# NOISE AND DYNAMIC RANGE OPTIMAL COMPUTATIONAL IMAGING

*Kalpana Seshadrinathan*[1], *Sung Hee Park*[1,2] *and Oscar Nestares*[1]

[1]Intel Labs, Santa Clara, CA, USA.
[2]Department of Electrical Engineering, Stanford University, CA, USA.

**ABSTRACT**

Computational photography techniques overcome limitations of traditional image sensors such as dynamic range and noise. Many computational imaging techniques have been proposed that process image stacks acquired using different exposure, aperture or gain settings, but far less attention has been paid to determining the parameters of the stack automatically. In this paper, we propose a novel computational imaging system that automatically and efficiently computes the optimal number of shots and corresponding exposure times and gains, taking into account characteristics of the scene and sensor. Our technique seamlessly integrates the use of multiple capture for both High Dynamic Range (HDR) imaging and denoising. The acquired images are then aligned, warped and merged in the raw Bayer domain according to a statistical noise model of the sensor to produce an optimal, potentially HDR and denoised image. The result is a fully automatic camera that constantly monitors the scene in front of it and decides how many images are required to capture it, without requiring the user to explicitly switch between different capture modalities.

***Index Terms***— HDR, denoising, SNR, mobile imaging.

## 1. INTRODUCTION

Digital cameras acquire images of the natural world that often do not do justice to capturing the scene due to limitations of the camera acquisition system. For example, dynamic range of natural scenes often exceeds the dynamic range of the image sensor resulting in loss of detail and camera images often suffer from significant noise. Many computational imaging techniques in the literature attempt to compensate for these drawbacks by capturing and merging image stacks for High Dynamic Range (HDR) Imaging [1, 2]. Recently several papers attempt to address optimal ways of acquiring these image stacks. Several methods determine the number and parameters of the optimal image stack based on estimates of the scene dynamic range [3, 4]. Other methods assume that the scene dynamic range are known and set up an optimization criterion based on SNR and/or capture time [5, 2]. These methods do not specify how the scene dynamic range can be estimated and do not consider the scene histogram in the optimization framework.

In this paper, we present a novel computational imaging system that captures the optimal image stack to represent the full dynamic range of the scene and reduce noise. Our system determines both the number of images that need to be acquired, and the corresponding exposure time and ISO gains, for optimum representation of the scene in terms of both dynamic range and noise. The number of images acquired can be one based on the scene, which is sufficient in many day-to-day situations. This case, corresponding to a traditional camera, does not suffer from many of the pitfalls of computational imaging such as capture/processing latency or visual distortions due to errors in the reconstruction. When multiple images are acquired, our system aligns them automatically to compensate for handshake. The aligned images are then merged in a noise optimal manner using a sensor noise model to create an HDR representation of the scene, which can be processed for display. Our primary contributions in this paper are estimation and utilization of scene and sensor specific information in determining the optimal image stack, a combined approach to HDR imaging and de-noising and a novel warping technique for images in the Bayer domain.

Section 2 presents the camera noise model and estimation of its parameters. The optimization criterion and algorithm is described in Section 3, followed by a description of alignment of multiple images to compensate for handshake in Section 4. We discuss reconstruction of the scene from multiple exposures in Section 5 and present images of real world scenes obtained using a prototype of our proposed system in Section 6. We conclude this paper in Section 7.

## 2. NOISE MODEL AND ESTIMATION

**Camera Noise Model**: Digital camera sensors suffer from different noise sources such as dark current, photon shot noise, readout noise and noise from analog to digital conversion (ADC) [6]. The noise model that we use is identical to the noise model proposed in [6], with the inclusion of ISO gain applied to the sensor similar to [5]. Let $I_j$ denote the number of electrons generated at a pixel site $j$ on the sensor per unit time, which is proportional to the incident spectral irradiance, resulting in $I_j t$ electrons generated over an exposure time $t$. We model the pixel value as a normal random variable $X_j$ :

$$X_j = (S_j + R)\alpha g + Q, S_j \sim N(I_j t + \mu_{\text{DC}}, I_j t + \mu_{\text{DC}})$$
$$R \sim N(\mu_R, \sigma_R^2), Q \sim N(\mu_Q, \sigma_Q^2) \quad (1)$$

