SweepCam — Depth-aware Lensless Imaging using Programmable Masks: supplementary material

Yi Hua *Student Member, IEEE*, Shigeki Nakamura, M. Salman Asif, and Aswin C. Sankaranarayanan, *Senior Member, IEEE*

Abstract—This document supplements the main paper with derivations, experiment details, and additional results. Section 1 explains fast implementation of forward operator for SweepCam. Section 2 includes implementation details of simulations and includes SNR plots for experiments in the main paper. Section 3 conducts more simulations to explore optimal operating parameter of Sweepcam. Section 4 elaborates on hardware experiments, including calibration and verification of convolution model. Section 5 shows additional experiments on a scene with resolution targets.

Index Terms—Lensless imaging, Computational Photography

1 Fast solution to full SweepCam reconstruction problem

Consider a scene $\{t_1, \dots, t_D\}$ consisting of depth planes $\{z_1, \dots, z_D\}$, with measurements $\{b_1, \dots, b_N\}$ obtained from masks translated in steps of Δ. We model the forward process as a sum of convolutions,

$$\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_N \end{bmatrix} = \begin{bmatrix} K_{z_1,a_1} & \cdots & K_{z_D,a_1} \\ \vdots & \ddots & \vdots \\ K_{z_1,a_N} & \cdots & K_{z_D,a_N} \end{bmatrix} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix} \equiv \boldsymbol{S}\boldsymbol{K} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix} \equiv \boldsymbol{A} \begin{bmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_D \end{bmatrix}, \tag{1}$$

where $\boldsymbol{K}$ is a block-diagonal matrix containing $D \times D$ blocks, and the (d, d) block effectively convolves with PSF at depth z_d; $\boldsymbol{S}$ is a matrix containing $N \times D$ blocks, and the block (n, d) effectively shifts by $n\nu_{z_d}$ (i.e., convolves with $\delta(x - n\nu_{z_d})$).

Let us consider the following least squares problem with an ℓ_2-norm regularization term:

$$l(\boldsymbol{t}) = \|\boldsymbol{A}\boldsymbol{t} - \boldsymbol{b}\|_2^2 + \frac{\lambda}{2}\|\boldsymbol{t}\|_2^2. \tag{2}$$

We can write the solution in the closed form as

$$\boldsymbol{t} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}, \tag{3}$$

which can be computed using an iterative method like conjugate gradients by supplying the operator $\boldsymbol{A}^T\boldsymbol{A}$.

- *Hua and Sankaranarayanan are with the ECE Department, Carnegie Mellon University, Pittsburgh.*
- *Nakamura is with Sony Semiconductor Solutions Corp. This work was done while he was a visitor scholar at Carnegie Mellon University.*
- *Asif is with the ECE Department, University of California, Riverside.*

1.1 Fast implementation of convolutions

To achieve a well-determined system, the number of measurements N should be equal to or greater than the number of depth planes D. Applying $\boldsymbol{A}$ and $\boldsymbol{A}^T$ separately requires $2ND$ convolutions, while applying $\boldsymbol{A}^T\boldsymbol{A}$ directly requires D^2 convolutions and $2ND > D^2$. We implement the (i, j)-th block of $\boldsymbol{A}^T\boldsymbol{A}$ as a convolution in the following manner:

$$\begin{aligned}
(\boldsymbol{A}^T\boldsymbol{A})_{ij} &= \sum_p (\boldsymbol{A}^T)_{ip}(\boldsymbol{A})_{pj} \\
&= \sum_{p=1}^{N} (\boldsymbol{A}_{pi})^T(\boldsymbol{A})_{pj} \\
&= \sum_{p=1}^{N} \boldsymbol{K}_{ii}^T\boldsymbol{S}_{pi}^T\boldsymbol{S}_{pj}\boldsymbol{K}_{jj} \\
&= \boldsymbol{K}_{ii}^T\boldsymbol{K}_{jj}\sum_{p=1}^{N} \boldsymbol{S}_{pi}^T\boldsymbol{S}_{pj} \\
&= \boldsymbol{K}_{ii}^T\boldsymbol{K}_{jj}\sum_{p=1}^{N} \mathcal{S}_{-p\nu_{z_i}}\mathcal{S}_{p\nu_{z_j}} \\
&= \boldsymbol{K}_{ii}^T\boldsymbol{K}_{jj}\sum_{p=1}^{N} \mathcal{S}_{p(\nu_{z_j}-\nu_{z_i})}.
\end{aligned}$$

Thus we can implement $(\boldsymbol{A}^T\boldsymbol{A})_{ij}$ operator as a convolution with a kernel k_{ij}. If the PSF at depth z_i, z_j are a_i and a_j, respectively, then k_{ij} is a kernel formed by cross-correlating a_j with a_i, and then summing its copies translated by $p(\nu_{z_j} - \nu_{z_i})$, for $p = 1, \dots, N$.

