Supplementary Document for "Compressive Single-Photon 3D Cameras"

Felipe Gutierrez-Barragan, Atul Ingle, Trevor Seets, Mohit Gupta, Andreas Velten
University of Wisconsin-Madison
{fgutierrez3, ingle, seets}@wisc.edu, mohitg@cs.wisc.edu, velten@wisc.edu

S. 1. Coding Matrix Properties

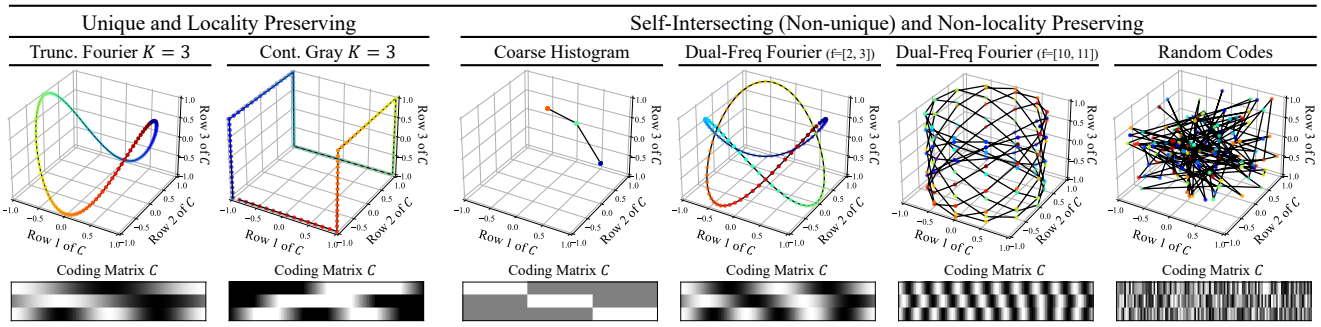
In this section we provide additional intuition for the coding matrix properties discussed in Section 4.1 of the main paper.

S. 1.1. A Geometric Perspective

Recall that the i^{th} column of the coding matrix, C, can be interpreted as a code word of length K that represents the i^{th} time bin. We can view this code word vector as a point in a K-dimensional space. Moreover, consider the curve that is traced by the N points (columns of C), denoted as the coding curve $\mathcal{C}$. In other words, the rows of C parameterize the coding curve $\mathcal{C}$, as illustrated in Suppl. Fig. 1.

Uniqueness Property (non self-intersecting): Suppl. Fig. 1 shows different coding curves parameterized by the coding matrices below them. Assuming a direct-only reflection model, i.e., $\Phi(t) = a\delta(t - t_z) + \Phi^{\text{bkg}}$, each point in the curve ($K \times 1$ code word vector) will correspond to the projection of a different time-shift t_z [5]. Therefore, the uniqueness property guarantees that each time-shift will map to a different point. If two or more different time-shifts map to the same point in the coding curve, that means the coding scheme will have ambiguities when estimating depths. For instance, as observed in the coarse histogram and dual-freq Fourier coding curves in Suppl. Fig. 1, the curves are self-intersecting at multiple points, which means that those points correspond to more than one true depth value.

Robustness Property (locality preserving): If we add a small perturbation to a point on a locality preserving coding curve, the perturbed point may map to a neighboring point *along* the curve. In Suppl. Fig. 1, neighboring points along the curve will have similar colors. For instance, in the Truncated Fourier curve, if we add a perturbation to a "reddish" point, then its nearest neighbor is likely to be another "reddish" point. On the other hand, a similar perturbation added to certain red points in the Dual-Freq Fourier curves may cause them to map to a blue point. These perturbations are similar to adding noise to the coded projection of the ground truth histogram, and mapping to non-neighboring points is similar to estimating a very different depth from the true depth. Hence, coding matrices with non-locality preserving curves are less robust to noise.



Supplementary Figure 1. **Example Coding Curves.** A coding curve is formed by plotting each column of the $K \times N$ coding matrix in K dimensional space. This figure shows examples with $K = 3$. Observe that unlike dual-frequency Fourier coding and random coding, truncated Fourier and Gray coding provide non-self-intersecting and locality preserving coding curves. This implies that the latter methods have a unique one-to-one mapping between the true (unknown) depth values and the coded measurements, and small perturbations (due to noise) do not cause large jumps in the compressed measurements.

S. 1.2. Coding Function Bandwidth Selection

The band-limit and indirect reflection properties, discussed in Sec. 4.1 of the main paper, give us a heuristic for choosing the frequency content of our coding functions (i.e., rows of C). We begin by analyzing the band-limit property.

Band-limit Property: Consider the discrete Fourier series of the system IRF, $h = (h_i)_{i=0}^{N-1}$, whose maximum harmonic is $b < N$.

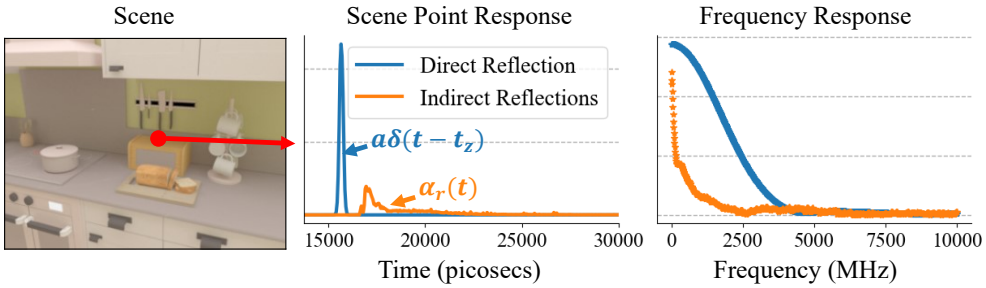
$$h_i = \sum_{k=0}^{b} H_k \exp\left(\frac{\mathrm{j}\, 2\pi k i}{N}\right) \tag{1}$$

where $\mathrm{j} = \sqrt{-1}$, $0 \leq i < N-1$, and H_k are the Fourier coefficients. Moreover, let $c = (c_i)_{i=0}^{N-1}$ be one coding function of C. The band-limit property says that c should not only be composed of frequencies larger than the system's band-limit (i.e., b). If c is only composed of frequencies that are not contained in b, the inner product of c and h will be zero, due to the orthogonality of sinusoids [12]. Hence, the coded measurement captured by c will not capture any information from the histogram photon flux waveform.

Indirect Reflection Property: Let $\alpha(t)$ be the impulse response of a point in the scene. Under the direct-only assumption, $\alpha(t) = a\delta(t - t_z)$, the impulse response is just the time-shifted direct reflection of that point. In the presence of diffuse indirect reflections, $\alpha(t)$ can be written as the sum of the direct reflection and the diffuse indirect reflections:

$$\alpha(t) = a\delta(t - t_z) + \alpha_r(t) \tag{2}$$

where $\alpha_r(t)$ are the diffuse indirect reflections. Suppl. Fig. 2 shows an example of a scene point impulse response with diffuse indirect reflections (orange line). As observed in the frequency response, the indirect reflections have a much smaller bandwidth than the direct reflection. Beyond 1 GHz, the direct reflection is the dominant component. Therefore, for coding functions mainly composed of frequencies above 1 GHz, the direct-only assumption continues to apply because the indirect component for frequencies above the threshold will be near 0. In general, the threshold is scene dependent [4], but scenes with relatively similar scales may have similar thresholds.



Supplementary Figure 2. **Diffuse Indirect Reflections Example.** Observe that the direct reflection has a narrow support in time and hence has much wider spectrum compared to the indirect response which, in this scene, does not go beyond 1 GHz.

S. 2. ZNCC Depth Estimation

Given a CSPH, $\widehat{B}$, acquired with a coding matrix C, we want to estimate the depth encoded by $\widehat{B}$. There exists a general depth decoding algorithm based on the zero-mean normalized cross-correlation (ZNCC) between $\widehat{B}$ and C, originally proposed for structured light [1, 9], and later applied to iToF imaging [6, 7]. Assuming a direct-only model, ZNCC decoding becomes optimal as noise goes to zero [9]. Moreover, even when the direct-only assumption does not hold, ZNCC continues to provide robust depth estimates [1, 6, 9].

Band-limited ZNCC Decoder [9]: Let C^h be the band-limited coding matrix, whose kth row, is defined as:

$$C^h_{k,i} = (h * C_{k,:})_i \tag{3}$$

where h is the system IRF, and $C_{k,:}$ is the kth row of the coding matrix. Let $C^h_{:,i}$ be the ith column of C^h. We can estimate depths, by finding $C^h_{:,i}$ that produces the maximum ZNCC with $\widehat{B}$ as follows:

$$\widehat{t_z} \propto \arg\max_i \frac{C^h_{:,i} - \text{mean}(C^h_{:,i})}{\left\|C^h_{:,i} - \text{mean}(C^h_{:,i})\right\|} \cdot \frac{\widehat{B} - \text{mean}(\widehat{B})}{\left\|\widehat{B} - \text{mean}(\widehat{B})\right\|} \tag{4}$$

Depths can be estimated from the time-shift t_z. Note that, although, we capture the CSPH using the input C matrix, we estimate distances using the band-limited coding matrix C^h. Therefore, Suppl. Eq. 4 assumes the system's IRF, h, is obtained in a prior one-time calibration step. Furthermore, if all coding functions in C are zero-mean, like the Fourier codes in Suppl. Fig. 1, then the above algorithm can be implemented as a normalized cross-correlation (NCC), where we do not need to subtract the mean in the numerator or denominator [9].

Suppl. Eq. 4 accounts for the system IRF (h) when estimating depths by using C^h instead of C. Not accounting for the IRF is equivalent to assuming that the system IRF is a dirac-delta function. In our experiments we find that not accounting for h can reduce the performance and sometimes introduce systematic depth errors in coding schemes that contain coding functions with frequency content outside of the band-limit of h.

