Differentiable Compound Optics and Processing Pipeline Optimization for End-to-end Camera Design

ETHAN TSENG*, Princeton University, United States
ALI MOSLEH* and FAHIM MANNAN*, Algolux, Canada
KARL ST-ARNAUD, AVINASH SHARMA, and YIFAN PENG, Algolux, Canada
ALEXANDER BRAUN, Hoschüle Dusseldorf, Germany
DEREK NOWROUZEZAHRAI, McGill University, Canada
JEAN-FRANÇOIS LALONDE, Université Laval, Canada
FELIX HEIDE, Princeton University, United States and Algolux, Canada

Fig. 1. We propose an end-to-end camera design scheme that jointly optimizes compound optics together with hardware and software image post-processors. Our approach allows us to cater lens systems and the hyperparameter settings of the entire imaging pipeline towards domain-specific applications, including but not limited to automotive object detection and natural image capture. We leverage existing traditional optics design tools such as Zemax, which can be easily integrated into our framework. We validate our method in simulations and in the real-world via manufactured prototypes, using an (optimized) ARM Mali-C71 ISP and cameras mounted onto an acquisition vehicle (left). When optimizing for perceptual image quality (center), the proposed method finds an optics and processing pipeline that improves visual detail across fields, outperforming conventional pipelines with a nominal lens resulting from *over one month* of Zemax-aided expert-design. When optimizing for a pedestrian-vehicle detection (right), the same method learns *different* optics with comparable spotsize but significantly higher speed (f/3.2) than the nominal optics (f/4.4), improving object detections in flux-limited image regions that are challenging to recover by fine-tuned conventional imaging and detection pipelines.

Most modern commodity imaging systems we use directly for photography—or indirectly rely on for downstream applications—employ optical systems of multiple lenses that must balance deviations from perfect optics, manufacturing constraints, tolerances, cost, and footprint. Although optical designs often have complex interactions with downstream image processing or analysis tasks, today's compound optics are *designed in isolation from these interactions*. Existing optical design tools aim to minimize optical aberrations, i.e., deviations from Gauss' linear model of optics, instead of application-specific losses, precluding joint optimization with hardware image signal processing (ISP) and highly-parameterized neural network processing. In this paper, we propose an optimization method for compound optics that lifts these limitations. We optimize *entire* lens systems *jointly* with hardware and software image processing pipelines, downstream neural network processing, and with application-specific end-to-end losses. To this end, we propose a learned, *differentiable forward model for compound optics* and an *alternating proximal optimization* method that handles function compositions with highly-varying parameter dimensions for optics, hardware ISP and neural nets. Our method integrates seamlessly atop existing optical design tools, such as Zemax. We can thus assess our method across many camera system designs and end-to-end applications. We validate our approach in an automotive camera optics setting—together with hardware ISP post processing and detection—outperforming classical optics designs for automotive object detection and traffic light state detection. For human viewing tasks, we optimize optics and processing pipelines for dynamic outdoor scenarios and dynamic low-light imaging. We outperform existing compartmentalized design or fine-tuning methods qualitatively and quantitatively, across *all* domain-specific applications tested.

*indicates equal contribution.

Authors' addresses: Ethan Tseng, eftseng@cs.princeton.edu, Princeton University; Ali Mosleh, mosleh.ali@gmail.com, Algolux; Fahim Mannan, fahim.mannan@algolux.com, Algolux; Karl St-Arnaud, karl.st-arnaud@algolux.com, Algolux; Avinash Sharma, avinash.sharma@algolux.com, Algolux; Yifan Peng, evan.y.peng@gmail.com, Algolux; Alexander Braun, alexander.braun@hs-duesseldorf.de, Hoschüle Dusseldorf; Derek Nowrouzezahrai, derek@cim.mcgill.ca, McGill University; Jean-François Lalonde, jflalonde@gel.ulaval.ca, Université Laval; Felix Heide, fheide@cs.princeton.edu, Princeton University and Algolux.

0730-0301/2021/8-ART0 $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

CCS Concepts: • **Computing methodologies → Computational photography**;

Additional Key Words and Phrases: compound optics, computational imaging, end-to-end image processing, optics design, deep learning

ACM Reference Format:
Ethan Tseng, Ali Mosleh, Fahim Mannan, Karl St-Arnaud, Avinash Sharma, Yifan Peng, Alexander Braun, Derek Nowrouzezahrai, Jean-François Lalonde, and Felix Heide. 2021. Differentiable Compound Optics and Processing Pipeline Optimization for End-to-end Camera Design. *ACM Trans. Graph.* 40, 0, Article 0 (August 2021), 18 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Nearly all commodity camera systems rely on compound optics. These cascades of individual lens elements are designed to focus light reflected from scene surfaces onto sensor pixels. Combined with real-time processing in image signal processing (ISP) hardware, conventional camera systems are the foundation for ubiquitous applications in personal photography, communication, surveillance, and, with image data consumed by computer vision software, emerging applications in robotics and autonomous driving.

While processing algorithms and hardware have developed rapidly during the last decades, achieving impressive image reconstruction capabilities [Chen et al. 2018; Hasinoff et al. 2016], existing optical systems are still conceived in isolation. Indeed, optical systems are designed to reduce optical aberrations—i.e., deviations from linear optics [Gauss 1843]—independently of downstream tasks such as detecting objects for autonomous driving, or generating visually pleasing images for mobile photography. Existing optics optimization methods rely on a rich ecosystem of optimization tools [Garrard et al. 2005; Geary 2002], including 0^{th}- and 1^{st}-order methods, but do not allow for joint optimization over the entire imaging pipeline—including the optics, ISP and *all* subsequent software processing—since the number of parameters to optimize becomes prohibitively large. Indeed, conventional compound optics, hardware ISPs and downstream convolutional neural network (CNN) architectures combine to form a complex, high-dimensional parameter space—with both categorical and continuous variables—with tens to millions of parameters, depending on the domain-specific application.

Recent end-to-end optics designs, using differentiable single-phase plate optics and software post-processing [Sitzmann et al. 2018], target this gap. These methods, however, require manufacturing complex photo-lithographic phase plates which are not applicable to commodity optical systems. Such systems are limited to a catalog of standard optical surfaces and a restricted number of ground, or injection-molded, aspherical elements. These constraints result from mass-market manufacturing processes and the high throughput required by real-time applications. As such, compound optics strike a careful balance between the availability of glasses, tolerance design in manufacturability, and compartmentalized manufacturing expertise perfected in the industry.

We propose the first approach to allow for joint optimization over the space of manufacturable compound optics, hardware or software ISPs, and downstream tasks. We propose an alternating proximal optimization method that combines 1^{st}-order optimization for deep neural network detectors and other processing methods with large parameter spaces, non-differentiable compound lens parameters, and hardware ISP hyperparameters. Our method is the first to bridge traditional design with proprietary tools, such as ZEMAX, and stochastic 1^{st}-order optimization. We can readily import existing ZEMAX designs and export to this industrial standard format. To this end, we build atop traditional tolerance design.

We validate our method across application-specific camera system designs and tasks. Specifically, we co-design automotive camera optics together with hardware ISP post-processing and object detection. We also optimize alternative optics and post-processing for human viewing, and for low-light imaging. Our approach qualitatively and quantitatively outperforms existing compartmentalized design or fine-tuning on *every* domain-specific application we tested.

Specifically, our work makes the following contributions:

- A joint end-to-end optimization framework for compound lens models combined with realistic sensor models, hardware (or learned software) image processing, and CNN computer vision modules. We jointly optimize the parameters and hyperparameters of this heterogeneous pipeline, using domain-specific losses.
- Our framework builds atop traditional tolerance analysis and integrates seamlessly with standard optics design methods.
- We combine existing raytracing algorithms with deep learning to construct accurate differentiable approximations of compound optics that are not limited to small fields of view within the paraxial regime.
- We analyze our method in simulation for optics and image processing tasks in human viewing and analytic downstream tasks.
- We *build five real prototype lens designs* based on our optimization framework, and validate the proposed approach on challenging real-world automotive data acquired on a test vehicle.

2 RELATED WORK

We review prior art most related to our methods and contributions.

Compound optics designs. While modern photography technologies have evolved to provide high-quality imaging performance across diverse devices and applications, the optical design process still follows classic aberration theory [Smith 2005] established over a century ago. Optical aberrations describe the deviations in focus—based on Gauss' linear optics model [Gauss 1843]—resulting from optical path offsets of light as it travels across lens regions from varying incident directions [Fowles 1989]. In this way, stacks of several optical elements of varying surface shapes and materials are introduced to combat both monochromatic and chromatic aberration [Kingslake and Johnson 2009; Sliusarev 1984]. This design methodology, when combined with the increases in sensor resolutions and shrinking form factors, results in commercial optic designs composed of many (i.e., dozens) spherical or aspherical elements.

State-of-the-art optics design software, such as ZEMAX or CODE V [Garrard et al. 2005; Geary 2002; Walker 2008], can optimize surface profiles of refractive lenses. These tools use mid-level metrics—so-called *merit functions*—which aim to strike a compromise across many criteria [Malacara-Hernández and Malacara-Hernández 2016], such as spatially-varying point spread function (PSF) size over target wavelength bands. Their (proprietary) optimization methods scale only to a dozens of continuous optical parameters. Open design tools, such as Lens Factory [Sun et al. 2015], also admit these limitations and are not able to match commercial tools in their breadth of supported designs. These tools are not suited for joint optimization of neural networks and/or hardware ISPs with discrete and continuous parameters.

Forward lens simulation models. A variety of approaches in computer graphics have been proposed for accurately simulating the behavior of complex lens assemblies, including ray-based models [Kolb et al. 1995], spectral models [Steinert et al. 2011], and efficient Monte Carlo sampling [Hanika and Dachsbacher 2014]. Simulated lens models have been used to optimize optical inspection systems [Retzlaff et al. 2016], or to fit predicted PSFs to captures using first-order approximations [Shih et al. 2012]. These forward models are however not end-to-end differentiable, and are restricted to spherical lenses. Handling richer designs—including aspherical lenses—requires modeling scattering, flare, vignetting, and propagation properly, which is not differentiable and requires expensive raytracing operations [Walker 2008]. Researchers have demonstrated that deep learning can closely approximate complex functions such as ISPs [Tseng et al. 2019] and physical simulators [Grzeszczuk et al. 1998], however, it is non-trivial to directly learn an *accurate* image-to-image model of compound optics that allows for gradient feedback as the PSFs are spatially varying and encode differences in the optical design parameters through subtle changes. We propose to combine deep learning with existing raytracing algorithms to create a differentiable module that accurately reproduces the behavior of complex lenses as a function of the optical parameters.