2 Simulation details

2.1 Preprocessing of 2001 Middlebury stereo dataset

We quantized depth from 5 scenes in the 2001 Middlebury stereo dataset, so that generating many measurements with

TABLE 1
Depth quantization thresholds used on Middlebury dataset for simulations.

scene	depth planes	quantization thresholds
sawtooth	3	[0.15, 0.285]
bull	3	[.1437, .2440]
tsukuba	7	[0.3138, 0.3766, 0.4393, 0.5021, 0.6276, 0.6903]
poster	6	[.13735 .23 .3 .44 .51365]
venus	3	[.1453 .2549]

translated mask pattern is fast and scalable for our simulation. The number of depth planes we used into are listed in the table above, with the threshold values for quantization.

Additionally, we pad each scene with zero boundary so that contribution from each pixel in the scene does not go out side sensor boundary with maximum amount of translation of $p$mm on the mask in each direction. The ratio of the scene occupying the field of view is calculated by

$$1 - 2p\frac{d + z_{min}}{w z_{min}}, \tag{4}$$

where d is sensor to mask distance, z_{min} is the depth of closest plane in the scene, and w is the width of sensor in mm. The maximum amount of translation is calculate for all operating points in each plot, with the maximum being 96 LCoS pixels, or equivalently $p = 3.45$mm.

2.2 Noise Generation

Photon noise and read noise are simulated in all the measurements with parameters taken from the sensor used in our hardware experiments, Sony IMX174, with full well capacity $F = 30500$ electrons and $R = 71.7$dB.

2.3 RSNR and PSNR of Experiments

For completeness we include the RSNR and PSNR plots for simulations experiments conducted in Section 5 of our paper evaluating effect of number of measurements in Fig. 1, different baseline in Fig. 2, and light level in Fig. 3. Our mask is a positive-negative M-sequence pattern that preserves information for high frequencies, but it does not preserve the DC component of the image, which contains a lot of energy but is not very informative of the scene itself. Therefore the SNR of reconstructions are low, especially in full reconstructions where the DC component of the whole volume can be arbitrarily distributed between depth planes.

3 Additional Experiments on Operating Parameters

3.1 Arranging Aperture Locations in 2D

We also evaluate the effect of sweep pattern, i.e. the spatial arrangement of aperture locations on the 2D mask. We use two different types of arrangements, 1D and 2D, where both of them use the same number of measurements (e.g. 9×1 versus 3×3 for N=9) for the uniform step sweep patterns

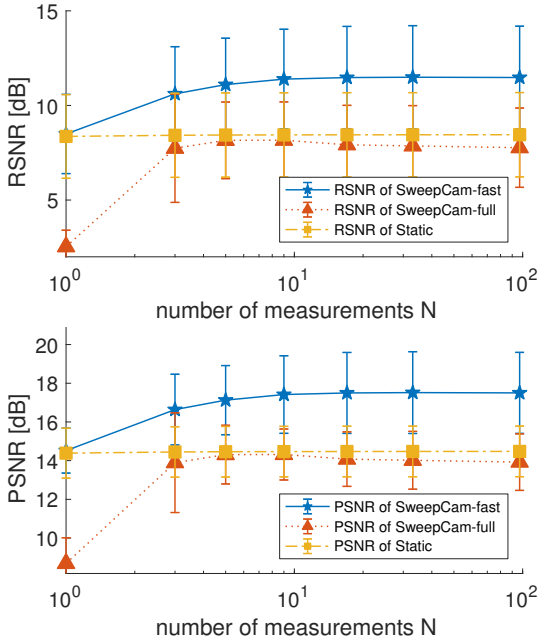

Fig. 1. RSNR and PSNR of image quality for different number of measurements.