Depth Estimation for Full-Resolution Histogram: To compute depths for FRHs we used matched filtering [16]. Interestingly, matched filtering is similar to applying Eq. **??** when C is an $N \times N$ identity matrix.

S. 3. Monte Carlo Simulations and Isometric Compression Analysis

In this section we provide further details on the Monte Carlo simulations performed to quantify the performance of each coding scheme. We also present additional mean and median depth errors results and isometric compression results for a wide range of SBR and photon count levels at various compression ratios. Finally, we present a similar analysis but for the case of a wide Gaussian pulse.

S. 3.1. Quantifying Coding Scheme Performance

A high-performance coding scheme should be able to provide high depth precision for the full depth range and at a wide range of signal and background light levels. One way to quantify the performance of a coding scheme is to compute the *mean expected absolute depth error* (MDE) [5, 7] at a wide range of signal and background light levels. In this paper we report the relative MDE, which is the MDE metric divided by the maximum possible depth value. Moreover, since depths are proportional to time shifts, the relative depth errors will be equivalent to relative time shift errors, hence we use these interchangeably when computing the relative MDE.

Monte Carlo MDE Computation: For a given coding scheme, fixed average photon counts ($\sum_{i=0}^{N} \Phi_i$), signal-to-background ratio (SBR $= \frac{\sum_{i=0}^{N} \Phi_i^{\text{sig}}}{N \Delta \Phi^{\text{bkg}}}$), histogram length ($N$), and system IRF ($h(t)$), we compute the MDE as follows. First, we generate $h(t)$ at D equispaced time shifts over N, and scale and vertically shift the noiseless $h(t)$ according to the input SBR and mean photon counts. Second, we draw a poisson sample at each time bin (i.e., add noise). Third, we encode the noisy histogram as in Eq. 4 of the main paper. Then, we compute the D time shifts as in Suppl. Eq. 4, and take the absolute difference with the ground truth. Finally, the above process is repeated S times, and the expected errors are computed for all $S \cdot D$ time shifts, averaged to obtain the MDE, and divide by N to obtain the relative MDE. Moreover, we also report the median expected depth error which is in the same way as the MDE, but instead of averaging the $S \cdot D$ time shifts, we take the median. For all the simulations in this paper we use $N = 1024$, $S = 1000$, and $D = 64$.

S. 3.2. Isometric Compression Analysis

In this section we present the mean and median depth error results and the isometric compression analysis for all coding schemes at different compression ratios. We compare the relative MDE of the FRH (ε_{FRH}) and the CSPH ($\varepsilon_{\text{CSPH}}$) to determine isometric compression. Since Continuous Gray coding is only defined up to $K \leq \log_2(N) = 10$ for $N = 1024$, we defined a coding scheme that extends Gray coding higher K values using a similar intuition as Gray-based Fourier coding in the main paper, and we denote it as Fourier-based Gray coding:

- **Fourier-based Gray Coding:** Suppl. Fig. 5 shows a Fourier-based Gray coding matrix for $K = 16$. The first $\lfloor \log_2(N) \rfloor$ rows of this matrix are the same as Gray coding. The second $\lfloor \log_2(N) \rfloor - 2$ codes are the 90 degree shifted version of each Gray code, excluding the first two gray codes. For the remainder codes we sample binary square functions at increasing frequencies that have not been sampled, in the same way the Gray-based Fourier coding samples Fourier components. The Fourier-based Gray coding matrix at $K > 2\lfloor \log_2(N) \rfloor - 2$, becomes a binary form of the Gray-based Fourier matrix with the initial rows in a different order.

Main Observations: Suppl. Fig. 3 and 4 show the isometric compression results for compression ratios of 64 ($K = 16$) and 16 ($K = 64$). At high compression ratios of 64x only Gray-based Fourier and Fourier-based Gray coding are the only coding schemes that are able to achieve isometric compression with $\varepsilon_{\text{diff}} < 0.01$ for a wide range of SBR and photon count levels. As we use a greater number of codes (reduce compression), Truncated Fourier coding starts approaching the performance of the proposed coding schemes in the isometric compression regions with low $\varepsilon_{\text{diff}} < 0.01$. Overall, commonly used approaches such as coarse histograms and timestamp transfer, consistently exhibit lower performance than a CSPH.

- **Gray-based Fourier vs. Fourier-based Gray:** Overall, the performance of these two coding schemes is very similar at $K > \log_2 N$ and across SBR and photon count levels. In the main paper we focus our discussion on Gray-based Fourier coding due to its marginally better performance at low SBR and its interpretability due to its use of Fourier functions. Nonetheless, Fourier-based Gray coding is an interesting coding scheme because at $K > \log_2 N$ it becomes fully binary, which may have a simple hardware implementation.

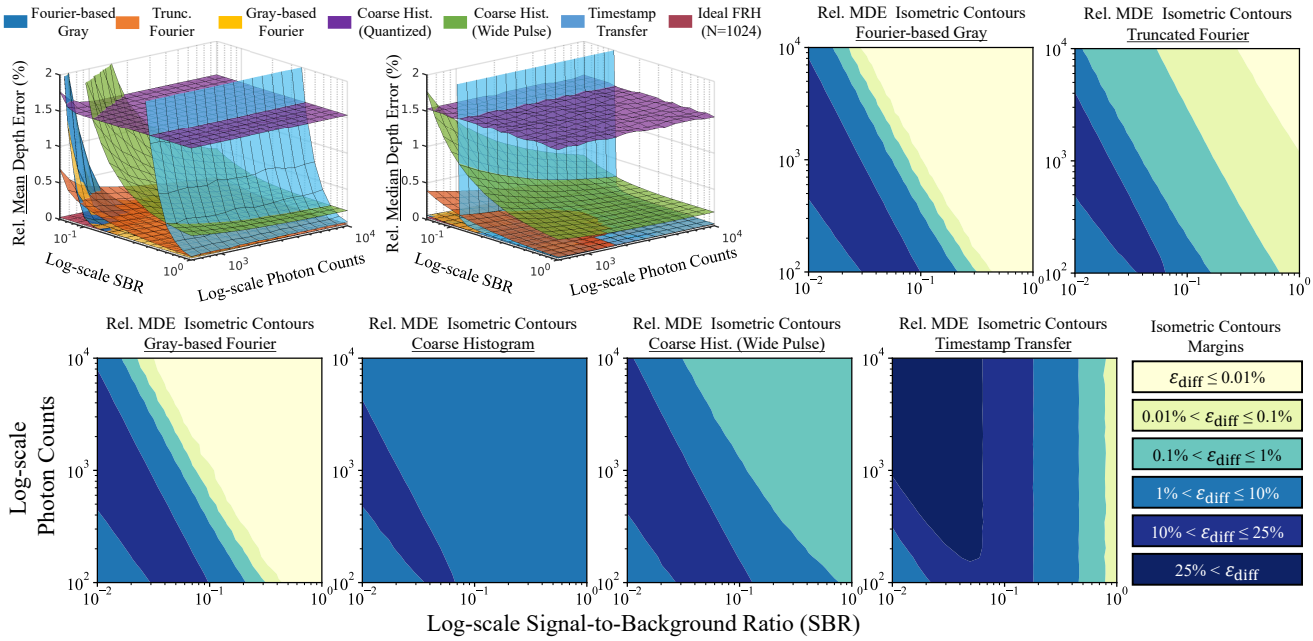
- **Rate of Performance Decay:** As we decrease SBR and the photon count levels (approach lower left corner of isometric countour plots), the performance of all methods begins to degrade at different rates. Gray-based Fourier coding is able to maintain low relative MDE up to a certain SBR and photon count level, and then its relative MDE quickly degrades faster than the ideal FRH, creating small isometric contours for $0.01\% < \varepsilon_{\text{diff}} < 1\%$. On the other hand, the performance of

Truncated Fourier coding progressively degrades, creating larger isometric contour regions for $0.01 < \varepsilon_{\text{diff}} < 1$, albeit at higher SBR and photon count levels. At $K = 16$ the progressive performance decay of Truncated Fourier, allows it to achieve isometric compression with lower $\varepsilon_{\text{diff}}$ for some SBR (< 0.1) and photon count levels (< 1000). Unfortunately, this advantage of Truncated Fourier coding is encountered around $\varepsilon_{\text{diff}} \approx 1\%$, which for a 10m depth range is an relative MDE difference of 10cm. Therefore, the better choice at those challenging SBR and photon count levels is to increase K for Gray-based Fourier Coding.

