Optimizing Metalenses - Erik Szakiel, Nicola Kubzdela

Team members: Erik Szakiel and Nicola Kubzdela

Table of Contents

Introduction

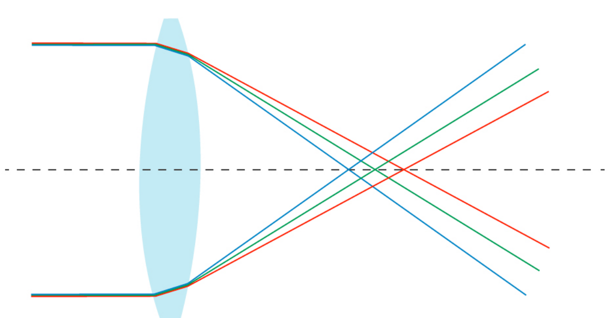
In this project, we investigate the inverse design and optimization of metalenses to correct for common optical aberrations. Using a method for inverse design proposed by Colburn and Majumdar (2021) which combines rigorous coupled-wave analysis (RCWA) with automatic differentiation (AD), we optimize a metasurface with the goal of creating an achromatic lens suited for visible light.

With an optimized structure, we then simulate the behavior of the metalens for input light of varying polarization, wavelength, and incident angle. We compute point spread functions (PSFs), modulation transfer functions (MTFs), geometric distortions, and relative illuminations of the lens for wavelengths across the visible spectrum. With this data, we then use ISETCam to simulate realistic imaging with our optimized metalens, and compare its performance to diffraction-limited optics.

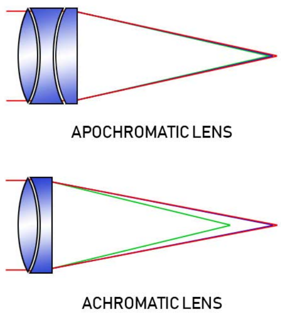
Our overall goal is to compare the performance of our metalens to a diffraction-limited system, and see how we can build from a new base technology to computation of metrics which image system engineers can use to evaluate its performance.

Background

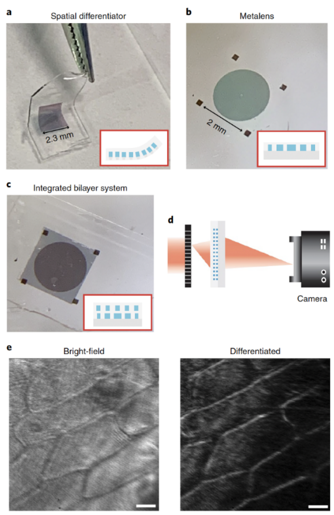
Traditional lenses are known to have chromatic aberration, or the variance in focal length based on frequency, due to the wavelength-dependent refractive index of the lens material. This causes color fringes on an image, as seen below.



Certain correction attempts can be made to improve the lens's chromatic aberration. Achromatic lenses, for example, use two optical components to bring two wavelengths into focus on the same plane. Using three optical components, one can build an apochromatic lens which will bring three wavelengths into focus on the same plane. However, as more corrections are made, these lens systems can both become very thick and require more precise alignment between individual optical components.

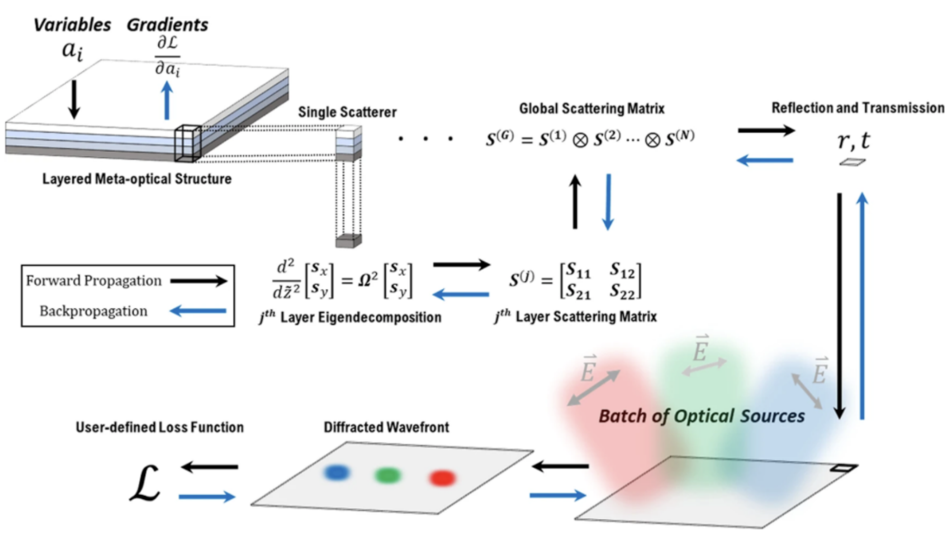


Metamaterials are ultrathin film materials engineered to have properties not found in naturally occurring materials. They use arrays of miniature, anisotropic light scatterers (e.g. resonators like optical antennas) which are on the sub-wavelength scale for multifunctional control of light. They are simple building blocks that allow for tremendous design flexibility with simultaneous electrical and optical functions, allowing metals to manipulate light, conduct currents, apply fields, etc. Their ability to mold optical wavefronts into arbitrary shapes with sub-wavelength resolution lends them to a diversity of applications, including gratings, polarizers, differentiators, sensors, lenses, etc.



