Deep Denoising of Flash and No-Flash Pairs for Photography in Low-Light Environments

Zhihao Xia[1], Michaël Gharbi[2], Federico Perazzi[3], Kalyan Sunkavalli[2], Ayan Chakrabarti[1]
[1]Washington University in St. Louis [2]Adobe Research [3]Facebook
{zhihao.xia,ayan}@wustl.edu, {mgharbi,sunkaval}@adobe.com, fperazzi@fb.com

Abstract

We introduce a neural network-based method to denoise pairs of images taken in quick succession, with and without a flash, in low-light environments. Our goal is to produce a high-quality rendering of the scene that preserves the color and mood from the ambient illumination of the noisy no-flash image, while recovering surface texture and detail revealed by the flash. Our network outputs a gain map and a field of kernels, the latter obtained by linearly mixing elements of a per-image low-rank kernel basis. We first apply the kernel field to the no-flash image, and then multiply the result with the gain map to create the final output. We show our network effectively learns to produce high-quality images by combining a smoothed out estimate of the scene's ambient appearance from the no-flash image, with high-frequency albedo details extracted from the flash input. Our experiments show significant improvements over alternative captures without a flash, and baseline denoisers that use flash no-flash pairs. In particular, our method produces images that are both noise-free and contain accurate ambient colors without the sharp shadows or strong specular highlights visible in the flash image.

1. Introduction

Flash photography has long been a convenient way to capture high-quality images in low-light conditions. A flash illuminates the scene with a bright burst of light at the time of exposure, allowing the camera to acquire a photograph with a much higher signal-to-noise ratio than would be possible under the dim ambient lighting alone and without introducing any motion or defocus blur. The flash addresses the problem of limited illumination at its root—by adding light to the scene. However, flash illumination is not without drawbacks. An on-camera flash often creates unappealing flat shading and harsh shadows, resulting in images that fail to capture the true mood and ambience of the scene.

Researchers have considered combining pairs of flash and no-flash images—captured in quick succession with and without the flash—to create a single enhanced photograph that is both noise-free and accurately represents the scene under ambient lighting. This is achieved by merging information about the ambient scene appearance from the noisy no-flash image, with high-frequency surface image details revealed by the flash [9, 29]. However, these methods assume moderate levels of noise in the no-flash image, and that the flash and no-flash pair are, or can be, aligned.

In extremely low light, the no-flash image can be very noisy, especially when using mobile phone cameras with small apertures. This precludes the use of traditional

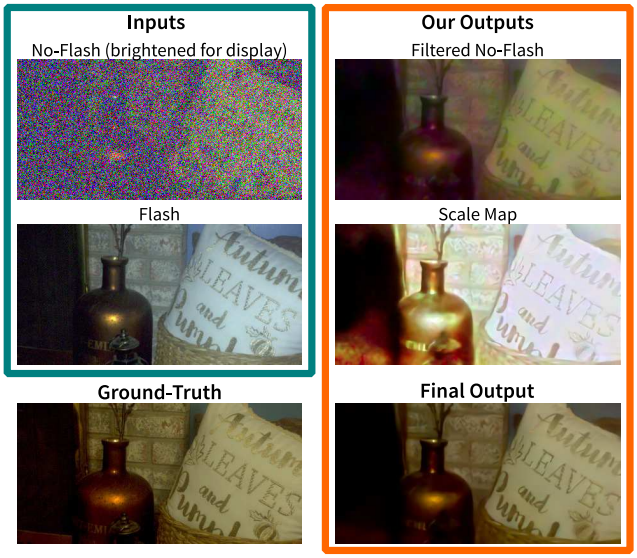

Figure 1: Given a pair of images of low-light scenes captured with and without a flash (left), our method produces a high-quality image of the scene under ambient lighting (right). This output is generated by filtering the no-flash image with a predicted field of kernels—to capture a smoothed stimate of scene appearance under ambient lighting, followed by multiplication with a scale map that introduces high-frequency detail illuminated by the flash.

flash/no-flash methods, since the noise obscures even the low-frequency shading information in the no-flash image and makes automatic alignment of the pair unreliable. In comparison, modern neural network-based denoising methods [5, 34, 40, 41] can produce reasonable estimates from a noisy no-flash image alone—although at high noise-levels, they still struggle to reconstruct high-frequency detail.

In this work, we leverage both the ability of modern neural networks to encode strong natural image priors, and the unique combination of appearance information available in a flash and no-flash image pair. Specifically, we consider the task of producing a high-quality image of the scene under ambient lighting given a flash and no-flash pair as input. We focus on extremely low-light scenes such that the no-flash image shows significant noise, and the appearance of the no-flash image is entirely dominated by the flash illumination. We further assume unknown geometric misalignment between the image pair, due to camera movement typically observed in hand-held burst photography [33].