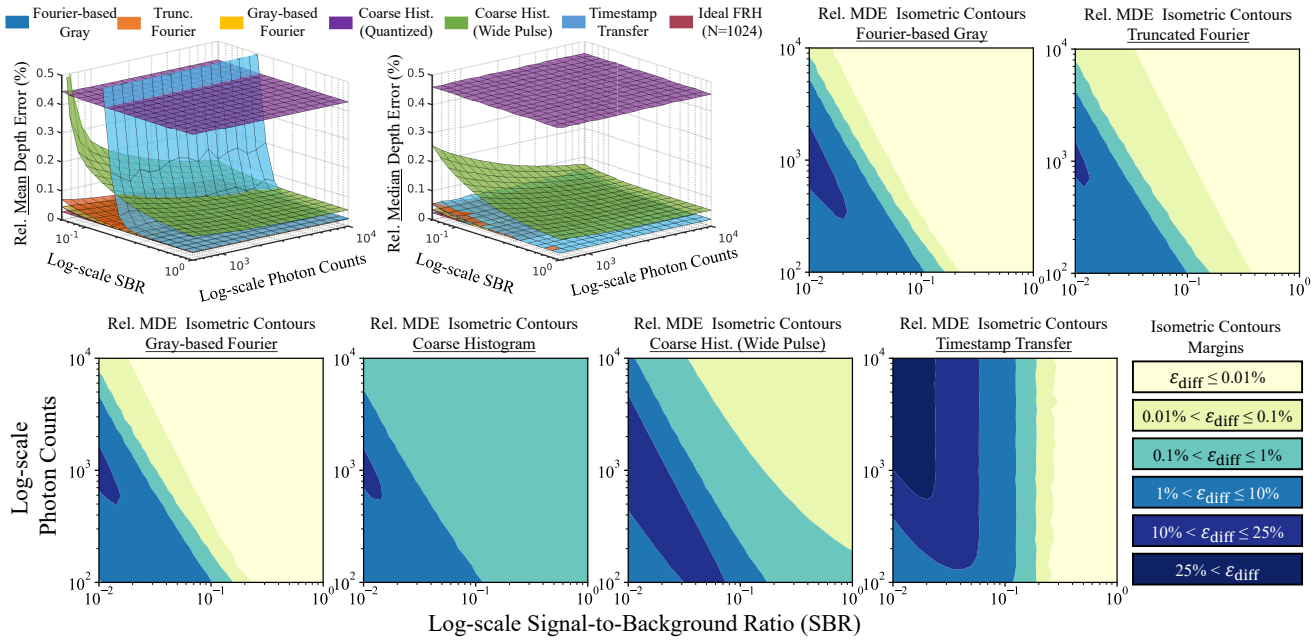
- **Mean vs. Median Depth Errors:** Comparing the relative mean and median errors surface plot trends allow us to learn useful things about each coding scheme. Coding approaches where the trends are similar produce depth errors of similar magnitudes. In the case of Truncated Fourier coding or a Coarse Hist (Wide Pulse), this means that they will make a lot of small errors, even at high SBR. On the other hand, approaches such as Timestamp transfer and Gray-based Fourier coding, can produce high relative mean depth errors while maintaining near-zero median depth errors. This is because, as we decrease SBR and photon count levels, these approaches either estimate the depth with near-zero error or produce a very high depth error (outliers).

In Appendix S. 8 we show additional results for compression ratios of 128, 32, and 8.

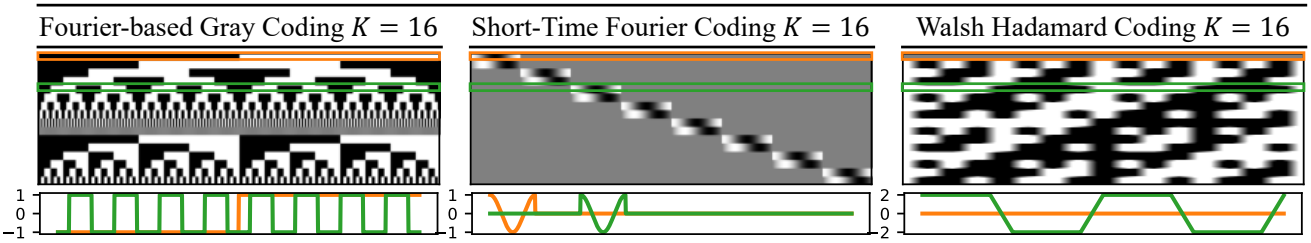
Summary: Our extensive Monte Carlo simulations and isometric compression analysis provide a guide for selecting the appropriate CSPH coding scheme when we have prior knowledge on SBR and photon count levels. Our results suggest that at $K < 32$, Gray-based Fourier coding is a good choice that can provide high compression without sacrificing depth precision in a wide range of scenarios. At higher K, Truncated Fourier becomes a competitive choice that can perform comparably to Gray-based Fourier. Overall, carefully designed coding matrices can significantly outperform a coarse histogram or transferring timestamps.



Supplementary Figure 3. **Isometric Compression Analysis at 64x Compression ($K = 16$) .** The surface plots show the relative mean and median depth errors computed as described in Sec. S. 3.1 for coding schemes with $K = 16$. The isometric compression contour plots are created by taking the element-wise absolute difference of each coding scheme relative MDE with the ideal FRH, and then classifying the difference into various margins ($\varepsilon_{\text{diff}}$). As we decrease SBR and photon count levels, the isometric compression margin achieved by each method increases. Finally, as we approach the lowest SBR and photon count levels, even the FRH starts failing too which makes $\varepsilon_{\text{diff}}$ smaller again. Although, the range of SBR and photon count levels for the isometric contour plots are 0.01-1 and 100-10000, the surface plots show SBR and photon count levels of 0.05-1 and 500-10000, for visualization purposes.



Supplementary Figure 4. **Isometric Compression Analysis at 16x Compression ($K = 64$) .** The surface plots show the relative mean and median depth errors computed as described in Sec. S. 3.1 for coding schemes with $K = 64$. The isometric compression contour plots are created by taking the element-wise absolute difference of each coding scheme relative MDE with the ideal FRH, and then classifying the difference into various margins ($\varepsilon_{\text{diff}}$). As we decrease SBR and photon count levels, the isometric compression margin achieved by each method increases. Finally, as we approach the lowest SBR and photon count levels, even the FRH starts failing too which makes $\varepsilon_{\text{diff}}$ smaller again. Although, the range of SBR and photon count levels for the isometric contour plots are 0.01-1 and 100-10000, the surface plots show SBR and photon count levels of 0.05-1 and 500-10000, for visualization purposes.



Supplementary Figure 5. **Additional Coding Matrices.** In addition to the coding schemes presented in the main paper we also evaluate these coding schemes. Fourier-based Gray coding is evaluated in Sec. S. 3.2. Short-time Fourier and Walsh Hadamard coding are evaluated in Sec. S. 3.3.

S. 3.3. Comparisons with Additional Coding Schemes

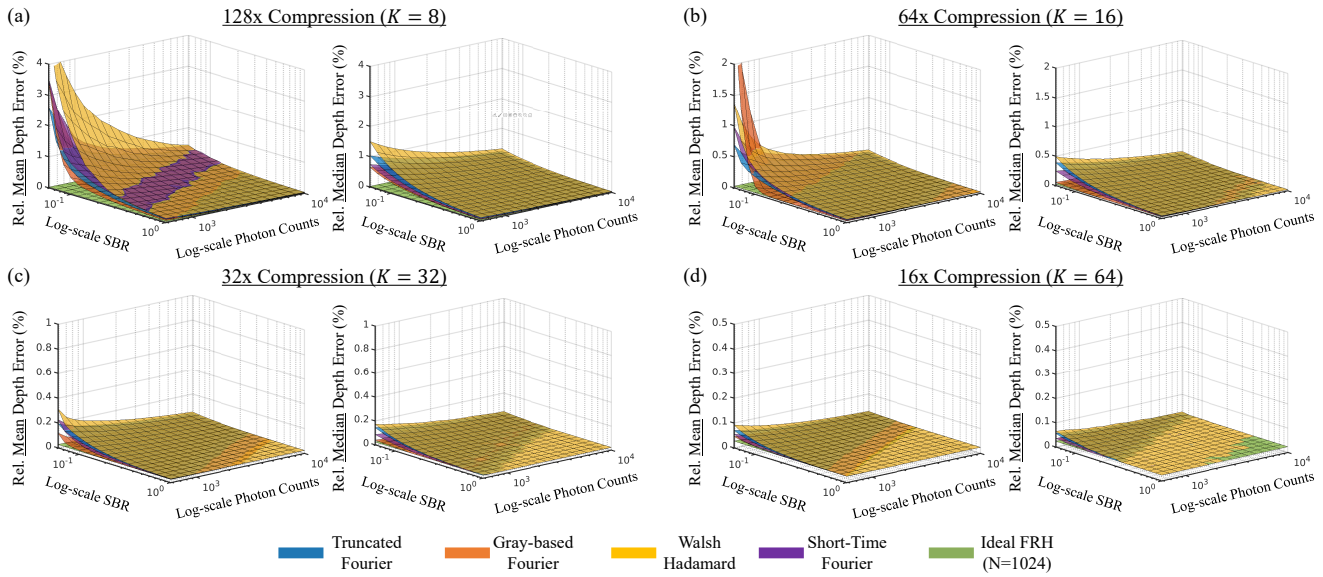
In this section we evaluate two additional CSPH coding schemes and compare them with the coding schemes presented in the main paper:

- **Short-time Fourier:** The coding matrix is similar as the coarse histogram, but instead of setting the rectangular window at each row to all 1's, we set the values inside the window to a single period of a sinusoidal function. As illustrated in Suppl. Fig. 5, the odd and even rows use a cosine and sine function respectively. We refer to this method a short-time Fourier, due to its similarity to a short-time Fourier transform.
- **Walsh Hadamard:** To generate C, we first generate a $K \times K$ Hadamard matrix using scipy's Hadamard function [17], transpose it, and then linearly interpolate each row to make it length N.

To evaluate the coding schemes we use the relative mean and median depth errors metrics described in Sec. S. 3.1.

Main Observations: Suppl. Fig. 6 compares the performance of Short-time Fourier and Walsh Hadamard coding with the coding schemes used in the main paper, at multiple compression ratios. Walsh Hadamard coding consistently achieves the lowest performance. Short-time Fourier, achieves lower relative median depth errors than Truncated Fourier but higher relative median depth errors, while having similar performance trends. Gray-based Fourier coding achieves outperforms all coding schemes at most SBR and photon count levels.

Summary: Although, the new coding schemes evaluated in this section were not able to outperform Gray-based Fourier coding for the most part, their performance was competitive and well above a coarse histogram method.



Supplementary Figure 6. **Performance of Additional Coding Schemes.** Relative mean and median depth errors at different compression ratios. As we decrease compression, all coding schemes quickly approach the performance of the ideal FRH (note the z-axis is different at each compression ratio), and the performance difference across coding schemes becomes small. However, at high compression ratios, the choice of coding scheme is crucial at most SBR and photon count levels.