Merit functions. Existing optical designs are optimized for intermediate merit functions, such as the PSF size or wavefront errors [Smith 2005]. While these traditional design guidelines have led to high-quality camera lens designs, current merit functions are blind to downstream operations in the camera image processing pipeline. While the goal of a camera is to record a "perfect"—typically post-processed—image for human consumption [12233:2017 2017; 1858-2016 2017; Phillips and Eliasson 2018], imaging systems may also introduce new artifacts. Typical spatial IQ metrics focus on a single aspect of image quality, such as sharpness, noise, or blur [Baxter et al. 2012].

Image processing pipeline design. Direct sensor measurements suffer from many sources of degradation including, but not limited to, the aforementioned aberrations, color filter subsampling, photon shot noise, cross-talk effect, and read-out noise. Custom optimized hardware ASICs are used—in part due to the performance critical nature of real-time systems built atop these imaging modules—to realize the low-level image processing needed to reconstruct high-quality images from the measurement polluted by these degradations [Hegarty et al. 2014; ON Semi MT9P001 2017; Ramanath et al. 2005; Shao et al. 2014; Zhang et al. 2011].

Any ISP designs that deviate from this model are typically limited, in their deployment, to off-line tasks. Optimization-based methods [Heide et al. 2014] operate orders of magnitude slower than real-time ISPs. Machine learning-based methods focus on specific tasks, such as demosaicking [Gharbi et al. 2016], tonemapping [Gharbi et al. 2017], low-light denoising [Chen et al. 2018] and other common processing operators [Chen et al. 2017; Fan et al. 2018; Xu et al. 2015]. These methods require high-end GPUs with high power consumption (i.e., ≥100 Watts). Despite this, such data-driven approaches remain attractive in their ability to be tailored to specific end tasks, but they remain limited to simpler single-element optical designs. We will demonstrate joint optimization of both hardware and software ISPs with complex multi-element optics.

Recent approaches that explore camera pipeline optimization cannot jointly optimize the ISP with downstream applications [Nishimura et al. 2018], are limited to optimizing individual differentiable blocks [Li et al. 2018], or tackle end-to-end ISP optimization without any consideration for the optics [Tseng et al. 2019]. While Li et al. [2018] does provide a differentiable optical model using raytracing, their optical design strategy entirely relies on intermediary heuristics without consideration of the final endpoint loss. Mosleh et al. [2020] recently proposed an ISP optimization method using hardware-in-the-loop, however, this approach is currently impractical for optics design as new lenses would need to be manufactured and tested at each iteration of the optimization.

End-to-end optical design. Recent works explore the applicability of diffractive optical elements [Chang et al. 2018; Metzler et al. 2020; Sitzmann et al. 2018; Stork and Gill 2014; Sun et al. 2020] for imaging in photography and other vision-based applications, here still with a single optical element. This single element restriction, coupled with an approximate forward Fresnel propagation model, is needed to render their joint automated design optimization task tractable. These simplifications, however, come at a cost: the realizable diameter and field of view of the resulting lens are limited, and resulting imaging quality lags behind that of commodity multi-element compound lens designs. Moreover, the resulting designs are not suitable for production systems, as no tolerance analysis is considered. Peng et al. [2019] address some of these limitation with a single, handcrafted free-form element. This hand-tuned design does not support joint optimization with image processing hardware and software, nor does it support multi-element compound lenses. Our work lifts these important remaining limitations, incorporating traditional tolerance analysis to design compound optics in an end-to-end fashion. Parallel work from Sun et al. [2021] addresses compound lens optimization. While their work relies on differentiable ray-tracing, the proposed method does not require a differentiable forward model, allowing us to integrate our method with existing lens design tools such as Zemax.

3 IMAGING PIPELINE STAGES

We present our overarching image formation model and imaging pipeline stages. Our imaging pipeline is divided into five (5) core stages (Fig. 2): the scene, compound camera optics, sensor, ISP (hardware or software), and downstream tasks. We detail each, below.

3.1 Scene representation

We treat scenes as all-in-focus RGB images and assume that all scene content lies beyond the hyperfocal distance, a representation widely used in existing optical design works [Chang et al. 2018; Sitzmann et al. 2018]. It is similarly suitable here, since we target

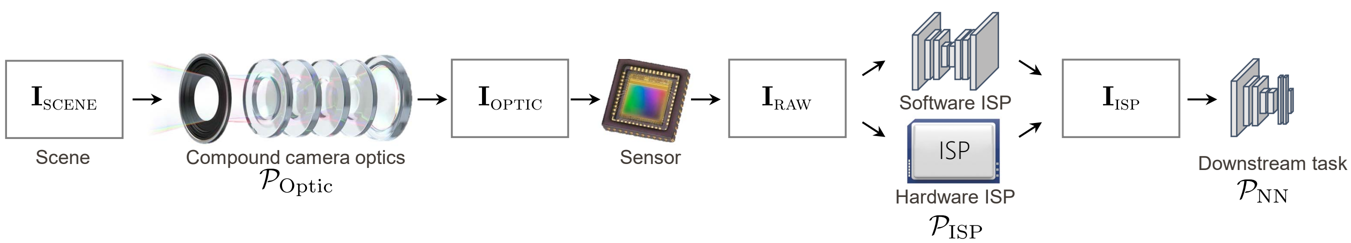


Fig. 2. Overview of the imaging pipeline stages. The scene light field (approximated by an image $\mathbf{I}_{\text{SCENE}}$) is captured by compound camera optics, forming $\mathbf{I}_{\text{OPTIC}}$. This is subsequently imaged by a sensor to produce $\mathbf{I}_{\text{RAW}}$. The RAW image is converted to $\mathbf{I}_{\text{ISP}}$ by a hardware or software ISP. The resulting image may feed further downstream processes, e.g., task-specific deep neural network architectures. The optics, ISP and task are parameterized by $\mathcal{P}_{\text{OPTIC}}$, $\mathcal{P}_{\text{ISP}}$, and $\mathcal{P}_{\text{NN}}$.

fixed focus (e.g., automotive imagers for object detection [Geiger et al. 2013]) rather than depth-sensitive applications (e.g., mobile photography) for which auto-focus is necessary. Treating the entire scene as being in focus also allows us to downsample scene data to suit downstream hardware and software ISPs, providing scene-scale invariance that decouples acquisition optics from image post-processing. RGB images allow us to build upon existing RGB training data and established methods for RGB image processing, while being computationally less costly than, e.g., computing multispectral PSFs.

3.2 Compound camera optics

Existing differentiable optical design approaches [Chang and Wetzstein 2019; Sitzmann et al. 2018] rely on the *paraxial approximation*, reducing the optical response to a single PSF and enabling a compact differentiable Fourier propagation model based on wave optics. This approximation, however, only holds for small fields of view (FOV $\approx 5^\circ$), whereas full ray-tracing is required to accurately model optics for larger FOV. Moreover, these methods only design a single optical element, whereas consumer and industrial optical systems commonly consist of a sequence of many such elements.

Our work develops a framework for the design of such compound optical systems, that is, optical systems consisting of multiple optical elements. Our model is not limited to the paraxial regime, and so, can handle wide FOVs. We achieve this by simulating *spatially-varying* PSFs that describe the features produced by complex optical pipelines – including Seidel aberrations and vignetting – which cannot be described using a single, spatially-invariant PSF.

We parameterize an optical system (with a fixed number of lens elements) by the set $\mathcal{P}_{\text{OPTIC}}$, which includes surface thicknesses t, intervals l, refractive indices η_λ, and surface parameters s for every element, as well as the stop position. We assume the f-number and back-focal length of the optical system are given as fixed design constraints during optical design. Fig. 3 illustrates a three-element system; we will introduce several alternatives in Sec. 6.

Assuming scene content that lies at infinity (Sec. 3.1), the spherical light rays from a scene surface point enter the pupil of the optical system in parallel at angle Θ. An *ideal* optical system transfers a wavefront at angle Θ, i.e., the position of equal phase, to a perfect inverted spherical wave centered at the image plane (polar) coordinate r. In this ideal system and assuming a rotationally symmetric optical system, sources at infinity produce images at $r = F\tan\theta$ with F as the focal distance to the image plane. Deviations from this ideal behavior are typically measured as an optical path difference

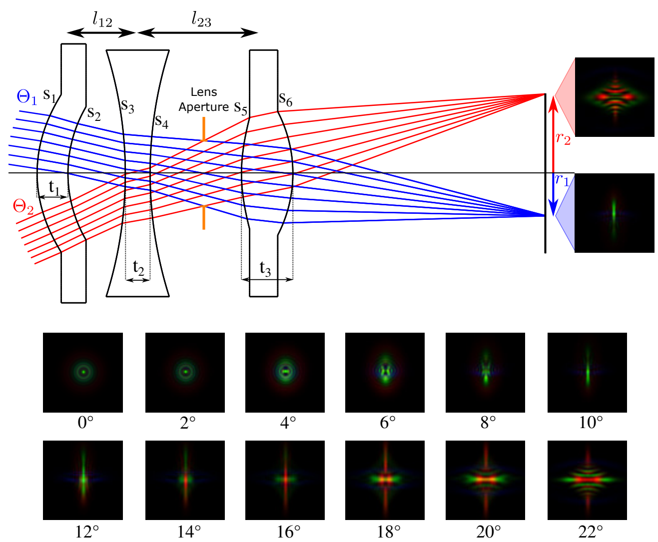


Fig. 3. Non-paraxial compound optics model. Compound optical systems consisting of several elements require full ray-tracing to determine the spatially-varying point spread functions (PSFs). Design parameters, such as the physical distance between elements (l_{12} and l_{23}, above), affect these PSFs. We assume rotationally symmetric optics and parameterize the spatially-varying PSFs by their incident angle, corresponding to spatial locations on the sensor plane, shown as Θ_1, Θ_2 and r_1, r_2, above.

(OPD) between the ideal and the system wavefronts, expressed as a function $f_{\text{OPD}}(\mathbf{p}, r, \lambda; \mathcal{P}_{\text{OPTIC}})$ of the exit pupil plane position $\mathbf{p}$, image coordinate r, and wavelength λ, for optics parameters $\mathcal{P}_{\text{OPTIC}}$.

We model the spatially-varying PSF response of a compound optical system as the following function f_{OPTIC}:

$$\begin{aligned}\text{PSF}_\lambda(\mathbf{x}, r; \mathcal{P}_{\text{OPTIC}}) &= \left| \int A(\mathbf{p})\, e^{i f_{\text{OPD}}(\mathbf{p}, r, \lambda; \mathcal{P}_{\text{OPTIC}})}\, e^{i 2\pi \mathbf{p}\mathbf{x}} \mathbf{dp} \right|^2 \\ &= f_{\text{OPTIC}}(r, \mathcal{P}_{\text{OPTIC}}),\end{aligned} \tag{1}$$

where $\mathbf{x}$ is the spatial coordinates in the PSF. The spatially-varying PSF for a given radial position r is thus the magnitude of the inverse Fourier transform over the exit pupil. We assume the amplitude of the exit pupil to be unattenuated, $A(\cdot) = 1$. Note that traditionally the OPD is optimized by lens designers where the polynomials for r and p are evaluated, such as the 3rd-order Seidel aberrations. So-called merit functions are also placed on these individual aberrations.

