Supplementary Material: A Light Transport Model for Mitigating Multipath Interference in Time-of-flight Sensors

Nikhil Naik[1,2] Achuta Kadambi[1] Christoph Rhemann[2] Shahram Izadi[2]
Ramesh Raskar[1] Sing Bing Kang[2]

[1]MIT Media Lab [2]Microsoft Research

{naik,kadambi,raskar}@mit.edu {chrheman,shahrami,sbkang}@microsoft.com

Supplemental Contents

This supplement is organized as follows:

- Section 1 describes the capture process and radiometric calibration (see line 478 of main document).
- Section 2 shows results of MPI correction after post-filtering (see line 583 of main document).
- Section 3 expands on the approximate global illumination model used to correct MPI (see line 731 of main document).
- Section 4 analyzes numerical stability (see line 743 of main document).

1. Capture Process

We perform separation of direct and global components of light transport using the method introduced by Nayar et al. [2]. We illuminate the scene using multiple high frequency patterns and the direct and global components of illumination are obtained from the acquired images using the Nayar method, which is a closed-form, per-pixel solution. We use the DLP Lightcommander projector in near-infrared mode to illuminate the scene using a 64×64 pixel checkerboard pattern. The pattern is shifted 5 times by 7 pixels each time in X and Y dimensions to capture a total of 25 images. We calculate two images α^+ and α^- from these patterns, by computing the maximum and minimum intensity values observed at each pixel across all 25 images. We also capture α_T, the total radiance, by projecting an all-white image on the scene. Specifically, the following equations are used to obtain the direct (α_D) and global(α_G) components:

$$\alpha_G = \frac{2}{1-b^2} \cdot (\alpha^- - b \cdot \alpha^+), \tag{1}$$

$$\alpha_D = \alpha_T - \alpha_G, \tag{2}$$

where b is the ratio between the brightness of the deactivated and activated state of the projector pixel. We set b to 0.08.

1.1. Radiometric Calibration

In our experimental setup, the internal light source of the Kinect One sensor is approximately 8 times brighter than the external projector. Since both sources project different amounts of radiance on the scene, the direct and global components obtained using the external projector (Equations 1, 2) need to be radiometrically corrected before they can be used for obtaining the corrected phase $\widehat{\varphi}_D$ in Equation 13 of the main text. To perform radiometric calibration, we first precompute a radiometric calibration mask and multiply images by this calibration mask before projection. The captured images are scaled with a constant to complete the radiometric correction process. We now provide details on computing the calibration mask.

978-1-4673-6964-0/15/$31.00 ©2015 IEEE

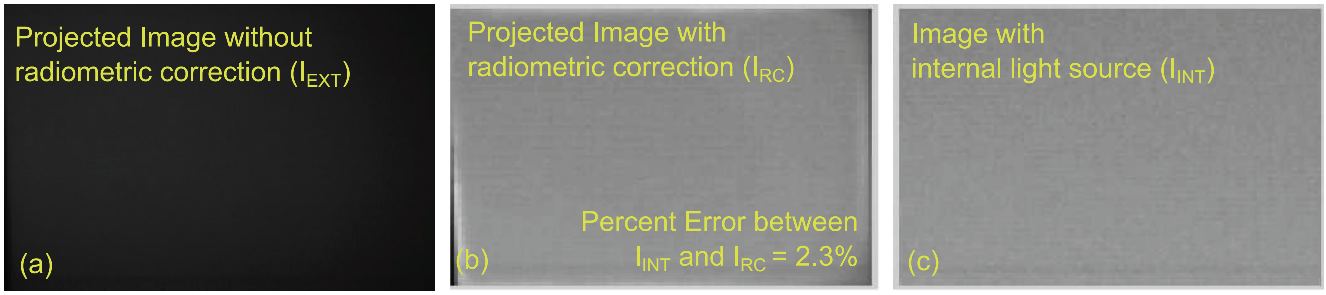
Projected Image without radiometric correction (I_{EXT})
(a)
Projected Image with radiometric correction (I_{RC})
Percent Error between I_{INT} and I_{RC} = 2.3%
(b)
Image with internal light source (I_{INT})
(c)

Figure 1. Intermediate images from radiometric correction. (a) A scene illuminated with the external projector without any radiometric correction. In (b) the radiometric corrected projection. Finally, in (c) we show the same scene with the Kinect light source. Note that the projected image in (a) is much dimmer than the observed image under Kinect lighting shown in (c).