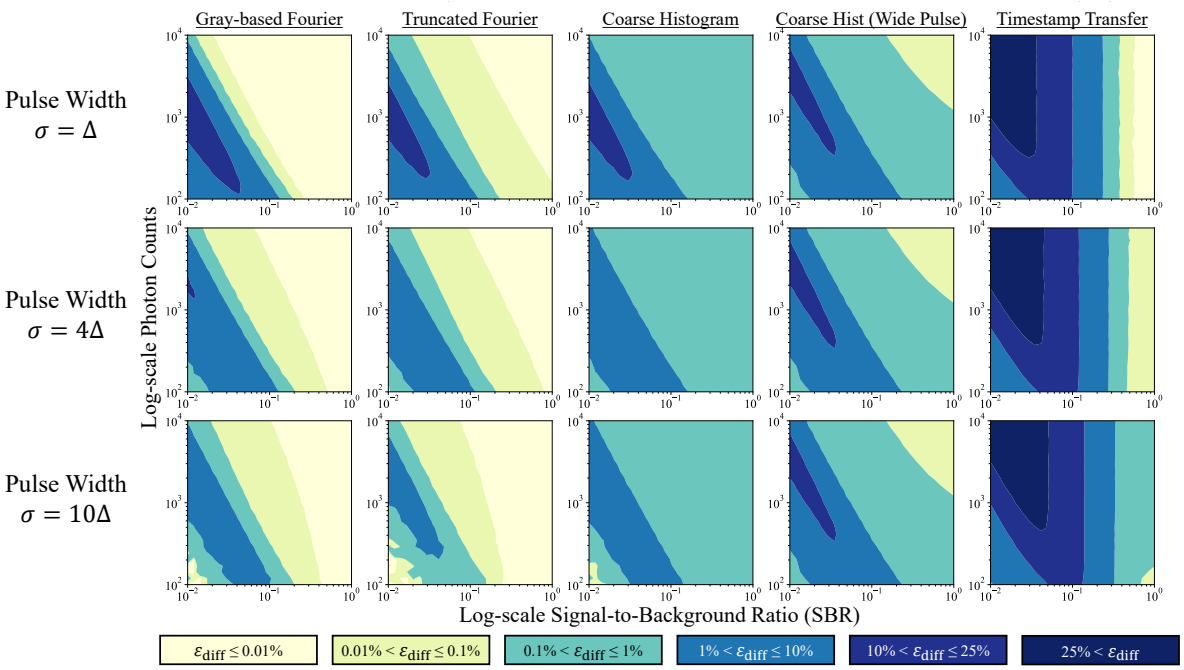
S. 3.4. Isometric Compression with Wide Gaussian Pulses

In this section we analyze how isometric compression trends change as we increase the system IRF pulse width.

Suppl. Fig. 7 shows the isometric compression contour plots for multiple coding schemes with system IRFs set to Gaussian pulses with varying widths. The relative MDE isometric contour plots are generated using similar MDE Monte Carlo simulations described in Sec. S. 3.1.

Main Observations: As we increase the pulse width used, the isometric contours with low $\varepsilon_{\text{diff}} \leq 1\%$ increase for Truncated Fourier, Gray-based Fourier, and coarse histograms. In particular, the overall performance difference between Truncated Fourier and the ideal FRH shrinks. This is because increasing the pulse width, decreases the bandwidth of the photon flux waveform, which makes Truncated Fourier more efficient. For medium SBR and photon count levels, the performance of Gray-based Fourier coding decreases, as the pulse width increases, because some of the high-frequency coding functions get zero-ed out (band-limit property). Moreover, the performance of an FRH with a wider pulse also decreases at a faster rate as we decrease SBR and photon counts. Therefore, at the lowest SBR and photon count levels, the FRH performs as poorly as the CSPH methods, leading to isometric compression with small $\varepsilon_{\text{diff}}$ again. Finally, the timestamp transfer baseline is one of the methods that is impacted the most as the pulse width increases. In Suppl. Sec. S. 9 we show additional contour plots for compression ratios of 64 and 16.

Summary: Truncated Fourier coding is the better CSPH for a SPAD-based 3D camera that has a pulse width significantly wider than the SPAD's time resolution (Δ). At narrower pulse widths, Gray-based Fourier coding performs better because these narrow pulses have useful high-frequency content that Gray-based Fourier coding samples. Overall, the system IRF and consequently the laser pulse waveform, play an important role in the choice of the CSPH coding scheme.



Supplementary Figure 7. **Wide Pulse Isometric Compression at 32x Compression ($K = 32$).** The Gaussian pulses used as $h(t)$ are proportional to: $\exp -\frac{t^2}{\Delta}$ (1st row), $\exp -\frac{t^2}{4\Delta}$ (2nd row), $\exp -\frac{t^2}{10\Delta}$ (3rd row). The jagged edges observed in the isometric contours at low SBR and low photon counts are caused by the variance of the Monte Carlo simulations which adds small variance to the relative MDE of all methods.

S. 4. Low Photon Count Regime

In this section we analyze the performance of different coding schemes in the low photon count regimes where less than 100 photons are detected per pixel, on average. In practice, this is important for scenarios such as low scene albedo, long standoff distances, extremely short acquisition times, or in case of constrained laser power budget.

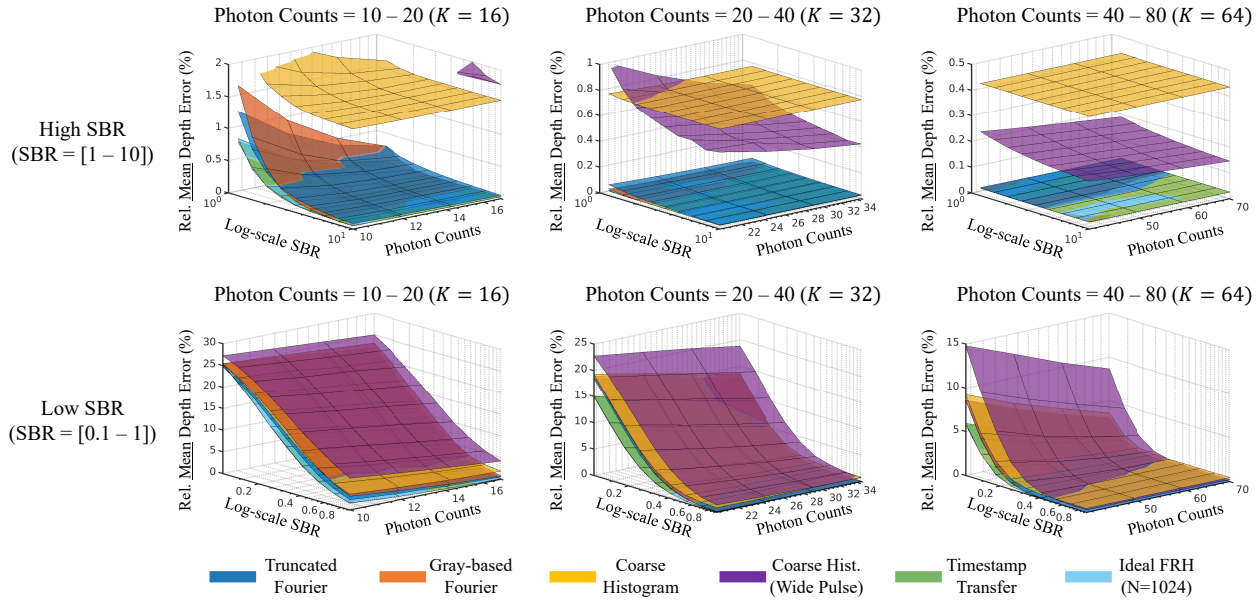
Suppl. Fig. 8 shows the mean depth errors for different coding schemes as a function of SBR and total number of photons. The total number of photons includes both signal (laser) and background (ambient) photons. The relative MDE Monte Carlo simulations were performed with the same parameters as in Sec. S. 3.2. The number of coding functions used for each photon count range is chosen to roughly match the expected number of photons (e.g., for a photon count range of 10-20, we use 16 coding functions). Note that in some cases the compressive algorithm may require *more* data bandwidth than timestamp transfer (*e.g.* when there are only 10 photons captured and the compressive method uses $K = 16$ Fourier coefficients.) Moreover, in the low count regime, timestamp transfer is nearly equivalent to the FRH that transfers all time bins.

Main Observations:

- In the high SBR regime the proposed Fourier-coding based methods provide some benefits over näive coarse histogramming approaches. When the number of photons captured is > 20, compressive methods perform almost as well as FRH and timestamp transfer and provide almost perfect depth reconstruction. The extent of compression achieved *vis a vis* transferring raw timestamps is limited due to the low number of photons captured in each pixel.

- In the low SBR regime most of the photons captured by the pixel are ambient photons, hence, the overall relative MDE remains high. In this regime, neither compressive approaches nor the baseline approaches (coarse histogramming, FRH and timestamp transfer) provide usable depth maps, with relative MDEs as high as 25% due to Poisson noise. In such scenarios it may become necessary to either increase the laser power, increase the total acquisition time, or resort to other denoising approaches (*e.g.* spatio-temporal filtering) [8, 11, 13, 14] that can complement the compressive capture strategies discussed in this paper.

Summary: In the low photon count regime it is necessary to combine the compressive acquisition methods described in this paper with denoising approaches that operate on raw timestamp data or on compressed histogram measurements.



Supplementary Figure 8. **Photon-Starved Mean Depth Error Results.** (Top Row) In the high SBR regime, compressive approaches provide better relative MDE as compared to coarse histogramming or raw timestamp transfer. (Bottom Row) In the low SBR regime compressive acquisition alone is not sufficient to achieve low MDE and other spatio-temporal denoising methods may be necessary to combat Poisson noise in these photon-limited regimes.

