Polka Lines:
Learning Illumination Patterns and Reconstruction for Active Stereo

Supplemental Document

Seung-Hwan Baek Felix Heide
Princeton University

In this Supplemental Document, we present additional details and analysis. Specifically, we provide

- Additional prototype details of diffractive optical elements and stereo cameras.
- Details on the experimental real-time capture system.
- Optimization details of designing a DOE phase map for a specific illumination pattern.
- Examples of the stereo NIR dataset.
- Calibration details including geometric parameters of the stereo cameras and the illumination module and radiometric calibration of the DOE-laser module.
- Details on self-supervised finetuning and the proposed network architecture.
- Analysis on the DOE initialization in the proposed end-to-end training.
- Insights from Polka Lines illuminations.
- Additional results on environment-specific end-to-end optimization.
- Comparison to existing illumination patterns and conventional DOE designs.

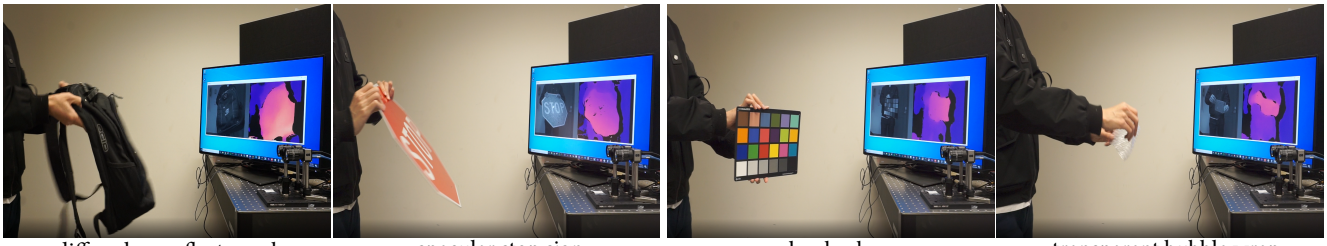
1. Additional Prototype Details

1.1. Diffractive Optical Element

We use a conventional photolithography process from *HoloOr* to prototype three learned DOEs for different ambient light powers. As we use the four-step lithography that produces 16 discrete height levels, we discretize the continuous height maps of the learned DOEs into discrete versions. Figure 3 shows the simulated illumination pattern before and after the discretization, demonstrating that the overall structure in the pattern remains the same except for the amplified zeroth-order diffraction. This zeroth-order diffraction pattern is also observed in the illumination image of the fabricated DOEs shown in the main paper which is partially handled by our self-supervised finetuning. We believe that these fabrication inaccuracies could be mitigated in a commercial photolithography process, e.g., the Intel RealSense D415 pattern does not exhibit a zeroth-order inaccuracy. We also note that our learned DOE design may be difficult to fabricate on a large scale due to its specialized structure. To remedy this, fabrication constraints could be incorporated into the proposed end-to-end design method in the future.

1.2. Stereo Cameras

While the exact stereo configuration is a system design choice, we chose the parameters of our system to match that of the Intel RealSense D415. Specifically, we chose the camera focal lengths, sensor specs, and the baseline to match the ones of the Intel RealSense D415 camera. Indeed, we originally planned to replace only their illumination DOE with our new DOE designs, however, this was not practical because of their proprietary system design. We also considered placing our



diffuse low-reflectance bag | specular stop sign | colorchecker | transparent bubble wrap

Figure 1. We demonstrate a real-time capture system from our Polka Lines prototype, reconstructing depth for several challenging objects in motion. We refer readers to the Supplemental Video.

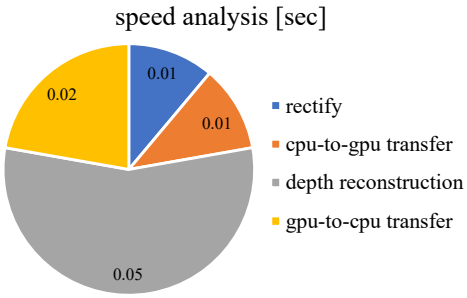


Figure 2. We measure the per-frame processing time for each stage of the live-capture program.

illumination module next to the Intel RealSense D415's stereo camera to enable effective active-stereo imaging using their highly-optimized camera. Unfortunately, this was also infeasible due to the required position of the illumination in between the stereo cameras which is unfortunately occupied by the Intel projector. Due to these challenges, we chose to build our own academic prototype with affordable elements.

2. Real-time Capture Demonstration

We developed a live-capture system that acquires stereo images and estimates a disparity map at 10 frames per second (FPS) as shown in Figure 1. Target objects are a low-reflectance diffuse bag, a highly-specular red stop sign, a ColorChecker, and a transparent bubble wrap. Note that even though we did not include such diverse reflectance in the training dataset, our reconstruction network together with the learned illumination enables effective reconstruction of the objects. We refer to the Supplemental Video for the video results. It is also worth to note that we observe temporal consistency between the estimated depth maps at different frames without employing any temporal smoothing loss in the reconstruction network.

We use a desktop computer with an NVIDIA GeForce RTX 3080 and the input 12-bit images are fed to our reconstruction network. Our capture program is written in Python with multi-threaded programming for simultaneously capturing images and reconstructing a depth map. Specifically, the program consists of capturing the stereo images using the camera APIs, rectifying the images with the calibration data, and estimating a disparity map using our reconstruction network. To quantify the latency of our live-capture program, we measure the elapsed time for each stage by averaging over 50 frames. Figure 2 shows the latency for each stage.

Note that the current capture software is not optimized. C++ implementation instead of the high-level Python API could provide speedup. We also expect that employing recent inference-dedicated network libraries such as NVIDIA TensorRT can further reduce the inference time of the neural network.