Here, $N(\mu, \sigma^2)$ represents the normal distribution with mean $\mu$ and variance $\sigma^2$. $S_j$ is a random variable denoting the number of electrons collected at pixel site $j$, which is typically modeled using a Poisson distribution whose mean is determined by the number of electrons generated and the dark current $\mu_{\text{DC}}$ [6]. We approximate the Poisson distribution using a normal distribution, which is reasonable for a large number of electrons [2]. We do not model the dependence of dark current on exposure time and temperature and assume a constant value estimated using typical exposure times for outdoor photography. $R$ denotes the readout noise and $Q$ denotes noise from ADC, both of which are modeled as Gaussian random variables. $g$ denotes the ISO gain applied to the sensor and $\alpha$ denotes the combined gain of the camera circuitry and the analog amplifier in units of digital number per electron. We do not model pixel response non-uniformity in the sensor. Square of the signal to noise ratio (SNR) is

then defined as:

$$\text{SNR}(I_j, t, g)^2 = \frac{(I_j t \alpha g)^2}{(I_j t + \mu_{\text{DC}} + \sigma_R^2)\alpha^2 g^2 + \sigma_Q^2} \quad (2)$$

**Parameter Estimation**: We capture a number of raw (Bayer domain) dark frames $D(k)$ by covering the lens of the camera and using all the different ISO gains $g(k)$ permitted by the image sensor. We set the exposure time to 1/30 sec, which is in the upper ranges of typical exposure times used in outdoor photography. We then compute the sample mean $\hat{\mu}_D(k)$ and variance $\hat{\sigma}_D^2(k)$ of each of the dark frames. Setting $I = 0$ for dark frames, we have:

$$\hat{\mu}_D(k) = (\hat{\mu}_{\text{DC}} + \hat{\mu}_R)\hat{\alpha} g(k) + \hat{\mu}_Q \quad (3)$$

$$\hat{\sigma}_D^2(k) = (\hat{\mu}_{\text{DC}} + \hat{\sigma}_R^2)\hat{\alpha}^2 g(k)^2 + \hat{\sigma}_Q^2 \quad (4)$$

We treat Eq. (3) as a linear function of $g(k)$ with two unknown variables, $\hat{\mu}_0 = (\hat{\mu}_{\text{DC}} + \hat{\mu}_R)\hat{\alpha}$ and $\hat{\mu}_Q$, which we estimate using least squares (LS) estimation. Similarly, we estimate the unknown parameters, $\hat{\sigma}_0^2 = (\hat{\mu}_{\text{DC}} + \hat{\sigma}_R^2)\hat{\alpha}^2$ and $\hat{\sigma}_Q^2$, in Eq. (4) using LS estimation. To estimate $\hat{\alpha}$, we display flat field images at different brightness levels on an LCD monitor and image this with the camera sensor using different exposure times and ISO gains under dark ambient conditions. While neutral density filters placed in a uniform light source box or uniform reflectance cards may provide more accurate calibration [6], we found that reasonable calibration accuracy can be achieved using our simple method. Let $F(k)$ represent a flat field raw image acquired using an illumination level $I(k)$, exposure time $t(k)$ and ISO gain $g(k)$.

$$\hat{\mu}_F(k) = [\hat{\mu}_{\text{DC}} + \hat{\mu}_R + I(k)t(k)]\hat{\alpha} g(k) + \hat{\mu}_Q$$

$$\hat{\sigma}_F^2(k) = [\hat{\mu}_F(k) - g(k)\hat{\mu}_0 - \hat{\mu_Q}]\hat{\alpha} g(k) + g(k)^2\hat{\sigma}_0^2 + \hat{\sigma}_Q^2$$

We used 4 different exposure times, 4 ISO gains and 5 illumination levels to acquire a set of images for calibration. $\hat{\mu}_F(k)$ and $\hat{\sigma}_F^2(k)$ were estimated using the sample mean and variance of $F(k)$ in a central region of the frame that suffered from less than $1\%$ of vignetting distortion. Using previously estimated values of $\hat{\mu}_0$, $\hat{\mu}_Q$, $\hat{\sigma}_0^2$ and $\hat{\sigma}_Q^2$, we used a LS fitting procedure to estimate $\hat{\alpha}$. We also determine the digital value at which the sensor saturates, denoted using $S$, during calibration. We estimate $\text{SNR}(I_j, t, g)^2$ in Eq. (2) from the pixel value at site $j$ denoted as $x_j$ using:

$$\hat{\text{SNR}}(I_j, t, g)^2 = \frac{[x_j - \hat{\mu}_Q - g\hat{\mu}_0]^2}{\hat{\alpha} g(x_j - \hat{\mu}_Q - g\hat{\mu}_0) + g^2\hat{\sigma}_0^2 + \hat{\sigma}_Q^2}$$