Zhou, Y., Zheng, H., Kravchenko, I.I. *et al.* 2020

When looking to design a new metamaterial structure for a specific purpose, intuition is often quite difficult since the performance of the structure is dictated by sub-wavelength optics. As a result, the field of metamaterial research has often turned to inverse design and optimization to create metamaterial structures. Inverse design allows researchers to first specify some overall set of desired performance characteristics and run an optimization to create a structure which achieves these design goals. Typically, inverse design is carried out via gradient descent using the adjoint method. This requires either analytical expressions for gradients of the system or numerical estimates of the gradients – the former of which is often constrained to very specific geometries and the latter of which can be computationally inefficient. In order to efficiently calculate gradients for any arbitrary metasurface, Colburn and Majumdar (2021) proposed a method which combines rigorous coupled-wave analysis (RCWA) with automatic differentiation (AD). This yields fast calculations of gradients with no need for derivation of state and adjoint equations for the metasurface. The authors have provided an open-source repository, *rcwa_tf*, which implements this method using TensorFlow in Python to optimize metamaterial topologies in gratings and lenses. The flow of this optimization is pictured below, with forward propagation in black and backward propagation in blue. By defining a loss function to be optimized, the repository can be used to design multilayered metamaterial structures to meet the design constraints of the user.

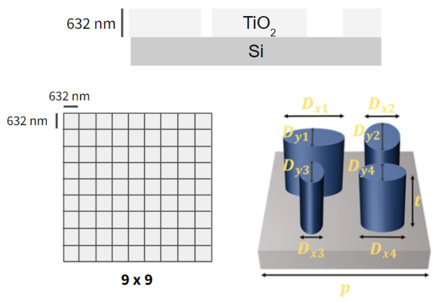


Methods

Optimization of Metalens

Using Colburn and Majumdar's (2021) *rcwa_tf* library, we run code attempting to optimize the performance of a metalens across the visible spectrum. Our goal is to minimize the chromatic aberration of this metalens and focus light across all visible wavelengths down to the same focal plane.

Our first step was to specify a metamaterial geometry which the library could optimize, which is pictured below. We chose a 632 nm layer of TiO_2 on silicon, arrayed into a 9 x 9 grid of unit cells. Each unit cell was 632 nm x 632 nm, and was set to include four coupled cylindrical nanoposts. The major and minor axis of each nanopost in the cell could be adjusted by the optimization, and each cell of the 9 x 9 array could be adjusted independent of the others. This left a total of (4 nanoposts) x (2 parameters per nanopost) x (81 unit cells) = 648 parameters which could be tuned during the optimization of the metalens.



S. Colburn and A. Majumdar, 2021

With the geometry specified, we then had to select a set of inputs which with to test the metalens during the optimization. We selected an array of wavelengths across the visible spectrum, *λ = [400, 450, 500, 550, 600, 650, 700] (nm)* along with two orthogonal polarizations of the light, *e = [TE, TM]* such that we could accurately simulate the kinds of inputs the lens might see in a consumer camera setting. Due to memory and time constraints, we elected for only a single incident angle of light – normal incidence. This left a set of (7 wavelengths) x (2 polarizations) x (1 angle of incidence) = 14 total system inputs.

Finally, we needed to define our loss function. In this case, we want all batch inputs to have the same focal point. Thus, we simulate the device, propagate the input fields to a chosen focal plane, and select the intensity of each batch input at the center of the focal plane. Since we would like equally strong performance across all inputs, our chosen figure of merit is the product of all center spot intensities across all inputs. Our loss function is the figure of merit with its sign flipped to negative, since we must phrase the problem as a minimization. Note that, since we're optimizing the field intensity at some *chosen* distance away from the surface of the lens, our choice of the lens focal length is defined implicitly through the loss function. In our case, we select a focal length of 8.532 micron (corresponding to a lens f/# = 1.5 with an implicit metalens aperture of 9 x 632 nm = 5.688 micron).

$$\text{Loss} = -\prod_{\lambda,\hat{e}} |E_{\lambda,\hat{e}}(0,0)|^2$$

Now that our problem is fully defined, we can leverage the *rcwa_tf* library to both optimize the metalens and simulate its performance once optimized. Please see the "Code" Section for optimization and simulation scripts.

Diffraction-Limited Optics

In order to obtain the point spread function (PSF) of a diffraction-limited optical system, we used an analytical expression derived for a diffraction-limited square aperture. In this case, a square aperture was most appropriate since our metalens implicitly contains a square aperture, being defined by a 9 x 9 square grid in the xy-plane.

The PSF is given by:

$$I(x,y) \propto \text{sinc}^2(\frac{W}{\lambda f}x)\text{sinc}^2(\frac{L}{\lambda f}y)$$

where W and L are the width and length of the aperture, f is the focal length, and λ is the wavelength. Alternatively, phrased more succinctly in terms of the f-number of the lens,

$$I(x,y) \propto \text{sinc}^2(\frac{1}{(f/\#)\lambda}x)\text{sinc}^2(\frac{1}{(f/\#)\lambda}y)$$

where f/# is the f-number of the lens.