3. DOE Phase Design

In addition to solving for optimal illumination patterns, the proposed differentiable image formation model can be repurposed for designing DOEs that produce specific target illumination patterns. We formulate this as an optimization problem of minimizing the difference between the target pattern image I_{target} and the simulated illumination image I_{illum} for a given phase map of the DOE ϕ as

$$\underset{\phi}{\text{minimize}}\, \mathsf{MSE}(I_{\text{illum}}(\phi), I_{\text{target}}), \tag{1}$$

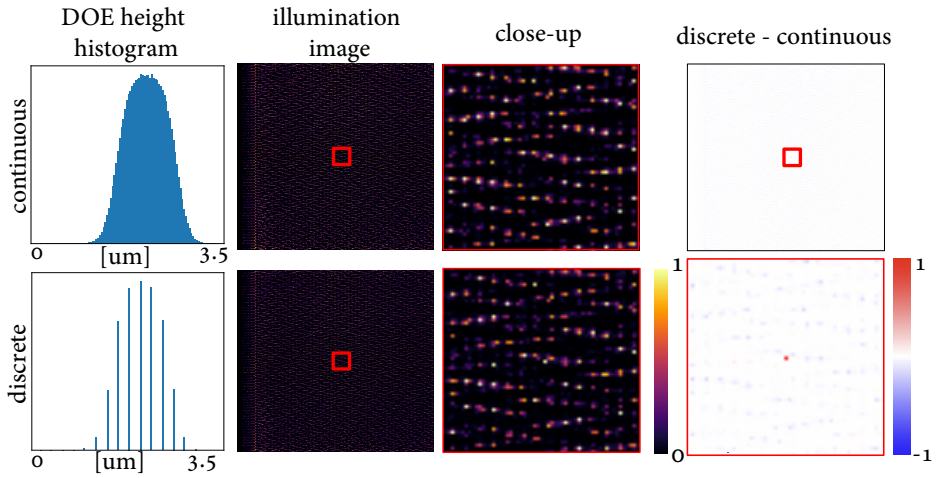


Figure 3. We discretize the optimized DOE height into 16 levels for photolithography fabrication. In simulation, the structure of the illumination image is maintained after the discretization process, except for the amplified zeroth-order diffraction.

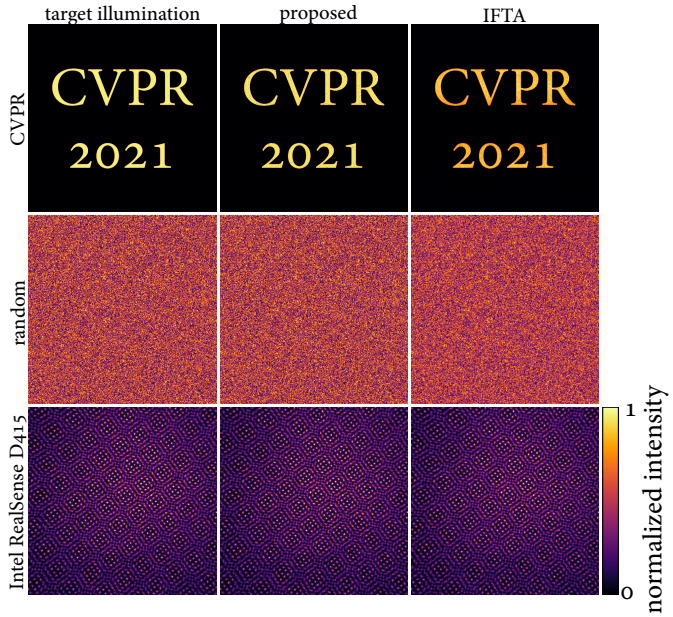


Figure 4. Our differentiable image formation can be used for designing a DOE that produces a desired target illumination pattern. Our method outperforms the commonly employed iterative Fourier transform method [1] and possesses additional advantages through design flexibility on the image formation and the loss function.

where MSE is the mean squared error. As computing the illumination image $I_{\text{illum}}(\phi)$ consists of differentiable operations based on our image formation model, we can solve this problem by relying on automatic differentiation using the Adam optimizer. Figure 4 shows target images and our reconstructions. We compare our method to the state-of-the-art iterative Fourier transform method [1] that indirectly solves the optimization problem. Our method not only outperforms this baseline in terms of reconstruction accuracy but also provides design flexibility by changing the image formation model and the loss function on demand.

4. NIR-stereo Dataset

Our method uses two NIR-stereo datasets, one for training in simulation and the other for finetuning the experimental prototype. For the synthetic training, we modify the RGB-stereo dataset [3] as described in the main paper, resulting in 21718 training images and 110 testing images. For finetuning, we capture 76 real-world stereo images of indoor scenes.