Under these conditions, we train a deep neural network to take noisy, misaligned no-flash/flash image pairs as input, and output a denoised image of the scene under the scene's ambient illumination. Rather than directly predicting the denoised image, our network outputs a kernel field used to filter the no-flash image, and a scale map that is multiplied with this filtered output to incorporate high-frequency image details from the flash image. To use the regularizing effect of kernels to effectively filter out the high levels of noise in the no-flash input while overcoming its significant memory and computational costs [26], our network combines a recent kernel basis prediction approach [35] with efficient kernel up-sampling. The use of a scale map is inspired by classical flash/no-flash approaches [9, 29] that adopt multiplicative combination based on a view of factorizing images into albedo and shading, where the former is common across the input pair while the latter is not.

We evaluate our approach extensively under different ambient light levels and spatial misalignment, and demonstrate state-of-the-art results for low-light denoising (see example result in Fig. 1). Our method outperforms denoising without a flash—when using a single or burst of two no-flash images. This demonstrates that a flash input, despite often representing drastically different shading, is still informative towards ambient appearance. Our method also outperforms other standard denoising approaches trained directly on flash/no-flash pairs, highlighting the importance of the formulation and design of our network architecture.

Code and pre-trained models for our method are available at https://www.cse.wustl.edu/~zhihao.xia/deepfnf/.

2. Related Work

Image Denoising. Early works reduced image noise using regularization schemes like sparse-coding [21] and low-rank factorization [13] to model the local statistics of natural images. Other classical approaches have exploited the recurrence of natural image patterns, averaging pixels with similar local neighborhoods [2, 8, 23, 28, 30, 31, 38]. Current state-of the-art denoisers use deep neural networks. Burger *et al*. [3] were the first to show the ability of even shallow multi-layer perceptrons to to outperform traditional methods such as BM3D [7], and more recent approaches utilizing deeper networks and complex architectures [24, 34, 36, 39, 40, 41] have since led to further improvements in reconstruction accuracy.

Burst Image Denoising. Burst imaging can achieve impressive denoising results, by capturing multiple frames in quick succession. Recent burst denoising algorithms have focused on circumventing the frame misalignment that exists in a real burst. Some methods estimate pixel-wise displacement [14, 15, 16, 19, 25]; others only require coarse global registration, and rely on neural networks to account for the remaining displacement. Amongst the latter group, kernel prediction networks [12, 26, 27] have demonstrated superior ability to recover from misalignment. Our method builds upon Xia *et al*. [35], which predicts a low-dimensional kernel decomposition, using a linear basis and mixing coefficients, to efficiently realize larger kernels, and to induce regularization leading to improved reconstructions. However, unlike Xia *et al*. [35] that operate on a burst of images, our approach works on misaligned *flash* and *no-flash* image pairs, and uses a scale map rather than filtering to extract information from our no-flash input, and an up-sampling approach to realize even larger kernels.

Flash Denoising. Flash photography enables the capture of low-noise images in low-light environments using short exposure times and low ISO settings. But, the additional source flash light drastically changes the mood of the scene captured. To remedy this while benefiting from the flash image's higher signal-to-noise ratio, several approaches have used the flash as reference to denoise a noisy ambient (no flash) image. Petschnigg *et al*. [29] and Eisemann *et al*. [9] use the flash photo to guide a joint-bilateral filter that denoises the ambient image, transferring high-frequency content from the flash photo. Krishnan and Fergus [20] exploit the correlations between dark flash images and visible light to denoise the ambient image and restore fine details. Yan *et al*. [37] combine gradients from the flash image with the no-flash image for denoising. These methods all use hand-crafted heuristics to decide which image features to preserve from each of the flash and no-flash inputs.

More recent work [22, 32] have replaced these heuristics with deep neural networks. Li *et al*. [22] use the (aligned) flash photo as guidance to denoise ambient images. Wang *et al*. [32] address some of the shortcomings of dark flash photography by adding a stereo RGB image to the capture