These PSFs were calculated analytically in MATLAB. Scripts available in "Code" Section.

Modulation Transfer Functions (MTFs)

The modulation transfer function (MTF) of an optical system is just the Fourier Transform of its point spread function (PSF).

In order to calculate MTF, we calculate the 2D FFT of the PSF and take a slice of the 2D MTF along a single axis. For simplicity, we slice along the x=0 axis (and avoid the complications that the square aperture brings to variation with azimuthal angle).

```
“MTF = abs(fftshift(fft2(psf)));”
```

Scripts available in "Code" Section.

ISET Simulation of Images

In order to render optical images formed by the optimized metalens and diffraction-limited optics, we used ISETCam. For the diffraction-limited system, ISETCam provides a simple and easy configuration of diffraction-limited optics with a given f-number. For the optimized metalens, we used simulated data of point spread functions, geometric distortions, and relative illuminations across different field heights and wavelengths to instantiate a ray-trace optics model. This model was then used to simulate imaging for standard scenes, such as point arrays and a Macbeth chart.

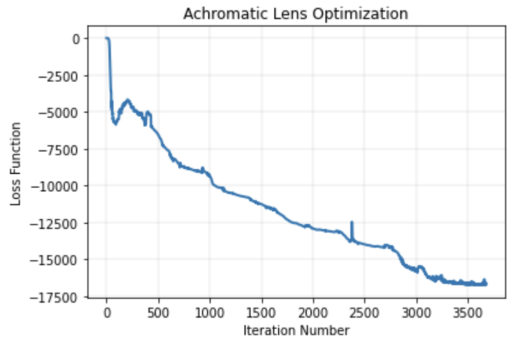
Simulating the true size of the metalens proved to be too computationally intense using the ISETCam ray-trace optics model. Thus, in order to generate some comparison between diffraction-limited optics and the optimized metalens, we scaled the metalens geometry up by a factor of 100 while keeping the same geometry of the PSFs. This allowed us to complete an imaging simulation of the optimized metalens and qualitatively evaluate the effects of suboptimal PSFs and nonuniform relative illuminations directly in image quality.

Scripts available in "Code" Section.

Results

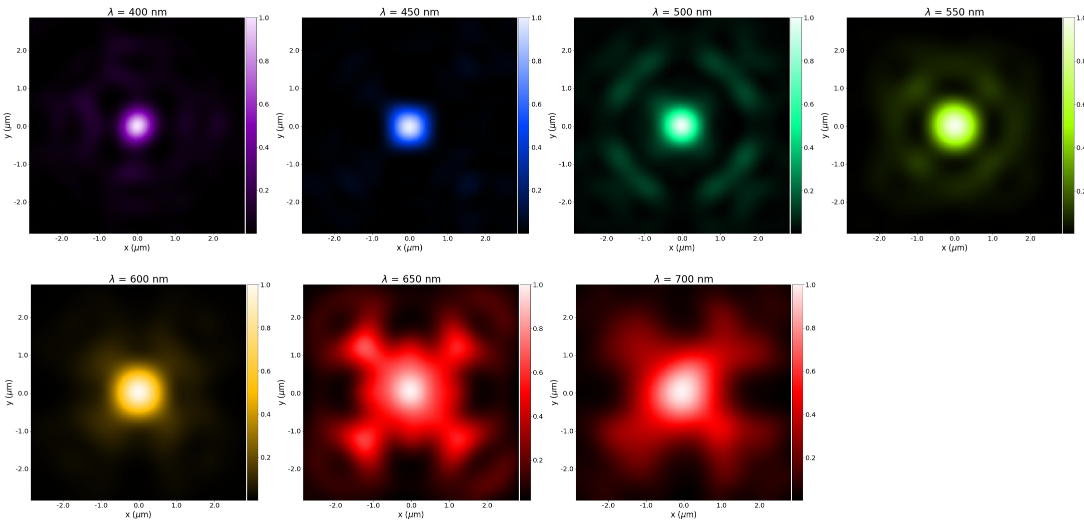
Metalens Optimization – Loss Function

Optimization of the metalens was carried out on the Sherlock High-Performance Computing Cluster run by Stanford University. Using a node instance with 1 CPU and 2 GPUs, we optimized for a total of roughly 30 hours.

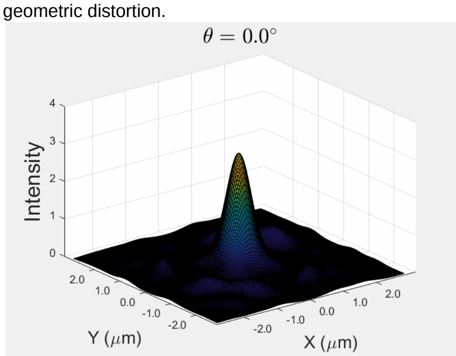


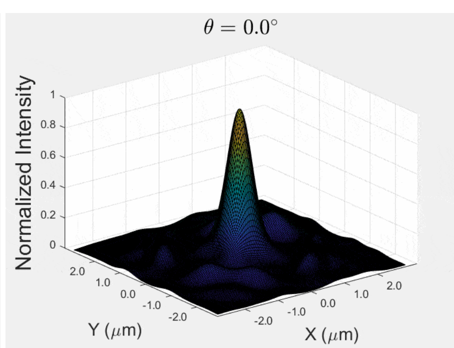
Focal Plane Intensity

We then simulated the focal plane intensity of the metalens under illumination at normal incidence. Pictured below, we see a tight, uniform focal spot at 400 nm which slowly grows to an X shape at wavelengths above 600 nm.



