EUROGRAPHICS 2013 / I. Navazo, P. Poulin
(Guest Editors)

Volume 32 (2013), Number 2

Sharpening Out of Focus Images using High-Frequency Transfer

Michael W. Tao, Jitendra Malik, and Ravi Ramamoorthi

Univesity of California, Berkeley

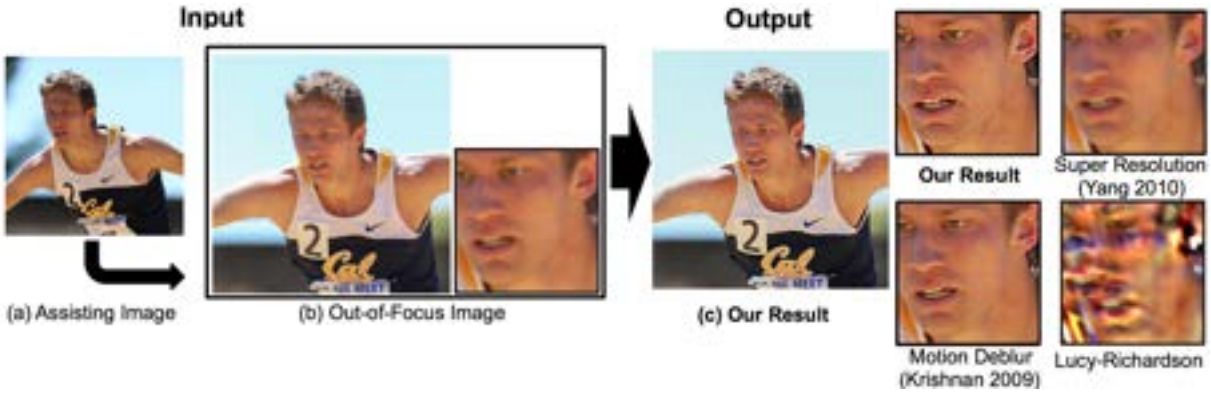

Figure 1: *Owing to out-of-focus errors, (a) and (b) are focused differently and capture different moments. There is significant motion between the images (notice the arms), as can also be seen by flipping between them in the HTML. With the assisting image (a) and our high frequency blending techniques, our result (c) uses the additional information to reduce ringing and add details. Readers are encouraged to zoom into the PDF, didactic video, and supplementary HTML images and comparisons.*

Abstract

Focus misses are common in image capture, such as when the camera or the subject moves rapidly in sports and macro photography. One option to sharpen focus-missed photographs is through single image deconvolution, but high-frequency data cannot be fully recovered; therefore, artifacts such as ringing and amplified noise become apparent. We propose a new method that uses assisting, similar but different, sharp image(s) provided by the user (such as multiple images of the same subject in different positions captured using a burst of photographs).

Our first contribution is to theoretically analyze the errors in three sources of data—a slightly sharpened original input image that we call the target*, single image* deconvolution *with an aggressive inverse filter, and* warped *assisting image(s) registered using optical flow. We show that these three sources have different error characteristics, depending on image location and frequency band (for example, aggressive deconvolution is more accurate in high-frequency regions like edges). Next, we describe a practical method to compute these errors, given we have no ground truth and cannot easily work in the Fourier domain. Finally, we select the best source of data for a given pixel and scale in the Laplacian pyramid. We accurately transfer high-frequency data to the input, while minimizing artifacts. We demonstrate sharpened results on out-of-focus images in macro, sports, portrait and wildlife photography.*

1. Introduction

It is common in sports or macro-photography for a desired image to be out-of-focus and blurred, because of camera or rapid subject movement (Figs. 1(b) and 2(top left)). Professional photographers often compensate by capturing a number of images, using the burst mode on a camera. With moving subjects, some frames may be in focus, while others are

not.[†] Photographers rely on luck to obtain a sharp frame that is suitable. However, none of the sharp frames may capture the exact moment we seek (Fig. 1(a)). In this paper, we use these "assisting images" to sharpen the blurred out-of-focus image, eliminating the compromise between timing or composition and sharpness.

Our algorithm combines three sources of data: (1) The original input image, to which we apply a conservative deconvolution to obtain a slightly sharper but still artifact-free baseline (*target*), (2) an aggressive *deconvolution* that preserves high frequencies but introduces ringing, and (3) the warped *assisting* image(s), registered to the original using optical flow. The target is accurate at low frequencies, where blur has minimal impact. High frequencies can be transferred from the deconvolved image at sharp features like edges, but this may lead to ringing in smooth or textured areas. Optical flow is usually accurate in textured areas, and the warped assisting image can be used in these regions. This paper formalizes these insights theoretically, and develops a practical algorithm to combine the three data sources. The complete pipeline for our method is shown in Fig. 2.

We begin in Sec. 3 with a theoretical analysis of the expected errors at each frequency band for each data source. This analysis quantifies the different error characteristics of each source, and is validated numerically on real examples. Next, we develop a practical implementation of the error calculation (Sec. 4), given that we do not know ground truth to estimate error, nor can we consider the global frequency domain (since different spatial pixels will be best reconstructed with different data sources). Instead of the Fourier domain, we consider a Laplacian pyramid of the image. Instead of ground truth, we use the warped assisting image as a reference, to compute errors for the original input and aggressively deconvolved images. Finally, for each pixel in the Laplacian pyramid, we select the data source with the lowest error, introducing high-frequencies where possible. Several practical details are discussed in Sec. 5, summarized in Fig. 2.

We evaluate our technique both on images where we have ground truth and apply a synthetic blur to simulate defocus (the football player example used to illustrate our algorithm first shown in Fig. 2, and Fig. 8), as well as on actual out-of-focus images from multiple photographers' datasets (Figs. 1 and 7). The assisting images are often quite different in composition from the target. Our method significantly improves results from either simple image warping, or blind deconvolution, as well as in comparison to other super-resolution and de-blurring methods (Figs. 1, 7, 8, and 9). Our supplementary HTML includes results on a dozen examples, each with several comparison methods. Our algorithm sharpens consistently a range of natural imagery with almost no artifacts.

