End-to-end optimization of a lensless imaging system

Cindy Nguyen
Electrical Engineering
Stanford University
cindyn@stanford.edu

Abstract

Lensless imaging systems have been designed to capture photorealistic images with features such as depth and 3D structure at a fraction of the camera volume. This information can be captured through encodings generated from propagating light through a pseudorandomly coded mask placed directly in front of the camera sensor. However, current reconstruction algorithms for lensless imaging systems typically require solving a well-posed inverse problem with respect to the pre-designed phase mask, which limits the solution space of image reconstruction. We propose an end-to-end optimization framework that optimizes the phase mask and reconstruction algorithm jointly. Our results demonstrate a deep learning approach to improving lensless imaging capture and reconstruction.

1. Introduction

Modern imaging systems can be bulky, expensive, and relatively complex to manufacture. Notably, lenses take up significant volume in a camera not only due to the thickness of the glass of the lens, but also through the distance between the sensor and lens required for light focusing. Lensless imaging systems using a coded aperture, where the diffractive optical element (DOE) is placed directly in front of the sensor, offer an attractive alternative for reducing camera volume, while also reducing the associated costs and steps to manufacture a camera with a glass lens. These benefits can enable improved applications of camera systems in areas where size can be a limiting factor, such as *in vivo* microscopy, depth sensing on autonomous vehicles, and cell phone cameras.

We propose a new way of optimizing lensless imaging systems using end-to-end optimization. Our contribution is a network to learn a phase mask height map and appropriate Wiener filter for reconstruction.

2. Related Work

2.1. Lensless Imaging

A typical camera will use a lens to map points on a scene to a point on a CMOS sensor. In a lensless imaging system, computational reconstruction algorithms are required for demultiplexing the signal as each point source of the scene as each point is mapped to multiple points on the sensor array.

One example of a lensless imaging system is the FlatCam. The FlatCam places a binary coded aperture almost directly in front of the sensor array, where the coded aperture modulates the incoming light with a Hadamard-Walsh pattern of transparent and opaque features [3]. For a surface S in the field of view of the sensor, the sensor measurements *y* are described as

$$y = \Phi x + e \tag{1}$$

in which Φ represents the transfer matrix of the phase mask and *x* represents the image unrolled, and *e* represents noise in detection. To ensure a well-conditioned system that would allow for stable inversion and reduced computational complexity, their system utilizes separable phase masks that allow (1) to be rewritten as

$$Y = \Phi_L X \Phi_R^T + E \tag{2}$$

in which $X \in \mathcal{R}^{N \times N}$ represents the scene, $Y \in \mathcal{R}^{M \times M}$ represents the sensor measurements. Φ_L and Φ_R represent the separable mask as a outer product of two one-dimensional patterns. The separable masks were designed to reduce computational complexity. The image reconstruction algorithm is then based on solving the least-squares problem

$$\hat{X}_{LS} =_x ||\Phi_L X \Phi_R^T - Y||_F^2, \tag{3}$$

typically with a regularization term.

Although the system shows impressive results given a computationally simple reconstruction algorithm, the linear demultiplexing leads to noise amplification and reduced resolution. There is also about a 100 ms delay between capture

and image display, which may not be as acceptable in real-time systems such as virtual reality.

Recent advances on the FlatCam reconstruction use a prior-based reconstruction algorithm (such as the aforementioned Tikhonov regularized least squares problem) combined with a two-stage generative adversarial network. The first stage maps FlatCam measurements into an intermediate space. The second stage refines the mapping with a U-Net, an encoder-decoder convolutional neural network [7]. These results show more photorealistic images and show the promise in using deep learning to approach lensless imaging reconstruction.

Another lensless imaging system is the DiffuserCam, which provides single-shot volumetric imaging under incohrerent light using a diffuser placed over a sensor [1]. Each point source on the 3D surface creates a unique pseudorandom pattern on the sensor. With the help of compressed sensing algorithms, 3D renderings can be generated from a diffuser phase mask in front of a 2D sensor array. Contrary to the FlatCam, the calibration process does not require precise alignment, and the system is more light efficient than those that use amplitude phase masks. For the phase mask, the system uses a thin transparent diffuser with varying thickness. DiffuserCam uses similar matrix inversion problem as the FlatCam as part of its reconstruction algorithm combined with a CNN.