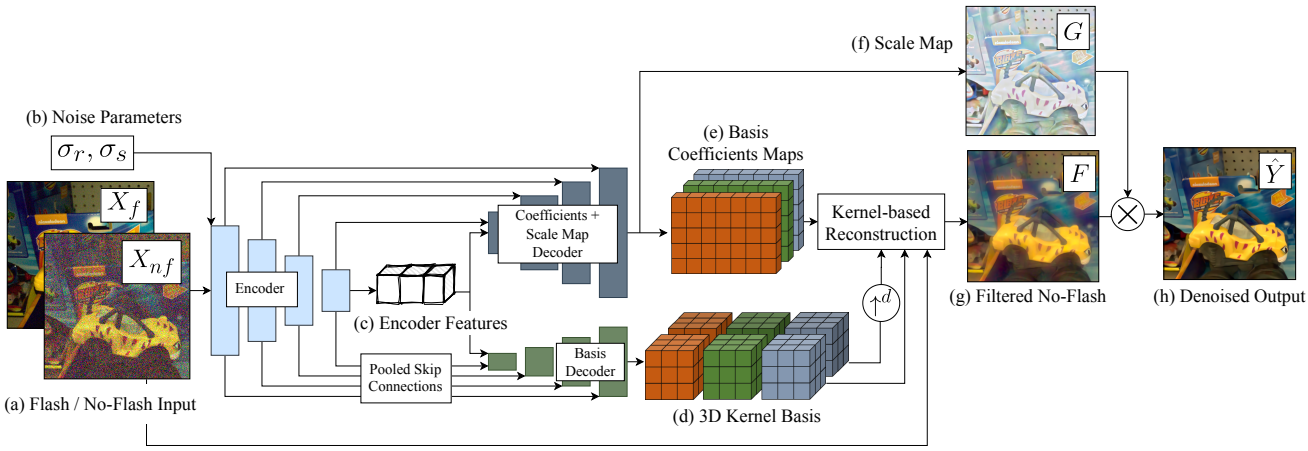

Figure 2: **System Overview.** The denoising network takes as input a pair of flash, no-flash images (a) together with the noise parameters (b). After encoding, the resulting features (c) are decoded into a multi-scale basis (d), a set of pixel-wise coefficients (e) and a scale map (f). The no-flash image is filtered using the reconstructed kernels (g) and multiplied by the scale map to produce the final denoised output (h).

setup. After being registered and aligned the two images are fused using recent techniques for hyperspectral image restoration and fast image enhancement [6, 11]. Unlike these methods, our approach handles motion between the flash/no-flash pairs and large noise levels by using large, learned denoising kernels robust to misalignment.

Beyond denoising, flash photography has also been used for other applications such as deblurring [42], shape estimation [4], and to separate shading from different ambient illuminants present in a scene [17].

3. Proposed Approach

Our goal is to estimate a noise-free color image $Y[n] \in \mathbb{R}^3$ of the scene under ambient illumination, from a pair of flash and no-flash images $X_f[n] \in \mathbb{R}^3$ and $X_{nf}[n] \in \mathbb{R}^3$, where $n \in \mathbb{Z}^2$ denotes the pixel location. Since both images are of the same scene, they represent observations of the same surfaces, with the same material properties, but under different illuminations, and with a potential change in viewpoint due to hand motion between the two shots.

3.1. Observation Model and Problem Formulation

In a chosen reference frame, we denote the appearance of the scene under ambient-only illumination as $S_a[n] \in \mathbb{R}^3$, and under flash-only illumination as $S_f[n] \in \mathbb{R}^3$. Further, we model the geometric transformations from the reference to the flash and no-flash images as 2D warps $\mathcal{T}_f(n)$ and $\mathcal{T}_{nf}(n)$, respectively. Then, the *noise-free* versions $\tilde{X}_{nf}[n]$ and $\tilde{X}_f[n]$ of our no-flash and flash inputs are given by:

$$\begin{aligned} \tilde{X}_{nf}[\mathcal{T}_{nf}(n)] &= S_a[n], \\ \tilde{X}_f[\mathcal{T}_f(n)] &= \alpha_f \ (S_f[n] + S_a[n]), \end{aligned} \tag{1}$$

where $\alpha_f \leq 1$ is a scalar that captures the effect of a possibly shorter exposure time for the flash image. Note that since the flash is typically much brighter than the ambient lighting ($S_f[n] \gg S_a[n]$), the contribution of the flash-only appearance is dominant in the flash image $\tilde{X}_f[n]$.

As in [27, 35], we assume a heteroscedastic Gaussian noise model [10] to account for both read and shot sensor noise. The observed input flash and no-flash pair relate to their ideal noise-free version (Eq. (1)) as:

$$\begin{aligned} X_f[n] &\sim \mathcal{N}(\tilde{X}_f[n], \sigma_r^2 + \sigma_s^2 \tilde{X}_f[n]), \\ X_{nf}[n] &\sim \mathcal{N}(\tilde{X}_{nf}[n], \sigma_r^2 + \sigma_s^2 \tilde{X}_{nf}[n]), \end{aligned} \tag{2}$$

where σ_r^2, σ_s^2 are read and shot noise parameters, which we assume are known. Given $X_f[n]$ and $X_{nf}[n]$, and the values of σ_r^2 and σ_s^2, we seek to estimate $Y[n] := S_a[n]$.

Flash vs. No-flash as reference. Note that in formulation above, we make a distinction between the target output $Y[n]$ and the noise-free no-flash image $\tilde{X}_{nf}[n]$, because they differ by the warp $T_{nf}(n)$. We may wish to use either of the two inputs (flash or no-flash) as the *geometric reference*. If for instance, the no-flash image is the reference, we assume $T_{nf}(n) = n$ is the identity transformation, and $Y[n] = \tilde{X}_{nf}[n]$. Conversely, if we choose the flash image as reference, $T_f(n) = n$ is choosen to be the identity mapping. In Section 4, we analyze the effect of this design choice on the output image quality, finding that in most settings, the choice of the no-flash image as reference yields more accurate reconstructions on average.

3.2. Enhancement Network

We use the basis prediction approach of Xia et al. [35], which was designed for burst denoising, as the starting point

for our model design. Our network differs in two crucial aspects: (a) rather than predicting kernels to filter both the flash and no-flash inputs and summing the result, we filter only the no-flash image and multiply a predicted per-pixel three-channel scale map to form our final output; and (b) we propose an efficient approach to predict larger kernels through upsampling, which is necessary in our setting, because we are filtering a single, highly noisy image. We show an overview of our approach in Figure 2, and include a complete description of our network in the supplement.