## 3. CAPTURE OPTIMIZATION

**Estimation of scene dynamic range**: We first determine the dynamic range of the scene using a procedure similar to [4]. We continuously stream images from the sensor to determine a short exposure $(t_{\text{min}}, g_{\text{min}})$ and a long exposure $(t_{\text{max}}, g_{\text{max}})$ that determine the dynamic range of the scene. We compute the histogram of each image streaming from the sensor in the range $[0, h_{\text{max}}]$ that depends on the bit depth of the sensor. We define $(t_{\text{min}}, g_{\text{min}})$ as the exposure level where approximately $P_{\text{min}} = 95\%$ of the pixels are not saturated and fall below $Q_{\text{min}} = S/h_{\text{max}}$. To determine $(t_{\text{min}}, g_{\text{min}})$, we set an initial exposure of ($t_{\text{ini}}$ =30ms,$g_{\text{ini}} = 1$) and determine the $P_{\text{min}}$-percentile value denoted as $a_{\text{min}}$. If $a_{\text{min}} \notin [(P_{\text{min}}-0.5)/100, (P_{\text{min}}+0.5)/100]$, we change the total exposure, which is the product of the exposure time and ISO gain, by a factor $Q_{\text{min}}/(100 * a_{\text{min}})$ for the next iteration. We continue this procedure for a fixed number of iterations or until the convergence criterion is met. A similar procedure is used to determine $(t_{\text{max}}, g_{\text{max}})$, which is determined as the exposure level where approximately $P_{\text{max}}$=99% of the pixels in the image are bright and larger than $Q_{\text{max}} = 0.2 * h_{\text{max}}$.

**Estimation of scene histogram**: We estimate the histogram of $I$ with two raw images $x^{\text{min}}$ and $x^{\text{max}}$ captured using parameters $(t_{\text{min}}, g_{\text{min}})$ and $(t_{\text{max}}, g_{\text{max}})$. Note that $x^{\text{min}}$ and $x^{\text{max}}$ are two different estimates of $I$ obtained using different exposures and can be used to generate an estimate of $I$, whose histogram can then be calculated. However, these exposures may not be aligned due to the handheld camera and alignment is a computationally expensive operation. We hence estimate $I$ using $x^{\text{min}}$ and $x^{\text{max}}$ separately and combine the resulting histograms. $\hat{I}$ can be estimated from $x^{\text{min}}$ as $\hat{I}_j = (x_j^{\text{min}} - \hat{\mu}_Q - g_{\text{min}}\hat{\mu}_0)/\hat{\alpha} t_{\text{min}} g_{\text{min}}$. We generate the histogram of $I$ by averaging the two histograms within overlapping regions and using the histogram of the available exposure in non-overlapping regions. We computed the histogram $H_I(l)$ using $L = 256$ bins in the range $[1/\hat{\alpha} t_{\text{max}} g_{\text{max}}, h_{\text{max}}/\hat{\alpha} t_{\text{min}} g_{\text{min}}]$, with $I(l)$ denoting the left endpoints of the bins. We also experimented with computing the histogram on a logarithmic scale in this range and obtained very similar final results. The histogram generated using $x^{\text{min}}$ and $x^{\text{max}}$ may not be accurate for scenes whose dynamic range exceeds twice the usable dynamic range of the sensor.