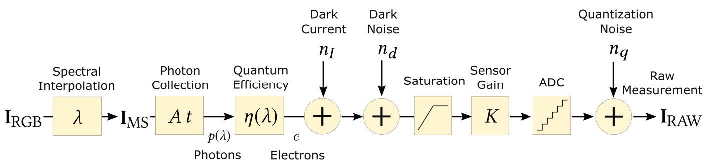


Fig. 4. Block diagram of our fully differentiable CFA sensor model. We first interpolate multispectral channels between the RGB channels of an input RGB image from a dataset. Photons arriving at the sensor plane follow a Poisson distribution with rate determined by this multispectral image. We convert photons into electrons using wavelength-dependent quantum efficiency, before adding dark current. Finally, we convert the analog signal into a digital readout. We detail derivations for each block in the text.

Given an input RGB scene $\mathbf{I}_{\text{SCENE}}$, we use the resulting spatially-varying PSFs to simulate the modulated radiance image $\mathbf{I}_{\text{OPTIC}}$ that would appear on the sensor surface after traversing the compound optics. We convolve locations (x, y) with their associated spatial PSF, given by the distance $r = \sqrt{x^2 + y^2}$ from the center:

$$\mathbf{I}_{\text{OPTIC}}(x, y) = \mathbf{I}_{\text{SCENE}}(x, y) * f_{\text{OPTIC}}\left(\sqrt{x^2 + y^2}, \mathcal{P}_{\text{OPTIC}}\right). \quad (2)$$

Unfortunately, performing a per-pixel convolution with spatially-varying PSFs is computationally costly. We compromise by modifying the overlap-add (OLA) method to calculate space-variant linear filters [Hirsch et al. 2010]. The image is split into overlapping patches and the borders of each patch are damped with a windowing function. Then, each patch is convolved with its corresponding PSF, and subsequently added to reconstruct the image $\mathbf{I}_{\text{OPTIC}}$. The compact $\mathbf{I}_{\text{OPTIC}} = f_{\text{OPTIC}}(\mathbf{I}, \mathcal{P}_{\text{OPTIC}})$ will henceforth refer to that process.

3.3 Sensor model

Our sensor model relies on a differentiable approximation of standard color filter arrays (CFAs) used by conventional imagers. Given an image $\mathbf{I}_{\text{OPTIC}}$ produced by the optics, our sensor model outputs a single channel RAW image $\mathbf{I}_{\text{RAW}}$ resulting from the sequence shown in Fig. 4. To this end, we propose a variant of the widely adopted EMVA 1288 [EMVA 1288 2016] model, with differentiable Poisson sampling and a multispectral interpolation model.

Specifically, the model first determines the quantity of photons arriving at the sensor plane. Each detector in a CFA is tailored to capturing a narrow band of wavelengths and is highly sensitive to the spectrum of incoming light. As such, we first extend the incoming RGB image $\mathbf{I}_{\text{OPTIC}}$ to a 50-band multi-spectral image $\mathbf{I}_{\text{MS}}$ using quadratic interpolation between the RGB color channels. The photon count per wavelength λ arriving at the detector at position (x, y) on the sensor follows a Poisson distribution[1] with mean

$$\mu_p(x, y, \lambda) = \mathbf{I}_{\text{MS}}(x, y, \lambda) \cdot \frac{\pi A t \lambda^2}{hc(1 + (2N)^2)}, \quad (3)$$

where A is the pixel area, t is exposure time, N is the f-number, $h = 6.626\text{e}{-34}$ $[\text{m}^2 \cdot \text{kg} \cdot \text{s}^{-1}]$ is Planck's constant, and $c = 2.998\text{e}8$ $[\text{m} \cdot \text{s}^{-1}]$ is the speed of light in a vacuum.

[1] Our differentiable Poisson sampler implementation requires careful thought—please consult the supplemental document for details.

Photons are then converted into electrons using the detector quantum efficiency $\eta(x, y, \lambda) = e(x, y, \lambda) \big/ p(x, y, \lambda)$, where $e(x, y, \lambda)$ is the number of electrons generated when $p(x, y, \lambda)$ photons arrive at a sensor location (x, y) for wavelength λ.

Other factors also lead to electron generation, such as temperature and electronic imperfections. Our sensor model includes dark noise $n_d \sim \mathcal{N}(\mu_d, \sigma_d)$ (electron noise generated in the absence of light) as well as dark current n_I (electron noise dependent on the sensor temperature T), which follows a Poisson distribution with mean

$$\mu_I = \mu_{I.\text{ref}} \cdot 2^{(T - T_{\text{REF}})/T_d} \cdot t_{\text{exp}}. \quad (4)$$

Here, $\mu_{I.\text{ref}}$ is the average dark current measured at a reference temperature T_{REF}, T_d is the temperature interval that causes a doubling of the dark current, and t_{exp} is the exposure time.

Finally, we convert electrons to digital values by clipping electron quantities at the maximum well capacity e_{SAT}, and scaling by a gain factor K, before quantizing and adding black level b. Note that clipping is commonplace in machine learning, e.g., with ReLU activations. To permit differentiability of the quantization step, we simulate quantization with uniform noise $n_q \sim \text{Uniform}(-0.5, 0.5)$. Thus, the digital readout $\mathbf{I}_{\text{RAW}}$ at position (x, y) when $p(x, y, \lambda)$ photons arrive at the sensor is given by

$$\mathbf{I}_{\text{RAW}}(x, y) = b + n_q + K \min\left(e_{\text{SAT}}, n_d + n_I + \textstyle\sum_\lambda p(x, y, \lambda)\eta(x, y, \lambda)\right). \quad (5)$$

3.4 Hardware imaging pipeline stages

Hardware ISPs, such as the ARM Mali C71, are becoming increasingly ubiquitous due to their real-time performance, power efficiency, and high resolution, all of which are critical to dynamic applications such as automotive imaging. ISPs operate directly on RAW images captured on camera sensors and, after a series of individual image processing stages, they output an image ready for human consumption or for further downstream processing. We describe these individual processing blocks below:

(1) **Color-correction, gain**: Variations in quantum efficiencies cause CFA filters to treat wavelengths non-uniformly. As such, color-correction (e.g., white-balance) and gain-adjustment are often applied after black level removal [Ramanath et al. 2005].
(2) **Demosaicking**: Sensor detectors are commonly arranged in an alternating R-G-G-B Bayer mosaic pattern. A demosaicking stage reconstructs missing color information at each pixel to produce trichromatic RGB images. Bilinear interpolation between neighboring pixels is a common strategy, here [Zhang et al. 2011].
(3) **Denoising**: Noise can be reduced using methods like edge-preserving filters [Choi et al. 2014; Tomasi and Manduchi 1998] or non-local patch matching [Dabov et al. 2007; Zhang et al. 2016].
(4) **Color, tone correction**: Image adjustments can be performed to improve overall appearance. These include both global (e.g., gamma correction) and local operations (e.g., edge sharpening).
(5) **Colorspace conversion, compression, further processing**: Finally, the image can be converted into an output colorspace (e.g., sRGB or HSV), compressed (e.g., to jpeg) for transfer or storage, or further processed by downstream image processing applications or pipelines (e.g., object detection).

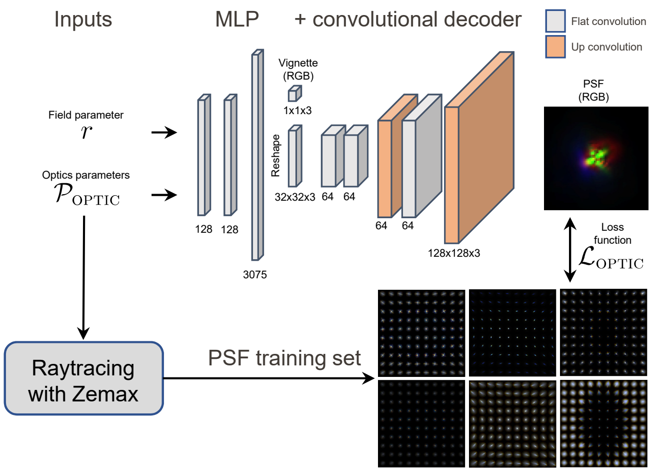


Fig. 5. Overview of our optics meta-network training procedure. We train our optics meta-network to map $\mathcal{P}_{\text{OPTIC}}$ and r to spatial PSFs. We use spatial PSF training data acquired by driving Zemax's ray tracer with $\mathcal{P}_{\text{OPTIC}}$.

3.5 Software image processing and analysis

Much of modern image processing and computer vision is performed in software, allowing for flexible algorithm design. There are many software ISPs, e.g., bilateral filtering, non-local patch denoising, and deep neural image processing approaches. The latter have been applied to a broad range of imaging tasks, including demosaicking [Gharbi et al. 2016], denoising [Chen et al. 2018], and tone-mapping [Gharbi et al. 2017]. Recent work has also demonstrated the ability of deep neural networks to replicate existing ISP pipelines [Chen et al. 2017; Fan et al. 2018; Tseng et al. 2019; Xu et al. 2015].

Furthermore, they are increasingly used for post-capture downstream tasks, such as face filtering, scene understanding, reconstruction, and object detection. Indeed, state-of-the-art performance for these vision tasks has been achieved with deep neural networks.

4 COMPOUND OPTICS PIPELINE OPTIMIZATION

We detail an end-to-end, differentiable model that implements each step of our image formation model (Sec. 3): compound optics, sensing, low-level and high-level image processing. We use our pipeline to jointly optimize optical design parameters $\mathcal{P}_{\text{OPTIC}}$ and image post-processing parameters, which can include—but are not limited to—hardware ISP parameters $\mathcal{P}_{\text{ISP}}$ and/or neural network weights $\mathcal{P}_{\text{NN}}$ (see Fig. 2), in an end-to-end fashion.

4.1 Compound optics modeling

Although optics design software, such as Zemax and Code V, incorporate powerful optical simulators, their non-differentiable black-box nature prevents us from directly incorporating them into our end-to-end differentiable pipeline. We circumvent the non-differentiability of these systems by modeling them with a neural network f_{OPTIC} that we train to predict spatially-varying PSFs and vignette given optics parameters $\mathcal{P}_{\text{OPTIC}}$. Specifically, from Eq. (1), we approximate the true optical function f_{OPTIC} with a network $\tilde{f}_{\text{OPTIC}}$ parameterized by weights $\mathcal{W}_{\text{OPTIC}}$,

$$(\tilde{\varphi}(r), \tilde{v}(r)) = \tilde{f}_{\text{OPTIC}}(r, \mathcal{P}_{\text{OPTIC}}; \mathcal{W}_{\text{OPTIC}}), \tag{6}$$

where $\tilde{\varphi}(r)$ and $\tilde{v}(r)$ are the (estimated) PSF and vignette factors at field r. The neural network $\tilde{f}_{\text{OPTIC}}$, illustrated in Fig. 5, separately outputs an energy-preserving PSF (unit sum across RGB channels in PSF_{EST}) for a given radial position r and a 3-vector (RGB) vignette. Finally, we scale the PSF channel-wise by the vignetting factor. We observe that separately predicting the normalized PSFs and vignette factors increases performance compared to direct prediction.