2.2. Deep Optics

To diverge away the typical pipeline of designing the phase mask first and then solving a modified inverse problem, we propose to take an end-to-end optimization approach. This method involves the joint optimization of an optical element design and the reconstruction algorithm over a large set of images. The reconstruction can be extended with training neural networks as part of the reconstruction method. This method has been successfully been demonstrated in optimizing achromatic extended depth of field, super-resolution imaging, and high dynamic range (HDR) imaging [9, 6].

3. End-to-end Lensless Imaging

Our task entails modeling a point spread function (PSF) to convolve images with and optimizing parameters of our reconstruction step. We used Wiener filtering as our reconstruction method as it has only single scaling factor to optimize, simplifying our model greatly.

3.1. Modeling Point Spread Functions

Point sources were first modeled from optical infinity, approaching the phase mask as a planar wave (Fig. 2). The phase mask is modeled to induced a phase delay of a complex-valued wave field, giving us

$$\phi(x, y) = \frac{2\pi\Delta n}{\lambda} h(x, y) \tag{4}$$

Here, Δn represents the refractive index difference between air and the phase mask material. Our phase mask is assumed to be polydimethylsiloxane (PDMS) with a refractive index of 1.432 [8]. Also, λ represents the wavelength of light, and $h(x, y)$ represents the thickness or height map of the phase mask.

We model a wave field U to have an amplitude A and phase ϕ_s incident on the phase mask as

$$U(x', y', z = 0) = A(x', y')e^{i(\phi(x',y')+\phi_s(x',y'))}, \tag{5}$$

where $U(x', y', z = 0)$ represents the light field coming out of the phase mask.

We then model the Fresnel propagation from the phase mask to the sensor using convolution [4]:

$$U(x, y, z) = \frac{e^{ikz}}{i\lambda z} \int\int U(x', y', 0) e^{i\frac{\pi}{\lambda z}[(x-x')^2+(y-y')^2]} dx'dy'. \tag{6}$$

After propagating to the sensor, we acquire the point spread function (PSF) from taking the intensity of the complex-valued wave field $|U(x, y, z)|^2$. To form the image recorded on the sensor based on a given image source, we perform a shift-invariant convolution of the given image and generated PSF p to give us the resulting image $I_\lambda \in \mathbb{R}^{3\times H\times W}$, the image propagated through a Fresnel lens designed for wavelength λ.[2].

3.2. Image Reconstruction

We utilize Wiener filtering, assuming circular boundary conditions by using symmetric padding. The Wiener filter operation is given by

$$\tilde{I} = \mathcal{F}^{-1}\left\{\frac{\bar{p}*}{|\bar{p}|^2 + \gamma}\mathcal{F}\left\{I_\lambda\right\}\right\}, \tag{7}$$

where $\bar{p}$ is the optical transfer function and γ is a learnable damping factor. This operation is applied to the convolved image I_λ to give our image reconstruction $\tilde{I}$.

3.3. End-to-End Framework

We optimize over a subset the Semantic Boundaries Dataset [5] for training and testing (Fig. 1). The training set is 8,498 images, and the validation set is 2,857 images. We use a batch size of 4. The images were center cropped and upsampled to 512×512 resolution. The height map and sensor were assumed to be the same resolution, each with a pixel pitch of $2\mu m$, which is on the smaller end of manufacturing limits and provides greater opportunity for better