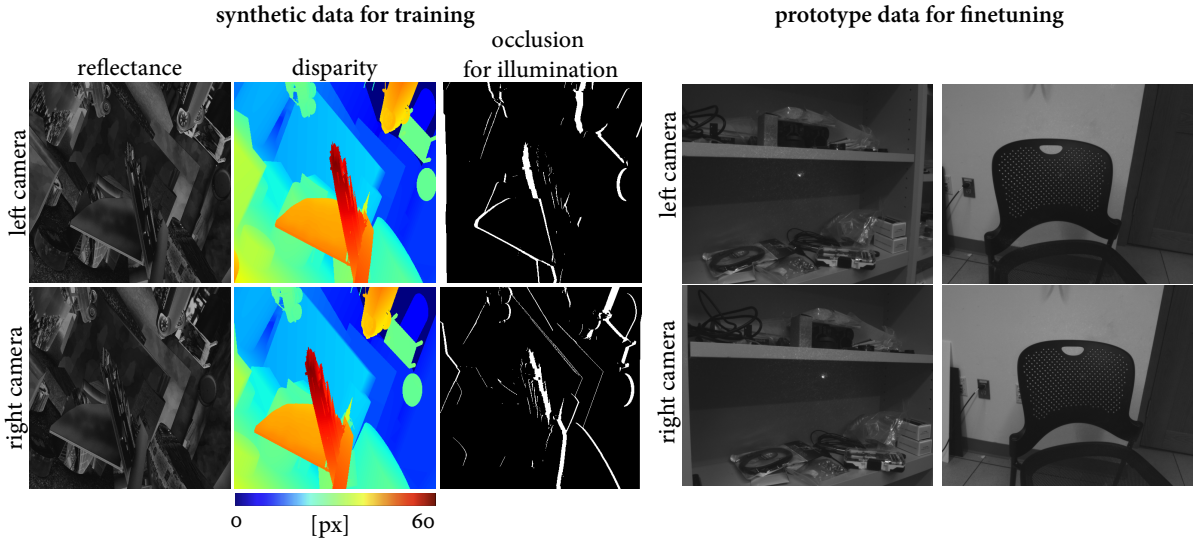


Figure 5. Examples of our NIR-stereo datasets for synthetic training and finetuning.

Figure 5 shows a sample from each dataset with varying reflectance and geometric complexity.

5. Calibration

5.1. Geometric Calibration

We calibrate our experimental prototype for efficient stereo matching in the rectified domain. We capture a checkerboard at different positions and obtain the camera intrinsics, the distortion coefficients, and the extrinsic between the stereo cameras. The average reprojection error was 0.6 pixels. For each input stereo frame, we rectify the stereo images using the calibration data and feed them to the reconstruction network.

Then, we obtain the illumination images of the fabricated DOEs. For each DOE, we illuminate a white wall at a distance of 50 cm from the camera, while ensuring that the intensity of the illumination pattern is in the observable dynamic range of the stereo cameras. We capture the stereo images of the wall with and without the structured-light illumination. Using the no-illumination images as background, we compute the illumination images at the stereo viewpoints. Undistortion and rectification are applied to the illumination images. This procedure provides a high-quality illumination image at the rectified illumination viewpoint which can be used for the reconstruction network.

5.2. Radiometric Calibration

In order to ensure a fair comparison between different illumination patterns, we use the same illumination power across different patterns. In synthetic experiments, this is achieved by using the same parameter value of the laser power β. For the Intel RealSense D415 pattern, we obtain the power-normalized illumination pattern to apply the laser power β. To this end, we estimate the optimal illumination power β that reconstructs the captured Intel RealSense D415 pattern as

$$\underset{\beta,\phi}{\text{minimize}}\, \mathsf{MSE}(I_{\text{illum}}\left(\phi,\beta\right), I_{\text{target}}). \tag{2}$$

Once the optimization converges, we normalize the D415 illumination image with the estimated β and use the same illumination power for its illumination image as the parameter value used for our end-to-end learning.

For real-world experiments, we use an integrating sphere of 12 mm input aperture (Thorlabs S142C) to measure the average illumination power of the Intel RealSense D415 illumination and our illumination. Our pattern exhibits higher peak power (not average power) than the Intel pattern, because the Intel RealSense D415 pattern has larger Gaussian-shaped dots while our pattern consists of smaller dots. Figure 7 shows the zoom-ins of Figure 11 in the main paper. Hence, for the same average power, our pattern features higher peak power with sparser dots.

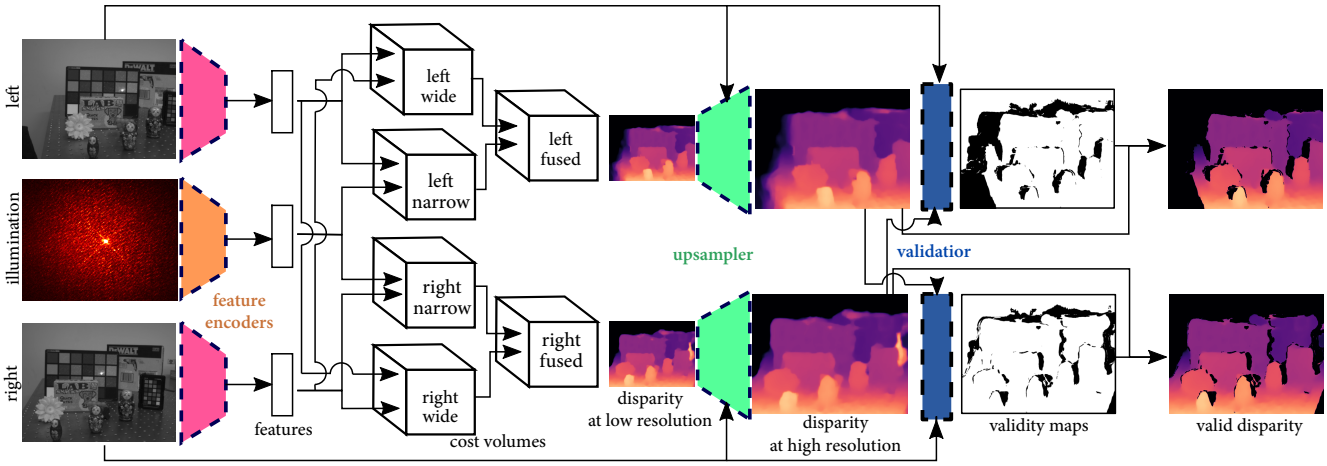


Figure 6. For finetuning, we extend the network architecture. We extract features for the left/right stereo images and the illumination image using the convolutional feature encoders. For each view, two cost volumes are constructed in narrow and wide baselines, fused into a multi-baseline cost volume. We estimate a disparity map for this cost-volume at a low spatial resolution which is upsampled to a original-resolution disparity using an edge-aware convolutional upsampler. The estimated disparity maps of the left and right views are then used for estimating validity maps that account for occlusion using a convolutional validator. The final disparity maps are obtained by removing the invalid region from the disparity estimates.