3.2.1 Input data

Our network takes a twelve channel tensor as input, with six channels containing the observed flash X_f and no-flash X_{nf} pair (Equation 2) themselves, and another six encoding the expected per-pixel standard deviation of noise in these inputs, computed using the (known) values of σ_s^2 and σ_r^2 and the observed noisy intensities as: $\sqrt{\sigma_r^2 + \sigma_s^2 \max(0, X^i[n])}$, for each channel $i \in \{R, G, B\}$ and $X = X_f$ and X_{nf}.

3.2.2 Predicting a global kernel basis

Like [35], our network features a common encoder whose output is fed to two decoders. The first decoder outputs a global low-rank kernel basis. Unlike [35], we do not constrain our kernels to be positive and unit-normalized. The second decoder outputs per-pixel mixing coefficients to combine the predicted basis elements and form per-pixel kernels. In another departure from [35], the second decode also outputs a 3-channel scale map. We include skip-connections from the encoder to both decoders, using global pooling for connections to the basis decoder as in [35].

3.2.3 Large kernels by interpolation

A key innovation in our method over [35] is that our basis encodes larger kernels using a 2-scale representation and an interpolation-based reconstruction scheme. This is crucial in our application where these kernels are used to smooth only one image—the noisy no-flash input—rather than a burst of images as in [35].

Specifically, our basis decoder outputs a set of J basis elements, each consisting of a pair of three-channel kernels $\{(A_j, B_j)\}_{j=1}^J$, where each $A_j, B_j \in \mathbb{R}^{K \times K \times 3}$. We interpret the second kernel B_j of each pair as a low-frequency term: a large kernel downsampled by a factor d, with an effective $((K-1)*d+1) \times ((K-1)*d+1)$ footprint. The j^{th} element of our basis is then given by $A_j + (B_j \uparrow^d)$, where $\uparrow^d$ denotes bilinear upsampling by a factor d, so that A_j can add fine high-frequency details to the kernel center. In our experiments, we use a basis with $J = 90$ kernels, with a base size $K = 15$ and upsampling factor of $d = 4$ resulting in an effective kernel size of 57×57.

3.2.4 Final reconstruction

Denoting the per-pixel coefficients from the second decoder as $\{c_j[n]\}_{j=1}^J$, we first filter the no-flash input image as:

$$F[n] = \sum_{j=1}^{J} c_j[n] \left(X_{nf} * (A_j + B_j \uparrow^d)\right)[n]. \quad (3)$$

where $*$ denotes per-channel convolution between three-channel images and kernels. Note that the filtering with upsampled kernels can be carried out efficiently, by pre-filtering the no-flash image and using dilated convolutions:

$$F[n] = \sum_{j=1}^{J} c_j[n] \left((X_{nf} * A_j)[n] + (X_{nf}^h *_d B_j)[n]\right), \quad (4)$$

where $X_{nf}^h[n] = (X_{nf} * h)[n]$ is the result of smoothing the no-flash input with a $(2d-1) \times (2d-1)$ tent kernel $h[n]$, and $*_d$ represents dilated convolution with a factor of d.

Recovering high-frequency detail with a scale map. The result $F[n]$ of this filtering step will typically encode a noise-free (and in the case of the flash as reference, an aligned) estimate of scene appearance under ambient illumination. However, due to the lower signal-to-noise ratio of X_{nf}, this filtering step cannot recover the high-frequency details that are illuminated only resolved in the flash image. To recover these, our full-pixel decoder also produces a scale map $G[n] \in \mathbb{R}^3$. Our final output $\hat{Y}[n]$ is given by the element-wise product of this scale map and the filtered no-flash image:

$$\hat{Y}[n] = F[n] \odot G[n]. \quad (5)$$

This formulation is inspired by classic flash/no-flashing denoising methods [9, 29] that add high-frequency details from the flash image in the *log domain*, i.e., corresponding to a product in our linear domain. In Section 4, we show this outperforms the alternative of using kernels to jointly denoise the no-flash and flash images.

3.3. Training details

While our network accepts raw linear sensor measurements as input and produces an estimate of linear intensities in $Y[n]$, it is trained to maximize image quality in a color and gamma-corrected sRGB space. In particular, we assume that for each training sample (X_f^t, X_{nf}^t, Y^t), we also have a scalar gain α^t (representing a desired target brightness level), and a 3×3 color transform matrix C^t based on camera sensor parameters and white-balance settings,

such that the mapping to sRGB is given by $f_t(Y[n]) = \gamma(\alpha C^t Y[n])$, where $\gamma(\cdot)$ is a gamma correction curve.

We train our model to minimize the sum of a squared L_2 pixel loss, and a L_1 gradient loss between the estimated and ideal rendered images:

$$L = \frac{1}{T}\sum_{t=1}^{T} \|f_t(\hat{Y}_t) - f_t(Y_t)\|^2 + \eta|\partial_x * (f_t(\hat{Y}_t) - f_t(Y_t))| + \eta|\partial_y * (f_t(\hat{Y}_t) - f_t(Y_t))|, \quad (6)$$

where ∂_x and ∂_y are horizontal and vertical gradient filters.