[†] http://www.robgalbraith.com/bins/multi_page.asp?cid=7-8740-9068
http://photography-on-the.net/forum/showthread.php?t=491657

2. Related Work

Deconvolution: A single image can be sharpened directly. However, blur due to autofocus or camera shake is typically non-invertible, and deconvolution therefore involves trade-offs between ringing artifacts and sharpness [YSQS08, KF09].The success of these methods also relies heavily on accurate estimation of the blur kernel.

The majority of recent works on deblurring address motion blur, but perform poorly on out of focus blur [LWDF11, HSGL11]. Cho et al. and Li et al. [LKJ*10,CWL12] propose sharpening motion-blurred video frames using sharp frames. Our algorithm works for larger displacement and deformation of objects between frames as opposed to close video frames. The motion blur kernel is often sparse; motion deblurring assumes sparse kernel distributions, making these methods ineffective for out-of-focus blur. We address this point in our comparisons to the motion de-blurring method of [KF09] in Figs. 1, 7 and 8. [YSQS07] and [AAB09] propose using blur and no-blur image pairs to find image alignment and estimate the motion blur kernel. The alignment requires subjects to be mostly rigid (e.g., buildings). In contrast, our approach uses additional information from assisting frames through optical flow to more robustly transfer high frequencies, relaxing the need for accurate kernel estimation and the need for rigid objects.

Deconvolution methods addressing out-of-focus blurs typically introduce ringing artifacts [Ric72]. [TTBL09] present a natural image statistics prior for deconvolution to enhance details. Because of the limited image statistical priors, the method does not work for general natural images, as stated by the authors. Unsharp mask is commonly used to sharpen images. However, the overshoot and undershoot effects can often introduce unfavorable ringing.

A number of recent papers have designed special photographic methods and blur kernels that are invertible such as coded apertures [RAT06, LFDF07]; and plenoptic cameras to recover light fields [NLB*05], but our method is designed to work with standard camera and focus mechanisms.

Combining Registered Images: Using multiple images to de-blur photographs has also been studied, but current methods require the subject or camera movements to be static, and the images to already be registered. Agarwala et al. [ADA*04] describe a method to use graph cuts for extended depth-of-field or image stacking. However, using graph cuts requires the image stack to have minimized subject or camera movements. Yuan et al. [YSQS07, YSQS08] use image pairs. Other work such as flash no-flash also enhances images with a specified image-pair or multiple images [PSA*04, MKR07]. However, in fast action photography such as sports, creating such image pairs is impractical.

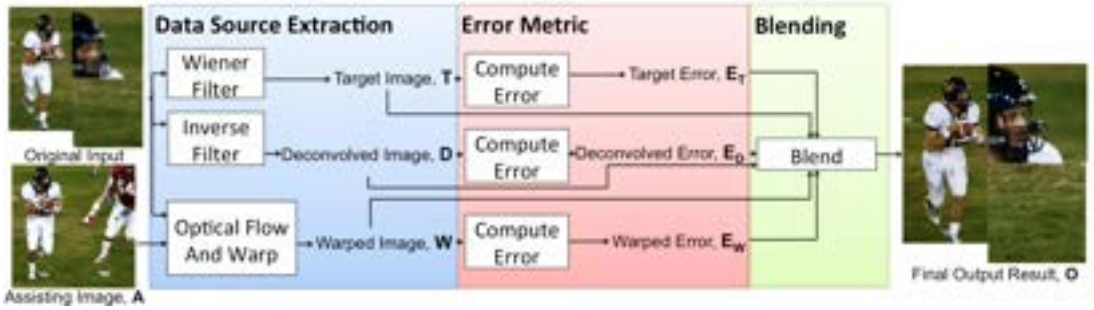

Figure 2: *Pipeline for our method. Our algorithm takes two inputs, the blurry image to sharpen and a similar but sharp assisting image(s). We sharpen the input using a conservative Wiener filter to obtain a baseline target T, and apply an aggressive inverse deconvolution filter to obtain D. We also use optical flow and warping to get W from the assisting image. With the three registered data sources, T, D, and W, we compute their errors: E_T, E_D, and E_W. We then blend to produce output O.*

Mahajan et al. [MHM*09] proposed a visually plausible interpolation method. Using linear interpolation between two frames has limitations, since the subject and camera do not always move linearly. Therefore, using the original blurred image as one of the inputs is essential.

Model-Based Super Resolution: Example-based super-resolution can be used for de-blurring [FJP02,Fat07,GBI09, YWHM10,FF11]. The most prominent limitation is that they require a large number of image priors and handle small blur sizes. Photographers usually do not have access to large datasets. With only a few assisting images, artifacts and softness persist in the final result (see Figs. 1, 7, and 8). While super resolution techniques typically handle blurs up to one or two pixels, we deal with larger blur kernels that are on the order of 8 pixels (see Fig. 9). Generally, out-of-focus images exhibit larger blur kernels than just smaller resolutions. [GBI09,FF11] propose using similar patches within a single image or other similar images. Because of the statistical priors that are used in these algorithms, vectorization artifacts are present. [SJK*12] uses temporally close video frames to achieve the super-resolution effect. Our proposed method alleviates the temporal and super-resolution limitations.

[JMAJ10] proposed personal photo enhancement. They estimate the blur kernel by using face priors to deblur the targeted image. However, the system requires a training set of faces. Because they use patch-based image replacement from multiple images, the results show changes of facial features; we focus on a more generalized approach that maintains composition and low frequency features of the out of focus image.

3. Theory: Error Analysis of Data Sources

The input is the out-of-focus blurred original image, and one or more sharp assisting images A_i. From this, we create three data sources as shown in Fig. 2. We apply a conservative deconvolution to the out-of-focus input to get a slightly sharper image, that will serve as the target T for the rest of this paper—we do this to have a realistic baseline, which is the best we can easily do without introducing artifacts. Our second source of data is an aggressive single image deconvolution D. Since we will combine sources, we can afford to be much more aggressive than previous work, preserving high frequencies at the risk of introducing artifacts. Finally, as we will describe later, the assisting images are combined and warped to the target image, to obtain the warped image W. Examples of T, D, and W are shown later in the paper in Figs. 5. In this section, we theoretically analyze the errors of T, D and W, and provide empirical validation. In the following section, we will describe practical implementation details of each of these steps. We will use a hat on variables to represent frequency domain quantities.

