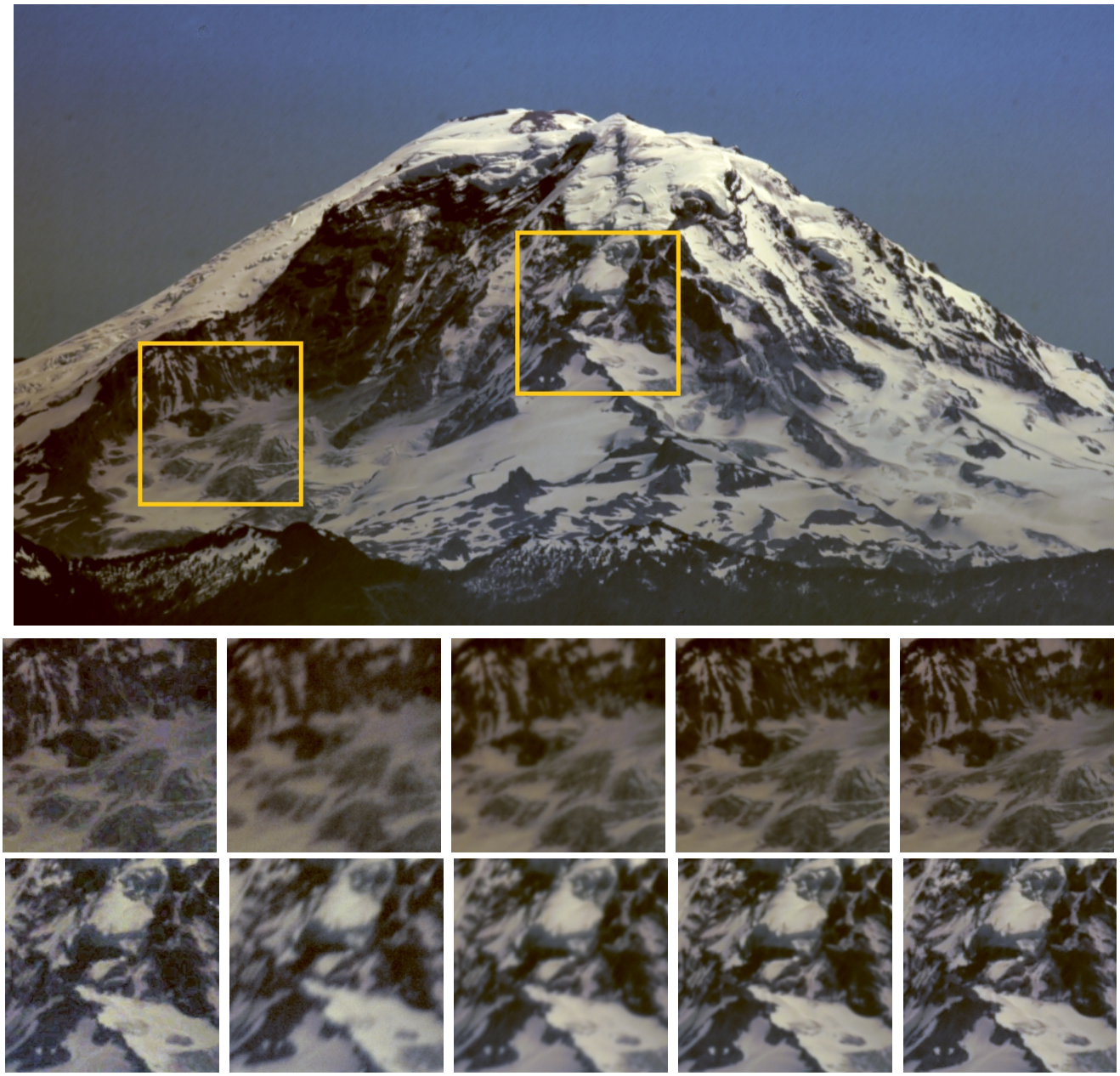
[9] C. D. Mackay, J. Baldwin, N. Law, and P. Warner. High-resolution imaging in the visible from the ground without adaptive optics: new techniques and results. volume 5492, pages 128–135. SPIE, 2004.

[10] S. Narasimhan and S. Nayar. Removing weather effects from monochrome images. In *CVPR '01*, pages II:186–193, 2001.

[11] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *PAMI*, 12(7):629–639, 1990.

[12] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE TIP*, 12(11):1338–1351, 2003.

[13] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *CVPR '05*, pages 860–867, 2005.



(f) Single Image (g) Mean (h) Global (i) Global+Local (j) Weighted

Figure 5. Detailed Dehazing Results: Cropped zoom-in regions, as indicated by yellow boxes on our final result (top), show the progressing increase in image quality, as a function of decreasing noise and increasing sharpness, from single to multi-image dehazing with various stages of alignment and weighting of the images. Our final result, with full alignment and our novel per-pixel weights, is significantly sharper than any of the other results.

[14] Y. Y. Schechner, S. G. Narasimhan, and S. K. Nayar. Instant dehazing of images using polarization. In *CVPR '01*, volume 1, pages 325 – 332, June 2001.

[15] H.-Y. Shum and R. Szeliski. Construction of panoramic image mosaics with global and local alignment. *Int. J. Comput. Vision*, 36(2):101–130, 2000.

[16] E. Simoncelli and E. Adelson. Noise removal via bayesian wavelet coring. *Image Processing, 1996*, 1:379–382 vol.1, 1996.

[17] R. Szeliski. Image alignment and stitching: a tutorial. *Found. Trends. Comput. Graph. Vis.*, 2(1):1–104, 2006.

[18] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. *Computer Vision, 1998*, pages 839–846, 1998.

[19] J. Tumblin, A. Agrawal, and R. Raskar. Why i want a gradient camera. In *CVPR '05*, pages 103–110, Washington, DC, USA, 2005. IEEE Computer Society.