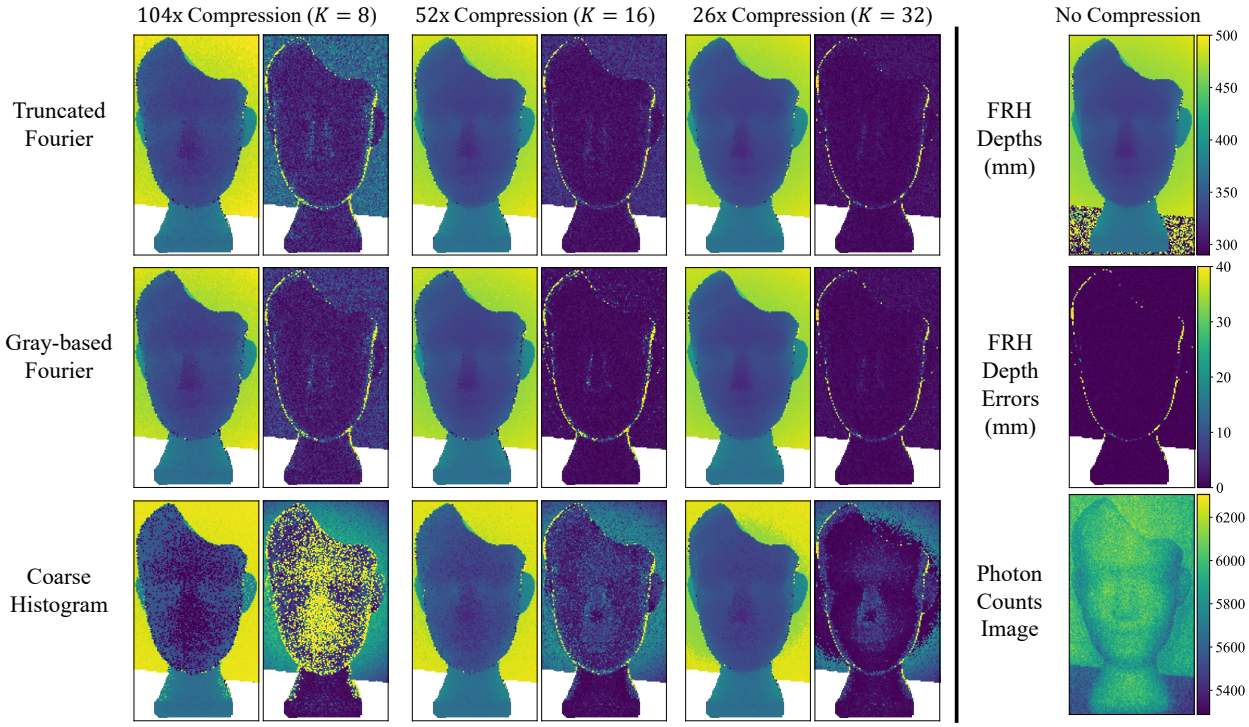
S. 5. Additional Real-world Scanning-based System Results

In this section we present additional results from the real-data acquired with the scanning-based system from [3].

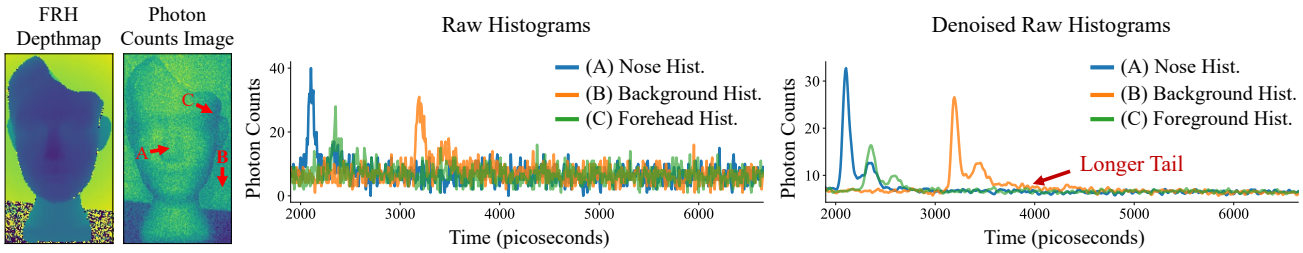
Suppl. Fig. 9 and 12 show the two scans that we downloaded and pre-processed from [3]. The pre-processed per-pixel full-res histograms have $\Delta = 8$ps and $N = 832$. To obtain ground truth depths we applied Gaussian denoising ($\sigma = 0.75$) to the full-resolution histogram images of each scene, and computed depths using matched filtering. As observed in the FRH depth images in Suppl. Fig. 9 and 12, there are some points whose SNR is too low to obtain a reliable depth even after denoising, therefore we manually mask those points when visualizing the depths and depth errors (white regions). The system IRF, $h(t)$, is extracted from on of the points near the center of the denoised histogram, and then is further pre-processed to remove most background photons and center it around $t = 0$.

Main Observations: As observed in Suppl. Fig. 9 and 12, at $K = 8$, all CSPH methods make systematic depth errors in the background points. These errors are caused by the longer tail observed in the background points histogram shown in Suppl. Fig. 10. At $K = 16$, Gray-based Fourier coding contains sufficient coding functions with high-frequency content that mitigate the effects of the longer tail and estimate reliable depths in the background points. Finally, at $K = 32$ Gray-based Fourier and Truncated Fourier are able to essentially match the performance of the FRH. Coarse histograms can achieve sub-bin precision because $h(t)$ spreads over multiple coarse time bins. Nonetheless, the depth precision is lower, and the systematic errors in the background points continue to affect it, even at $K = 32$.

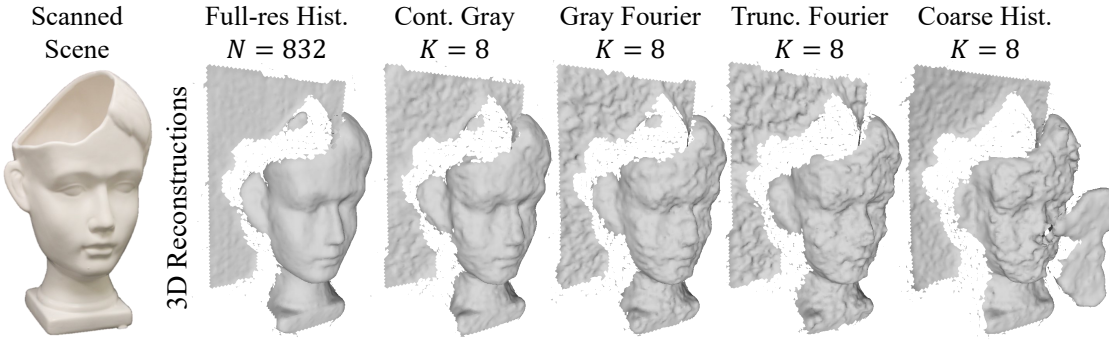
Summary: Despite the non-idealities encountered in real data (long-tailed $h(t)$ and indirect reflections), CSPH coding approaches, such as Truncated Fourier and Gray-based Fourier, can provide significant compression without sacrificing depth precision.



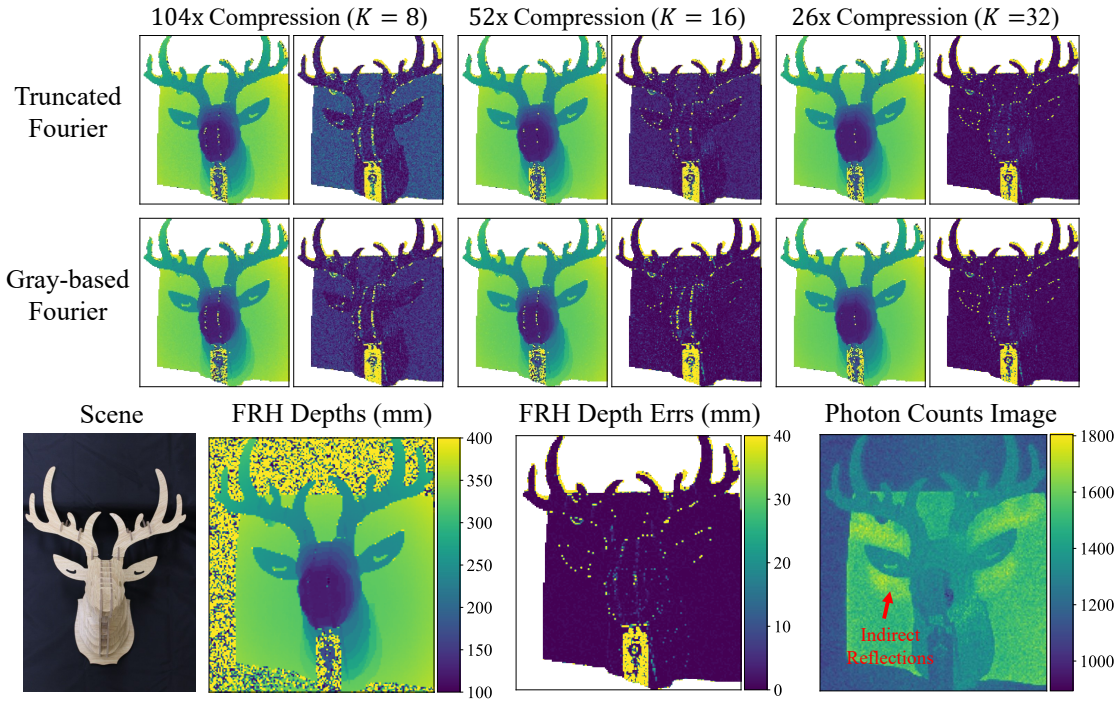
Supplementary Figure 9. **Scan Data Results at Different Compression Ratios.** As we decrease the compression (increase K) the errors for all coding schemes decrease. Truncated Fourier at $K \leq 16$ is susceptible to the systematic errors caused by the longer tail that appears in the background histograms (Suppl. Fig. 10). At $K = 32$ both Gray-based Fourier and Truncated Fourier essentially match the performance of using the full-res histogram. Comparatively, coarse histogram have low depth precision and are quite susceptible to the longer tail of the background points histogram, even at $K = 32$.