**Optimization Criterion**: Using the estimated scene histogram, we perform a joint optimization for the number of shots and the exposure values of each shot using:

$$\arg\max_{N,(t_1,g_1,\ldots,t_N,g_N)} \sum_{k=1}^{N}\sum_{l=1}^{L} H_I(l)\text{SNR}[I(l), t_k, g_k]^2$$

$$\text{subject to} \sum_{l:\sum_{k=1}^{N}\text{SNR}[I(l),t_k,g_k]^2 > 10^{T/10}} H_I(l) > \beta \sum_{l} H_I(l)$$

We express the SNR of a merged shot as the sum of individual SNR's [5]. Further, we require that more than a fraction $\beta = 0.9$ of pixels be above a threshold of $T = 20$ dB in SNR. Due to difficulty in solving this optimization analytically, we use an iterative method and initialize the number of shots to 1 and the corresponding parameters to $(t_1 = t_{\text{max}}, g_1 = g_{\text{max}})$. We then do a coarse search about the total exposure value by multiplying it with factors of (1/8,1/4,1/2,2,4,8) to determine the value that maximizes the objective function. We then perform a fine search around this maximum value by repeatedly multiplying with $2^{0.25}$ until a local maximum is determined. If the constraint is not satisfied, we add another shot to the optimization and repeat this procedure until the maximum allowable number of shots. We allowed a maximum of 3 shots in our experiments to keep latency reasonable. The initial conditions were chosen to be $(t_1 = t_{\text{max}}, g_1 = g_{\text{max}})$ for $N = 1$, adding $(t_2 = t_{\text{min}}, g_2 = g_{\text{min}})$ for $N = 2$ and adding $(t_3 = \sqrt{t_{\text{min}} t_{\text{max}}}, g_3 = \sqrt{g_{\text{min}} g_{\text{max}}})$ for $N = 3$. We perform the optimization for the total exposure level, which is the product of exposure time and ISO gain. We assume a maximum exposure time of 60ms to avoid motion blur, beyond which the ISO gain is increased.

## 4. IMAGE ALIGNMENT AND WARPING

The raw images that are acquired using the parameters estimated in the optimization stage described in Section 3 need to be aligned to account for handshake. We use a robust, multi-resolution, gradient-based registration algorithm based on a modification of [7] that uses a pure 3D camera rotation model, which has proven accurate to compensate for camera shake. For alignment, we use a version of the green channel where the missing samples in the Bayer pattern are

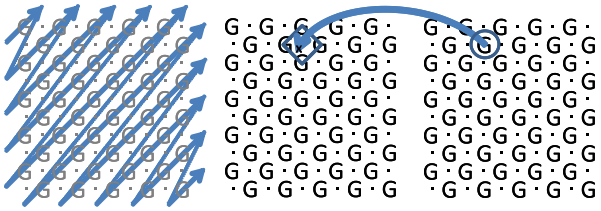

**Fig. 1**: Warping G channel in Bayer pattern using a grid rotated 45 deg: IIR filter scanning (left); neighbor selection in the original image (middle) to interpolate a given pixel in the warped image (right).

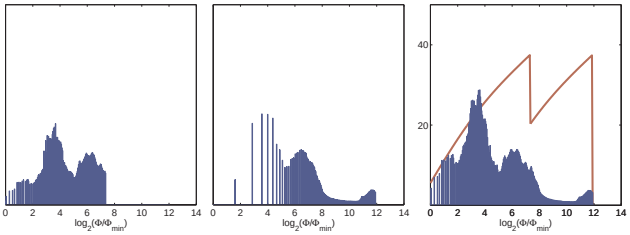

**Fig. 2**: Histograms of images obtained using the long and short exposure (left) and the SNR of the merged shot superimposed on the estimated scene histogram (right). These plots correspond to the images in Fig. 3.

linearly interpolated. Intensity conservation is achieved by normalizing the linear raw intensities by the exposure time and gain. A robust M-estimator is used to minimize the impact of outliers due to independently moving objects and local illumination changes.

Once the alignment has been estimated, the images are warped to a common frame of reference. We use backwards interpolation directly on the raw images arranged in a Bayer pattern to avoid demosaicing the input images which is computationally expensive and since we merge in the raw domain. We use shifted linear interpolation [8], which provides similar or superior quality to bicubic interpolation at reduced cost. Shifted linear interpolation starts preprocessing the image with a separable IIR filter (or its equivalent in 2D) that requires a regular sampling grid. Red (R) and blue(B) channels can be directly processed ignoring the missing samples in the Bayer pattern. For the green (G) channel we use a scanning strategy that is rotated by 45 deg to obtain a regular sampling grid, as is illustrated in Fig. 1 (left). We then apply bilinear interpolation with shifted displacements, which can be directly applied to the R and B channels. For the G channel, we again use the grid that is rotated by 45 deg to select the nearest neighbors needed for the interpolation as shown in Fig. 1 (middle and right).