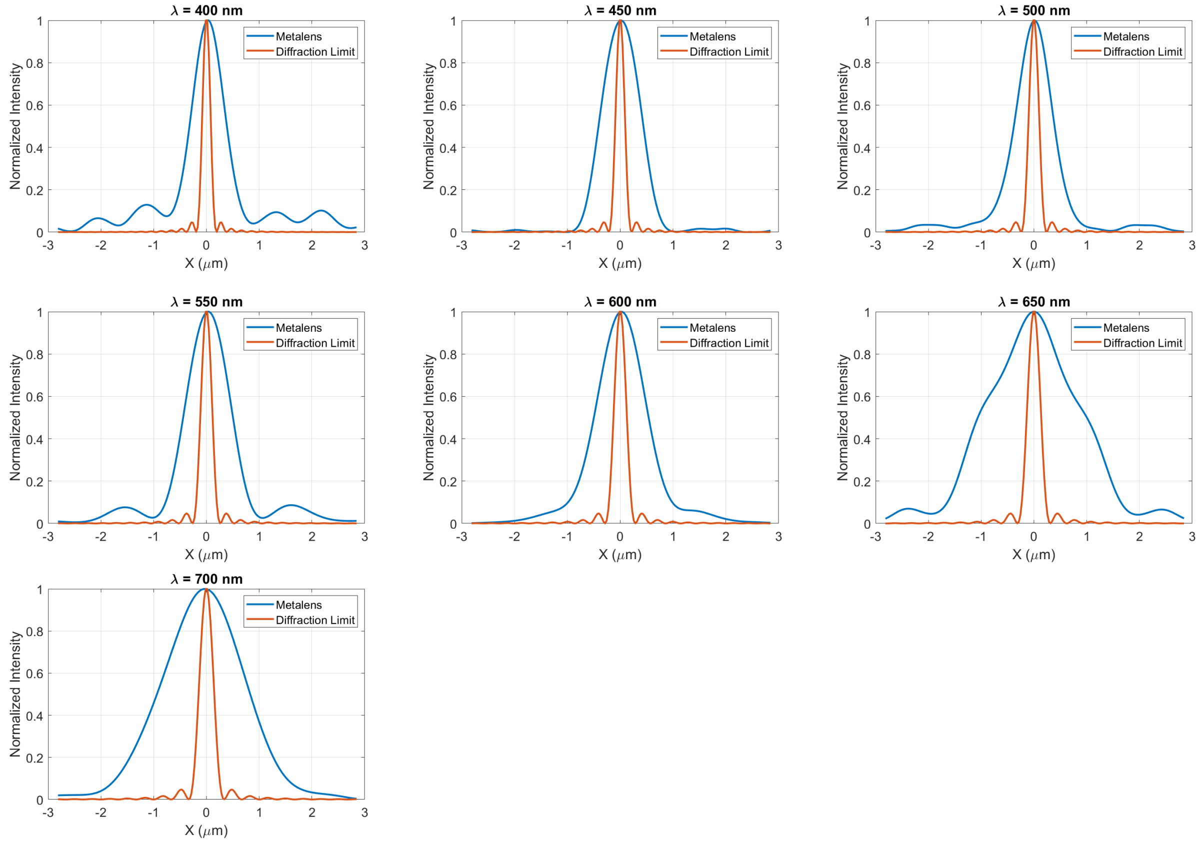
We also simulated the point-spread functions of the metalens under different incident angles of light. Below are two GIFs illustrating these point spread functions as the polar angle (theta) is swept. The left plot shows the raw PSFs (including effects of relative illumination) while the right plot is normalized to show only effects of geometric distortion.



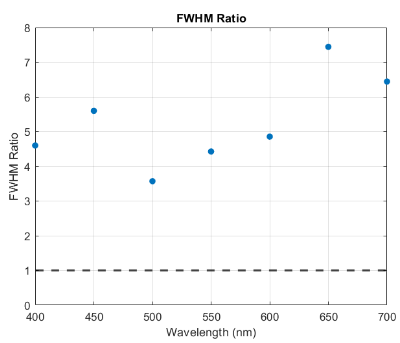


PSF Comparison – Metalens vs. Diffraction-Limited Optics

We see that the PSF for diffraction limited optics is narrower and more ideal than the meta-lens we optimized, including at lower wavelengths.