Supplementary Figure 10. **Denoised Raw Histograms.** We apply a 3D Gaussian filter to the histogram image of the face scan and visualize histograms at different scene points. Note that the histogram extracted from the background point (orange line) exhibits a longer tail than the histograms obtained from points on the face. This longer tail is likely due to indirect reflections.



Supplementary Figure 11. **3D Reconstruction Visualizations.** To generate the 3D reconstructions in this paper we apply a 3×3 median filter to the recovered depth images, generate point clouds, and use MeshLab to estimate normals and perform Poisson surface reconstruction [2].



Supplementary Figure 12. **Scan Data Results for Deer Scene.** The first and second row show the depth and depth error image pairs obtained from each CSPH coding method. As we decrease the compression (increase K) the errors for all coding schemes decrease. Truncated Fourier at $K = 8$ is susceptible to the systematic errors caused by the longer tail that appears in the background histograms. Increasing K resolved these errors are mitigated by using coding functions with higher frequencies.

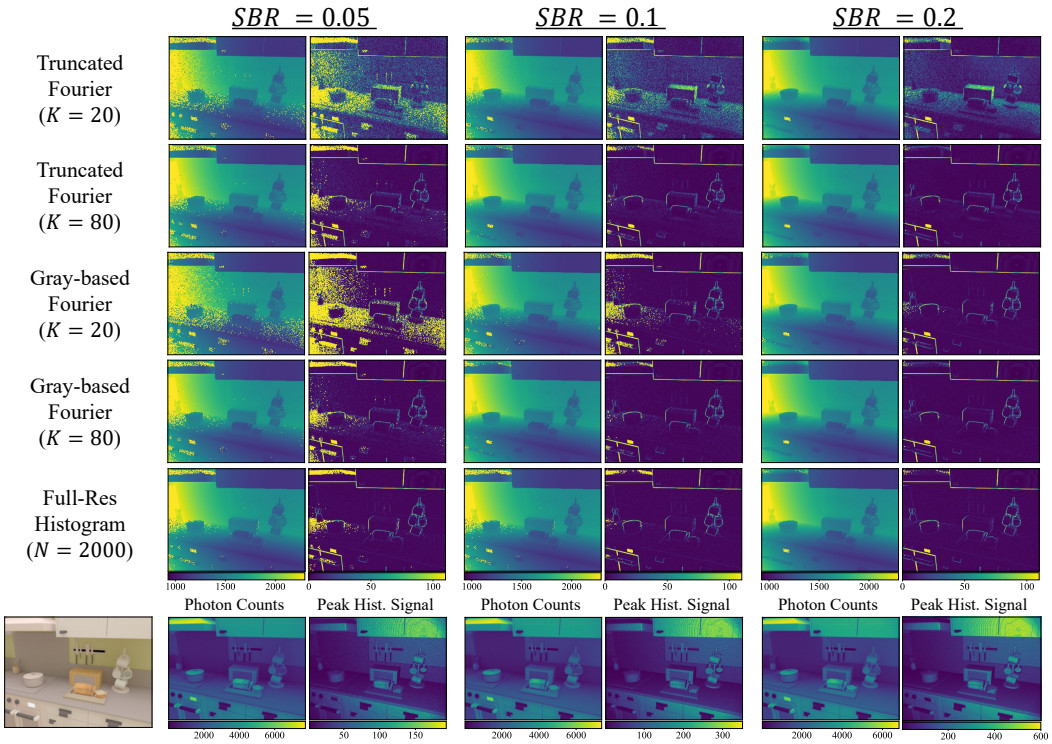
S. 6. Additional Simulated Flash Illumination System Results

In this section we evaluate different CSPH strategies applied on a flash illumination scenario at varying SBR levels. The flash illumination system data is simulated from physically accurate histogram images rendered with MitsubaToF [10]. The scenes input to MitsubaToF are obtained from [6]. The histogram image parameters were $\Delta = 50$ps and $N = 2000$ time bins. As described in the main paper, we use the R channel of the corresponding RGB image of the scene as an approximation of the per-pixel background photons. Then to simulate the FRHs we set the mean SBR and mean photon counts of the overall scene, and use those input values to scale and the histogram image and vertically shift it (add background). Similar to the relative MDE simulation results, we use an $h(t) \propto \exp -\frac{t^2}{\Delta}$ which is convolved with the rendered ground truth per-pixel histograms.

Main Observations:

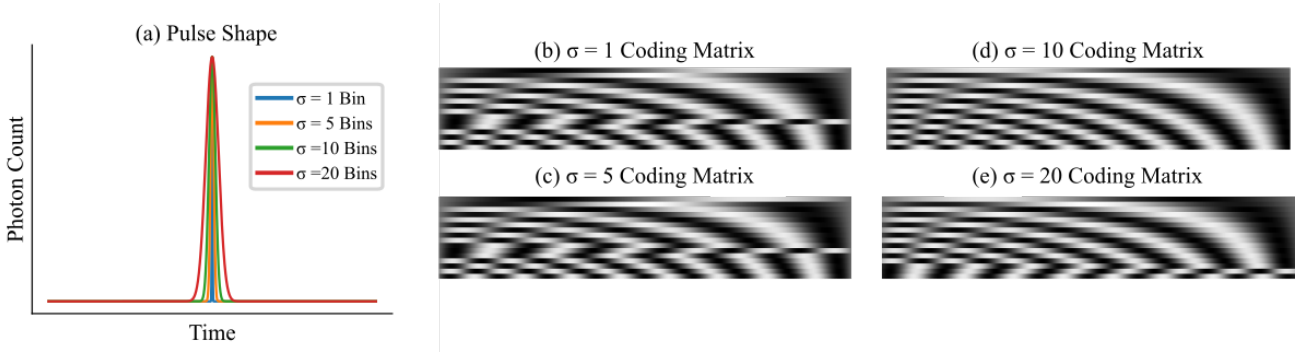
- **Indirect Reflections Errors:** At $K = 20$ Truncated Fourier is susceptible to systematic depth errors caused by diffuse indirect reflections. For Truncated Fourier coding, the only way to overcome these errors is to increase K and use higher frequencies. Gray-based Fourier coding can mitigate these errors with only 20 coding functions.
- **Outlier Depth Estimates:** At $K = 20$, in low SBR regions, Gray-based Fourier coding produces depth estimates with either near-zero or very high (outlier) errors. This is consistent with the observations made in Sec. S. 3.2, where the mean depth errors were significantly higher than the median (due to outliers). Increasing K helps Gray-based Fourier coding mitigate these outliers and perform comparably to a FRH.

Summary: Truncated Fourier coding at low K is susceptible to errors due to indirect reflection, regardless of SBR or photon count levels. This means that a Truncated Fourier CSPH on a flash illumination system may require higher K (lower compression) to avoid trading-off depth precision. Gray-based Fourier coding, on the other hand can use lower K (high compression) and still achieve high depth precision, as long as the SBR level is sufficiently high. Overall, having prior knowledge on the SBR and photon count levels can help select the correct K to use in a CSPH.



Supplementary Figure 13. **Flash Illumination Results at Multiple SBR.** We simulate the same scene with different mean SBR levels, and fixed mean photon counts to 2000. A mean scene SBR of 0.05 means that if we look at the per-pixel SBR and average them, we will obtain approximately 0.05. As observed in the bottom row, as we increase the SBR, the photon count image remains unchanged (because mean scene photon counts is fixed at 2000), but the peak of the per-pixel histogram increases.

S. 7. PCA Codes



Supplementary Figure 14. **PCA Codes at Different Pulse Widths.** In (a) two Gaussian pulses at different pulse widths (σ) are shown. (b-e) shows PCA coding matrix ($K = 16$) for $\sigma = 1, 5, 10, 20$. All four pulse widths yield similar PCA codes that look like Fourier components. Note that some PCA codes in (b,c,e) are multiplied by negative one when compared to (d) or the Fourier codes.

This section will describe how we found the principal component analysis codes used in the main paper and why they relate to Fourier codes. To numerically find the PCA codes, we generate many Gaussian LiDAR pulses at different signal-to-background ratios and with different shifts representing the binned return signal at the camera sensor. Let $(\Phi_{i,s,b})_{i=0}^{N-1}$ be the binned photon flux waveform from Eq. 2 in the main text when $\Phi^{\text{sig}}(t)$ is a Gaussian pulse centered at time s and $\Phi^{\text{bkg}} = b$. Therefore, $(\Phi_{i,s,b})_{i=0}^{N-1}$ is the binned return signal from a Gaussian laser pulse, from an object at distance $\frac{cs}{2}$, and signal-to-background ratio of $\frac{1}{b}$.