As shown in Fig. 5, our neural network architecture comprises a multi-layer perceptron (MLP) encoder combined with a convolutional decoder. We output the (per-PSF) vignette factor directly from the MLP, while the decoder generates the PSF. We obtain optical network weights $\hat{\mathcal{W}}_{\text{OPTIC}}$ by minimizing a loss $\mathcal{L}_{\text{OPTIC}}$ between the network predictions and ground truth $\mathbf{O}$:

$$\hat{\mathcal{W}}_{\text{OPTIC}} = \underset{\{\mathcal{W}_{\text{OPTIC}}\}}{\arg\min} \sum_{i,j}^{M,K} \mathcal{L}_{\text{OPTIC}}\left(\tilde{f}_{\text{OPTIC}}\left(r^{(j)}, \mathcal{P}_{\text{OPTIC}}^{(i)}; \mathcal{W}_{\text{OPTIC}}\right), \mathbf{O}^{(i,j)}\right). \tag{7}$$

Here, the sum is over M optical designs, each with K PSFs. We center and normalize optical parameters by their mean and standard deviation (across the M optical designs) before passing them to $\tilde{f}_{\text{OPTIC}}$. In our experiments, we uniformly sample spatial distances r across $K = 13$ locations along the vertical axis of the input image and use $M = 5.5 \times 10^4$ designs. We opt for a discrete sampling instead of a dense continuous sampling of every pixel for computational efficiency, see supplemental document for details. The image $\mathbf{I}_{\text{OPTIC}}$ is reconstructed by rotating each $\tilde{\varphi}(r)$ to obtain PSF predictions for all locations at distance r from the center of the scene, and using the procedure from Sec. 3.2 (see Eq. (2)).

Our optics meta-network requires that the cardinality of input parameters be fixed. Lifting this requirement is a direction of future research. Nevertheless, training a suite of optics meta-networks for several different parameter sets is feasible as the time for network training is around 6 hours. Note that robustness to manufacturing errors can be incorporated by adding noise to the input parameters, however, we did not find this necessary.

Training data generation. We obtain ground truth PSFs with traditional optics design software, e.g., OpticStudio by Zemax. Zemax allows us to accurately compute optical path differences with a time-consuming ray-tracer [Hanika and Dachsbacher 2014; Harvey et al. 2015; Schrade et al. 2016; Steinert et al. 2011], including aspherical surfaces, scattering, flare and diffraction. From a basic lens system design (e.g., a 3-element design; Fig. 3), we randomly sample within predefined ranges for each parameter to generate the superset of M optical parameters $\mathcal{P}_{\text{OPTIC}}^{(i)}$ in Eq. (7).

Of note, however, is that we face additional constraints when determining plausible ranges for each parameter. Slight changes in parameters can greatly affect the performance of the optical system, and we are additionally bound by constraints imposed by the manufacturing process as we wish our lens designs to be physically realizable. We perform a tolerance sensitivity analysis in Zemax to enforce this. Once we assign a viable tolerance to each component, and subsequently determine parameter ranges, we generate thousands of random variations of the compound lens.

For each lens variation we uniformly sample FOVs, and we obtain their corresponding PSFs (projected onto the sensor plane) with ray-tracing and PSF simulation in ZEMAX. Since we assume rotational symmetry in the compound lens designs (Sec. 3.2), we only simulate PSFs sampled from the positive vertical axis of the FOV. Also, during PSF simulation, our sampling accounts for the target sensor resolution. In practice, training the optics PSF representation model with super-resolved PSF data leads to more accurate fits: e.g., in Fig. 3, we sample the target FOV uniformly, and we simulate the corresponding PSFs for 128×128 micrometer sensor areas while the target sensor pixel size is 5.86 µm. For PSF simulation, we rely on Huygens PSF calculation of the optical system [Sun 2016]. Huygens PSF calculation is computationally more expensive than FFT PSF calculation but is more accurate.

Loss function. We compute the loss $\mathcal{L}_{\text{OPTIC}}$ from Eq. 7 on the estimated (energy-preserving) PSF $\tilde{\varphi}$ and vignette factor $\tilde{v}$ as

$$\begin{aligned}\mathcal{L}_{\text{OPTIC}}(\tilde{\varphi},\tilde{v},\varphi^*,v^*) &= \mathcal{L}_1(\tilde{\varphi},\varphi^*) + \mathcal{L}_1(\mathcal{F}(\tilde{\varphi}),\mathcal{F}(\varphi^*)) \\ &+ \sum_d \mathcal{L}_1(\nabla_d\tilde{\varphi},\nabla_d\varphi^*) + \mathcal{L}_1(\tilde{v},v^*)\,, \end{aligned} \tag{8}$$

where $\mathcal{F}(\cdot)$ is the Fourier transform, and ∇_d is the forward difference operator in direction d. Here, superscripts $(^*)$ indicate ground truth values. Please refer to our supplemental document for more details on network architecture, training, and datasets.

Validating the optics meta-network. Before detailing the remaining components of our method, we illustrate the capabilities of our optics network $\tilde{f}_{\text{OPTIC}}$ in Fig. 6 for accurately parameterizing PSFs on a real compound lens—here the Kowa 1/2" LM6NCL lens [Kowa 2020], which contains 7 optical elements. We measured PSFs with a Trioptics ImageMaster using broad spectrum and a "photopic eye" filter, which transmits light in proportion to the human eye's natural response [Galvoptics 2020]. We measured PSFs at five distances from the image plane, $\pm\{0, 20, 40\}$µm and train $\tilde{f}_{\text{OPTIC}}$ to reproduce these measured PSFs given the distance to the image plane. The qualitative results in Fig. 6 demonstrate that our optics network can accurately reproduce spatially-varying PSFs, even with minute details encoded depending on the design parameters. We provide further validation of the optics network in our supplemental document.

4.2 Differentiable sensor and ISP model

We implement our differentiable sensor model f_{SENSOR} as described in Sec. 3.3: it accepts the post-optic image $\mathbf{I}_{\text{OPTIC}}$ as input and outputs the sensor-produced RAW image $\mathbf{I}_{\text{RAW}} = f_{\text{SENSOR}}(\mathbf{I}_{\text{OPTIC}})$. We feed this RAW image into the post-processing ISPs. As mentioned in Sec. 3.4, our model supports both hardware and software ISPs.

Hardware ISPs. We simulate hardware ISPs (Sec. 3.4) using the approach of [Tseng et al. 2019], who learn to approximate the behavior of an ISP using a deep UNet-style CNN. Similarly, we use a network to learn the mapping from an input RAW image $\mathbf{I}_{\text{RAW}}$ and ISP parameters $\mathcal{P}_{\text{ISP}}$ to the ISP output image $\mathbf{I}_{\text{ISP}} = \tilde{f}_{\text{ISP}}(\mathbf{I}_{\text{RAW}}, \mathcal{P}_{\text{ISP}}; \mathcal{W}_{\text{ISP}})$, where $\mathcal{W}_{\text{ISP}}$ are trainable network weights obtained by minimizing

$$\hat{\mathcal{W}}_{\text{ISP}} = \underset{\{\mathcal{W}_{\text{ISP}}\}}{\arg\min} \sum_{i=1}^{M} \mathcal{L}_{\text{ISP}}\left(\tilde{f}_{\text{ISP}}\left(\mathbf{I}^{(i)}, \mathcal{P}_{\text{ISP}}^{(i)}; \mathcal{W}_{\text{ISP}}\right), \mathbf{O}^{(i)}\right), \tag{9}$$

on a set of M input/output $(\mathbf{I}^{(i)}/\mathbf{O}^{(i)})$ training pairs (see [Tseng et al. 2019]). We combine an $\mathcal{L}_1$ loss on the image domain and a perceptual loss (from a pre-trained AlexNet [Zhang et al. 2018]).

We base the network architecture for $\tilde{f}_{\text{ISP}}$ on a UNet, which accepts a multi-channel tensor as input, with the input RAW image $\mathbf{I}_{\text{RAW}}$ as the first channel, and the remaining channels are the ISP parameters (with each parameter replicated to fill an entire channel). This mirrors [Tseng et al. 2019], with the exception that we prepend a non-trainable bilinear demosaicking layer to the network to handle varying CFA patterns. This layer converts the single channel RAW sensor image into an RGB tensor. Note that we are not limited to bilinear demosaicking and that any differentiable demosaicker can be used for this step. The trained ISP proxy is appended to the remainder of the pipeline, obtaining $\mathbf{I}_{\text{ISP}} = \tilde{f}_{\text{ISP}}(\mathbf{I}_{\text{RAW}}, \mathcal{P}_{\text{ISP}}; \mathcal{W}_{\text{PROXY}})$.

Software ISPs. Software ISPs (Sec. 3.5), particularly those parameterized as deep neural networks, are trivial to employ in our pipeline in a differentiable manner. We similarly append software ISPs f_{NN} parameterized by $\mathcal{P}_{\text{NN}}$ to our pipeline as $\mathbf{I}_{\text{NN}} = f_{\text{NN}}(\mathbf{I}_{\text{RAW}}; \mathcal{P}_{\text{NN}})$.

Note that here, we do not employ superscripts $(\tilde{\ })$ as the network is not used to approximate a physical process (as $\tilde{f}_{\text{OPTIC}}$ and $\tilde{f}_{\text{ISP}}$ approximated both physical optics and ISP respectively).

5 JOINT OPTIMIZATION

5.1 Fully differentiable imaging pipeline

Our full end-to-end pipeline using a hardware ISP is given as

$$\mathbf{O} = \tilde{f}_{\text{ISP}}\left(f_{\text{SENSOR}}\left(\tilde{f}_{\text{OPTIC}}(\mathbf{I}, \mathcal{P}_{\text{OPTIC}}; \mathcal{W}_{\text{OPTIC}})\right), \mathcal{P}_{\text{ISP}}; \mathcal{W}_{\text{ISP}}\right), \tag{10}$$

where $\mathbf{I}$ is the input RGB image, $\mathbf{O}$ is the output image, and $\tilde{f}_{\text{OPTIC}}$ produces the post-optic image as described in Sec. 4.1.