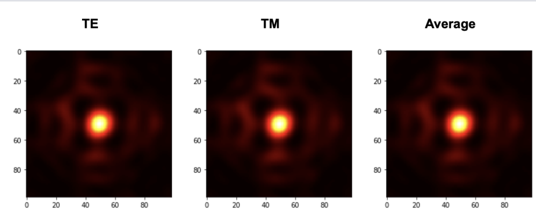


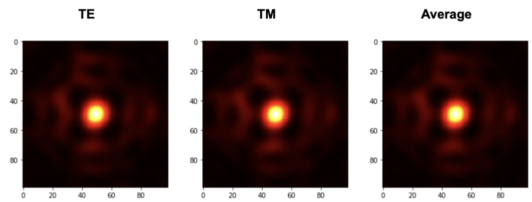
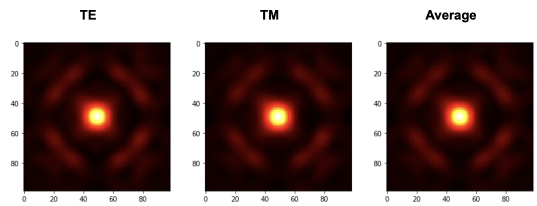
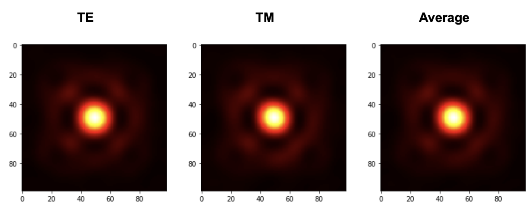
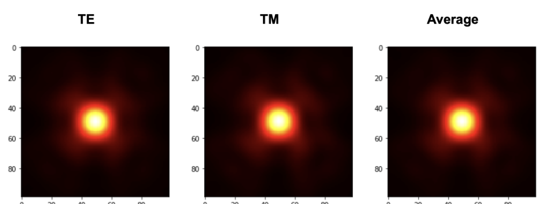
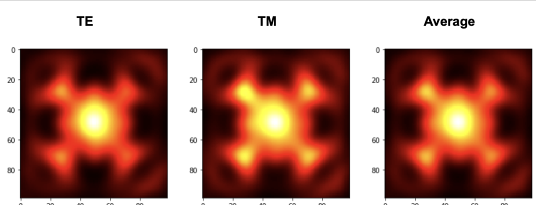
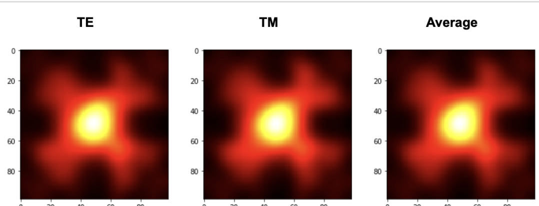
To summarize this data, we can plot the relative full-width half-maximum (FWHM) across wavelengths. The FWHM ratio is defined in this case as the FWHM of the metalens PSF divided by the FWHM of the diffraction-limited system. We can see that the metalens PSF is on average 4-5 times wider than a diffraction-limited system.



Polarization Effects

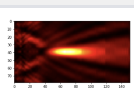
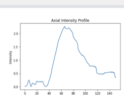
Overall, we find that the metalens is very polarization-insensitive, with TE and TM focal plane intensities looking nearly identical for any wavelength across the visible spectrum.

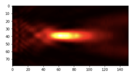
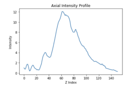
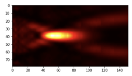
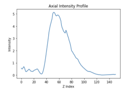
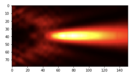
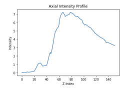
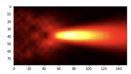
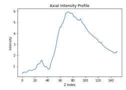
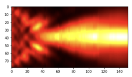
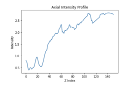
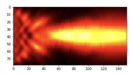
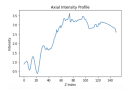
Wavelength	Focal Plane Intensity
400nm	