In order to run PCA we generate $(\Phi_{i,s,b})_{i=0}^{N-1}$ for many s and b values and stack them into a large matrix. Specifically, we use 1000 s values in increments of $\Delta_s = \frac{\tau}{1000}$ from 0 to τ, that is $s \in \{0, \Delta_s, 2\Delta_s, \dots \tau\}$. At each s value we use 50 b values with log spacing with increments of $\Delta_b = \frac{2}{50}$ from 10^{-2} to 1, that is $b \in \{10^{-2}, 10^{-2+\Delta_b}, 10^{-2+2\Delta_b}, \dots 1\}$. The final matrix, M, is given by,

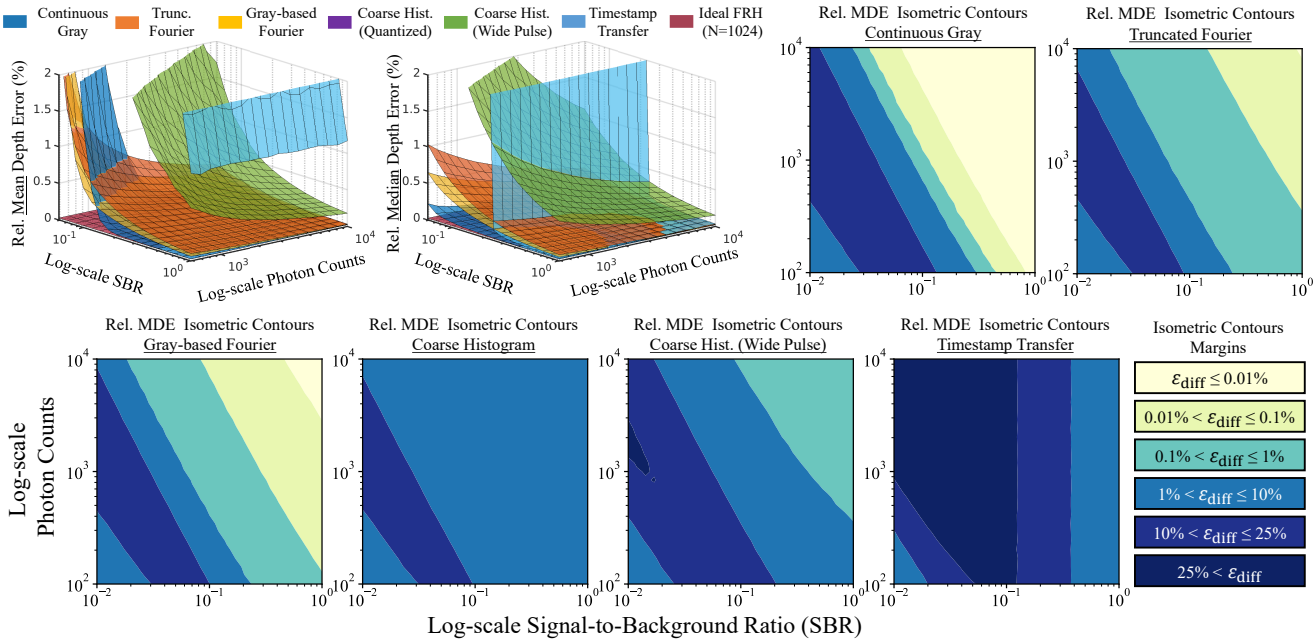
$$M(b) = \frac{1}{1+b} \begin{bmatrix} \Phi_{0,0,b} & \Phi_{1,0,b} & \dots & \Phi_{N-1,0,b} \\ \Phi_{0,\Delta_s,b} & \Phi_{1,\Delta_s,b} & \dots & \Phi_{N-1,\Delta_s,b} \\ \Phi_{0,2\Delta_s,b} & \Phi_{1,2\Delta_s,b} & \dots & \Phi_{N-1,2\Delta_s,b} \\ \vdots & \vdots & & \vdots \\ \Phi_{0,\tau,b} & \Phi_{1,\tau,b} & \dots & \Phi_{N-1,\tau,b} \end{bmatrix} \tag{5}$$

$$M = \begin{bmatrix} M(10^{-2}) \\ M(10^{-2+\Delta_b}) \\ \vdots \\ M(1) \end{bmatrix} \tag{6}$$

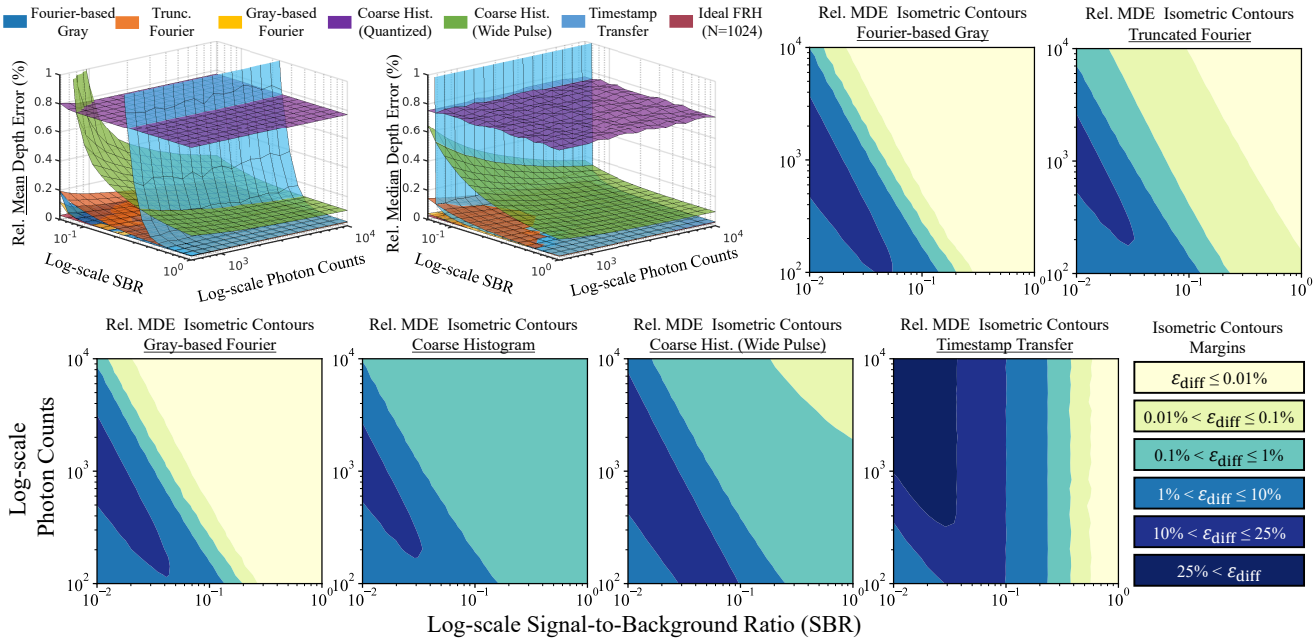
Where the factor of $\frac{1}{1+b}$ ensures that the rows of $M(b)$ sum to 1. We then use PCA on M and use the found PCA components as the PCA codes. Some of our found PCA codes for four different pulse widths are shown in Suppl. Fig. 14. Interestingly, the resulting PCA codes where very similar to Fourier codes.

To understand why PCA codes are similar to Fourier codes consider a single b value and set Δ_s to the length of a single time bin, then $M(b)$ will be a circulant matrix. It turns out that eigenvectors and thus PCA components of circulant matrices are given by Fourier components [15]. To understand why recall that circulant matrices implement convolutions. Convolutions are simply multiplications in the Fourier domain which implies that the eigenvectors of a convolution and thus circulant matrices are the Fourier components. Although, we do not use Δ_s equal to the length of a time bin, the $M(b)$'s we use are close to a circulant matrix so our resulting PCA components are very close to Fourier components.

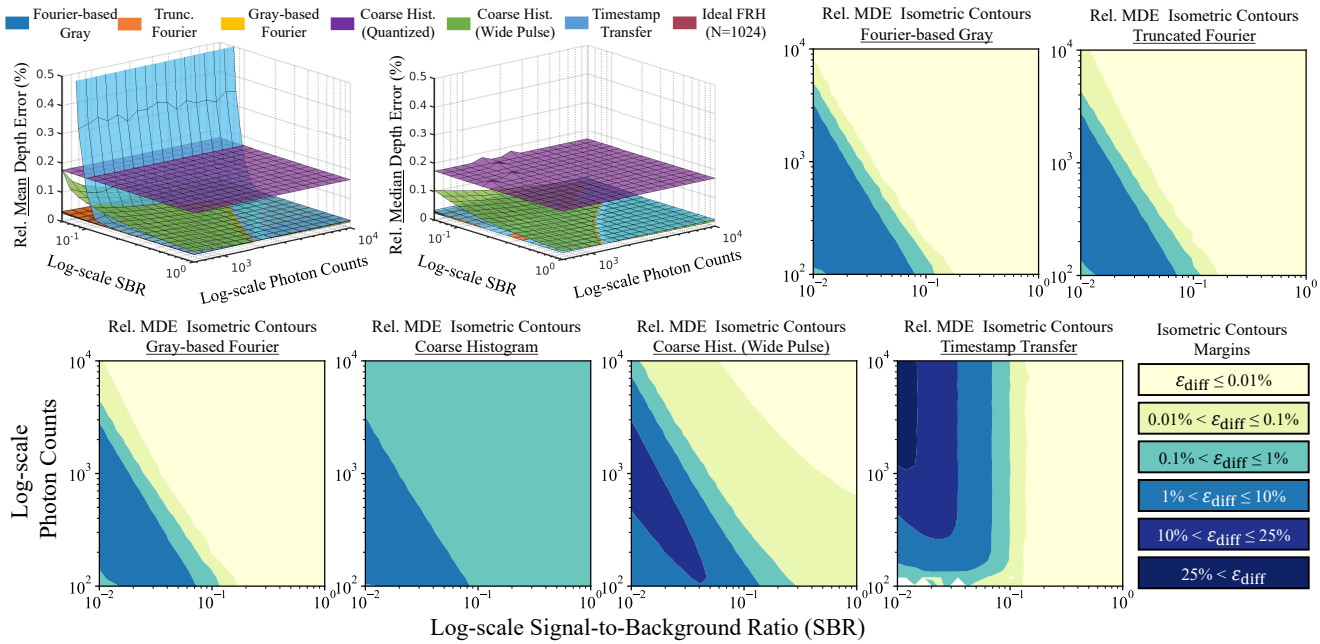
S. 8. Appendix A: Additional Isometric Compression Plots