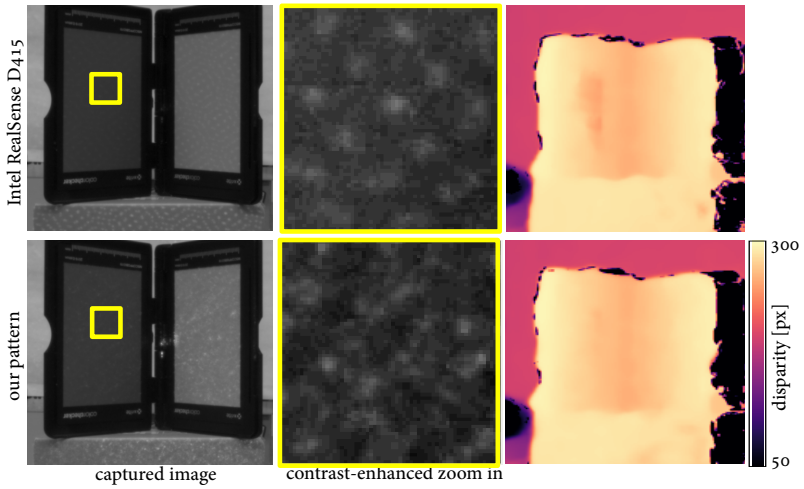


Figure 7. Radiometric comparison. Our pattern consists of small slanted dots with varying intensities while the Intel RealSense D415 pattern projects larger regular dots, hence average intensity is similar.

6. Reconstruction

6.1. Self-supervised Finetuning

To handle the domain gap between the simulation and the real-world inputs, we apply self-supervised finetuning for our reconstruction network. Figure 6 shows the overview of the trinocular reconstruction network for finetuning. There are two major differences to the network used in simulation. First, we estimate disparity maps for both left and right views. This is implemented by computing the right view disparity in the same way as computing the left view disparity which is described in the main paper. Second, we introduce a validation network that estimates validity maps of the estimated disparity. Inspired by left-right consistency approaches [2], we warp the estimated left/right disparity maps to the other view and compute the difference with the original disparity maps. This difference and the stereo images are fed to the validation network as inputs. In summary, the changes of the network architecture and the loss function enables effective handling of challenging regions such as large occlusion and strong specularity which are often observed in the real-world inputs. Our finetuning is specifically

Feature Extractor		
Name	Type	Channels
input	camera or illumination image	1
down	3×(conv-k5-s2-d2-p2)	32
res	6×(res-k3-s1-d1-p1-BN-LRelu)	32
conv	conv-k3-s1-d1-p1	32
output	feature	32
Cost-Volume Filter		
input	3D cost volume	32
conv3	4×(conv3-k3-s1-d1-p1-BN-LRelu)	32
conv3	conv3-k3-s1-d1-p1	1
output	filtered 3D cost volume	1
Convolutional Upsampler		
input	high-res. image and bilinear-upsampled disparity	32
conv	conv-k3-s1-d1-p1-BN-LRelu	32
res1	conv-k3-s1-d1-p1-BN-LRelu	32
res2	conv-k3-s1-d2-p1-BN-LRelu	32
res3	conv-k3-s1-d4-p1-BN-LRelu	32
res4	conv-k3-s1-d8-p1-BN-LRelu	32
res5	conv-k3-s1-d1-p1-BN-LRelu	32
res6	conv-k3-s1-d1-p1-BN-LRelu	32
conv	conv-k3-s1-d1-p1-Relu	1
output	high res. disparity	1
Validation Network for Finetuning		
input	image and warped image from the other view	2
conv	conv-k3-s1-d1-p1-BN	32
res	5×(res-k3-s1-d1-p1-BN-LRelu)	32
conv	conv3-k3-s1-d1-p1-Sigmoid	1
output	invalid mask from zero to one	1

Table 1. Network architectures. conv/res-k(x)-s(y)-d(z)-p(q) describes a convolution or residual layer with a kernel of $x \times x$ window, stride y, dilation rate z, and padding q.

formulated as the following optimization problem,

$$\begin{aligned}
&\underset{\theta,\vartheta}{\text{minimize}}\, \mathcal{L}_u + \tau\mathcal{L}_v + \kappa\mathcal{L}_d, \\
&\mathcal{L}_u = \mathsf{MSE}\left(J^{\text{L/R}} \odot V_{\text{est}}^{\text{L/R}}\left(\vartheta\right), J_{\text{est}}^{\text{L/R}}\left(\theta\right) \odot V_{\text{est}}^{\text{L/R}}\left(\vartheta\right)\right), \\
&\mathcal{L}_v = \mathsf{CE}\left(V_{\text{est}}^{\text{L/R}}\left(\vartheta\right), \mathbf{1}\right), \\
&\mathcal{L}_d = \mathsf{MSE}\left(\nabla D_{\text{est}}^{\text{L/R}}\left(\theta\right)\right),
\end{aligned} \tag{3}$$