Calibration Mask Computation To compute the calibration mask, we first obtain an image of a white plane using the Kinect TOF sensor with the internal source illuminating the scene (I_{INT}). Next we turn the internal source off and acquire another image (I_{EXT}) with the external projector illuminating the scene with an all-white image. We then establish pixel correspondence between the internal and external projector using gray codes. We project 10-bit gray codes and their complements, both in vertical and horizontal dimensions, capturing 40 images in total. Using the 20 images along X and 20 images along Y, we determine the pixel correspondences between the 1024×768 projector pixels and the much smaller image captured by the camera, whose size is determined by the distance of the scene from the capture hardware. Using these correspondences, we obtain the radiometric calibration mask $I_R = I_{INT}/I_{EXT}$. We repeat this exercise at three depth values (95, 100, and 105 cm) that approximately span the scene volume—we set I_R to the mean value across the three depths.

Next, we convert I_R to an 8-bit image as $I_{R8} = (\frac{255}{max(I_R)} \cdot I_R)$ for projection. We multiply every checkerboard pattern by I_{R8} before projection. The image acquired by the Kinect sensor is post-multiplied by a scalar factor $\frac{max(I_R)}{255}$ to obtain the final, radiometrically corrected image I_C. We validate the radiometric correction by measuring the percent intensity error between I_{INT} and I_C for the three white planes used to compute I_R and find that the mean absolute error is ~2.3%. After radiometric correction of all the projected checkerboard images, the direct (α_D) and global (α_G) components are used in Equation 13 of the main text.

It is necessary to synchronize the camera exposure and projector projection time to capture an 8-bit image. Since the Kinect One sensor does not allow synchronization with an external device, we project the 8 bit planes of I_{R8} separately and perform a weighted sum of these 8 images in post-capture.

Note that the radiometric calibration mask needs to be acquired only once for a given camera-projector setup. In a commercial implementation, the same light source can be used to project both the modulated illumination and high-frequency coded illumination needed for direct-global separation, thus eliminating the need for radiometric calibration.

2. Post-filtering for Smoothing

We pointed out in the main document that our MPI-corrected results were noisy and jagged as compared to the measured shape (see Figure 3 of the main document). In line 583 of the main document, we made the argument that our noisy results were preferred to the MPI-corrupted results. This is because the former can be enhanced with standard smoothing filters.

We now substantiate our claim. As illustrated in Figure 2, the measured shape distorts the geometry of the original model, while the MPI-corrected model imparts noise. The latter artifacts can be corrected. In Figure 2d, the corrected shape after bilateral filtering resembles the bowl.[1]

We must mention that such smoothing is a common step in today's depth sensing pipeline [1]. It would serve as an interesting follow-up to incorporate joint computation, where we combine our closed-form approach with spatial processing to obtain a higher quality result.

3. Approximating Global Light Transport

Recall from the main document that our model approximates the amplitude of all bounces of global light transport. Our results empirically substantiate the validity of this approximation.

[1] Bilateral filter parameters: half-window size 2, spatial domain standard deviation of 3 and intensity domain standard deviation of 0.1

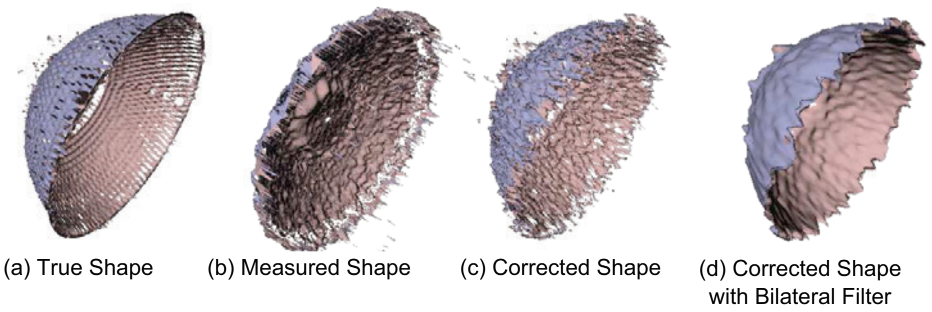

Figure 2. From left: (a) the true shape, (b) the measured, MPI-corrupted shape, (c) the result after our MPI-correction algorithm, and (d) bilateral filtering of our corrected shape in (c). The take-home point: after MPI-correction, noise can be smoothed. But without MPI-correction, a filter cannot recover the original geometry.

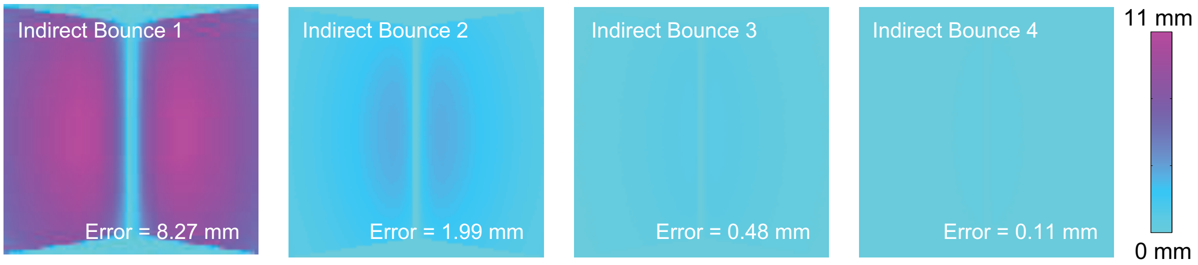

Figure 3. Here we use the raytracing software to simulate a 90 degree corner, and compute the error induced by different indirect bounces. Note that the contribution to RMSE of the fourth indirect bounce is just 1.3% of that of the first indirect bounce.