3.1. Basic Framework

In general, the target and deconvolved images can be written,

$$T = H \otimes X + V \tag{1}$$

$$D = G \otimes T = (G \otimes H) \otimes X + G \otimes V. \tag{2}$$

X is the original signal, that we seek to recover and V is additive noise, that is assumed to be independent of the signal. H is the original blur from defocus, that we assume is constant over the image T (more precisely, it is the original blur followed by a small sharpening filter for the conservative deconvolution). G is the deconvolution filter that we apply to compute D. More complex deconvolution algorithms and noise models are possible, that involve spatially-varying kernels. The actual errors E_T and E_D can then be computed,

$$\begin{aligned} E_T &= (I - H) \otimes X + V \\ E_D &= (I - G \otimes H) \otimes X + G \otimes V, \end{aligned} \tag{3}$$

where I is the identity operator, and we continue to use the positive sign on V (since noise is assumed to have zero mean). In practice, we do not know the ground truth X nor the filter H or noise V. As described later, we use the warped image W as a reference instead of the unknown ground truth X (Sec. 4); we estimate the blur kernel H from the out-of-focus and assisting images (Sec. 5), and we use a constant expected value for the noise amplitude. However, the theory

in this section is best developed directly from equation 3, assuming X and H are given, and V is known.

We first study the Fourier domain to develop the key insights; our actual algorithm operates entirely in the spatial domain using a multiscale Laplacian pyramid [BA83].

Frequency Analysis: We compute errors in the Fourier domain,

$$E_{\hat{T}} = (1-\hat{H})\hat{X} + \hat{V} \qquad E_{\hat{D}} = (1-\hat{G}\hat{H})\hat{X} + \hat{G}\hat{V}. \quad (4)$$

The frequency domain also allows us to compute expected errors, based on the spectral power density $S(f)$ of the signal X, as well as $N(f)$ for the noise V. In particular,

$$\begin{aligned} \left\langle E^2_{\hat{T}}(f) \right\rangle &= \left|1-\hat{H}(f)\right|^2 S(f) + N(f) \quad (5) \\ \left\langle E^2_{\hat{D}}(f) \right\rangle &= \left|1-\hat{G}(f)\hat{H}(f)\right|^2 S(f) + |\,\hat{G}(f)\,|^2\, N(f), \end{aligned}$$

where we have made the dependence on frequency f explicit.

Discussion: The derivation so far is similar to that used to derive the optimal Wiener filter [Wie64]. However, we seek to estimate errors for the target and aggressively deconvolved images, rather than finding an optimal $\hat{G}$. There is no single ideal filter in our case, given different image content like textures, edges and low frequency regions. Fortunately, we also have assisting sharp images, and can be more aggressive in choosing $\hat{G}$.

3.2. Error Analysis for T and D

To preserve edges, we choose $\hat{G} = \hat{H}^{-1}$, which produces the sharpest results. However, applying $\hat{G}$ will amplify noise and lead to ringing; the ringing will eventually be suppressed in the output by using other data sources in those regions.

Now, consider the errors in equation 5. For the target T, the first term (*signal or data error*) will dominate, if noise is low. On the other hand, the second term (*noise amplification*) is the source of error for the deconvolved image since $\hat{G} = \hat{H}^{-1}$. A key factor is the signal-to-noise ratio $S(f)/N(f) = \mu(f)$. If we divide equation 5 by $N(f)$ to obtain the normalized relative error per unit of noise, we obtain

$$\begin{aligned} \left\langle \bar{E}^2_{\hat{T}}(f) \right\rangle &= \left|1-\hat{H}(f)\right|^2 \mu(f) + 1 \quad \in [1, \mu+1] \\ \left\langle \bar{E}^2_{\hat{D}}(f) \right\rangle &= \frac{1}{|\,\hat{H}(f)\,|^2} \quad \in [1, \infty]. \quad (6) \end{aligned}$$

Discussion: At low frequencies, where $\hat{H}(f) \approx 1$, both errors above are comparable, and are close to 1 (proportional to noise). Now, consider mid-frequencies, such as where $\hat{H}(f) = 1/2$. In this case, $\left\langle \bar{E}^2_{\hat{T}}(f) \right\rangle \sim (1/4)\mu(f) + 1$, while $\left\langle \bar{E}^2_{\hat{D}}(f) \right\rangle \sim 4$. The relative error depends on the signal to noise ratio μ. If that is large (greater than 12 in this example), the target image will have higher errors at that frequency—signal error due to attentuation from blurring will exceed noise amplification error. For lower signal to noise ratios, the converse is true. For high frequencies, $\hat{H}(f)$ approaches 0. While the maximum target error is $\mu+1$, the error for the deconvolved image will be higher.

Spatial Domain Insights and Analysis: Frequency analysis provides insights but practical algorithms are local in the spatial domain. Fortunately, the above insights carry into the spatial domain. Most critically, the *target input image will have high errors in sharp edges*, when the signal-to-noise ratio is large in the high frequencies; deconvolved will perform better at those pixels. Conversely, the *target input will have lower error in smooth regions.* To numerically verify these observations, we directly compute errors using equation 3.

It is also advantageous to use a Laplacian pyramid representation [BA83] to separate out different frequencies, while still keeping locality. Since Laplacian construction involves linear operators, error computation and the construction can be done in either order. (Specifics of the implementation for our final algorithm are discussed in Sec. 4.)

Fig. 3 shows the real measured errors (left column), with respect to ground truth X, align closely with our computation of E_T and E_D using equation 3. The predicted error for data and noise is shown in the right two columns. As expected, the E_D exhibits ringing in some pixels, from sharp edges near the sides of the image. In contrast, the target error E_T is lower in most areas of the image, that are smooth, but larger when there are high-frequency edges (and the signal-to-noise is therefore more in high frequencies). E_W is discussed in the next section.