## 5. SCENE RECONSTRUCTION AND PROCESSING

Let $\{x(k), k \leq N\}$ denote the aligned images with exposure parameters $\{[t(k), g(k)], k \leq N\}$. $x(k)$ corresponds to samples from a Gaussian distribution with mean and variance given by:

$$\mu_j(k) = \alpha g(k)(I_j t(k) + \mu_{\text{DC}} + \mu_R) + \mu_Q \quad (5)$$

$$\sigma_j(k)^2 = \alpha^2 g(k)^2 (I_j t(k) + \mu_{\text{DC}} + \sigma_R^2) + \sigma_Q^2 \quad (6)$$

Poisson shot noise and dark noise introduce a dependence between the mean and variance of the Gaussian distribution in the noise model, which makes it difficult to obtain a closed form solution to the maximum likelihood (ML) estimate of $I$ based on $x(k)$. Although iterative procedures can be used for this estimation [2], we simply estimate $\hat{\sigma}_j(k)^2$ using the pixel value to reduce computational cost:

$$\hat{\sigma}_j(k)^2 = \hat{\alpha} g(k)[x_j(k) - g(k)\hat{\mu}_0 - \hat{\mu}_Q] + g(k)^2 \hat{\sigma}_0^2 + \hat{\sigma}_Q^2$$

Considering $\mu_j(k)$ as a function of $I_j$, it is then easy to show that the maximum likelihood (ML) estimate of the HDR scene $I$ is:

$$\hat{I}_j = \frac{\sum_{k=1}^{N} [x_j(k) - g(k)\hat{\mu}_0 - \hat{\mu}_Q] \hat{\alpha} g(k) t(k) / \hat{\sigma}_j(k)^2}{\sum_{k=1}^{N} \hat{\alpha}^2 g(k)^2 t(k)^2 / \hat{\sigma}_j(k)^2} \quad (7)$$

To avoid saturated pixels while merging, we use the value at which the sensor saturates and set the weight to 0 for all four pixels in the Bayer 4-pack if any of them are saturated. This avoids creating color artifacts in sensors where the different color channels saturate at different levels. We also ensure that the darkest exposure is always used in the merging to handle the corner case where all exposures are saturated. The values of $\hat{I}$ lie in the range $[0, h_{\max}/\hat{\alpha}\min_k\{g(k)t(k)\}]$ and we convert the HDR image to 16-bit unsigned integer format by linearly re-scaling $\hat{I}$ to this range. We then process the HDR image using an HDR camera pipeline consisting of demosaicing, white balancing and color correction, tonemapping, followed by SRGB gamma transformation. We use a global tonemapping method for reasons of computational efficiency and use a sigmoid function to generate the tone curve. Let $L(I)$ denote the luminance of $I$. The tonemapped image $H$ is then obtained via $H = IL(H)/L(I)$ where

$$L'_j(H) = \frac{1}{1 + be^{-sL_j(I)}}, L_j(H) = \frac{L'_j(H) - L'_{\min}(H)}{L'_{\max}(H) - L'_{\min}(H)}$$

Here, $L'_{\min}(H)$ corresponds to a luminance of 1 and $L'_{\max}(H)$ corresponds to a luminance of $2^{16} - 1$. Values beyond the 5- and 95-percentile values of the histogram of $H$ determined by combining all the color channels are clipped and the remaining values are linearly expanded to the range [0,1]. Finally, the sRGB gamma transformation is applied to generate 8-bit images that can be displayed on standard monitors.

## 6. RESULTS

We implemented our proposed system on an Intel ATOM-based smartphone running the Android Gingerbread OS with an Aptina MT9E013 8MP, 1/3.2-inch CMOS image sensor. The following parameters were estimated for the noise model for this sensor: $\hat{\alpha}$=0.146, $\mu_0$=-0.040, $\mu_Q$=41.337, $\sigma_0^2$=0.133 and $\sigma_Q^2$=7.080. The initial stages of dynamic range and histogram estimation and optimization described in Section 3 were performed by streaming VGA (640x480) frames from the sensor to improve speed. The raw image captures were then performed using full image sensor resolution and the resulting images were aligned and merged.