If we instead employ a software ISP then our full pipeline becomes

$$\mathbf{O} = f_{\text{NN}}\left(f_{\text{SENSOR}}\left(\tilde{f}_{\text{OPTIC}}(\mathbf{I}, \mathcal{P}_{\text{OPTIC}}; \mathcal{W}_{\text{OPTIC}})\right); \mathcal{P}_{\text{NN}}\right). \tag{11}$$

Due to the differentiability of each of pipeline stage, we can concatenate arbitrarily many image post-processing stages. One such example is for automotive object detection, where the sensor RAW image is first processed by a hardware ISP before being fed into an object detection network. In this case, our full pipeline is

$$\mathbf{O} = f_{\text{NN}}\left(\tilde{f}_{\text{ISP}}\left(f_{\text{SENSOR}}\left(\tilde{f}_{\text{OPTIC}}(\mathbf{I}, \mathcal{P}_{\text{OPTIC}}; \mathcal{W}_{\text{OPTIC}})\right), \mathcal{P}_{\text{ISP}}; \mathcal{W}_{\text{ISP}}\right); \mathcal{P}_{\text{NN}}\right). \tag{12}$$

At this point, $\tilde{f}_{\text{OPTIC}}$ and $\tilde{f}_{\text{ISP}}$ are fully trained, and so their weights $\{\mathcal{W}_{\text{OPTIC}}, \mathcal{W}_{\text{ISP}}\}$ are fixed (but included in Eq. (12) for completeness).

We can now minimize task-specific losses $\mathcal{L}_{\text{TASK}}$ with respect to the system parameters ($\mathcal{P}_{\text{OPTIC}}$, $\mathcal{P}_{\text{ISP}}$, and $\mathcal{P}_{\text{NN}}$) in order to determine what the best combination of optics, ISP and image processing parameters are in a task-dependent manner:

$$\{\mathcal{P}^*_{\text{OPTIC}}, \mathcal{P}^*_{\text{ISP}}, \mathcal{P}^*_{\text{NN}}\} = \underset{\{\mathcal{P}_{\text{OPTIC}}, \mathcal{P}_{\text{ISP}}, \mathcal{P}_{\text{NN}}\}}{\arg\min} \sum_{i=1}^{M} \mathcal{L}_{\text{TASK}}\left(\mathbf{O}^{(i)}, \mathbf{T}^{(i)}\right). \tag{13}$$

Here, example tasks include image-to-image translation, where the target $\mathbf{T}$ is a desired high-quality image, or an image-to-abstraction task with scene segmentation or a bounding box map targets.

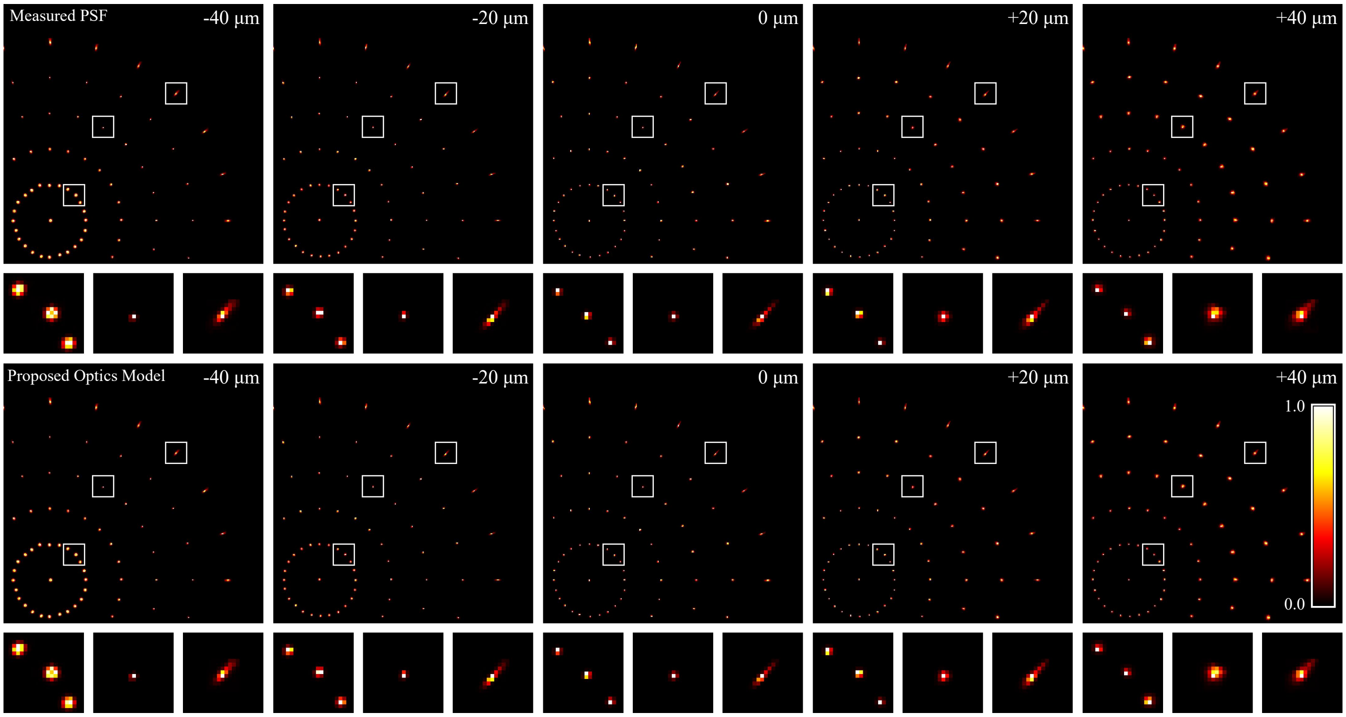


Fig. 6. Our proposed optics meta-network can learn diverse PSFs of real optical systems. Top row: Optical spatial PSFs from a Kowa 1/2” LM6NCL compound lens with seven (7) elements at $\pm\{0, 20, 40\}$µm from the image plane. Bottom row: Reproduced PSFs from the proposed optics meta-network. Please zoom into the electronic document to see details.

5.2 Proximal Compositional Optimization

Joint end-to-end optimization of several different processing blocks is challenging due to many local minima in the loss landscape and sensitivity to initialization states. We detail an optimization method that allows for efficient end-to-end design below. Note that while we do not provide formal guarantees, we validate the method extensively in Sec. 6.3.

Our method optimizes differentiable compositions of functions,

$$F\left(x,\, \bigcup_{i=1}^{K} \mathcal{P}_i\right) = f_K(f_{K-1}(\dots f_1(x,\, \mathcal{P}_1)\dots,\, \mathcal{P}_{K-1}),\, \mathcal{P}_K)\,, \quad (14)$$

with respect to a global loss $\mathcal{L}$, where differentiable functions f_i depend on parameters $\mathcal{P}_i$. Eq. (12) is one instance of this class of functions, where x is an RGB image, $f_1 = \tilde{f}_{\text{OPTIC}}$, $f_2 = f_{\text{SENSOR}}$, $f_3 = \tilde{f}_{\text{ISP}}$, $f_4 = f_{\text{NN}}$ and $\mathcal{P}_1 = \mathcal{P}_{\text{OPTIC}}$, $\mathcal{P}_2 = \emptyset$, $\mathcal{P}_3 = \mathcal{P}_{\text{ISP}}$, $\mathcal{P}_4 = \mathcal{P}_{\text{NN}}$. Our algorithm (see listing Alg. 1) operates as follows.

Initialization. Careful parameter initialization is key to exploring a diversity of possibilities, while avoiding local minima. Each $\mathcal{P}_i$ can be initialized through random sampling or from a pre-determined initialization. In our experiments, we initialize $\mathcal{P}_{\text{OPTIC}}$ with uniform random sampling. We always initialize ISP parameters $\mathcal{P}_{\text{ISP}}$ randomly and uniformly. Network software ISPs consist of many more parameters than the other stages and so, to avoid local minima, we first pre-train the software ISPs on synthetic training data.

Compositional optimization. We individually optimize each parameter set $\mathcal{P}_i$ for n_i steps in a round-robin fashion. All parameters are optimized with respect to the same task loss $\mathcal{L}_{\text{TASK}}$. We train for n_{C} cycles, where each cycle consists of $\sum_{i=1}^{K} n_i$ training steps. This alternating optimization scheme yields finer control over the optimization of individual function blocks.

Proximal regularization. If a certain stage evolves too rapidly, then it may be difficult to optimize the other stages in tandem. We propose and employ a proximal regularizer (inspired by Eq. (1.3b) of [Xu and Yin 2013]) to stabilize training. Specifically, our proximal regularization loss for a specific parameter vector $\mathcal{P}_i$ is

$$\mathcal{L}_{\text{PROX}}\left(\mathcal{P}_i(t), \beta_i\right) = \beta_i \left\|\mathcal{P}_i(t) - \mathcal{P}_i(t+1)\right\|_2^2\,, \quad (15)$$

where $\mathcal{P}_i(t)$ and $\mathcal{P}_i(t+1)$ are the current and next iterates, and β_i is a scalar weight. During compositional optimization we add $\mathcal{L}_{\text{PROX}}$ to the global task loss $\mathcal{L}_{\text{TASK}}$.

Fine-tuning. After n_{C} cycles, we train all parameters end-to-end—without alternating nor proximal regularization—for n_{F} cycles.

6 ANALYSIS AND SYNTHETIC VALIDATION

In this section, we first validate the utility and effectiveness of the proposed method using simulated optical designs.

6.1 Cooke triplet optimization

We first present a series of experiments that employ our optics network $\tilde{f}_{\text{OPTIC}}$ to determine the optimal task-specific parameters of a “Cooke triplet” [Kidger 2002], an established lens design composed of three optical elements which corrects Seidel aberrations. To minimize manufacturing efforts for the handful of lens systems