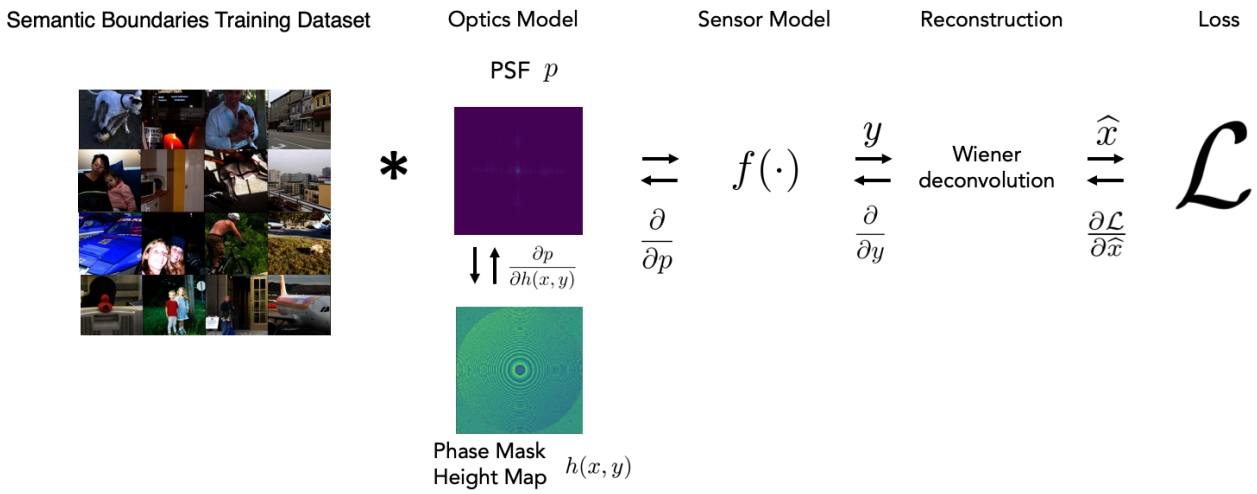

Figure 1. An end-to-end optimization pipeline for a lensless imaging system. Images are taken from the Semantic Boundaries training set and processed through an optics and sensor model before being Wiener filtered.

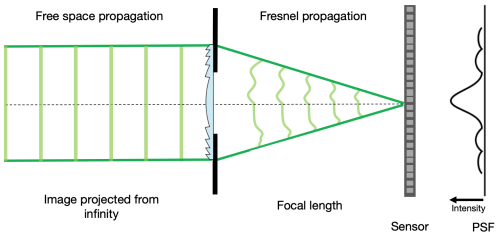

Figure 2. Modeling Point Spread Functions. We model a point light source of green wavelength coming from infinity through a Fresnel lens and use Fresnel propagation to propagate the light from the aperture to the sensor.

resolution in smaller focal lengths. Height maps were initialized with an in-focus Fresnel lens designed for a wavelength of 530 nm, which produces a PSF resembling a Dirac peak. We set our aperture to be as wide as possible.

We use an Adam optimizer with a learning rate scheduler that drops the learning rate after the loss difference does not exceed 1×10^{-4} for 10 iterations. We employed a mean-squared error loss

$$\mathcal{L}(\tilde{I}, I) = \sum_{c \in \{R,G,B\}} ||\tilde{I}_c - I_c||_2^2, \tag{8}$$

where $I \in \mathbb{R}^{3 \times H \times W}$ is the image from the dataset converted from sRGB to linear space (see Appendix). All processing operations were performed in linear space.

To validate our framework, we initialized the height map to resembled an in-focus Fresnel lens intended to focus a point source from optical infinity with a focal length of 50 mm. Without a reconstruction method, we set the loss to be the MSE between the ground truth and the resulting image. The height map ultimately converged to a Fresnel-like lens, which suggests that our optimal initialization provides the best solution for focusing the image.

3.4. Choosing Focal Lengths

To determine the optimal focal length defined by our aforementioned camera system parameters, we must find the distance at which the given pixel pitch on the phase mask gives a resolvable pixel pitch on the sensor of the same size. We wanted to optimize with the same pixel pitch on the phase mask and the sensor for simplicity. To do this, we use Bragg's law, $2d \sin\theta = n\lambda$, in which d represents the pixel pitch, n is a positive integer (we use 1), and λ is the wavelength of interest (we use 530 nm). The following θ represents our θ_{max}. By similar triangles we can determine the optimal focal length by using

$$\tan\frac{\theta_{max}}{2} = \frac{w/2}{f}, \tag{9}$$

where w represents the height of the spatial light modulator, and solving for f.

We can calculate the resolvable pixel pitch by using

$$p = 2f\tan(\frac{\theta_{max}}{2}) \tag{10}$$

to confirm that our system has a resolvable pixel pitch the same size as the pixel pitch of our phase mask. In our system, the optimal focal length is 7.7 mm. We test 25 mm, 5 mm, and 1 mm for comparison.