Wavelength	Focal Plane Intensity
450nm	
500nm	
550nm	
600nm	
650nm	
700nm	

Axial Intensity vs. Wavelength

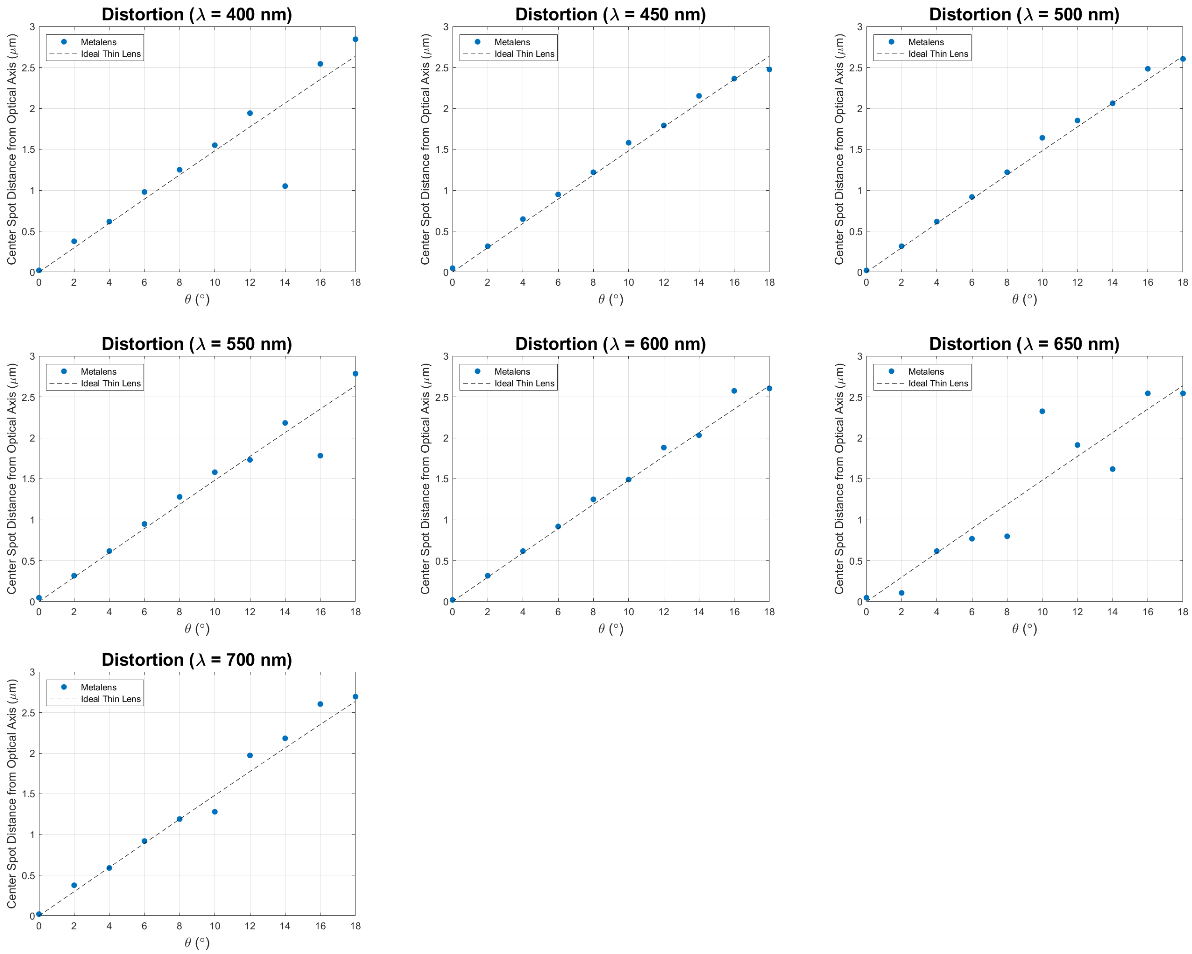
The system clearly had trouble optimizing longer wavelengths >= 650 nm. This is likely due to the fact that the device layer is only 632 nm thick – for any wavelengths longer than 632 nm, the device layer is sub-wavelength and will likely have relatively little focusing power.

Wavelength	Axial Intensity (2D)	Axial Intensity (1D)
400nm		

Wavelength	Axial Intensity (2D)	Axial Intensity (1D)
450nm		
500nm		
550nm		
600nm		
650nm		
700nm		