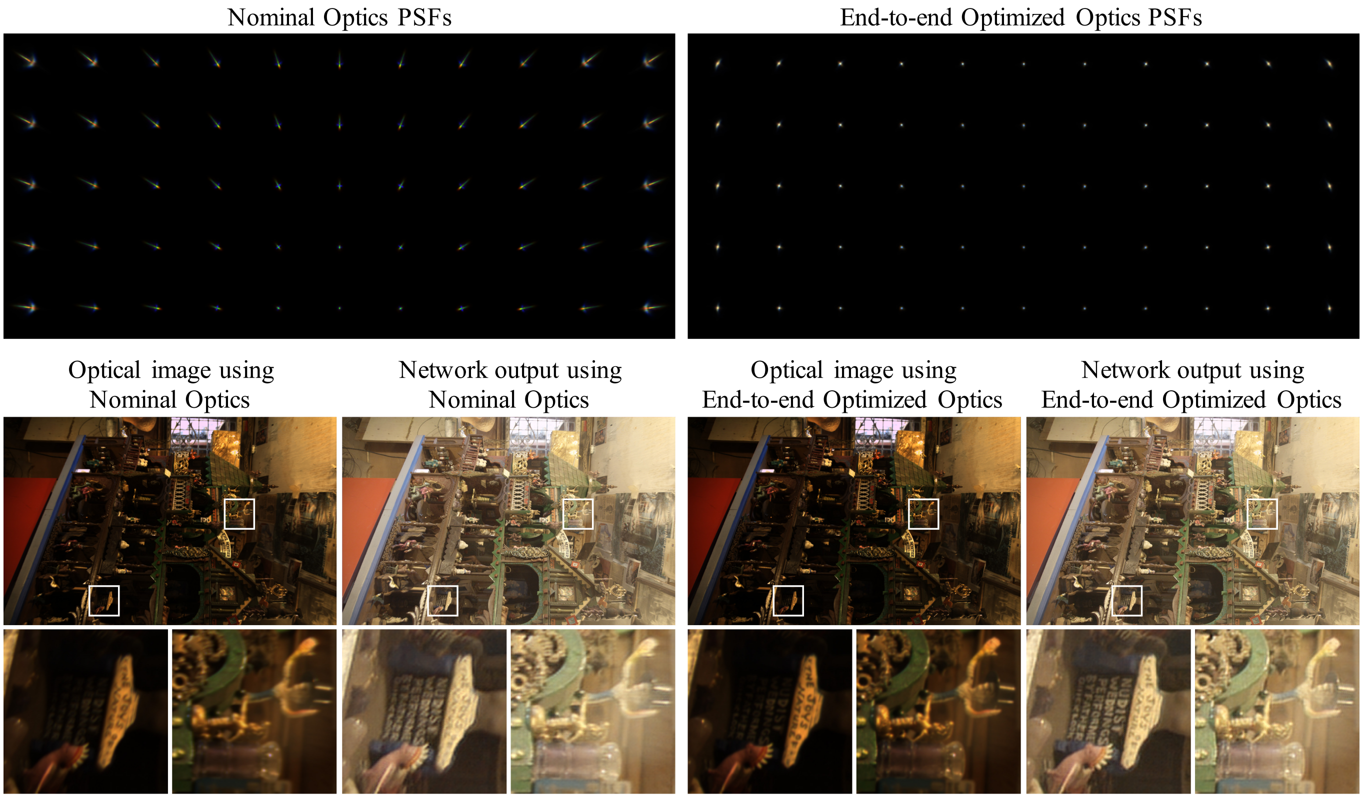


Fig. 7. Image quality with Cooke triplet and software ISP using simulated measurements. Images produced using the nominal optics (left) are blurrier and have more color artifacts than images produced using our optimized optics (right). Simulated PSFs from our optics meta-network are shown at the top. The end-to-end optimized optic trades off sharp central focus in return for more compact PSFs across the FOV, which enables the downstream software ISP to produce high-quality tone-mapped images although the spot size of the center PSF is slightly larger.

ALGORITHM 1: Proximal Compositional Optimization

$\bigcup_{i=1}^{K} \mathcal{P}_i$ = `Initialize()`
for $t = 1, \ldots, n_{\mathrm{C}}$ **do** // Compositional Optimization
  for $f_i \in \{f_1, \ldots, f_K\}$ **do** // Round-robin over f_i
    for $j = 1, \ldots, n_i$ **do**
      $\mathcal{L}' = \mathcal{L}_{\mathrm{TASK}} + \mathcal{L}_{\mathrm{prox}}(\mathcal{P}_i, \beta_i)$
      `Update` $\left(\mathcal{P}_i, \partial \mathcal{L}' \big/ \partial \mathcal{P}_i\right)$
    end
  end
end
for $t = 1, \ldots, n_{\mathrm{F}}$ **do** // Fine-tuning
  `Update` $\left(\bigcup_{i=1}^{K} \mathcal{P}_i, \partial \mathcal{L} \big/ \partial \bigcup_{i=1}^{K} \mathcal{P}_i\right)$
end

that we intended to fabricated for this work, we select an off-the-shelf bi-concave glass element (Thorlabs LD2297-A in BK7 material) as the center element. Not only does this allow us to employ two material types in our design (as our manufacturing facilities were limited to PMMA lens fabrication), it has the added benefit that this element is coated with an anti-reflective film, reducing lens flare. All experiments use the methodology presented in Sec. 4.1 for training $\tilde{f}_{\mathrm{OPTIC}}$, but each adapt the loss function $\mathcal{L}_{\mathrm{TASK}}$ from Eq. (13) and training set to different tasks. For our sensor simulation, we have calibrated a 2.3 megapixel Sony IMX249 sensor with IR cutoff filter (specifications BFLY-U3-23S6C-C) with exposure of 5 ms, see supplemental document for details.

For all experiments, the optics parameters to optimize $\mathcal{P}_{\mathrm{OPTIC}}$ are the ones shown in Tab. 1, where each lens element is denoted by its two surfaces. We consider the first and the sixth surfaces as rotationally symmetric polynomial aspheric surfaces, and optimize their spherical radius parameters and higher-order aspheric coefficients. While these parameters are all continuous, discrete parameters can be handled using the approach of Tseng et al. [2019] by using a continuous relaxation.

Nominal Optics Design. The experiments below compare optimized optical parameters against a "nominal" design, obtained using the Hammer optimization in Zemax with the primary goal of enforcing as focusing performance as much as possible across the field of view. To this end, we apply the default OPD (optical path difference) merit function with the following physical constraints: effective focal length, element thickness, air gap, and back focal distance. These constraints still permit a high degree of freedom, from which the Hammer optimization yields a high quality optic

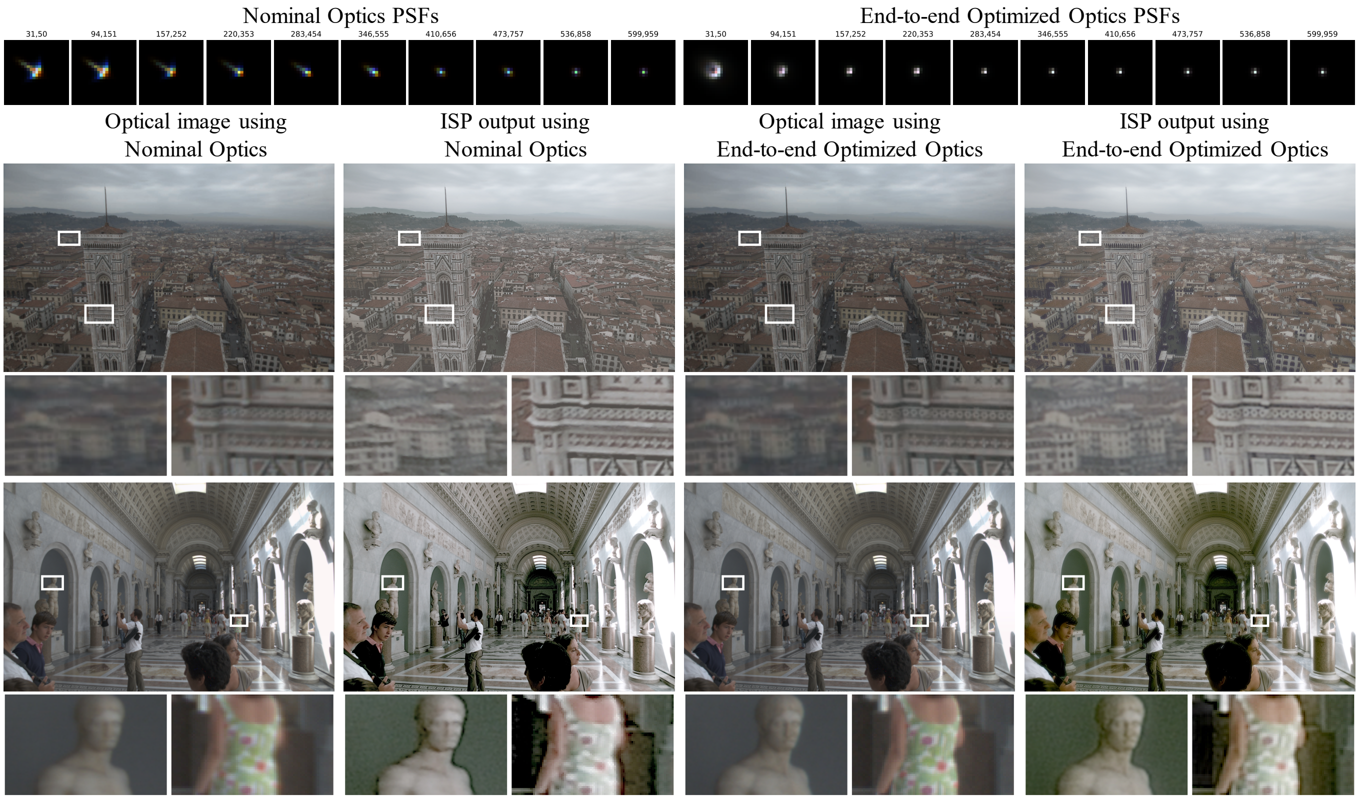


Fig. 8. Image quality with Cooke triplet and hardware ISP using simulated measurements. Although both PSFs have a support (number of non-zero entries on sensor) similar in size, the long streak component of the nominal optics (left) results in the ISP tends to overly unsharp mask which generates shadow artifacts and noise. The optical images produced using our optimized optics (right) exhibit a more circular distribution and the ISP output hence has less artifacts. Displayed optics PSFs have been resampled for the sensor array. We show the PSFs along the first half of the main diagonal of the 1200×1920 sensor. The center pixel coordinates (row, column) of the spatial PSFs are indicated above each PSF, where (0, 0) refers to the top-left corner.

design with an RMS spot radius of 10 microns. We set the min/max constraints described in Tab. 1 around this nominal optic.

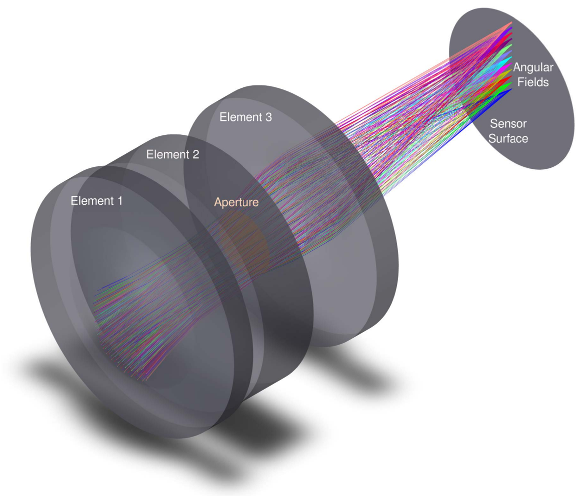
The process of obtaining a high-quality nominal baseline using *traditional optics engineering is a time-consuming manual effort.* For our particular configuration, in addition to the geometric design constraints, glass materials could not be used due to fabrication availability, and aspheric surfaces need to be optimized, making manual design a non-trivial task. In our first attempt, a human error in considering the manufacturing constraints led to a baseline lens design with an RMS spot radius of 30 micron after a two-week design process of optimizations and analyses of the design in Zemax. We refer to Sec. 7.2 of the supplemental document for the experiments we have performed considering this baseline design. The final nominal design took *one month effort* for an experienced optical engineer.