where $V_{\text{est}}^{\text{L/R}}$ are the estimated left/right validity maps and $D_{\text{est}}^{\text{L/R}}$ are the corresponding disparity maps. $\mathcal{L}_u$ computes the mean squared error between the input and the estimated sensor images via validity-weighted warping: $J_{\text{est}}^{\text{L/R}} = \mathsf{warp}\left(J^{\text{R/L}}, D_{\text{est}}^{\text{L/R}}\right)$. $\mathcal{L}_v$ is the cross-entropy loss on the validity maps to avoid the trivial solution of making the validity as zero. $\mathcal{L}_d$ is the disparity smoothness loss to cope with real-world challenges in correspondence matching. τ and κ are the balancing weights set as 0.01 and 0.0001. The parameters of the reconstruction network θ are finetuned, while the validation network parameters ϑ is trained from scratch. We train over 5 epochs during finetuning. For visualizations, we threshold with a validity map to handle large occlusion.

6.2. Network Details

We provide network architectures in Table 1 including feature extractor, cost-volume filter, convolutional upsampler, and validation network for finetuning.

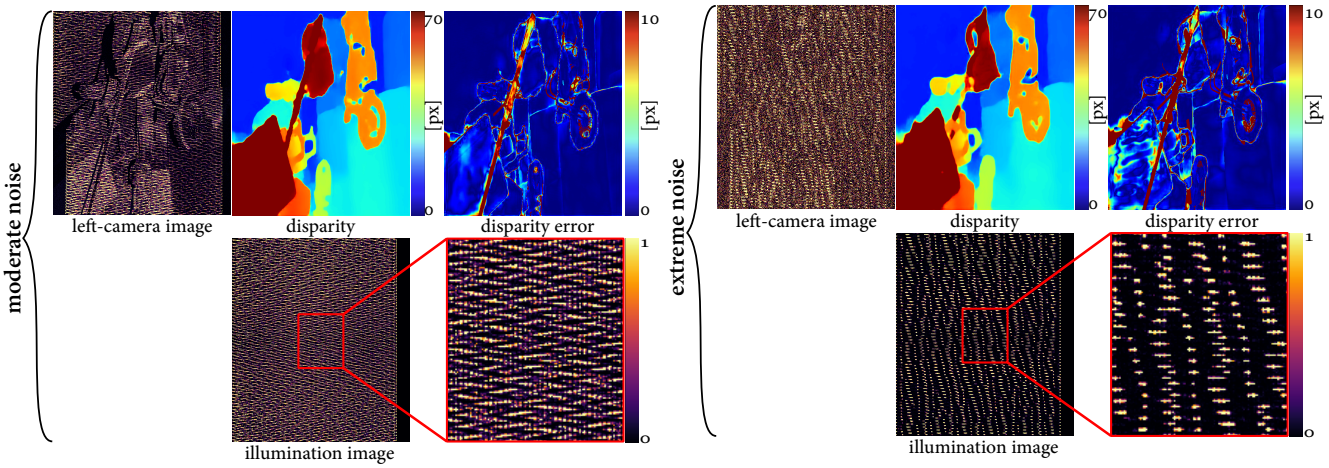


Figure 8. Optimized illumination and depth reconstruction for different noise levels. When the noise is moderate, the illumination pattern becomes dense with varying-intensity Polka Lines to provide dense correspondence cue. In contrast, severe noise, e.g., due to strong ambient illumination, makes the illumination pattern sparse with high intensities to stand out of the noise floor.