Fig. 3 shows a case where two raw images were captured and merged to produce an HDR image. Fig. 2 shows the histograms acquired using the long and short exposures determined during scene dynamic range estimation and the estimated scene histogram. Fig. 2 shows that the optimal exposure levels determined by our algorithm are able to capture the scene histogram. Fig. 3 shows 8-bit images obtained by processing the individual images through a regular camera pipeline (demosaicing, white balancing, color correction, SRGB gamma conversion) and the merged image processed using our HDR camera pipeline. Since we perform the merging in the raw domain

**Fig. 3**: HDR Imaging: Two images captured using parameters of (0.002sec,1) and (0.042sec,1) (left) and the merged image (right).

**Fig. 4**: De-noising: One of three input images captured using parameters of (0.06sec,8) (left) and the merged result (right).

and create an HDR image, we can generate new representations of the scene for interaction and editing (cropping to region of interest, zooming, etc), which is particularly advantageous as mobile images are often transported to systems with varying screen sizes and computational power. Future image signal processors (ISP) should provide increased bit depth and facilitate accessing/merging of multiple raw images to accelerate such applications on a mobile device. Under low light conditions, our system captures multiple images to meet the required SNR criteria and effectively de-noises the merged image. If the lowest and highest exposure levels differ by a factor less than 2, we disable the tonemapping algorithm since in this case only de-noising is performed. Fig. 4 shows one of the input images (left) and the result of merging three images (right) under low light.

Our current implementation uses the ATOM CPU and does not use the Image Signal Processor (ISP), which can considerably speed up the method. We provide rough estimates of the processing time: Scene dynamic range estimation=3 sec, Scene histogram estimation=3 ms, optimization=5ms, alignment and warping=5 sec/image and merging=3 sec/image. Time required for scene dynamic range estimation is primarily limited by the number of frames used for metering. The speed of execution can be improved using flexible camera architectures such as FCam [9].The HDR camera pipeline is also implemented in software and we use a simple adaptive demosaicing available as part of the Intel Performance Primitives (IPP) library. We found that de-noising results can be sensitive to the choice of demosaicing function, as more advanced demosaicing algorithms that attempt to preserve color edges also tend to preserve noise.

## 7. CONCLUSIONS AND FUTURE WORK

We presented a computational imaging system that performs HDR imaging and de-noising under low light conditions. Our system automatically determines when multi-image acquisition and processing are necessary using an SNR-based optimization criterion. SNR is computed using a probabilistic model of the image sensor noise, whose parameters are estimated once offline. We demonstrated HDR and de-noising results obtained using a prototype implementation on a phone. In the future, we would like to conduct a more extensive evaluation of our system against other HDR/de-noising methods and tackle the problem of de-ghosting when independently moving objects are present in the scene.

## 8. REFERENCES


[1] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *Proc. SIGGRAPH*, 1997.

[2] M. Granados, B. Ajdin, M. Wand, C. Theobalt, H.-P. Seidel, and H. Lensch, "Optimal HDR reconstruction with linear digital cameras," in *IEEE CVPR*, 2010.

[3] N. Barakat, T. Darcie, and A. Hone, "The tradeoff between SNR and exposure-set size in HDR imaging," in *IEEE ICIP*, 2008, pp. 1848–1851.

[4] N. Gelfand, A. Adams, S. H. Park, and K. Pulli, "Multi-exposure imaging on mobile devices," in *Proc. ACM Multimedia*, 2010.

[5] S. Hasinoff, F. Durand, and W. Freeman, "Noise-optimal capture for high dynamic range photography," in *IEEE CVPR*, 2010.

[6] G. Healey and R. Kondepudy, "Radiometric CCD camera calibration and noise estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 3, pp. 267–276, 1994.

[7] O. Nestares and D. J. Heeger, "Robust multiresolution alignment of MRI brain volumes," *Magnetic Resonance in Medicine*, vol. 43, no. 5, pp. 705–715, 2000.

[8] T. Blu, P. Thévenaz, and M. Unser, "Linear interpolation revitalized," *IEEE Trans. Image Process.*, vol. 13, no. 5, pp. 710–719, May 2004.

[9] A. Adams, E.-V. Talvala, S. H. Park, D. E. Jacobs, B. Ajdin, N. Gelfand, J. Dolson, D. Vaquero, J. Baek, M. Tico, H. P. A. Lensch, W. Matusik, K. Pulli, M. Horowitz, and M. Levoy, "The frankencamera: an experimental platform for computational photography," in *ACM SIGGRAPH*, 2010.