We train our model using the Adam optimizer [18], beginning with a learning rate of 10^{-4}, and going through two learning rate drops every time validation loss saturates, for a total of roughly 1.5 million iterations.

4. Experiments

We now describe experiments evaluating our approach, and comparing it to baseline methods for both denoising without a flash input, and to applying existing network architectures to a flash and no-flash pair. We also include ablations describing the effect of our kernel interpolation approach, and of choosing the no-flash vs. flash image as geometric reference.

4.1. Preliminaries

Dataset. We use the dataset of Aksoy *et al*. [1], which contains 16-bit well-exposed ambient-only and flash-only image pairs. These images were crowdsourced from users who were asked to capture images with hand-held mobile phones in real-world settings, and roughly had a 0.5-1 second delay between captures of the pair. We split the dataset as follows: 2519 images for the training set, 128 for validation and 128 for testing, considering 440×440 crops (random crops for training, and fixed central crops for validation and testing). We simulate a real low-light capture by dimming the linear ambient-only image by dividing with a random factor in $[2, 50]$, sampled uniformly in the log domain. This forms our *no-flash* input. We increase the exposure of the flash-only image by a constant factor 2, and add it to the no-flash input to obtain our *flash* input.

Misalignment and simulated noise. The image pairs provided by [1] were automatically aligned by finding correspondences with feature matching. Since this would be unrealistic in low-light images, we undo such alignment by warping the no-flash or flash image (the other is the reference) with a random homography. To obtain the homography parameters, we assume the camera's FOV is 90 degrees to get its intrinsic matrix. We perturb it with a random 3D rotation uniformly sampled in the range $[-0.5, 0.5]$ degrees in each axis, followed by random 2D scaling by a factor uniformly sampled in $[0.98, 1.02]$, and a random 2D translation of $[0, 2]$ pixels. The overall average per-pixel displacement between our flash and no-flash inputs ranges up to 20 pixels (Manhattan distance). Note that real-world non-idealities like parallax, occlusions, blur, etc. originally present in the data are preserved by undoing [1]'s alignment.

We use the same noise parameters for the flash and no-flash image, i.e., we assume they were captured with the same ISO setting. During training, we randomly sample the noise parameters σ_r and σ_s uniformly in the log-domain in the ranges: $\log(\sigma_r) \in [-3, -2]$ and $\log(\sigma_s) \in [-4, -2.6]$.

Losses and metrics. The preprocessing pipeline is executed on the original linear color space of the camera. To compute losses, we set the desired gain α^t in Sec. 3.3 to be the inverse of the factor we used to dim the image above, since the original images in [1] were well-lit. The database also includes a color transform matrix for each image which we use as C^t. We evaluate performance by computing PSNR and SSIM between the rendered versions of our estimate and the ground-truth.

Baselines. We compare to denoising without a flash input: using a single no-flash image denoised by a version of our architecture (without a scale map), and a burst of two (misaligned) no-flash inputs denoised using the state-of-the-art burst denoising method of [35] (which we refer to as BPN). For flash and no-flash image inputs, we compare our method to other standard architectures: a direct prediction network which simply regresses to the denoised output, and burst denoising methods KPN [27] and BPN [35] applied to the flash and no-flash pair. All of these methods were trained on our dataset, and provided information about noise standard deviation in an identical manner to our method. Further details are included in the supplement.

4.2. Evaluation

We begin by evaluating our method, choosing the ambient image as geometric reference (i.e., we assume $T_{nf}(n) = n$), on our test set of 128 images, and comparing it to the various baselines described above. We fix the noise level to $\log(\sigma_r) = -2.6$ and $\log(\sigma_s) = -3.6$, sample a random homography for each pair to be applied to the flash image (for the no-flash burst, this homography is applied to the second no-flash image), and repeat our evaluation with a discrete set of dimming factors: $[100, 50, 25, 12.5]$. Note that the factor 100 lies outside our training range, and demonstrates the robustness of our method.

Our method consistently outperforms all methods, regardless of the dimming factor, as seen by the quantiative results in Table 1. We also include example reconstructions in Fig. 4, where we see that our method reconstructs fine surface detail with higher fidelity than the other methods.

In Fig. 3, we take a closer look at the effect of misalignment. We take a subset of 64 flash and no-flash pairs

Method	100x Dimmed		50x Dimmed		25x Dimmed		12.5x Dimmed	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
No-flash input only								
Our Architecture	24.91 dB	0.779	27.23 dB	0.825	29.31 dB	0.865	30.98 dB	0.895
2-frame burst input (no flash)								
BPN [35]	25.58 dB	0.796	27.75 dB	0.839	29.65 dB	0.874	31.21 dB	0.899
Flash and no-flash input pair								
Direct Prediction	24.80 dB	0.773	27.06 dB	0.818	29.12 dB	0.857	30.84 dB	0.888
KPN [27]	25.87 dB	0.815	27.94 dB	0.852	29.69 dB	0.880	31.21 dB	0.901
BPN [35]	26.11 dB	0.815	28.04 dB	0.850	29.75 dB	0.880	31.21 dB	0.901
Ours	**26.75 dB**	**0.829**	**28.56 dB**	**0.860**	**30.14 dB**	**0.884**	**31.52 dB**	**0.903**

Table 1: **Quantitative results.** Thanks to the richer signal provided by the flash input, our method outperforms our single image denoising baseline, and a 2-frame burst denoising baseline. Comparisons to standard burst denoising approaches adapted to use flash–no-flash pairs show that our model architecture with its filtering/scale decomposition and larger kernels outperforms previous work. These results hold over a wide range of ambient light levels, shown here as dimming factors between the low-light no-flash input and a well-lit ground-truth target.