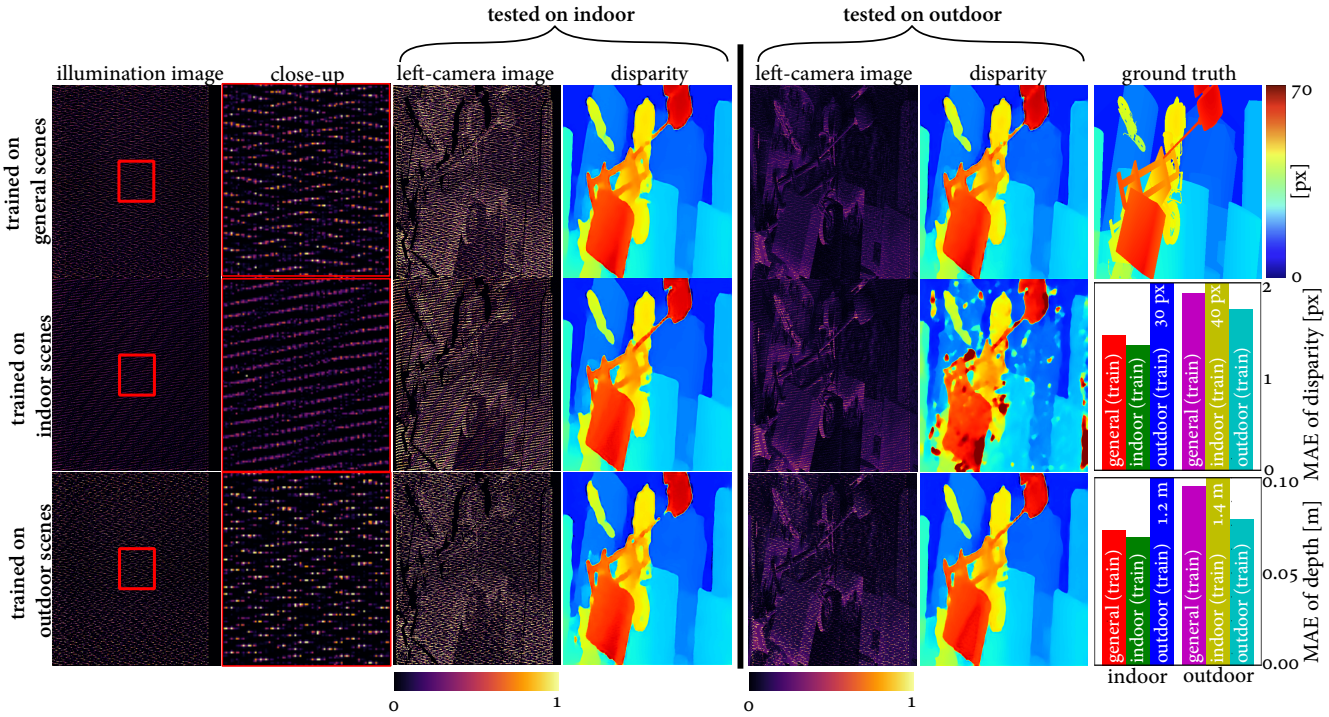


Figure 9. Our method enables us to design illumination patterns tailored for indoor, outdoor, or general environments.

7. Analysis

7.1. DOE Initialization and Evaluation

In our end-to-end training, we initialize DOE phase from a uniform random distribution from zero to 2π. We also tested two different DOE initializations, an all-zero phase initialization and a 2D diffraction grating phase initialization as seen in Figure 11. All initializations result in similar Polka Lines patterns as shown in Figure 10.

7.2. Polka Lines Illumination

Our learned Polka Lines illumination features high density slanted dotted-line structures, each of which consists of small-size dots. We note that this pattern has not been hand-engineered but was computationally found. These features are intu-

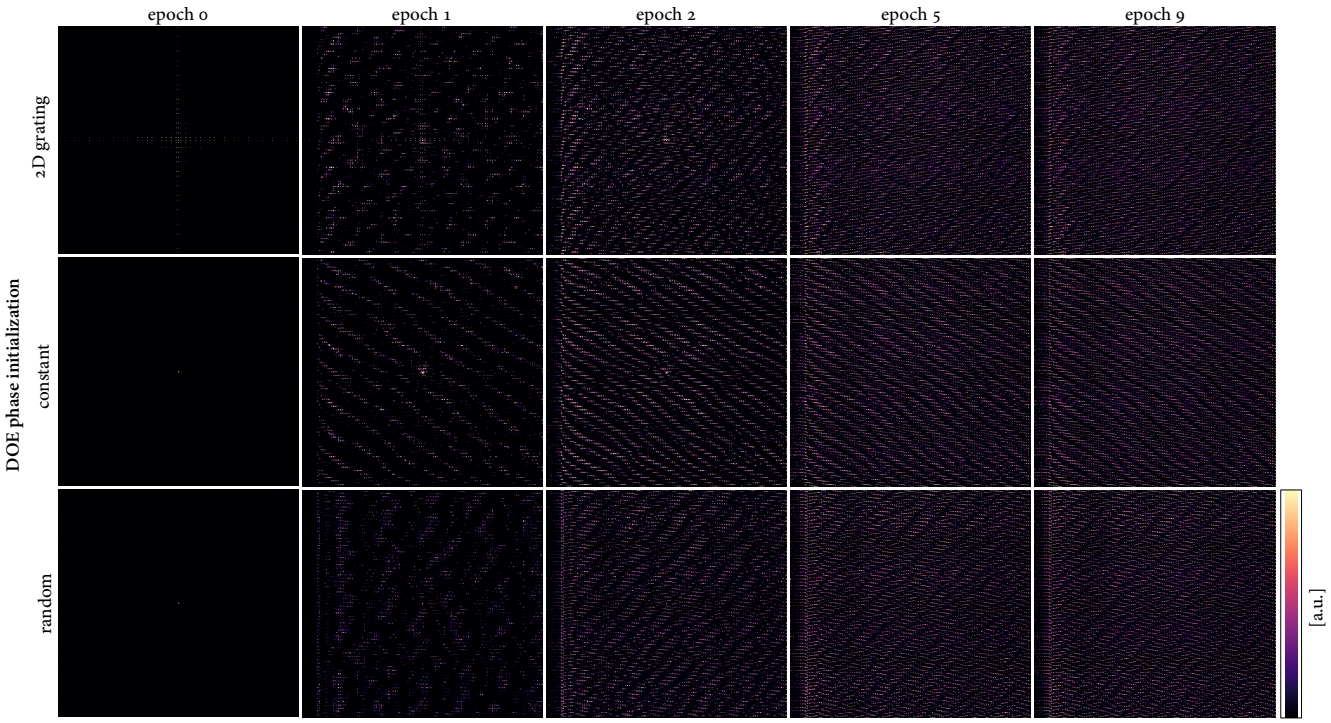


Figure 10. We test three different DOE initialization for our end-to-end training. The learned illumination images converge to similar Polka Lines patterns after completing training procedure.

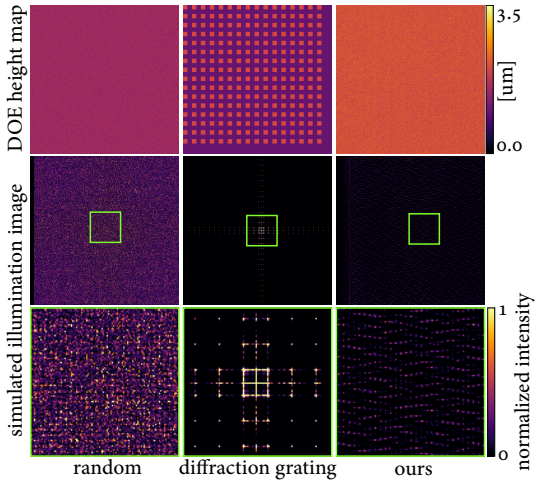


Figure 11. Our differentiable image formation can be applied to arbitrary DOE height maps, including random DOE height maps (left) and the 2D diffraction grating (middle), enabling the end-to-end design of illumination pattern for active-stereo systems.

itively helpful for active stereo imaging and some of them can be found in existing illumination patterns such as the Heptagon projector pattern in the Intel RealSense D415. Our end-to-end optimization method provides meaningful insights on how the trade-off between these properties should be maintained in the form of a DOE designed for a specific environment and imaging configuration.