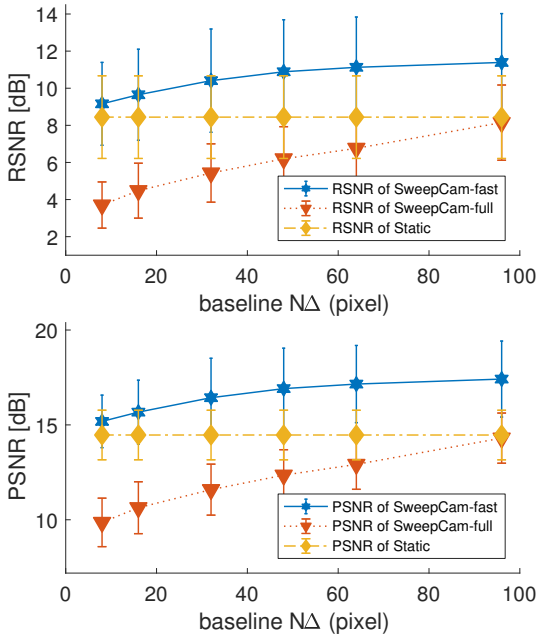

Fig. 2. RSNR and PSNR of image quality for different baseline.

under the same baseline of 96 pixels. Three scenes from the dataset of [1] are employed for the evaluation.

The effect of spatial arrangement is scene dependent, as shown in Fig. 4. For the Bull scene, which has a vertical and horizontal edges in its depth variations, 2D arrangement performs better than 1D with sufficient number of measurements ($N \geq 25$). This is because 2D arrangement can effectively mitigate the cross-plane interferences over edges with various angle, while 1D arrangement aquires only the horizontal parallax. Although 1D arrangement scores better than 2D for Sawtooth and Tsukuba, this is because these scenes have mainly vertical edges and thus fewer measurements of

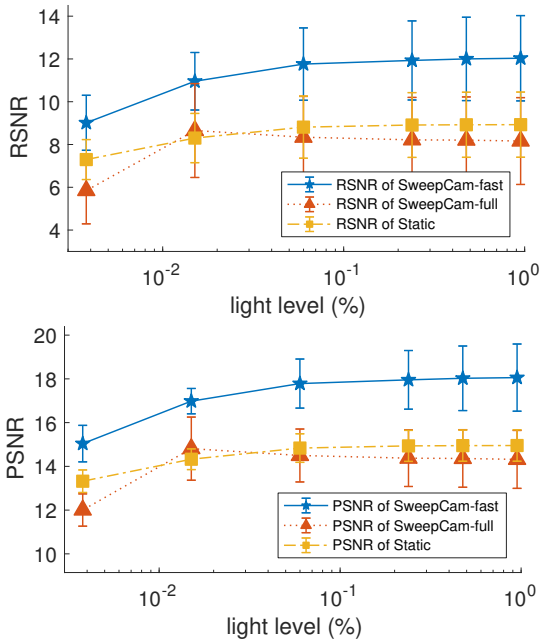

Fig. 3. RSNR and PSNR of image quality for different light level.

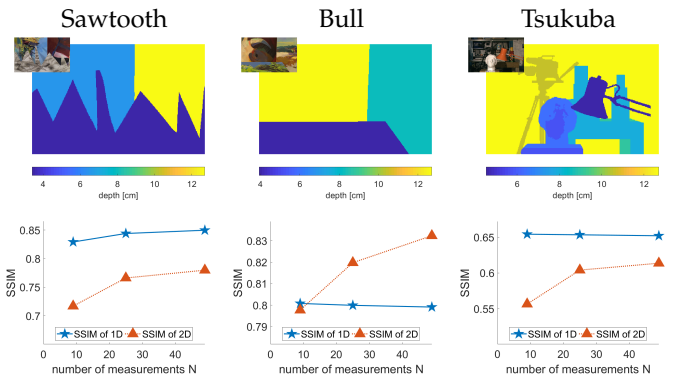

Fig. 4. *Reconstruction quality of SweepCam with 1D and 2D sweep pattern.* Top row shows the depth maps of scenes, with their texture in insets.Bottom row shows the transition of SSIM for each scene. Effects of different sweep patterns are scene dependent.

2D for the horizontal parallax causes the performance drops. In practice, we do not have prior knowledge on the scene, therefore it is desirable to acquire 2D measurements with sufficient sampling along both directions.

3.2 Length of M-sequence

Length of M-sequence affect the area of aperture, which determines the light efficiency of proposed camera. Therefore, we evaluate how the length of M-sequence effects on the quality of reconstructed image in simulation. We observe the performance transition by using various types of M-sequence, whose size is 15, 31, 63, 127 and 255 respectively, while other experimental setups follow those of *parameter A*. The measurement noise is applied considering the light efficiency which is decided by the size of each aperture size.