3.3. Error Analysis for W

The warped image is written as $W(x) = A(x + F(x))$, where $F(x)$ is the optical flow field from the input to the assisting image. (In practice, we compute the flow field from the assist A to the input image, and then flow all quantities of interest). For simplicity, we consider a single assisting image and a single variable x to index the image coordinates. Errors arise because of errors in the flow field, and are more pronounced at occlusions and edges in the source image [BBM09]. To derive an error metric, assume that $F(x)$ is incorrect and should actually be $F + dF$, where dF is a small change (so that we can apply differential methods),

$$\begin{aligned} W(x) + dW(x) &= A(x + F(x) + dF(x)) \\ dW(x) &= dF(x) \cdot \frac{dA}{dx}(x + F(x)). \quad (7) \end{aligned}$$

The image error is the product of two components: in the optical flow itself and the image derivative. If the flow is exact, there is no error. Equivalently, in a smooth region, the flow may not be accurate but the image gradient is very small and the pixel error is low. We approximate dF as the standard deviation σ_F of F in a 3×3 neighborhood (i.e., how much

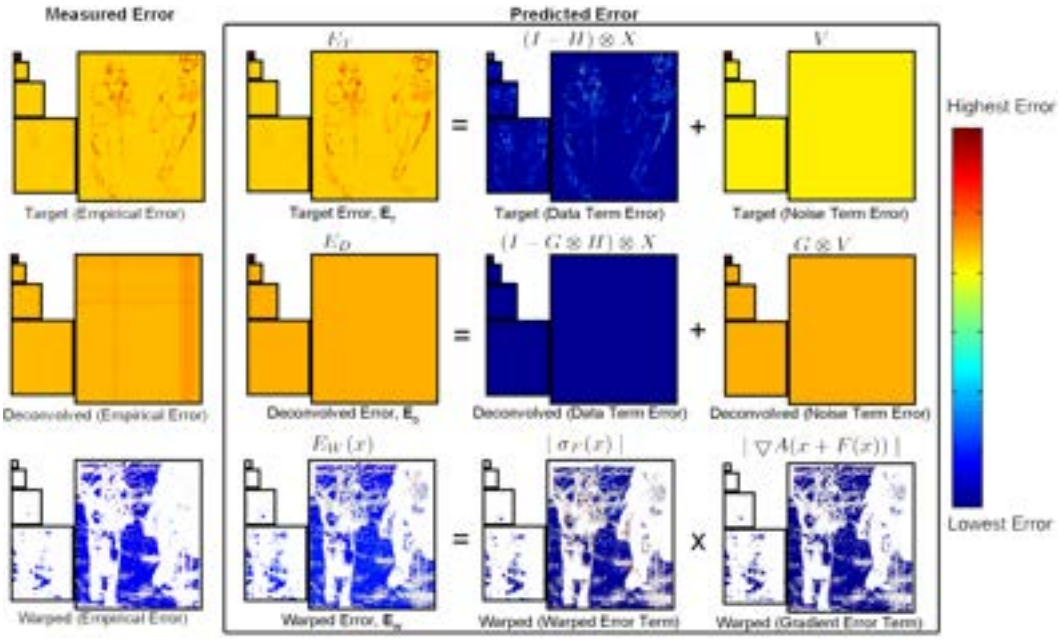

Figure 3: *Error Validation and Overview. In the left column, we compute the actual measured error, which is the normalized difference for each pixel against ground truth. Our theoretical errors are aligned closely with the empirical errors. Predicted error is computed using equation 3 for E_T and E_D, and equation 8 for E_W. The predicted error is composed of the two terms in the right columns, that are computed separately and combined. All images are 5 level Laplacian pyramids, and use the same normalized scale, shown in the color chart; this scale is used throughout the paper for errors. White regions are hole errors, since there is no data at those pixels.*

nearby flows differ, which will be large near edges and occlusions). Finally, since images are 2D, we use the magnitude of the relevant gradients,

$$E_W(x) \approx | \sigma_F(x) | \cdot | \bigtriangledown A(x+F(x)) | . \qquad (8)$$

Note that this error is undefined (infinite) when there are holes in the flow; Sec. 5 describes how we identify such regions. Holes are shown in white in Fig. 3 and elsewhere in the paper. The error estimate is good for high frequencies, but low frequencies will have systematic errors like lighting changes; we address this in Sec. 5 by using target or deconvolved images at coarser levels of the Laplacian pyramid.

The bottom row of Fig. 3 shows the warped error E_W on the same scale as E_T and E_D. The assisting images have much lower error when flow is accurate. However, there are holes, and the flow can be inaccurate near edges and occlusions. These observations motivate the use of assisting images to transfer high frequencies where flow is accurate, while using other data sources near edges, occlusions and holes. This example confirms that *target T has high errors near edges, deconvolve D has uniform errors, and warp W lower errors when flow is accurate.*

Combining Sources: We seek to combine T,D and W effectively. We use the source with the lowest error for each pixel in the Laplacian pyramid. In pixels that are holes for W, the error metrics above cannot be computed, and we default to using the target image T. This simple scheme is shown in Fig. 4 and gives a substantially sharper result than the original T but some artifacts remain. Section 5 describes a more sophisticated blending algorithm that transfers high frequencies while minimizing artifacts.

4. Practical Error Computation

The previous section has developed a theoretical error model for the three sources of data, T, D, and W. Pratically computing these errors poses two challenges: first, some of our analysis is in the frequency domain and, second, equation 3 assumes we know the ground truth X and blur kernel H. In practice, we replace the use of X with W. The kernel H is estimated from the inputs as discussed in Sec. 5.

Finally, the noise term is inherently random, but the practical error computation should use the expected value (since we have no way of knowing which pixels have more noise than others). In practice, we use a small constant value for $< V^2 >$. Since noise error is independent of signal error, we combine their magnitudes.