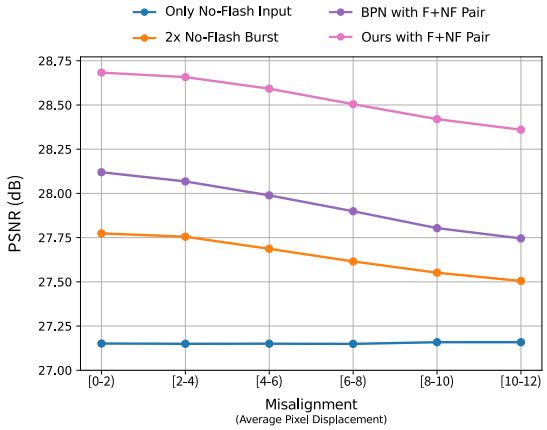

Figure 3: **Performancs vs. misalignment**. We show the performance profile of our method and select baselines as a function of average displacement between the two frames. Our model consistently delivers superior performance and is robust to large misalignment between its inputs.

from our test set (all dimmed with a factor of 50), and evaluate each set with different homographies that cause different average pixel displacements. We plot the PSNR of reconstruction by various methods for different degrees of displacement (for the single no-flash input baseline, these numbers are the same for all displacements). As expected, the accuracy of all methods decreases with greater misalignment. Nevertheless, we find that our method consistently outperforms all baselines, including the single no-flash input even with misalignment greater than 10 pixels.

Additional results, including results on different noise levels and combining flash denoising with burst denoising, can be found in the supplemental material.

Setting	**Flash Reference**	**No-Flash Reference**	**No-Flash w/o $\{B_j\}$**
100x dimmed	**26.83 dB**	26.75 dB	26.45 dB
50x dimmed	28.39 dB	**28.56 dB**	28.42 dB
25x dimmed	29.55 dB	**30.14 dB**	30.09 dB
12.5x dimmed	30.45 dB	**31.52 dB**	31.51 dB

Table 2: **Ablation study.** We compare the performance of our method to two ablations. One uses the flash image instead of the no-flash image as reference for the geometric transformation. The other uses a kernel basis without interpolation, leading to an effective kernel size of only 15×15.

4.3. Ablation

Section 3.1 considered two options for the alignment reference: with the output geometrically aligned with the flash input, or the no-flash image. In the previous section, we reported results with the no-flash input as the reference (for our method, as well as the other methods evaluated on the flash and no-flash pair). This was based on an evaluation of both alternatives, which we report in Table. 2.

We found that except for the lowest light level, the using the no-flash image as reference yields results that are quantiatively better (this is also true for the other baselines). However, looking at the actual reconstructions in Fig. 5, we find both images to be of similar visual quality—with the lower quantiative performance of the flash reference being largely due to slight, and largely imperceptible, alignment errors in low-frequency shading. However, as shown in ex-

Figure 4: **Qualitative comparison.** Our method uses flash/no-flash image pairs to denoise low-light images. It produces cleaner outputs than baseline flash/no-flash denoisers (*Direct (F+NF)*, *BPN (F+NF)*), as well as single-image (*Only No-Flash Input*) and burst denoisers (2× *No-Flash Burst*). We also visualize our intermediate filtered no-flash image and scale map.

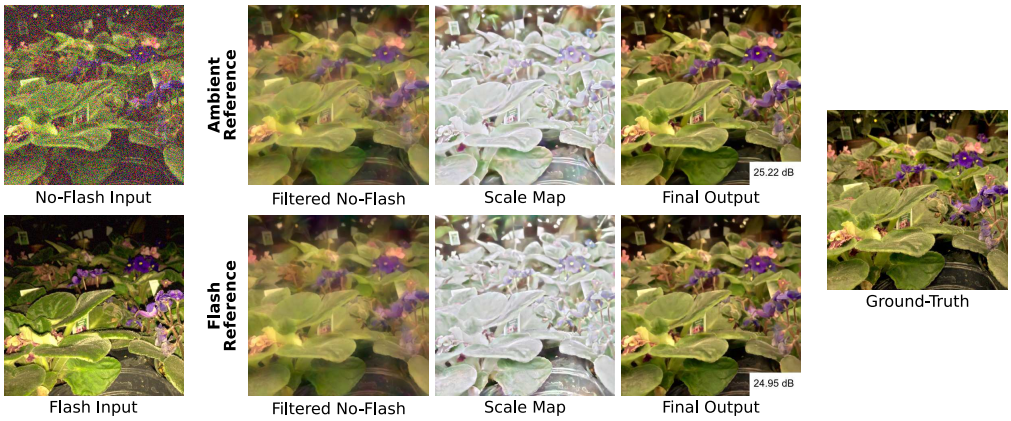

Figure 5: **Flash vs. no-flash as reference frame.** We use the ambient-only image as the reference frame for our reconstruction (*top*), i.e. the ground truth is aligned to the no-flash image. We found this choice leads to a lower error on average, compared to the alternative, using the flash as reference (*bottom*).