Image quality with software ISP. We begin with the common task of capturing images for human viewing. Here, we use a software ISP f_{NN} (Eq. (11)) to perform the tone mapping operation immediately after the sensor readout. A standard UNet architecture is used for f_{NN}. The training set is the MIT-Adobe FiveK dataset [Bychkovsky et al. 2011] and the RGB scenes used are under good-lighting conditions with high photon flux. The inputs are linear RGB images and the target tone-mapped images follow the tone-mapping performed by expert tuning. The simulated imaging pipeline from Eq. (11) is trained to produce the target tone-mapped images as closely as possible using a weighted perceptual quality loss. Specifically, we use $\mathcal{L}_{\text{TASK}} = \mathcal{L}_1 + \mathcal{L}_{\text{LPIPS}}$ where $\mathcal{L}_{\text{LPIPS}}$ is the perceptual loss using pre-trained AlexNet described by Zhang et al. [2018]. To simultaneously optimize for $\mathcal{P}_{\text{OPTIC}}$ and $\mathcal{P}_{\text{NN}}$, we employ the alternating scheme from Sec. 5.2. For fair comparison against the nominal optic, we optimize f_{NN} with the same task loss while keeping f_{OPTIC} fixed at the nominal parameter settings.

Through our end-to-end optimization process, we observe that the optimized optic design sacrifices the sharp focus at the center of the FOV in return for tight PSFs that minimize chromatic aberrations across the sensor FOV, see PSFs in Fig. 7. These traits enable superior joint performance with the software ISP as can be seen qualitatively for the images (post processing) in Fig. 7 and quantitatively in Tab. 2. Please see the supplemental document for additional results.

Image quality with hardware ISP. We perform the same experiment using a hardware ISP $\tilde{f}_{\text{ISP}}$ as a post-processor instead of the neural network f_{NN}. Here, we use the ARM Mali-C71 hardware ISP and train $\tilde{f}_{\text{ISP}}$ as in Sec. 4.2. We again employ alternating optimization from above to minimize $\mathcal{L}_{\text{TASK}} = \mathcal{L}_1 + 3\mathcal{L}_{\text{LPIPS}}$. We see that the

Table 1. Parameters for the three-element Cooke triplet. We follow the optics CAD terminology and denote each lens element by its two surfaces [Garrard et al. 2005]. Accordingly, we refer to the aperture and the imaging plane by surface 5 and surface 8, respectively. We enforce the min/max constraints for all optimized lenses and the nominal lens.



Parameter	Min	Max	Units	Description
s_1_radius	9.85	14.98	mm	radius of the 1st surface
s_1_conic	-0.49	0.29	-	conic constant of the 1st surface
s_2_radius	9.45	14.82	mm	radius of the 2nd surface
l_12	4.58	10.11	mm	distance between lens 1 and lens 2
l_2STO	1.03	9.22	mm	distance between lens 2 and aperture
l_STO3	0.0	9.86	mm	distance between aperture and lens 3
s_6_radius	13.38	18.17	mm	radius of the 6th surface
s_6_conic	-0.49	0.49	-	conic constant of the 1st surface
s_7_radius	-15.05	-11.50	mm	radius of the 7th surface
s_1_2nd	-4.99e-3	4.99e-3	mm^{-1}	2nd order coefficient of polynomial fit to 1st surface
s_1_4th	-9.06e-5	-1.45e-5	mm^{-3}	4th order coefficient of polynomial fit to 1st surface
s_1_6th	-5.10e-7	6.30e-7	mm^{-5}	6th order coefficient of polynomial fit to 1st surface
s_1_8th	-1.58e-8	-2.66e-11	mm^{-7}	8th order coefficient of polynomial fit to 1st surface
s_1_10th	-1.28e-10	1.08e-10	mm^{-9}	10th order coefficient of polynomial fit to 1st surface
s_6_2nd	-4.99e-3	4.99e-3	mm^{-1}	2nd order coefficient of polynomial fit to 6th surface
s_6_4th	-2.57e-4	-1.41e-4	mm^{-3}	4th order coefficient of polynomial fit to 6th surface
s_6_6th	-1.44e-6	1.73e-6	mm^{-5}	6th order coefficient of polynomial fit to 6th surface
s_6_8th	-4.54e-8	4.92e-7	mm^{-7}	8th order coefficient of polynomial fit to 6th surface
s_6_10th	-2.33e-8	8.80e-10	mm^{-9}	10th order coefficient of polynomial fit to 6th surface

Table 2. Quantitative evaluation of end-to-end design and nominal design using simulated measurements on an unseen validation set for image quality. In addition to PSNR and SSIM as conventional metrics, we also report 1 - LPIPS [Zhang et al. 2018] as a perceptual metric (higher is better).

Methods	1 - LPIPS	PSNR	SSIM
End-to-end with Neural Network	0.961	35.6	0.942
Nominal with Neural Network	0.914	32.0	0.899
End-to-end with Hardware ISP	0.811	21.2	0.892
Nominal with Hardware ISP	0.750	21.1	0.871

optimized optic features a similiar support but different distribution without the elongated streak of the nominal design. We compare against the nominal optic using the expert-tuned settings for the hardware ISP, qualitative and quantitative results are shown in Fig. 8 and Tab. 2 respectively. Please see the supplemental document for further results.

Single-image low-light imaging. Low-light imaging is another important task which is affected by both the optics used for capture and the post-processing algorithms. For this experiment, we feed in RGB images from the MIT-Adobe FiveK dataset and scale down the exposure of the simulated Sony IMX249 by a 100× factor to 5 µs, and then setting the gain factor in our sensor model f_{SENSOR} to compensate for this scaling difference. We use the same compound optics, software ISP architecture, and loss as in the "image quality with software ISP" experiment. Note that both networks are able to learn to deconvolve the jointly optimized PSFs. Even for such fine-tuned processing, shown quantitatively in Tab. 3 and qualitatively in Fig. 9, our end-to-end pipeline demonstrates improved perceptual quality with substantially more fine detail preserved compared to fine-tuning the network f_{NN} for the given the nominal optics. Please see the supplemental document for additional results.

Table 3. Quantitative evaluation of end-to-end design and nominal design using simulated measurements on an unseen validation set for low-light imaging.

Methods	1 - LPIPS	PSNR	SSIM
End-to-end with Neural Network	0.865	33.3	0.870
Nominal with Neural Network	0.827	30.9	0.829

Automotive object detection and traffic light state detection with hardware ISP. We now jointly optimize a full end-to-end pipeline for automotive object and traffic light detection, consisting of a Cooke triplet compound lens, ARM Mali-C71 ISP for image processing, and a Faster-RCNN [Ren et al. 2015] (with a ResNet-28 backbone) object detector, which we dub "FRCNN" for short in the following.

For object detection (OD), we rely on a training dataset created from BDD100K [Yu et al. 2020] by grouping different categories, resulting in 6 categories: car/van/suv, bus/truck/tram, person, bike, traffic lights, traffic signs. We use an additional 20000 images captured using a FLIR BFLY-23S6C-C camera with a Fujinon CF12.5HA lens to handle European scenes. For fair comparison against our end-to-end optimized pipeline, we fine-tuned the detector and ISP for the pipeline using the nominal optics on images simulated with the same nominal optics. We evaluate both pipelines on a validation set consisting of 10000 images from BDD100K and 10386 images from our additional captures. Quantitative metrics demonstrating improved object detection performance are shown in Tab. 4.

We also optimize the same pipeline for traffic light detection (TL). This time, we rely on the DriveU dataset [Fregin et al. 2018] and our own captures, again with fine-tuning for the nominal optics pipeline. For the labels, only 10 categories of front-facing traffic lights were considered: red circle/straight/left/right, green circle/straight/left/right, yellow circle, red-yellow circle. We evaluate both pipelines on a validation set consisting of 10905 images from DriveU and 1851 images from our additional captures. Quantitative metrics demonstrating improved traffic light detection performance are shown in Tab. 4. We refer to the supplemental document for qualitative results in simulation for both object detection and traffic light detection.

In both cases, the proposed method learns optics and processing pipelines that are *different* from the optics for perceptual image quality. These simulated optics follow the same trend as the manufactured prototype optics in Fig. 10, which we show here ahead of the detailed description in Sec. 7.1 for brevity. Specifically, both

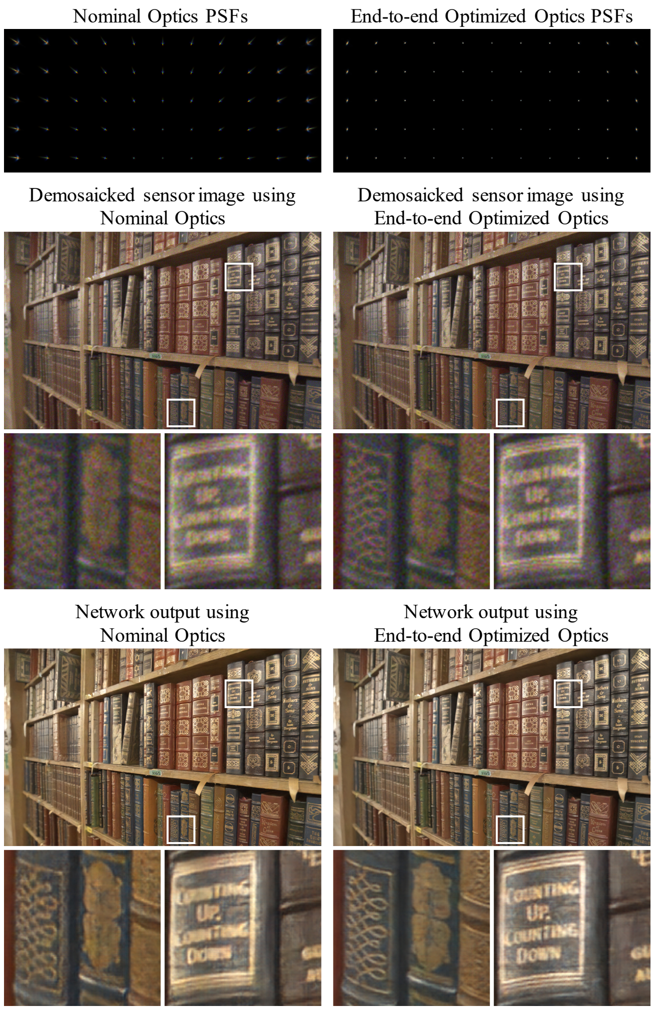


Fig. 9. Low-light imaging with Cooke triplet and software ISP using simulated measurements. We jointly optimize a Cooke triplet together with a neural network for image denoising. Input image intensities are reduced by 100× to 5 µs (Sony IMX249 sensor) to imitate low-light conditions and the sensor gain is set to compensate. The demosaicked sensor image is shown for both approaches to highlight the noise level in addition to the effect of the optics. Our optimization produces an optic with compact spatial PSFs that assist the neural network in recovering low-light image content.

Table 4. Mean Average Precision (mAP) for object detection (OD) and traffic light (TL) state detection (see text) for our optimized end-to-end pipeline versus the nominal system using simulated measurements. We use 20386 images for OD validation and 12756 images for TL validation.