The result is shown in Fig. 5, with the averaged scores among 5 different scenes, and errorbar showing the standard deviation for each M-sequence length. We can observe that we have a peak on SSIM score at the length of 63 and

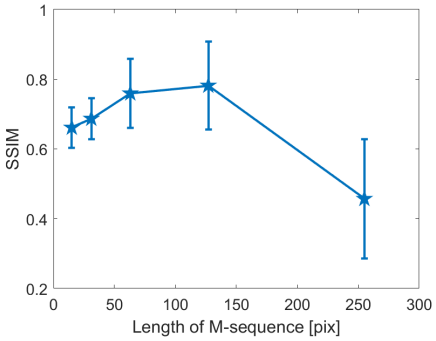

Fig. 5. Image quality of SweepCam over different length of M-sequence.

127. The transition is not monotonic due to two conflicting nature on the size of aperture. The longer the length of M-sequence is, the flatter the power spectrum becomes, which is desirable for the reconstruction performance. But this theory holds when we can ignore the effect of sensor boundary. In practice, too large an aperture leads to the performance deterioration since a significant portion of the measurement is cropped at the sensor boundary.

4 Calibration

4.1 Angle between Programmable Mask and Sensor

The focusing operator requires knowledge of the direction of mask rows and columns in the sensor coordinate. We try to align the mask parallel to the sensor, and estimate those directions via calibration. An LED is placed before the mask, and an image is captured when each row of the mask is turned on to transmit light. The direction of mask rows in sensor coordinate can be calculated from the lines detected in those images. The direction of mask columns in sensor coordinate is similarly obtained.

4.2 Point Spread Function (PSF)

We display a pattern on the mask and capture its PSF by moving a point light source, an LED, on a rail for different depth. Two more measurements were captured while translated patterns were displayed, and those were used to produce a focused image, which is cropped and used for reconstruction. We capture these images at six depths, and obtain the PSF at other depths by scaling the image captured at the nearest depth.

4.3 Distance between Mask and Sensor

The distance between mask and sensor can be solved from the scale of the PSF captured at different depths. We calibrate by setting up a ruler rail on the z-axis of the camera, moving a point light source at $z_1, \ldots, z_m$ on the rail, and recording the physical mask size l as well as corresponding PSF size $l_1, \ldots, l_m$. The first measurement gives a equation from similar triangle,

$$\frac{z_1 + z_0}{l} = \frac{z_1 + z_0 + d}{l_1}. \tag{5}$$

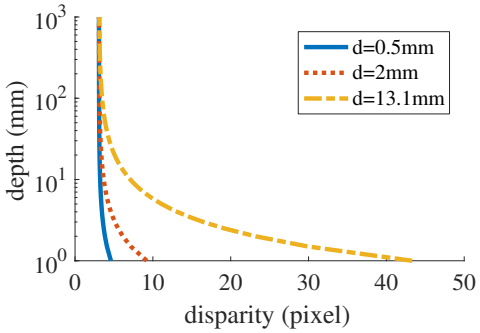

Fig. 6. *Scene point depth v.s. disparity for different distance* d. The vertical arrows indicate range of depth corresponding to 1 pixel change in disparity. Note larger d results in an larger range of indistinguishable depth, and close depth has smaller range of indistinguishable depth.

The distance between mask and sensor, d ,and the distance from start of ruler to mask, z_0, can be solved from the equation formed from m similar triangles for $m \geq 2$,

$$\begin{bmatrix} l_1 - l & -l \\ \vdots & \vdots \\ l_m - l & -l \end{bmatrix} \begin{bmatrix} z_0 \\ d \end{bmatrix} = \begin{bmatrix} -(l_1 - l)z_1 \\ \vdots \\ -(l_m - l)z_m \end{bmatrix} \tag{6}$$

The distance between mask and sensor affects observed disparity from the same depth. Calibration on our prototype yields 1.31cm between mask and sensor; its disparity from depth is plotted in yellow dashed line in Fig. 6. Decreasing the distance will make the prototype more suitable for microscopic applications.