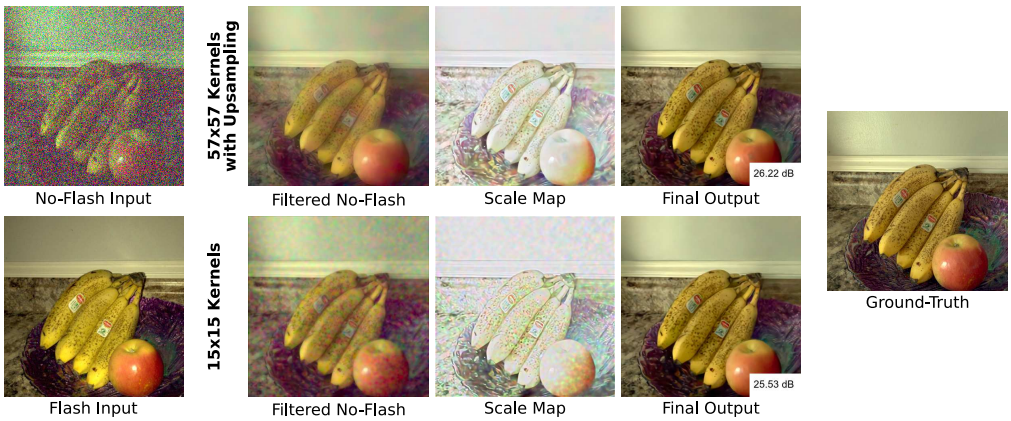

Figure 6: **Benefit of large kernels.** By using a 2-scale kernel decomposition, where the low-pass component is bilinearly upsampled, our model (*top*) can better denoise the ambient-only image. This leads to reduced residual chroma noise, which makes the scale map more effective at recovering fine details. Without it (*bottom*), the kernels are too small to effectively denoise the ambient image, so the scale map needs to compensate for the residual mid-frequency noise.

panded analysis in the supplement, using the flash image as reference sometimes yields visually sharper results

Table 2 also evaluates the benefit of using larger filters though our interpolation-based approach. We find that by allowing filters with a larger footprint (57×57), our two-scale kernel basis improves denoising quality, especially at low light levels. As show in Figure 6, large kernels yield a smoother filtering of the noisy no-flash image, so that the flash-driven scale map does not need to overcompensate for residual mid-frequency color noise, leading to better reconstructions in the final output.

5. Conclusion

This paper introduced a method to effectively leverage the unique mix of visual information available in a flash and no-flash image pair, and produce high-quality images in low-light environments. Our method preserves the warmth and colors of the ambient lighting while bringing out fine details thanks to the flash image. Drawing on traditional flash/no-flash techniques, our network architecture assembles its output from a filtered ambient-only image, and a scale map that encoded high-frequency details from the flash. Although it was not trained with any intermediate supervision, we found our network automatically learns to carry out both the necessary geometric alignment between the frames, and the photometric transfer needed to produce state-of-the-art reconstructions. Still, there remain situations where flash photography may be too obtrusive. Exploring how our model would fare with dark flash imaging [20, 32] is an interesting avenue for future research.

Acknowledgments. ZX and AC acknowledge support from NSF award IIS-1820693, and a gift from Adobe Research.

References

[1] Yagiz Aksoy, Changil Kim, Petr Kellnhofer, Sylvain Paris, Mohamed A. Elgharib, Marc Pollefeys, and Wojciech Matusik. A dataset of flash and ambient illumination pairs from the crowd. In *Proc. ECCV*, 2018. 5

[2] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A non-local algorithm for image denoising. In *Proc. CVPR*, 2005. 2

[3] Harold C Burger, Christian J Schuler, and Stefan Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *Proc. CVPR*, 2012. 2

[4] Xu Cao, Michael Waechter, Boxin Shi, Ye Gao, Bo Zheng, and Yasuyuki Matsushita. Stereoscopic flash and no-flash photography for shape and albedo recovery. In *Proc. CVPR*, 2020. 3

[5] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proc. CVPR*, 2018. 2

[6] Qifeng Chen, Jia Xu, and Vladlen Koltun. Fast image processing with fully-convolutional networks. In *Proc. ICCV*, 2017. 3

[7] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Color image denoising via sparse 3d collaborative filtering with grouping constraint in luminance-chrominance space. In *Proc. ICIP*, 2007. 2

[8] David L. Donoho. De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627, 1995. 2

[9] Elmar Eisemann and Frédo Durand. Flash photography enhancement via intrinsic relighting. *ACM Transactions on Graphics (TOG)*, 23(3):673–678, 2004. 1, 2, 4

[10] Alessandro Foi, Mejdi Trimeche, Vladimir Katkovnik, and Karen Egiazarian. Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008. 3

[11] Michaël Gharbi, Jiawen Chen, Jonathan T. Barron, Samuel W. Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG)*, 36(4):118:1–118:12, 2017. 3