Our final computed error for the target is,

$$\langle | L[E_T] | \rangle = \sqrt{L[(I-H) \otimes W]^2 + \left(L^2\right)[1] \left\langle V^2 \right\rangle}, \qquad (9)$$

where $L[\cdot]$ is the operator that creates the Laplacian pyramid, and $< \cdot >$ denotes the expected value, since noise is a

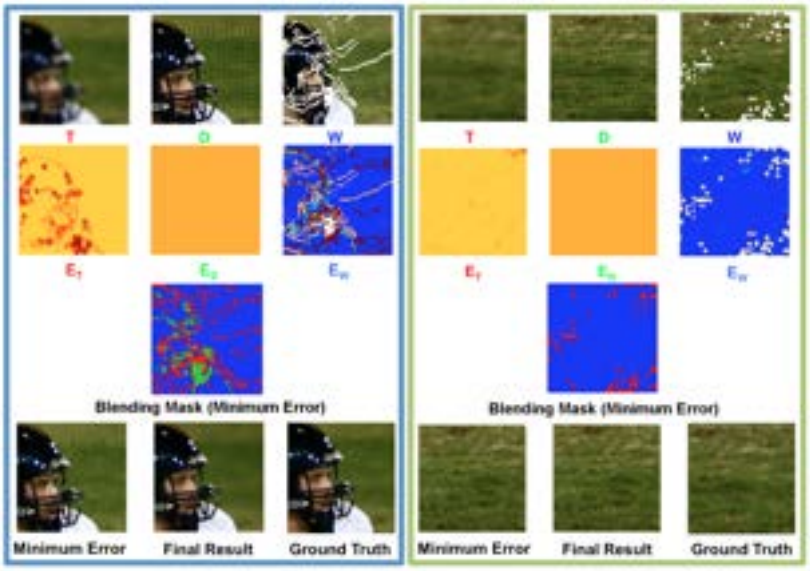

Figure 4: *Error Insight and Blending. The deconvolved image D exhibits low errors E_D in edge regions. The warped image W exhibits low errors E_W, in textured regions such as grass, while D contains ringing artifacts. However, the warped image also has holes (shown in white). The target image T is used when there are high error regions in both D and W. The blending masks shown are color coded with red being from target T, green from deconvolved D and blue from warp W. Blending using the minimum error of each pixel preserves both edges and textured regions while reducing ringing artifacts. Since these results were generated by using the ground truth as X, we describe how to replace X with the warped image in Sec. 4. We reduce sharp transition artifacts and hole residuals in Sec. 5. Our final result after these two stages is also shown.*

random process. Squaring involves squaring each pixel separately. The right hand side includes two orthogonal terms for the signal or data error, and noise amplitude. Since the Laplacian construction is a linear operator, it can be applied directly to the signal error. For the noise error, we have to consider the mean square error of the noise $< V^2 >$. The Laplacian combines noise from different pixels, so the coefficient in the variance involves summing the square of the Laplacian weights at each pixel (scale). This can be accomplished by using the squared Laplacian operator (L^2) applied to a unit image $(L^2)[1]$.

For the deconvolved image D, the expected signal error can be neglected, since we use the inverse of the blurring filter. In that case, the noise amplification error is,

$$\langle | L[E_D] | \rangle = \sqrt{(L \cdot G)^2[1] \langle V^2 \rangle}, \tag{10}$$

where $L \cdot G$ is a composition of the Laplacian, and the deconvolution filter G. Note that the result is essentially a constant expected error, since it comes solely from noise, not the signal.

Finally, for the warped image W,

$$\langle | L[E_W] | \rangle = L[| \sigma_F | \cdot | \bigtriangledown A |]. \tag{11}$$

Note that taking the absolute values does not strictly commute with applying the Laplacian in this case, but in practice, it still gives a good representation of the error.

Figure 5: *T and D compared to the input image (a). T uses a conservative Wiener filter, which shows minor improvements in high frequencies and minimal ringing (b). D uses a direct inverse of the blur kernel H, and exhibits significant ringing but large improvements in high frequencies (c).*

5. Implementation

As summarized in Fig. 2, we first estimate the blur kernel, and extract the data sources T, D and W (Sec. 5.1). Next, we compute the errors E_T, E_D and E_W as described above (Sec. 5.2). Finally, we blend the data sources based on these errors to compute the output (Sec. 5.3).

5.1. Extracting and Registering Data Sources

Estimating Blur Kernel: We use a similar approach as [KPCW11]. We find patch correspondences between the input and A_i. We use each patch pair to construct a non-negative least squares system, $Ck = b$. C encodes the convolution of the sharp patch from the assisting image, with

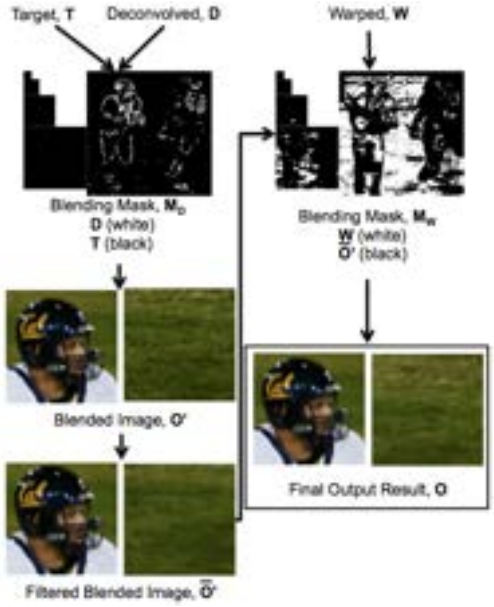

Figure 6: *With the three data sources, T, D, and W, we first blend D and T. We then use the bilateral filter to reduce ringing artifacts. Finally, we blend with the warped image to obtain the final output.*

the arbitrary kernel k, and b is the blurry patch from the input. Finally, the blur kernel H is obtained by combining k with a conservative Wiener filter to create T.