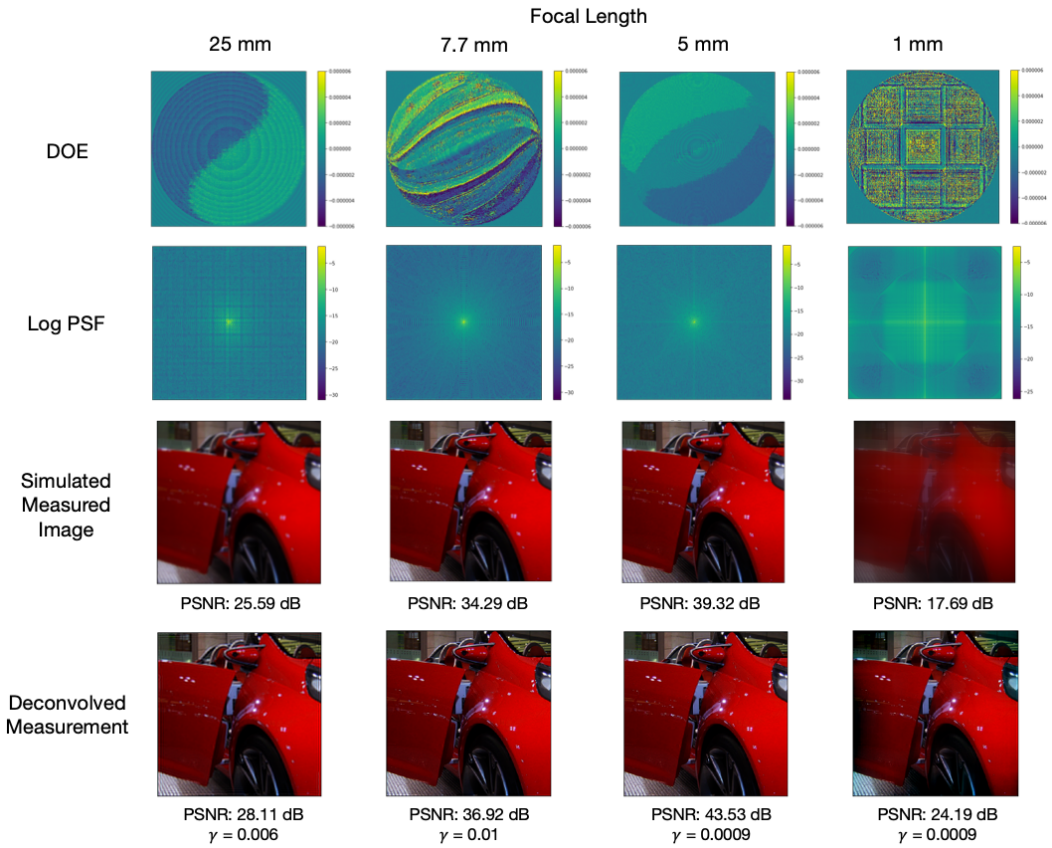

Figure 3. Optimizing the phase mask only. The height map is displayed in meters, and the PSF is displayed in logarithmic meters for visualization. The ground truth image is a test image from the Segmentation Boundaries dataset. We observe the highest PSNR for a Wiener filtered image at 5 mm focal length.

4. Results and Analysis

4.1. Optimizing the Phase Mask

We first choose to optimize the height profile of the DOE only. Figure 3 shows the learned DOE for each focal length, the PSF (displayed in logarithmic for easy visualization), the resulting image from being convolved with the PSF, and the resulting image after Wiener deconvolution with a hand-chosen damping factor. The damping factor was hand-chosen to optimize peak signal-to-noise ratio (PSNR) against the ground truth image.

Even though training at each focal length began with an in-focus Fresnel lens, we observed that training typically led the height map to diverge partially away from a perfect Fresnel lens. At lengths greater than 1 mm, we can still observe a Fresnel-like pattern underlying the height map. Interestingly, we see some patterns of opposing symmetry across these height maps. At a 1mm focal length, the network has difficulty converging on a height map that can properly given that the propagation distance is so small.

We note that the PSNR for both the simulated measured image (before reconstruction) and the deconvolved image is typically higher at 5 mm, which is not the optimal focal distance. However, we observed that at the start of training, the height map initialized at 7.7 mm provided smaller loss values before convergence. This result suggests that the height map might be easier to optimize at 5 mm.

4.2. Optimizing the Phase Mask and Wiener Damping Factor

We incorporated the Wiener damping factor γ into the learning process. We use the same height map initialization (in-focus Fresnel lens), and chose to initialize γ to values close to those hand-chosen from our previous experiment. We noticed that when we initialized γ with values further from these (e.g. 0.1 or 0.8), the height map took longer to converge and overall PSNR from the Wiener deconvolved image was not as high.