[12] Clément Godard, Kevin Matzen, and Matt Uyttendaele. Deep burst denoising. In *Proc. ECCV*, 2018. 2

[13] Qiang Guo, Caiming Zhang, Yunfeng Zhang, and Hui Liu. An efficient svd-based method for image denoising. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(5):868–880, 2016. 2

[14] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics (TOG)*, 35(6):192, 2016. 2

[15] Felix Heide, Steven Diamond, Matthias Nießner, Jonathan Ragan-Kelley, Wolfgang Heidrich, and Gordon Wetzstein. Proximal: Efficient image optimization using proximal algorithms. *ACM Transactions on Graphics (TOG)*, 35(4):84, 2016. 2

[16] Felix Heide, Markus Steinberger, Yun-Ta Tsai, Mushfiqur Rouf, Dawid Pająk, Dikpal Reddy, Orazio Gallo, Jing Liu, Wolfgang Heidrich, Karen Egiazarian, et al. Flexisp: A flexible camera image processing framework. *ACM Transactions on Graphics (TOG)*, 33(6):231, 2014. 2

[17] Zhuo Hui, Kalyan Sunkavalli, Sunil Hadap, and Aswin C. Sankaranarayanan. Illuminant spectra-based source separation using flash photography. In *Proc. CVPR*, 2018. 3

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[19] Filippos Kokkinos and Stamatis Lefkimmiatis. Iterative residual cnns for burst photography applications. In *Proc. CVPR*, 2019. 2

[20] Dilip Krishnan and Rob Fergus. Dark flash photography. *ACM Transactions on Graphics (TOG)*, 28(3):96, 2009. 2, 8

[21] Huibin Li and Feng Liu. Image denoising via sparse and redundant representations over learned dictionaries in wavelet domain. In *Proc. International Conference on Image and Graphics (ICIG)*, 2009. 2

[22] Yijun Li, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep joint image filtering. In *Proc. ECCV*, 2016. 2

[23] Michael Lindenbaum, M. Fischer, and Alfred M. Bruckstein. On gabor's contribution to image enhancement. *Pattern Recognit.*, 27(1):1–8, 1994. 2

[24] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *Advances in Neural Information Processing Systems*, pages 1680–1689, 2018. 2

[25] Ziwei Liu, Lu Yuan, Xiaoou Tang, Matt Uyttendaele, and Jian Sun. Fast burst images denoising. *ACM Transactions on Graphics (TOG)*, 33(6):232, 2014. 2

[26] Talmaj Marinč, Vignesh Srinivasan, Serhan Gül, Cornelius Hellge, and Wojciech Samek. Multi-kernel prediction networks for denoising of burst images. In *Proc. ICIP*, 2019. 2

[27] Ben Mildenhall, Jonathan T Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. In *Proc. CVPR*, 2018. 2, 3, 5, 6

[28] Pietro Perona and Jitendra Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, 12(7):629–639, 1990. 2

[29] Georg Petschnigg, Richard Szeliski, Maneesh Agrawala, Michael Cohen, Hugues Hoppe, and Kentaro Toyama. Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics (TOG)*, 23(3):664–672, Aug. 2004. 1, 2, 4

[30] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992. 2

[31] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Proc. ICCV*, 1998. 2

[32] Jian Wang, Tianfan Xue, Jonathan T. Barron, and Jiawen Chen. Stereoscopic dark flash for low-light photography. In *Proc. ICCP*, 2019. 2, 8

[33] Bartlomiej Wronski, Ignacio Garcia-Dorado, Manfred Ernst, Damien Kelly, Michael Krainin, Chia-Kai Liang, Marc Levoy, and Peyman Milanfar. Handheld multi-frame super-resolution. *ACM Transactions on Graphics (TOG)*, 38(4):1–18, 2019. 2

[34] Zhihao Xia and Ayan Chakrabarti. Identifying recurring patterns with deep neural networks for natural image denoising. In *Proc. WACV*, 2020. 2

[35] Zhihao Xia, Federico Perazzi, Michaël Gharbi, Kalyan

Sunkavalli, and Ayan Chakrabarti. Basis prediction networks for effective burst denoising with large kernels. In *Proc. CVPR*, 2020. 2, 3, 4, 5, 6

[36] Junyuan Xie, Linli Xu, and Enhong Chen. Image denoising and inpainting with deep neural networks. In *Advances in Neural Information Processing Systems*, pages 341–349, 2012. 2

[37] Qiong Yan, Xiaoyong Shen, Li Xu, Shaojie Zhuo, Xiaopeng Zhang, Liang Shen, and Jiaya Jia. Cross-field joint image restoration via scale map. In *Proc. ICCV*, 2013. 2

[38] Leonid P Yaroslavsky. Digital picture processing: an introduction. *Applied Optics*, 25(18):3127, 1986. 2

[39] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 2

[40] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. In *Proc. CVPR*, 2017. 2

[41] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. In *Proc. ICLR*, 2018. 2

[42] Shaojie Zhuo, Dong Guo, and Terence Sim. Robust flash deblurring. In *Proc. CVPR*, 2010. 3