4.4 Validation of Convolutional Model

We validate the convolutional model by placing an 8×8 LED in 1 inch array 5 cm in front of the hardware prototype, displaying a mask pattern that is the outer product of M-sequence of length 31, and capturing a measurement while one LED is turned on for each LED in the odd rows in the array. We annotate the center ofPSF from LED in row 5 column 5, predict the center of PSFs in other measurements based on disparity, and crop patches with those predicted centers. Those patches are shown in Fig. 7. The maximum value exceed 1 because cubic interpolation is used. The difference between patches extracted from other measurements and that from LED in row 5 column 5 is shown on the bottom image in Fig. 7. The small intensity in difference verify that translating a point light source results in a measurement with corresponding translated PSF, and the convolutional model holds.

5 Additional Experiments

5.1 Resolution Chart on Two Planes

We image two printed USAF charts located at different depth to demonstrate how SweepCam improves of resolution of lensless images, shown in Fig. 8. The near chart is 6.6 cm away containing group 0 and 1; the far one is 28 cm away containing group from -2 to 1. The static mask reconstructions is able to resolve 1.78 lp/mm on near chart and 0.44 lp/mm on far chart. The SweepCam full and fast reconstructions resolve 2.24 lp/mm on near chart and 0.70 lp/mm on far chart, as they can distinguish contributions from different depth planes.

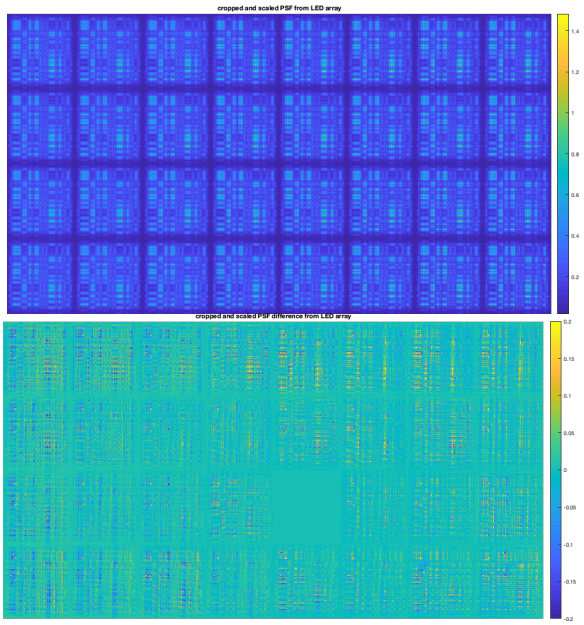

Fig. 7. Measurements from an LED array, aligned with predicted disparity. Small intensity in difference image verifies the convolution model.

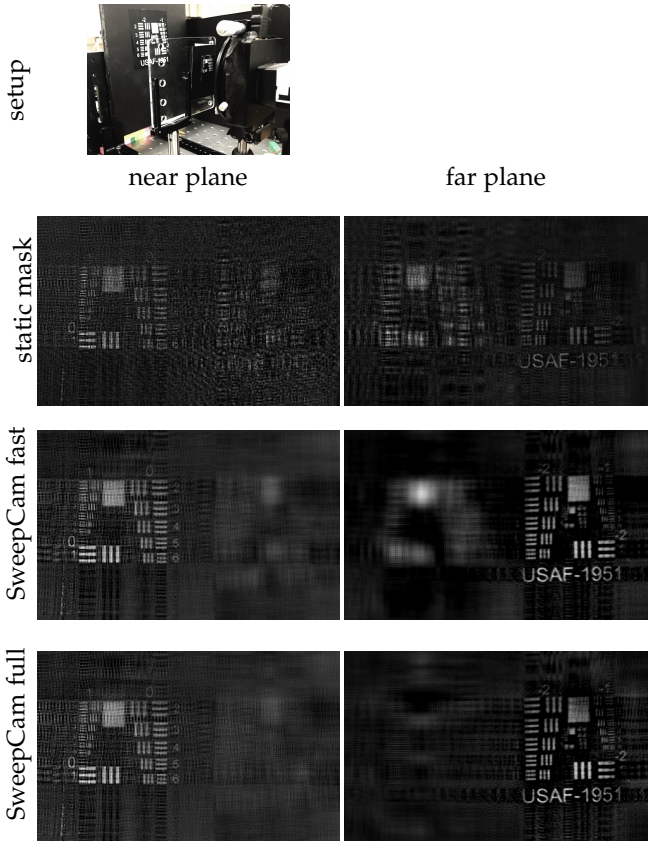

Fig. 8. Two USAF resolution charts are placed at different depths from the camera. SweepCam results are captured with 9×9 aperture locations across 0.4cm × 0.4cm.

References

[1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, Apr 2002. [Online]. Available: https://doi.org/10.1023/A:1014573219977