Supplementary Figure 15. **Isometric Compression Analysis at 128x Compression** ($K = 8$) **.**

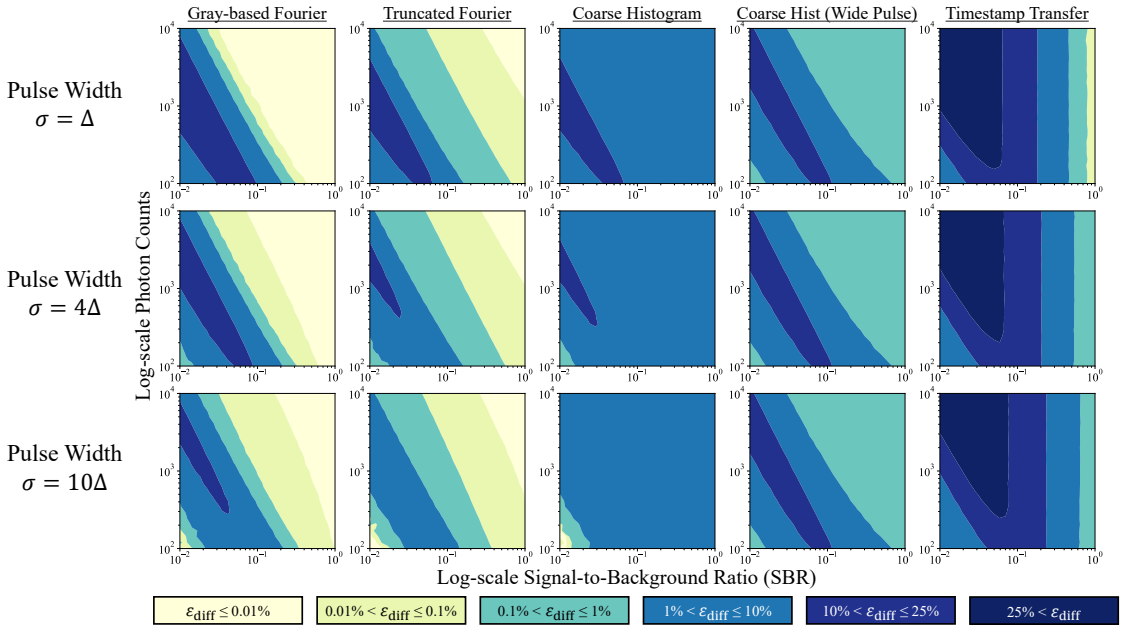


Supplementary Figure 16. **Isometric Compression Analysis at 64x Compression** ($K = 32$) **.**

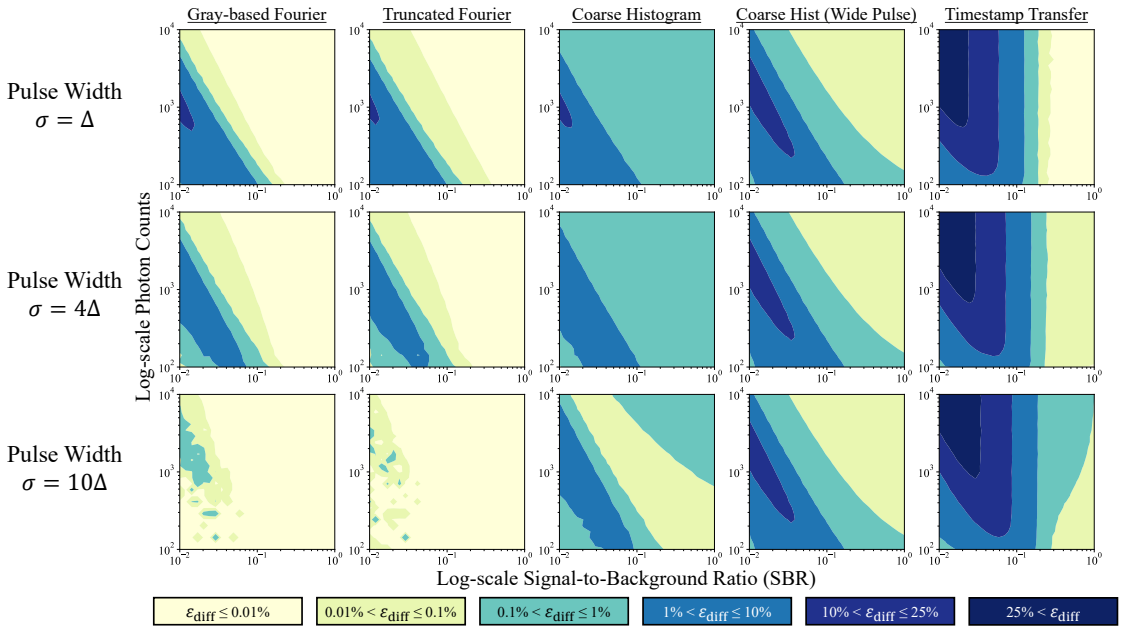


Supplementary Figure 17. **Isometric Compression Analysis at 8x Compression** ($K = 128$) **.**

S. 9. Appendix B: Additional Isometric Compression Results with Wide Pulses



Supplementary Figure 18. **Wide Pulse Isometric Compression at 64x Compression ($K = 16$).** The Gaussian pulses used as $h(t)$ are proportional to: $\exp -\frac{t^2}{\Delta}$ (1st row), $\exp -\frac{t^2}{4\Delta}$ (2nd row), $\exp -\frac{t^2}{10\Delta}$ (3rd row).



Supplementary Figure 19. **Wide Pulse Isometric Compression at 16x Compression ($K = 64$).** For $\sigma = 10$ the coarse histogram method follows a different trend from what is seen before. This is because at high SBR and high photon counts, the coarse histogram is quantization limited, while the FRH can achieves low depth errors. As SBR and photon counts decrease the FRH performance decreases, while the coarse histogram continues to be quantization limited. As SBR and photon counts continue to decrease the performance of the coarse histogram starts degrading more rapidly than the FRH. And finally, at the lowest SBR and photon count levels both methods are achieving very high depth errors (almost estimating depths at random), making their $\varepsilon_{\text{diff}}$ small again.

References

[1] Wenzheng Chen, Parsa Mirdehghan, Sanja Fidler, and Kiriakos N Kutulakos. Auto-tuning structured light by optical stochastic gradient descent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5970–5980, 2020. 3

[2] Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, Guido Ranzuglia, et al. Meshlab: an open-source mesh processing tool. In *Eurographics Italian chapter conference*, volume 2008, pages 129–136. Salerno, Italy, 2008. 11

[3] Anant Gupta, Atul Ingle, and Mohit Gupta. Asynchronous single-photon 3d imaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7909–7918, 2019. 10

[4] Mohit Gupta, Shree K Nayar, Matthias B Hullin, and Jaime Martin. Phasor imaging: A generalization of correlation-based time-of-flight imaging. *ACM Transactions on Graphics (ToG)*, 34(5):156, 2015. 2

[5] Mohit Gupta, Andreas Velten, Shree K Nayar, and Eric Breitbach. What are optimal coding functions for time-of-flight imaging? *ACM Transactions on Graphics (TOG)*, 37(2):13, 2018. 1, 4

[6] Felipe Gutierrez-Barragan, Huaijin Chen, Mohit Gupta, Andreas Velten, and Jinwei Gu. itof2dtof: A robust and flexible representation for data-driven time-of-flight imaging. *arXiv preprint arXiv:2103.07087*, 2021. 3, 12

[7] Felipe Gutierrez-Barragan, Syed Azer Reza, Andreas Velten, and Mohit Gupta. Practical coding function design for time-of-flight imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1566–1574, 2019. 3, 4

[8] David B Lindell, Matthew O'Toole, and Gordon Wetzstein. Single-photon 3d imaging with deep sensor fusion. *ACM Trans. Graph.*, 37(4):113–1, 2018. 9

[9] Parsa Mirdehghan, Wenzheng Chen, and Kiriakos N Kutulakos. Optimal structured light à la carte. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6248–6257, 2018. 3

[10] Adithya Pediredla, Ashok Veeraraghavan, and Ioannis Gkioulekas. Ellipsoidal path connections for time-gated rendering. *ACM Transactions on Graphics (TOG)*, 38(4):1–12, 2019. 12

[11] Joshua Rapp and Vivek K Goyal. A few photons among many: Unmixing signal and noise for photon-efficient active imaging. *IEEE Transactions on Computational Imaging*, 3(3):445–459, 2017. 9

[12] Walter Rudin. *Real and complex analysis*. McGraw-HiII Book Company, 1987. 2

[13] Dongeek Shin, Feihu Xu, Dheera Venkatraman, Rudi Lussana, Federica Villa, Franco Zappa, Vivek K Goyal, Franco NC Wong, and Jeffrey H Shapiro. Photon-efficient imaging with a single-photon camera. *Nature communications*, 7(1):1–8, 2016. 9

[14] Julián Tachella, Yoann Altmann, Nicolas Mellado, Aongus McCarthy, Rachael Tobin, Gerald S Buller, Jean-Yves Tourneret, and Stephen McLaughlin. Real-time 3d reconstruction from single-photon lidar data using plug-and-play point cloud denoisers. *Nature communications*, 10(1):1–6, 2019. 9

[15] Garry J. Tee. Eigenvectors of block circulant and alternating circulant matrices. 2005. 13

[16] George Turin. An introduction to matched filters. *IRE transactions on Information theory*, 6(3):311–329, 1960. 3

[17] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. 7