Creating Data Sources T, D and W: To obtain T, we sharpen the input using a conservative Wiener filter with noise-to-signal power ratio of the additive noise set to a value of 0.1. D is created using an idealized inverse filter with $\hat{G} = \hat{H}^{-1}$ in the frequency domain. T and D can be seen in Fig. 5. To create the warped image W, we use [BBM09] to "flow" the assisted image A to the reference frame of the target. We compute the flow from A to the input image. We compute our Laplacian pyramids using five layers.

Holes in W: Holes are caused by forward warping from A. We start with an initial binary mask M_h which is 1 for pixels with no correspondence and 0 otherwise. We also compute the L_2 YCbCr distance between D and W. If this distance is greater than a threshold (we use 0.05), mask $M_c = 1$. A pixel is confident ($M = 0$) only if both M_h and M_c are 0.

5.2. Error Metric

If $M = 0$, or confident regions, we compute the errors for E_T, E_D and E_W using equations 9, 10 and 11. We set the estimate of the squared noise amplitude $< V^2 >$ to 0.105.

If $M = 1$, or regions considered as holes, we make the conservative decision to use the target T, essentially setting E_T to a low value or 0, while setting E_W and E_D to large or infinite values.

5.3. Blending

The three sources, T, D, and W provide useful information in different regions. Using the lowest error at each pixel, as seen in Fig. 4, discontinuities and jitter appear. Instead, we develop a two-stage blend (Fig. 6), where we first combine the target and deconvolved then blend in the warped image.

Target and Deconvolved: We first create a mask image $M_D(l)$, where l is the Laplacian level. We use pixels from deconvolved D when $M_D(l) = 1$, and target T when $M_D(l) = 0$. $M_D(l) = E_D < E_T$. To reduce artifacts, we smooth the mask with a Gaussian filter ($\sigma = 0.5$ pixels) to obtain $\bar{M}_D$.

Finally, to blend T and D into an intermediate output O', we simply use a linear combination,

$$\begin{aligned} O'(l) &= \bar{M}_D(l)D(l) + (1 - \bar{M}_D(l))T(l) \\ E_{O'}(l) &= \bar{M}_D(l)E_D(l) + (1 - \bar{M}_D(l))E_T(l), \end{aligned} \quad (12)$$

where the multiplications are pixel-wise. We then reconstruct the image from the Laplacian pyramid to obtain the intermediate output image O' and error $E_{O'}$.

Some ringing artifacts can remain, so we clean up the image by using a bilateral filter to obtain $\bar{O}'$. The parameters are $\sigma_{\text{color}} = 0.1$ and $\sigma_{\text{spatial}} = 10$ pixels. Textures with low color differentiation (such as fur and grass) may also be blurred, which will be compensated by blending high frequency details from W.

Blending Warped Image W: W provides high frequency in textures. To avoid color shifts from the target, we use coarser Laplacian pyramid levels of $\bar{O}'(l)$ because W may exibit different color balance or exposure. We first compute M_W, which characterizes which pixels we are going to take from the warped image,

$$M_W(0) = E_W(0) < E_{O'}(0), \quad (13)$$

where we only consider the finest Laplacian level 0.

We then apply the same smoothing method as described to obtain $\bar{M}_D$ to obtain $\bar{M}_W$. Finally, analogous to equation 12,

$$\begin{aligned} O(0) &= \bar{M}_W(0)O(0) + (1 - \bar{M}_W(0))\bar{O}'(0) \\ O(l > 0) &= \bar{O}'(l), \end{aligned} \quad (14)$$

and we obtain the final output image O by inverting the Laplacian pyramid from $O(l)$. Figure 6 shows the various steps.

6. Results

We show examples from photographers' actual datasets; a range of real-world examples are in Figs. 1 and 7. Readers are encouraged to look into our results in the electronic document, our didactic video, and supplementary HTML, which also contain a number of additional results.

Images were taken with multiple cameras settings to emphasize the generality of our algorithm. We typically use

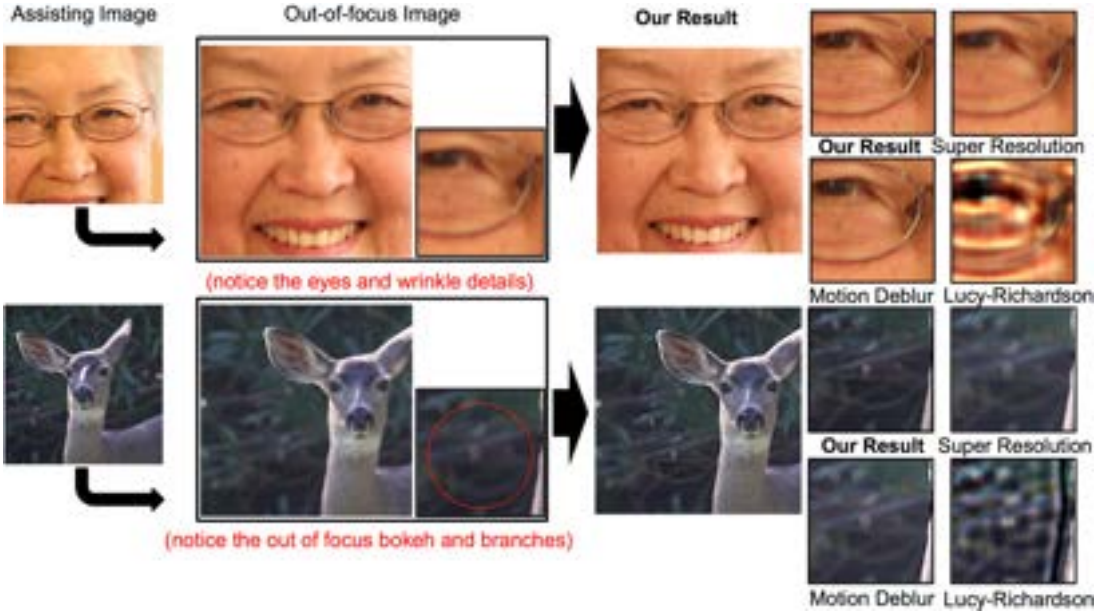

Figure 7: *Challenging real out-of-focus image comparisons. Using natural images, we compare our results against super-resolution, motion deblur, and iterative blind deconvolution [Ric72, KF09, YWHM10]. On top, our algorithm recovers fine details of the wrinkles and skin pores. The bottom example shows how our algorithm reduces the circle of confusion instead of emphasizing the edges. Our supplementary HTML contains comprehensive comparisons with more real examples, and including algorithms from Glasner et al. [GBI09], Freedman and Fattal [FF11], and unsharp mask.*

400×400 sized target. We tested our method on images up to 2 mega-pixels; a water polo example is shown in the supplementary HTML. As seen in our supplementary video and HTML, the assisting sharp images often have significant motion and deformation compared to the target out of focus image. *Portrait Photography:* In Fig. 7 (top), we show problems in portrait photography where timing and expression are crucial. With an assisting image, we can recover skin tones and wrinkle details in the out-of-focus image.