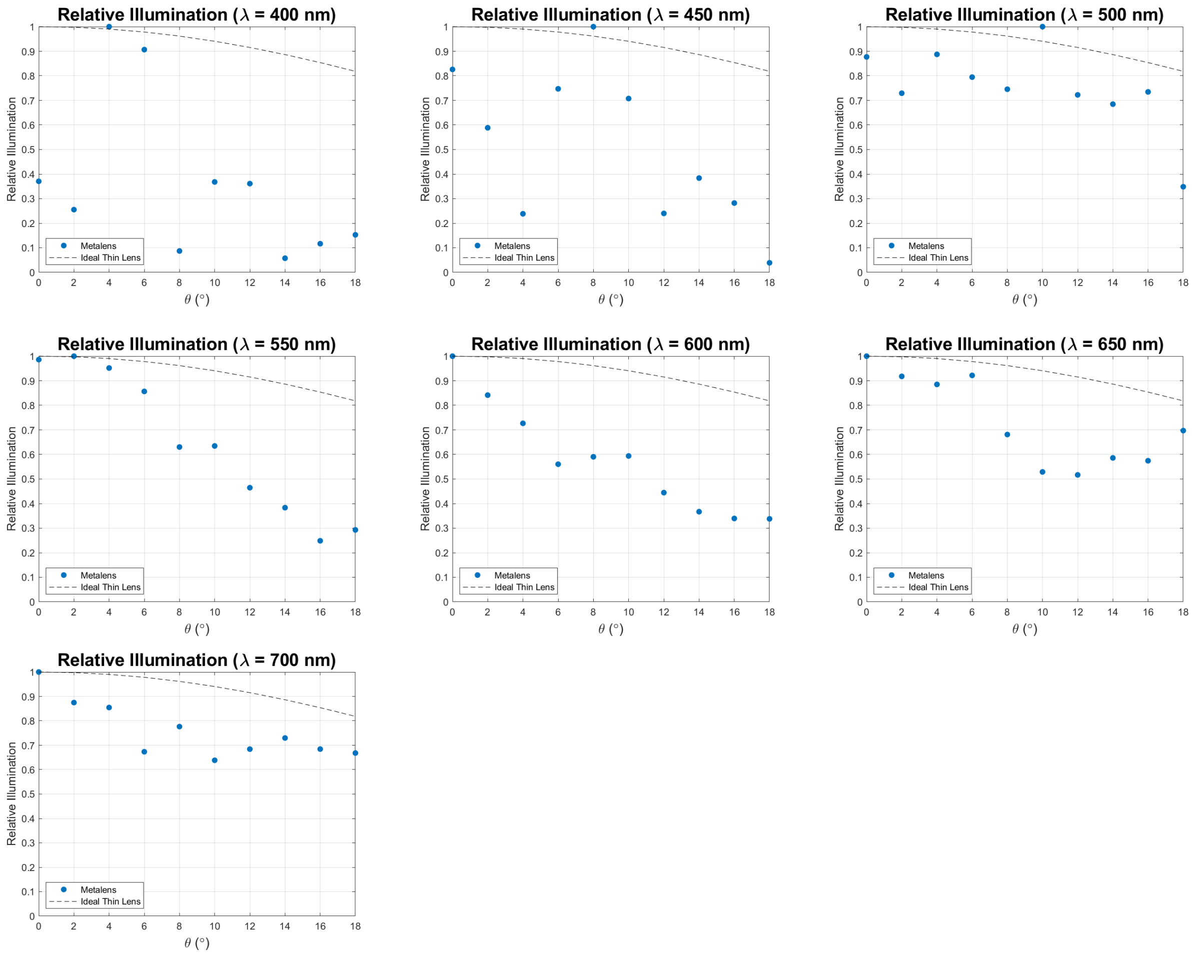
Geometric Distortion

We can also view how the maximum of the PSF shifts around depending on the angle of incidence of the input light. In an ideal, aberration-free system, the maximum of the PSF should be displaced as $f \sin\theta$, where f is the focal length of the lens. We find that, for most wavelengths, the geometric distortion of the lens is fairly well-behaved – most anomalies in otherwise straight curves are expected to be from changes to the PSF itself which cause poor fitting (and an unreliable location of the PSF maximum).



Relative Illumination

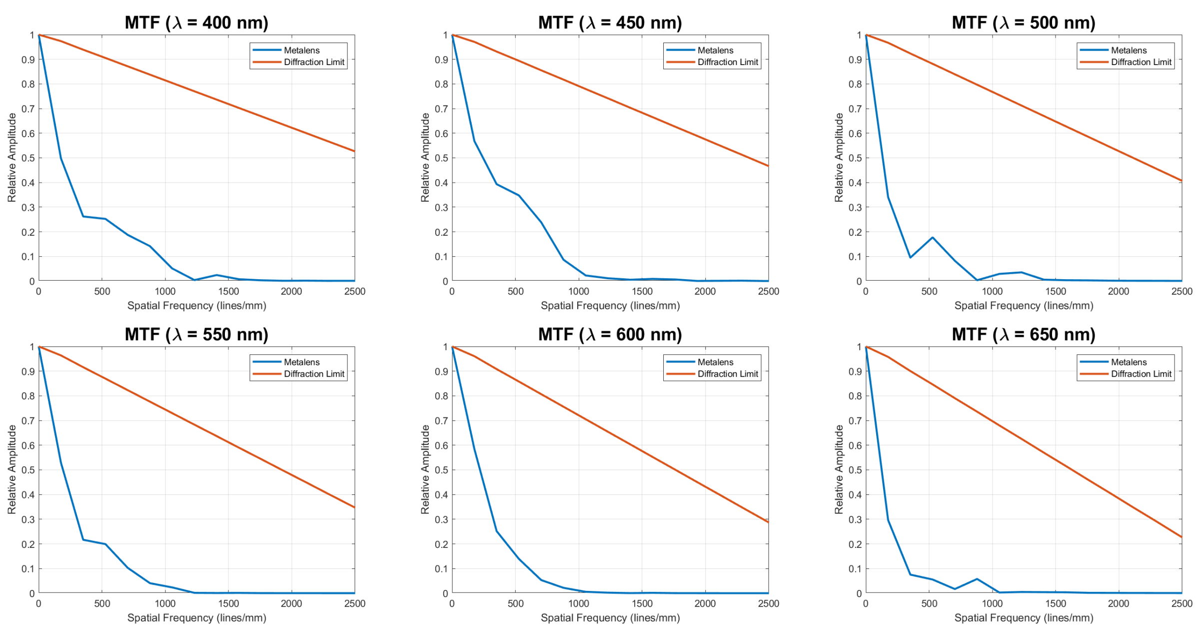
By plotting the relative maxima of the PSF peaks across different angles of incidence, we can view the relative illumination of the metalens. In an ideal system, this relative illumination should have a falloff which goes as $\cos^4(\theta)$. We can see that, regardless of wavelength, the metalens has very unpredictable and rough relative illumination.

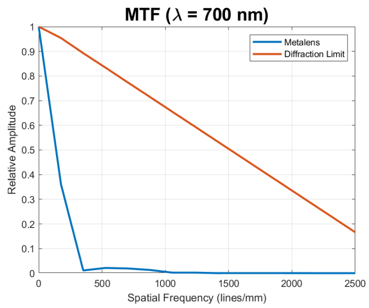


Modulation Transfer Functions (MTFs)

MTFs are used to evaluate and compare the performance of optical systems. We see here that the diffraction limited optics have much better MTFs and are performing better than the meta-lens. Note that the extremely sharp falloff of the metalens around 0 spatial frequency is due to coarse sampling of the PSF (and isn't indicative of the true MTF of the metalens). Overall, however, we see much worse performance in the metalens when compared to a diffraction-limited system. In this case, it's important to have a full image system design approach – what might count as "good enough" for a particular camera application will likely depend on the pitch of pixels in the underlying CMOS sensor array.