We now provide further intuition on this point. For generalized MPI, the measured phase takes the form

$$\widetilde{\varphi} = \arctan\left(\frac{\alpha_0 \sin\varphi_0 + \alpha_1 \sin\varphi_1 + \alpha_2 \sin\varphi_2 + ...}{\alpha_0 \cos\varphi_0 + \alpha_1 \cos\varphi_1 + +\alpha_2 \cos\varphi_2 + ...}\right), \tag{3}$$

where a_i and φ_i correspond to the amplitude and phase from the i-th optical bounce. The amplitude of second and higher indirect bounces $(\alpha_1, \alpha_2, ...)$ diminishes rapidly due to a combination of surface absorbance, the inverse square law, and Lambert's law. As a result, the contribution of higher bounces to the measured phase (Equation 3) reduces rapidly for most scenes.

We substantiate this claim by simulating MPI in presence of different number of global bounces for a 90 degree corner scene. Figure 3 illustrates error plots from four indirect bounces from the corner scene with associated errors. Here, the listed error corresponds to the error in the measured shape introduced by the associated bounce. For instance, the first indirect bounce induces an error of 8.27 mm to the measured shape. In contrast, the fourth indirect bounce induces an error of only 0.11 mm to the measured shape. Note that the contribution to RMSE of the fourth indirect bounce is just 1.3 percent of that of the first indirect bounce.

In summary, this explains why our model, which aggregates the amplitude of all indirect bounces, allows us to recover a close approximation of the true direct phase for a wide variety of natural scenes. Our method is not restricted to scenes with sparse reflections (see main document for results from subsurface scattering).

4. Numerical Stability

In line 743 of the main document, we deferred analysis of numerical stability to the supplemental material. Here, we use perturbation theory to analyze the conditioning of our closed-form solution. Recall from the main document that the closed-form solution to recover the direct phase is written as

$$\widehat{\varphi}_D = \arctan\left(\frac{\alpha_D\gamma + \alpha_G\left(\sin\left(\varphi_D - \varphi_G\right) + \gamma\cos\left(\varphi_D - \varphi_G\right)\right)}{\alpha_D + \alpha_G\left(\cos\left(\varphi_D - \varphi_G\right) - \gamma\sin\left(\varphi_D - \varphi_G\right)\right)}\right), \tag{4}$$

where $\gamma = \tan\left(\widetilde{\varphi}\right)$. We note that $\alpha_G = \left(1 - \alpha_D\right)$ and $\left(\varphi_D - \varphi_G\right) = \left(\frac{4\pi f_{\mathrm{M}}}{c} \cdot dz\right)$, where dz is the difference in depth