7.3. Environment-specific Illumination Design

Our method facilitates incorporating system and environmental parameters in the image formation model, allowing us to design an illumination pattern tailored to the given scene. Specifically, we evaluate the learned patterns in terms of ambient light and noise level.

Measurement noise is critical for robust depth estimation and becomes strong in challenging environments, e.g., low-reflectance scene objects, strong ambient illumination, and long-range objects. Figure 8 shows the optimized illumination

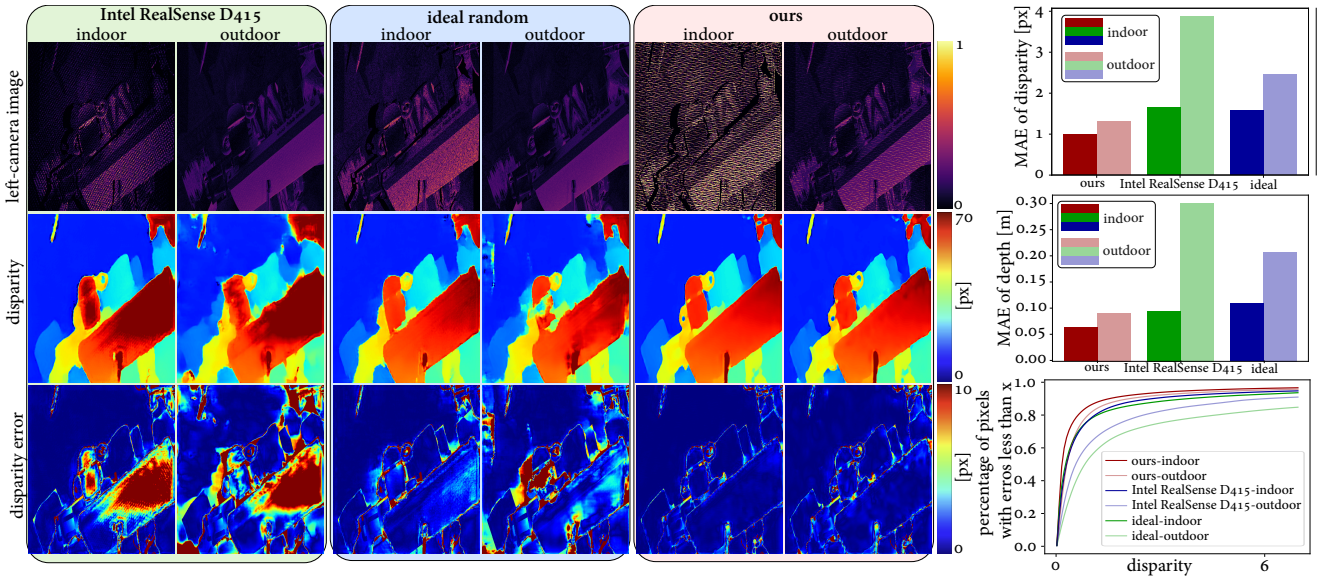


Figure 12. We compare the Intel RealSense D415 pattern, ideal random pattern, and our learned Polka Lines pattern in simulation. Our pattern outperforms the two hand-crafted illumination in all tested metrics.

images and corresponding depth reconstructions for the moderate and the extreme noise levels. The standard deviations of the Gaussian noise are 0.02 and 0.6 respectively. Extreme noise makes the illumination pattern sparse with high-intensity dots. In the moderate noise case, we obtain dense varying-intensity Polka Lines in the illumination image, providing high-quality depth reconstruction.

We also test varying ambient light power and laser power to simulate indoor and outdoor conditions by setting the parameter values of the ambient light power and the laser power as follows: indoor ($\alpha = 0.0, \beta = 1.5$), outdoor ($\alpha = 0.5, \beta = 0.2$), and general ($\alpha \in [0, 0.5], \beta \in [0.2, 1.5]$). We empirically chose the values of α and β by capturing the relative intensity differences of laser illumination and ambient light indoors and outside. We train a DOE and a reconstruction network for each of the configurations. Figure 9 shows the optimized illumination patterns and their performance tested on both indoor and outdoor environments. We learn dense Polka Lines in the indoor scenes to provide many features for correspondence matching. For the outdoor scenes, we obtain sparse high-intensity Polka Lines, providing robustness against the strong ambient light and relatively weak laser power. When training on general environments, we learn Polka Lines with varying intensities with moderate density.

We fabricated and placed these three DOEs for indoor, outdoor, and general conditions by mounting them on a manual rotation stage. In the future, we envision using mechanically interchangeable DOE configurations or multiple projectors to adapt to the environment. Overcoming high manufacturing cost of multiple illumination modules and reducing their form factor are exciting steps for future work.

7.4. Illumination Patterns of Conventional DOEs

Our image formation model for active stereo involves computing the illumination image for a given DOE profile. As a sanity check on our image formation model, we compute the illumination patterns for two conventional DOE designs: random height DOE and 2D diffraction grating. In theory, their illumination patterns are random dots and regular grid patterns with decaying intensity profile as the diffraction order increases. Figure 11 shows that our simulated illumination images contain these characteristics.