In Figure 4, we observe that, again, focal lengths of greater than 1 mm provide height maps that have some

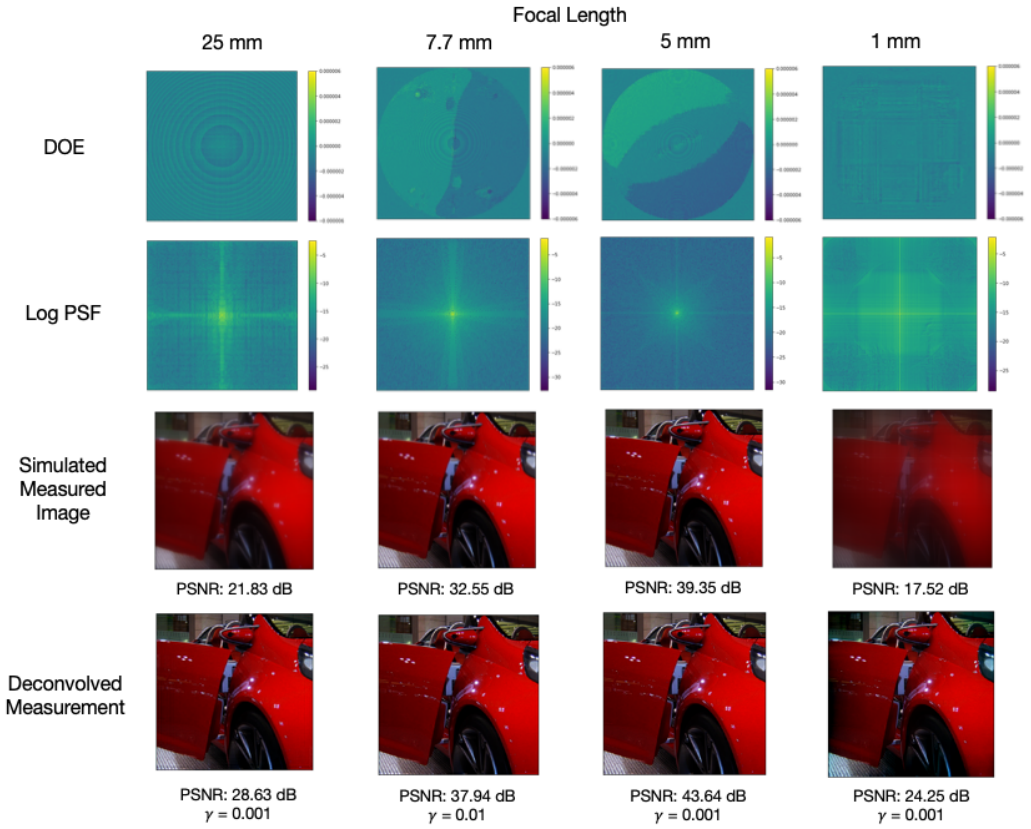

Figure 4. Optimizing the phase mask and Wiener damping factor. The height map is displayed in meters, and the PSF is displayed in logarithmic meters for visualization. We observed improved overall PSNR when the damping factor is learned jointly with the height map.

Fresnel-like structure. At 5 mm, we observe a visually very similar height map to that learned at 5 mm in optimizing the height map only. At 7.7 mm, we observe a simpler pattern than that learned in the previous optimization. At 25 mm, we no longer see the opposing symmetry but observe a more flattened Fresnel-like pattern. At 1 mm, we observe also a more flatten lens version as that of the previous 1 mm focal length optimization.

Notably, we see a decrease in PSNR in simulated measured images compared to those seen in the previous experiment. However, the deconvolved image provides higher PSNR than those in the previous experiment. These results suggest that the network is more heavily relying on the reconstruction method to provide a clearer image, perhaps providing a simulated measured image that, although has a lower PSNR, performs better with the specific convolution performed in Wiener filtering. We also observed that the damping factor would not diverge far $(< 1 \times 10^{-7})$ from our initialization during the learning process.

5. Discussion

This work utilizes the paradigm of co-designing the phase mask encodings and the method of reconstructing an image based on those encodings. We provide results of two simulation experiments: optimizing the phase mask only and optimizing both the phase mask and the reconstruction parameters. We observed an improvement in PSNR in images reconstructed with an optimized Wiener filtering parameter, showing the potential of end-to-end optimization in lensless imaging systems.