Low Light Photography: We show a high ISO example in Fig. 7 (bottom). Notice the sharpening in the branches and the fur of the deer.

Sports or Action Photography: The examples in Figs. 1, 2 and other examples in the supplementary HTML, show a problem with fast moving subjects. In Fig. 1, the steeplechaser moves rapidly, and the auto-focus may miss for some frames in a burst while producing a sharp image in other frames. Capturing the right position and moment is hard due to the fast movement. In these cases, assisting images can help greatly in sharpening the out-of-focus captured photograph. Note that the assisting image in this case is fairly different due to the rapid motion, as seen in our supplementary video.

The supplementary HTML shows additional real examples from wildlife photography of birds, where the motion is rapid and unpredictable. Our method considerably sharpens out of focus photographs, and reveals details on the eyes and feathers.

7. Evaluation and Comparisons

In our supplementary HTML, we present a detailed evaluation with more examples, each with comparison to deconvolution [Ric72], motion de-blurring [KF09], super-resolution ([YWHM10], [GBI09], and [FF11]), and unsharp masking.

Evaluation with Ground Truth: We tested our algorithm on several synthetic examples, where the defocused image was generated using a known blur kernel. In Fig. 8, our results show good agreement with ground truth.

Comparison to other De-Blurring Algorithms: We compare in Fig. 8 to iterative blind deconvolution ([Ric72]; the recent method of [LFDF07] is shown in Fig. 9 since it requires a known kernel), and state of the art techniques for motion de-blurring [KF09], and example-based super-resolution [YWHM10, FF11] (their code with default parameters given by the authors). We include comparisons against [GBI09] in our supplementary material.

In Fig 8, blind deconvolution algorithms produce results that vary by the kernel size and type, with ringing and artifacts often present. Motion-deblurring algorithms use kernel sparsity priors but out-of-focus kernels are dense. Even with large dictionaries, super-resolution methods perform inadequately, since down-scaling usually implies very small blur kernels. Our algorithm levarages the sharp assisting image both for blur kernel estimation and for introducing high frequencies, and therefore performs better.

In Figs. 1 and 7 (and accompanying didactic video),

Figure 8: *We compare our results with iterative blind deconvolution, motion de-blurring, and super-resolution [Ric72, KF09, YWHM10]. Our results consistently contain higher frequencies with less artifacts.*

we show similar comparisons for real images. Many additional examples are shown in our supplementary HTML. Our technique consistently performs better than each competing method across all of the examples.

Robustness with blur kernel size: As with all sharpening methods, the technique degrades with large blur kernel sizes. However, we preserve sharpness over a much larger operating range than most previous work. In Fig. 9, we evaluate the robustness of our algorithm against commonly used de-blurring methods. The graph emphasizes significant visual errors and ringing by considering the RMS of the largest pixel errors (while there is no perfect quantitative metric, we found this to be the most meaningful numerical value). We produce stable results even for relatively large blurs.

8. Limitations and Future Work

While the results are much sharper than the original, there is still a slight blur relative to the assisting images (or ground truth where available). Some minor color shifts may also occur from the assisting image, because we blend in different layers of the Laplacian.

When there is a lack of useful assisting images, more holes appear and, in those regions, we default to the target (Wiener filter, which can be subtituted with a state-of-the-art deconvolution algorithm for better default performance). Very large occlusions and deformations will result in higher errors and holes. The results could further be improved by hole-filling and texture synthesis techniques to sharpen even those areas. Holes from warping can also lead to a loss in visual quality where the image appears patchy, with increased sharpness where the assisting image provides information and blurred regions where there are holes. However, this effect is minimal in actual examples.

The HTML shows a portrait example with a large blur. All methods fail, but our method still performs best, consistent with Fig. 9.

9. Conclusion

We have presented a simple and flexible method for enhancing out of focus images. We first use optical flow and deconvolution to generate sources that will be useful for transferring high frequency data to the target image. We then combine these sources by analyzing different frequencies in the Laplacian pyramid. Our method improves the sharpness of images while preserving the low frequency information in the original photograph and minimizing artifacts. The algorithm is based on a principled analysis of error in multiple data sources, practical ways to compute the expected error, and effective techniques to combine multiple sources based on the errors. We have evaluated our method on several examples, demonstrating its benefit both numerically and visually. These ideas may have applications in many other image processing problems.

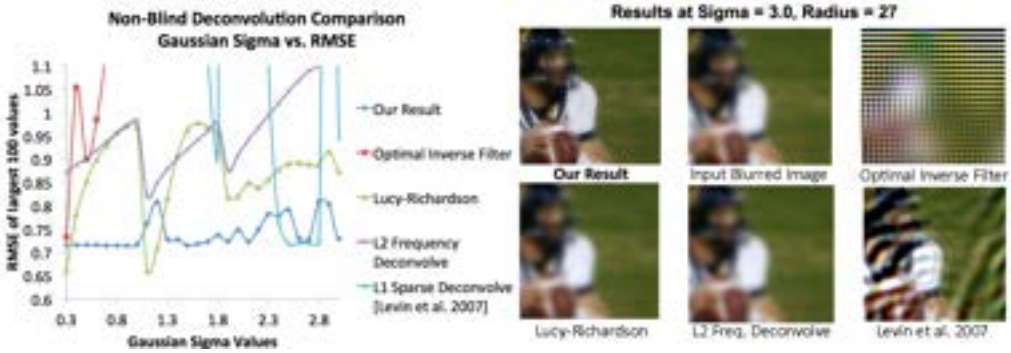

Figure 9: *Large blur kernel sizes lead to degraded performance. However, our method remains relatively stable for a range of blurs, as seen in the graph on the left. In the visual comparisons on the right for a larger Gaussian blur kernel, our result is sharper than alternative algorithms, and does not have the ringing artifacts of some previous methods in this range.*

Acknowledgements

We would like to thank The Daily Californian for providing the sports images. We acknowledge support from an NSF graduate fellowship, ONR PECASE grant N00014-09-1-0741, an Okawa Foundation Research Grant, and equipment, software donations and funding from Adobe and Intel.