Optimized Meta lens vs. Ideal thin lens:

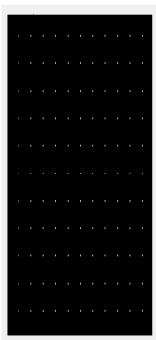
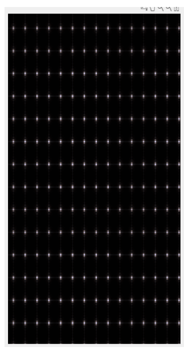
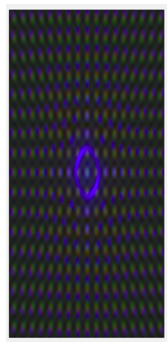




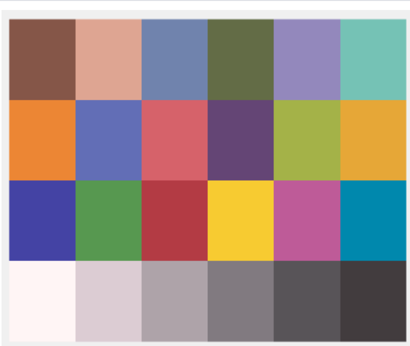
Simulated Images

Now we can visualize images as they would look through the diffraction limited lens and the optimized meta-lens, simulated via ISETCam ray-tracing.

Point Array:

Source Point Array	Diffraction-Limited Image	Metalens Image
		

Macbeth Chart:

Source Macbeth Chart	Diffraction-Limited Image	Metalens Image
		

Conclusions

The metalens still produces a large amount of chromatic aberration, and has much worse statistics than a diffraction-limited system. While the metalens we optimized was not better than traditional diffraction-limited optics, it was still really great to learn about and use a metamaterial design algorithm. While it might be relatively simple to optimize for a single performance metric, effective imaging (especially for consumer cameras) requires quite a few figures of merit to be well-optimized in any optical system. Metamaterial research will continue to yield exciting developments in the optics and photonics space, but image systems engineers should be cautious in the near term regarding the capabilities of metalenses. For every task at which an optimized metalens might excel, there could be other hidden performance metrics which would severely reduce its efficacy in real-world imaging applications.

There exist many avenues through which this work could be greatly improved. For example, we believe that the severe drop-off in performance of the metalens at wavelengths >= 650 nm was due to the fact that the designed structure was only 632 nm thick. Thus, in the future, a thicker lens might be able to have more diffracting power and better performance. Similarly, a different metasurface topology could be used. It could very well be that coupled cylindrical nanoposts is an inefficient design for a metalens application. Multilayer metamaterials could also yield more degrees of freedom with which the structure's optical performance could be improved.

Additionally, the rcwa_tf library seems to be primarily designed for gratings (since its geometries and calculations are specified on an xy plane). In the case of circularly symmetric lenses, there are likely large improvements to be made in terms of reducing the computational complexity of an optimization through symmetry. This would then free up more computational power to design more complex structures or test the system during optimization with more inputs.

This project also raises a question of how post-processing algorithms can work to reduce aberrations introduced by the optical system of a camera. Even if a chosen optical system has a particular set of aberrations, these might be effectively corrected and eliminated by the use of clever post-processing.

References

1. Colburn, S., Majumdar, A. Inverse design and flexible parameterization of meta-optics using algorithmic differentiation. *Commun Phys* 4, 65 (2021). https://doi.org/10.1038/s42005-021-00568-6
2. H. Ren and S. A. Maier, "3D meta-optics for high-bandwidth twisted light holography," in *OSA Imaging and Applied Optics Congress 2021 (3D, COSI, DH, ISA, pcAOP)*, H. Hua, B. Javidi, M. Martinez-Corral, O. Matoba, A. Stern, S. Thibault, T. Alieva, J. Ke, F. Willomitzer, F. Okten, P. Silveira, P. Banerjee, E. Stoykova, D. Chu, J. Park, F. Imai, C. Joo, M. Digman, D. Gardner, S. Gladysz, D. LeMaster, S. Basu, and O. Korotkova, eds., OSA Technical Digest (Optica Publishing Group, 2021), paper DF2C.5.
3. Yu, N., Capasso, F. Flat optics with designer metasurfaces. *Nature Mater* 13, 139–150 (2014). https://doi.org/10.1038/nmat3839
4. Zhou, Y., Zheng, H., Kravchenko, I.I. *et al.* Flat optics for image differentiation. *Nat. Photonics* 14, 316–323 (2020). https://doi.org/10.1038/s41566-020-0591-3

Code

(1) achromatic_metalens.ipynb – Metalens optimization in Python using the *rcwa_tf* library

(2) generate_psfs.ipynb – PSF generation in Python at different wavelengths, incident angles

(3) np_to_mat.ipynb – Converts .npy files to .mat using Python (so PSFs can be imported to MATLAB for analysis)

(4) psf_analysis.m – Imports psf files and generates plots for analysis using MATLAB

(5) s_opticsRTPSFCustom.m – Uses ISETCam to simulate imaging with metalens vs. imaging with diffraction-limited optics