Limitations and Future Directions

While training the damping factor, we observed that the network heavily relied on the factor's initialization to further optimize the height map, rather than altering the damping factor by more than 1×10^{-7} and learning possibly a completely different height map. This event occurred likely due to the network beginning training in a local optima. We would like to try different hyperparameters that would allow the network to explore the solution space better.

We would like to explore different loss functions based on other distortion metrics, in addition to regularizers that may allow for better deconvolution parameters. Additional regularizers could enforce the network to more heavily rely on the damping factor during reconstruction. We would also like to explore the possibility of altering the amplitude $A(x', y')$ involved in modeling the wave field to get better reconstruction.

We would also like to optimize a U-Net to provide additional image reconstruct after Wiener deconvolution. Due to time constraints, we were unable to fully optimize a U-Net to provide higher PSNRs than those provided from only Wiener deconvolution. We believe that, with data augmentation, we could get a sufficiently large training set to conduct this step.

Conclusion

We present an approach to perform end-to-end optimization of a lensless imaging system. With improved reconstruction algorithms, lensless cameras may provide great potential as a miniature system for everyday image capture. It lends itself to a wide array of applications given the cheaper cost, quicker manufacturing times, and smaller volume. Our work demonstrates the potential to co-design the mask and reconstruction algorithm. In future works, we hope to incorporate CNNs to learn features helpful for reconstruction.

Acknowledgments

I would like to thank Chris Metzler and Gordon Wetzstein for providing invaluable guidance over the course of this project. I would also like to thank the members of the Stanford Computational Imaging Lab for insightful discussions and suggestions.

References

[1] N. Antipa, G. Kuo, R. Heckel, B. Mildenhall, E. Bostan, R. Ng, and L. Waller. Diffusercam: lensless single-exposure 3d imaging. *Optica*, 5(1):1–9, 2018.

[2] N. Antipa, S. Necula, R. Ng, and L. Waller. Single-shot diffuser-encoded light field imaging. In *2016 IEEE International Conference on Computational Photography (ICCP)*, pages 1–11. IEEE, 2016.

[3] M. S. Asif, A. Ayremlou, A. Sankaranarayanan, A. Veeraraghavan, and R. Baraniuk. Flatcam: Thin, bare-sensor cameras using coded aperture and computation. *arXiv preprint arXiv:1509.00116*, 2015.

[4] J. W. Goodman. *Introduction to Fourier optics*. Roberts and Company Publishers, 2017.

[5] B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *International Conference on Computer Vision (ICCV)*, 2011.

[6] C. A. Metzler, H. Ikoma, Y. Peng, and G. Wetzstein. Deep optics for single-shot high-dynamic-range imaging. *arXiv preprint arXiv:1908.00620*, 2019.

[7] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[8] F. Schneider, J. Draheim, R. Kamberger, and U. Wallrabe. Process and material properties of polydimethylsiloxane (pdms) for optical mems. *Sensors and Actuators A: Physical*, 151(2):95–99, 2009.

[9] V. Sitzmann, S. Diamond, Y. Peng, X. Dun, S. Boyd, W. Heidrich, F. Heide, and G. Wetzstein. End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging. *ACM Transactions on Graphics (TOG)*, 37(4):1–13, 2018.

Appendix

The complex-valued field U is modeled as

$$U(x, y) = A(x, y)e^{i\phi}, \tag{11}$$

where A represents amplitude and ϕ represents phases. Cameras typically apply a gamma correction before compression in the image processing pipeline. For this reason, we apply the following sRGB to linear conversion to each image before the image is processed in the optics module:

$$\gamma^{-1}(u) = \begin{cases} \frac{25u}{323} & u \leq 0.040045 \\ (\frac{200u+11}{211})^{\frac{12}{5}} & \text{otherwise} \end{cases}, \tag{12}$$

where u represents the given image. This conversion allows us to perform processing operations in linear space, although the converted image does not follow the sensitivity of the human visual system.

We are also aware of the diffraction limit of the systems. Diffraction-limited spot sizes are defined by Abbe's law as

$$r = \frac{\lambda}{2NA}, \tag{13}$$

where λ is the wavelength and $NA = n\sin(\theta_{max})$ represents the numerical aperture. Here, n is the refractive index of the medium (in this case $n_{air} = 1$). θ_{max} is the maximum diffraction angle of the spatial light modulator, and we can calculate this using $\theta_{max} = \arcsin(\frac{\lambda}{2p})$, where p is the pixel size. In our case, r can be approximated by p.

Source code: `github.com/ccnguyen/lensless`