7.5. Comparison with Other Illumination Patterns

We compare our learned Polka Lines pattern to the Intel RealSense D415 pattern and the ideal random-dots pattern in simulation. Figure 12 shows that the Intel RealSense D415 pattern contains sparse and low highest intensity feature points with repeated structure, leading to reconstruction artifacts. It is worth noting that there is a disparity bias in the estimates of the Intel RealSense D415 pattern. The bias is at around 10 px which corresponds to the distance between the two nearest dot features. Based on this observation, we speculate that this failure may come from the characteristics of the two-scale Intel RealSense D415 features. That is, it has high-frequency 10 px-width dot features and also low-frequency

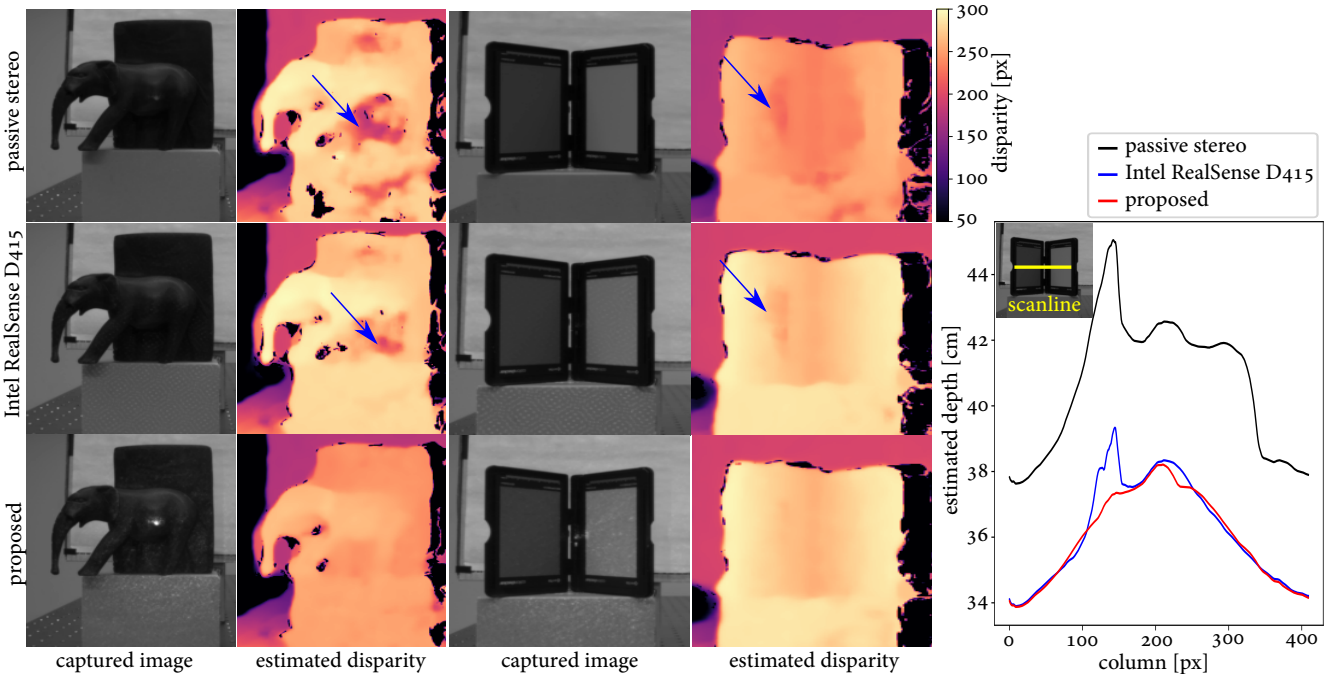


Figure 13. Our learned illumination pattern with varying-intensity dots outperforms passive stereo and the conventional fixed-intensity pattern (Intel RealSense D415 sensor) for high dynamic range of incident light. Blue arrows indicate estimation artifacts. We capture a V-shaped reflectance target (x-rite ColorChecker Pro Photo Kit) of which a scanline analysis reveals that the accurate reconstruction of the shape only by our Polka Lines pattern.

60 px-width heptagon features consisting of multiple dots. This could cause ambiguity when using the high-frequency or the low-frequency patterns used for correspondence matching, resulting in the disparity bias. Specifically, we observe that 10 px of disparity bias happens as the reconstruction network uses the high-frequency dots as matching features instead of the low-frequency components. Thus, this fundamental ambiguity in the Intel RealSense D415 pattern often leads to biased estimates. The ideal random-dot pattern provides high-quality depth reconstruction on average, however, reconstruction quality degrades under high ambient light conditions due to the scattered light energy by the random phase distribution. In contrast, our Polka Lines pattern provides accurate reconstructions with dense features and varying-intensity dots that we learn from end-to-end optimization with the goal of accurate depth reconstruction.

Figure 13 shows the real-world comparison of the passive stereo, the Intel RealSense D415 pattern, and our Polka Lines pattern. Our Polka Lines design provides accurate reconstruction on feature-less objects. For additional analysis of the illumination intensity, refer to Figure 7 validating that our pattern provides higher peak power while maintaining average power.

References

[1] Pei-Qin Du, Hsi-Fu Shih, Jenq-Shyong Chen, and Yi-Shiang Wang. Design and verification of diffractive optical elements for speckle generation of 3-d range sensors. *Optical Review*, 23(6):1017–1025, 2016.

[2] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–279, 2017.

[3] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. arXiv:1512.02134.