References

[AAB09] ANCUTI C., ANCUTI C., BEKAERT P.: Deblurring by matching. *Computer Graphics Forum* (2009). 2

[ADA*04] AGARWALA A., DONTCHEVA M., AGRAWALA M., DRUCKER S., COLBURN A., CURLESS B., SALESIN D., COHEN M.: Interactive digital photomontage. In *ACM SIGGRAPH* (2004). 2

[BA83] BURT P., ADELSON E.: The laplacian pyramid as a compact code. *IEEE Trans. Commun. COM-31* (April 1983), 337–345. 4

[BBM09] BROX T., BREGLER C., MALIK J.: Large displacement optical flow. In *IEEE CVPR* (2009). 4, 7

[CWL12] CHO S., WANG J., LEE S.: Video deblurring of handheld cameras using patch-based synthesis. In *ACM SIGGRAPH* (2012). 2

[Fat07] FATTAL R.: Image upsampling via imposed edges statistics. In *ACM SIGGRAPH* (2007). 3

[FF11] FREEDMAN G., FATTAL R.: Image and video upscaling from local self-examples. *ACM TOG* (2011). 3, 8

[FJP02] FREEMAN W. T., JONES T. R., PASZTOR E. C.: Example-based super-resolution. *IEEE Computer Graphics and Applications* (2002). 3

[GBI09] GLASNER D., BAGON S., IRANI M.: Super-resolution from a single image. In *ICCV* (2009). 3, 8

[HSGL11] HACOHEN Y., SHECHTMAN E., GOLDMAN D., LISCHINSKI D.: Nrdc: non-rigid dense correspondence with applications for image enhancement. *ACM TOG* (2011). 2

[JMAJ10] JOSHI N., MATUSIK W., ADELSON E. H., J. K. D.: Personal photo enhancement using example images. *ACM TOG* (2010). 3

[KF09] KRISHNAN D., FERGUS R.: Fast image deconvolution using hyper-laplacian priors. *Proc. of Neural Information Processing Systems* (2009). 2, 8, 9

[KPCW11] KEE E., PARIS S., CHEN S., WANG J.: Modeling and removing spatially-varying optical blur. In *ICCP* (2011). 6

[LFDF07] LEVIN A., FERGUS R., DURAND F., FREEMEN W. T.: Image and depth from a conventional camera with a coded aperture. In *ACM SIGGRAPH* (2007). 2, 8

[LKJ*10] LI Y., KANG S. B., JOSHI N., SEITZ S. M., HUTTENLOCHER D. P.: Generating sharp panoramas from motion-blurred videos. In *IEEE CVPR* (2010), pp. 2424–2431. 2

[LWDF11] LEVIN A., WIESS Y., DURAND F., FREEMAN W.: Efficient marginal likelihood optimization in blind deconvolution. In *IEEE CVPR* (2011). 2

[MHM*09] MAHAJAN D., HUANG F., MATUSIK W., RAMAMOORTHI R., BELHUMEUR P.: Moving gradients: a path-based method for plausible image interpolation. In *ACM SIGGRAPH* (2009). 3

[MKR07] MERTENS T., KAUTZ J., REETH F.: Exposure fusion. In *Pacific Graphics* (2007). 2

[NLB*05] NG R., LEVOY M., BRÉDIF M., DUVAL G., HOROWITZ M., HANRAHAN P.: Light field photography with a hand-held plenoptic camera. *Stanford University Computer Science Tech Report CSTR* (2005). 2

[PSA*04] PETSCHNIGG G., SZELISKI R., AGRAWALA M., COHEN M., HOPPE H., TOYAMA K.: Digital photography with flash and no-flash image pairs. In *ACM SIGGRAPH* (2004). 2

[RAT06] RASKAR R., AGRAWAL A., TUMBLIN J.: Coded exposure photography: Motion deblurring using fluttered shutter. In *ACM SIGGRAPH* (2006). 2

[Ric72] RICHARDSON W. H.: Bayesian-based iterative method of image restoration. *JOSA 62*, 1 (1972), 55–59. 2, 8, 9

[SJK*12] SUNKAVALLI K., JOSHI N., KANG S., COHEN M., PFISTER H.: Video snapshots: Creating high-quality images from video clips. *IEEE Transaction on Visualization and Computer Graphics 18*, 11 (2012), 1868–1879. 3

[TTBL09] TAI Y., TANG H., BROWN M., LIN S.: Detail recovery for single-image defocus blur. *IPSJ Transactions on Computer Vision and Applications* (2009). 2

[Wie64] WIENER B.: Extrapolation, interpolation, and smoothing of stationary time series. *The MIT Press* (1964). 4

[YSQS07] YUAN L., SUN J., QUAN L., SHUM H.: Blurred/non-blurred image alignment using kernel sparseness. In *ICCV* (2007). 2

[YSQS08] YUAN L., SUN J., QUAN L., SHUM H.: Progressive inter-scale and intra-scale non-blind image deconvolution. In *ACM SIGGRAPH* (2008). 2

[YWHM10] YANG J., WRIGHT J., HUANG T., MA Y.: Image super-resolution via sparse representation. *IEEE Transactions on Image Processing* (2010). 3, 8, 9