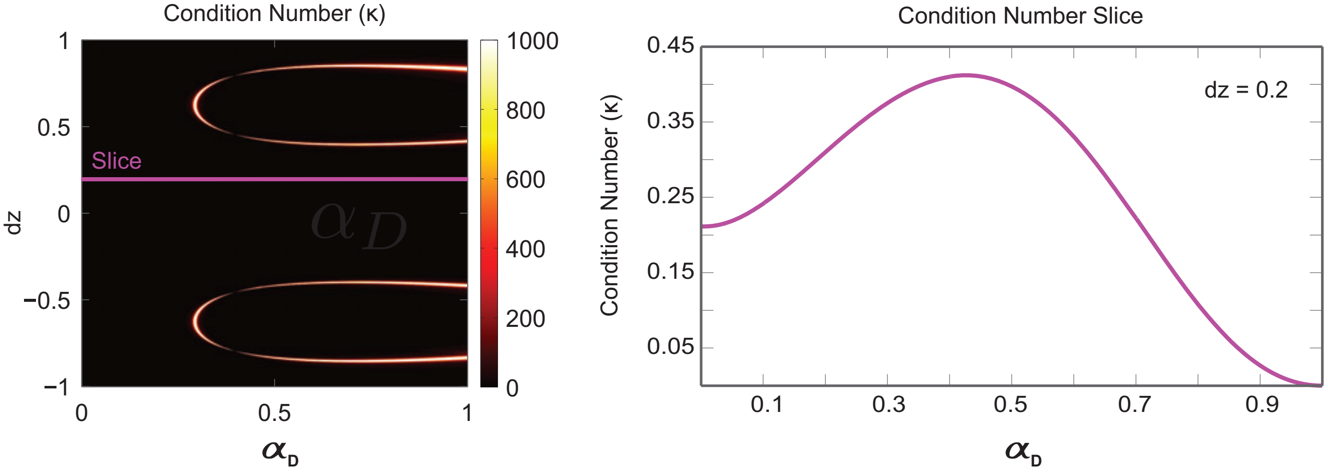

Figure 4. Analyzing numerical stability. We compute the condition number (see Equation 7) and plot its value for various values of α_D and dz. Except for the two symmetric bands with high condition number at large values of dz, the system is stable. At right, a horizontal slice of the condition number matrix, where $dz = 0.2$.

between the scene point contributing the direct bounce and the scene point contributing the indirect bounce. Therefore, this system is sensitive to changes in α_D and dz. We perform perturbation analysis of the condition number of the function using these two arguments. The condition number of a multivariate function $f(\mathbf{x})$ wrt. $\mathbf{x}$ is given by [3]

$$\kappa = \frac{\|J(\mathbf{x})\|}{\|f(\mathbf{x})\|/\|\mathbf{x}\|}. \tag{5}$$

We evaluate this formula for $\widehat{\varphi}_D$, computing the norm of the Jacobian as

$$\|J(\mathbf{x})\| = \sqrt{\left(\frac{\frac{4\pi f_{\mathrm{M}}}{c}(1-\alpha_D)\left((1-\alpha_D)-\alpha_D\cos(\frac{4\pi f_{\mathrm{M}}}{c}dz)\right)}{2\alpha_D^2+2(1-\alpha_D)(\cos(\frac{4\pi f_{\mathrm{M}}}{c}dz)+1)-1}\right)^2+\left(\frac{\sin(\frac{4\pi f_{\mathrm{M}}}{c}dz)}{2\alpha_D^2+2(1-\alpha_D)(\cos(\frac{4\pi f_{\mathrm{M}}}{c}dz)+1)-1}\right)^2}. \tag{6}$$

We replace $(1-\alpha_D)$ by α_G for brevity and simplify further to obtain the condition number κ, which takes the form of

$$\kappa = \frac{\sqrt{\alpha_D^2+dz^2}\left(\frac{(\frac{4\pi f_{\mathrm{M}}}{c})\alpha_G\left(\alpha_D\cos\left(\frac{4\pi f_{\mathrm{M}}}{c}dz\right)+\alpha_G\right)}{2\alpha_D^2+2\alpha_G(\cos(\frac{4\pi f_{\mathrm{M}}}{c}dz)+1)-1}\right)}{\arctan\left(\frac{\alpha_D\gamma+\alpha_G\left(\sin\left(\frac{4\pi f_{\mathrm{M}}}{c}dz\right)+\gamma\cos\left(\frac{4\pi f_{\mathrm{M}}}{c}dz\right)\right)}{\alpha_D+\alpha_G\left(\cos\left(\frac{4\pi f_{\mathrm{M}}}{c}dz\right)-\gamma\sin\left(\frac{4\pi f_{\mathrm{M}}}{c}dz\right)\right)}\right)}. \tag{7}$$

Although this is the simplified form of the condition number, it is too complex to cast immediate intuition. To obtain a sense of empirical intuition we plug in reasonable, physical values for the parameters and compute κ. In particular, we vary values of α_D between its physical limits of 0 and 1, and dz between -1 and 1 meters. We assume the following imaging scenario: a scene point one meter away, captured with Kinect camera parameters (e.g. f_{M} = 120 MHz).

The results are shown in Figure 4. On the left hand side we plot κ as a 2-D matrix for different entries of dz and α_D. Note that the condition number is low throughout the entire plot, except for the loop regions. These loops occur, because the denominator of the Jacobian norm reaches extremely low values, which causes the condition number to inflate. On the right hand side of Figure 4, we plot a slice through the matrix plot, for $dz = 0.2$, which is a physically reasonable value. Note that the condition number is less than 1 throughout this region, which indicates a well-conditioned problem.

In summary, our closed-form solution is well-conditioned, which is reflected in the quality of our results.

References

[1] A. Kadambi, A. Bhandari, and R. Raskar. 3d depth cameras in vision: Benefits and limitations of the hardware. In *Computer Vision and Machine Learning with RGB-D Sensors*, pages 3–26. Springer, 2014.

[2] S. K. Nayar, G. Krishnan, M. D. Grossberg, and R. Raskar. Fast separation of direct and global components of a scene using high frequency illumination. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 935–944. ACM, 2006.

[3] L. N. Trefethen and D. Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.