Methods	OD	TL
End-to-end with Hardware ISP and FRCNN	43	61
Nominal with Hardware ISP and FRCNN	34	53

the simulated and real learned optics have spot sizes comparable to the nominal optics but the learned designs are substantially faster lenses (f/3.2 for TL and f/3.3 for OD) compared to the nominal design (f/4.4). For OD and TL tasks, this design improves detections in low-intensity image regions that are challenging to denoise by the conventional (fine-tuned) ISP. The hardware ISP accentuates noise and shadow artifacts when attempting to compensate for the lower light efficiency of the nominal optics, which in turn results in reduced detection accuracy. In addition, the OD design favors a uniform PSF over all fields, while the TL lens exhibits a PSF with a slightly stronger peak component in the far periphery, aiding the detection of small details, e.g., traffic light arrows. Please see the supplemental document for qualitative visualizations.

Optical properties of optimized lens designs. Please see the supplemental document for further detail on the optimized optics and their design trade-offs compared to the nominal expert-designed optic.

6.2 Eight-element achromat experiments

We demonstrate that the applicability of our method extends to more complex compound lenses. Specifically, we repeat the "Image quality with hardware ISP" experiment for an 8-element achromat compound lens using the same optimization procedure as before. As this compound lens has many more degrees of freedom than the Cooke triplet, we expect to be able to learn nearly any spatial PSF suited towards our desired applications. The nominal lens design is well-corrected with small PSF spot sizes across all fields. Perhaps surprisingly, our experiments demonstrate that even for this complex lens system our approach retrieves compact PSFs that match and slightly improve upon the nominal PSFs, see qualitative results in Fig. 11. This is confirmed by the quantitative results in Tab. 5 which reports improvements in perceptual quality with the LPIPS an SSIM metric while keeping SNR the same. We refer to the supplemental document for additional results.

6.3 Validation of the optimization method

We demonstrate that our proposed optimization scheme described in Sec. 5.2 has superior optimization performance than vanilla end-to-end optimization for joint end-to-end optimization. In machine learning practice, the parameters of deep neural networks are often optimized using a vanilla stochastic gradient optimizer (e.g. SGD or Adam) which updates all trainable parameters with the same optimization settings (e.g. same learning rate) at each training iteration. While this is often sufficient for an isolated processing unit (with all others fixed), the proposed Alg. 1 is substantially less prone to local minima for our multi-stage imaging pipelines.

We perform a validation experiment by repeating the "Image quality with software ISP" and "Single-image low-light imaging" experiment from Sec. 6.1, but we now compare against the performance obtained when directly applying a vanilla stochastic gradient optimizer to all trainable parameters in a non-alternating fashion. For these comparison experiments we apply Alg. 1 by using the Adam optimizer with learning rate 10^{-2} for $\mathcal{P}_{\text{OPTIC}}$ and using the Adam optimizer with learning rate 10^{-4} for $\mathcal{P}_{\text{NN}}$. Quantitative comparisons are shown in Fig. 12. We observed that using the same optimizer negatively impacted optimization performance. For the "Image quality with software ISP" experiment the vanilla optimizer became stuck in a local minima whereas our proposed optimization scheme successfully optimizes the imaging pipeline. For the "Single-image low-light imaging" experiment the vanilla optimizer

Fig. 10. Real-world prototype captures for automotive object and traffic light detection with Cooke triplet and hardware ISP. The manufactured prototypes are tested in the wild and demonstrate that our optimization allows for higher accuracy object and traffic light detection and classification. Note that our traffic light detector is trained to recognize vehicle traffic lights and ignores pedestrian traffic lights. The optimized optics have greater light efficiency (smaller f-number) and more uniform blur across the field of view than the nominal optic, which leads to greater detection performance. The optic optimized for traffic light detection is slightly sharper in the center and the peripheries than the optic optimized for object detection due to the size of traffic lights. Note that the object detection captures were taken during daytime whereas the traffic light detection captures were taken at dusk.

manages to optimize the pipeline but fails to achieve the same performance as the proposed optimization. Although it is possible that the vanilla optimization could eventually converge to the same point as our proposed optimization, the experiment demonstrates that the proposed optimization achieves much faster convergence. Although our method converges within a day on a single GPU, we ran these experiments for more than one week for a fair comparison. We also use the same random initialization point for both Alg. 1 and the vanilla stochastic gradient optimizer in these comparison experiments.

7 EXPERIMENTAL VALIDATION

7.1 Lens prototype manufacturing

We validate our proposed method by manufacturing five physical lens prototypes (two iterations for the nominal design, see Sec. 6.1, three obtained with our optimization procedure) and testing them on three different applications. With the manufacturing constraints and sensors available to us, we opted for a typical mid-to-far range automotive camera configuration using Cooke triplets with a field of view of 25°, allowing us to analyze image quality and detection of small objects at a distance typically affected the most by aberrations or ISP processing settings. We refer to the supplemental document for simulations with a larger field of view. All fabricated Cooke triplets have an effective focal length of 25 mm, and real clear aperture size of 5 mm (although optimized designs can stray from these initial values slightly). The designs comprise a negative flint glass element (Thorlabs LD2297, N-SF11 Bi-Concave Lens with AR coating) in the center with a positive polymethyl methacrylate (PMMA)

Table 5. Quantitative evaluation of the eight-element lens designs from Fig. 11 using simulated measurements. As explained in the text, the margin of improvement is smaller than that of the Cooke triplet experiment because of the substantially greater degrees of freedom of the eight-element lens. Note that, nevertheless, the proposed optimization still manages to achieve slightly higher image quality. Note that the different field of views between the eight-element lens and the Cooke triplet results in different evaluation settings, thus these values are not comparable to those in Tab. 2.

Methods	1 - LPIPS	PSNR	SSIM
End-to-End with Hardware ISP	0.760	18.5	0.683
Nominal with Hardware ISP	0.728	18.5	0.675

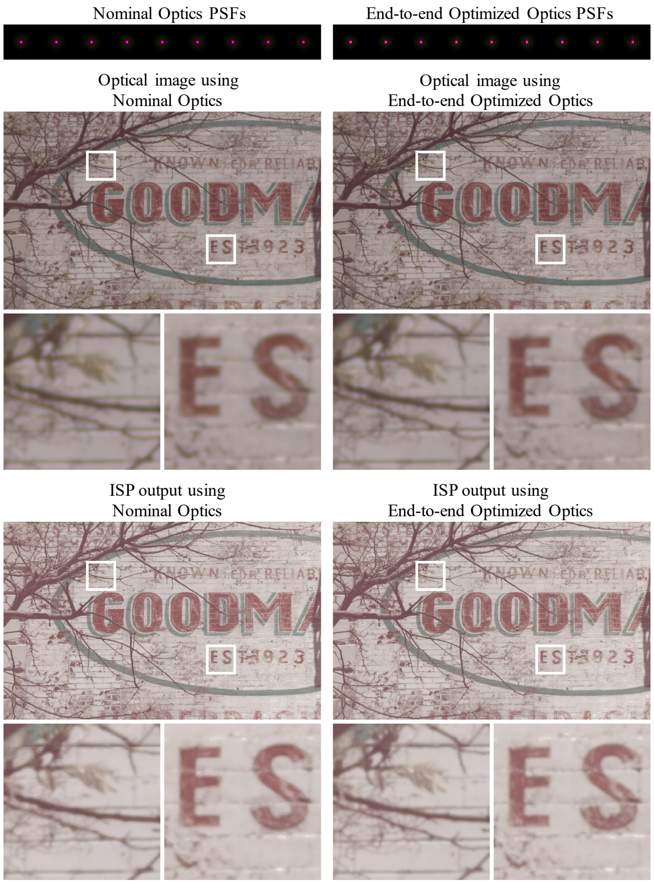
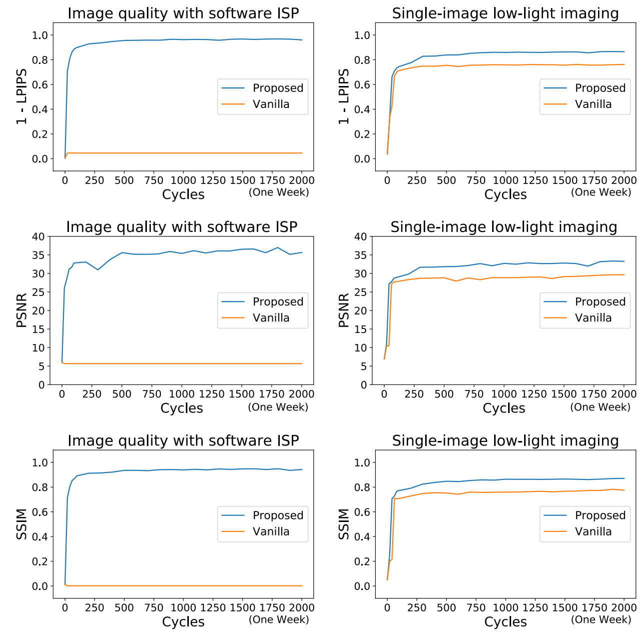


Fig. 11. Image quality with 8-element achromat compound lens and hardware ISP using simulated measurements. In addition to Cooke triplet optimization, we are also able to optimize an 8-element achromat compound lens for natural image capture. The PSFs produced by the optimized optic are slightly more compact than those of the expert-designed optic, demonstrating that our method is indeed applicable to complex optical systems. Please zoom into the electronic document to see details.



Methods	1 - LPIPS	PSNR	SSIM
Image quality with software ISP using proposed opt. (alg. 1)	0.961	35.6	0.942
Image quality with software ISP using vanilla opt.	0.045	5.67	0.001
Single-image low-light imaging using proposed opt. (alg. 1)	0.865	33.3	0.870
Single-image low-light imaging using vanilla opt.	0.762	29.6	0.776

Fig. 12. Quantitative comparison of proposed optimization method (alg. 1) against vanilla non-alternating backpropagation for "Image quality with software ISP" and "Single-image low-light imaging" using simulated measurements. Applying a vanilla stochastic gradient optimizer to all trainable parameters resulted in worse performance compared to using our algorithm. Note that, for fairness, all training runs start from the same random initialization point. To ensure convergence in the machine learning sense, we run the vanilla optimization for more than one week.

element on each side. The two positive elements comprise an aspherical and a spherical surface, whose parameters are optimized by our approach. The substrate PMMA has a refractive index of 1.493 at the principle wavelength of 550 nm. This combination of materials with different Abbe numbers mitigates chromatic aberrations, while the use of aspherical surfaces provides more degrees of freedom to the optimization for achieving the desired optical behavior. Refer to the supplemental document for detailed specifications of all lenses.

To fabricate our customized lenses, we use a CNC machining system that supports 5-axis single point diamond turning (Nanotech 350FG) [Fang et al. 2013; Peng et al. 2019]. This process supports a high precision ($\lesssim$ 1 µm) regarding the tolerance of the physical height of a continuous surface. We use standard mechanical turning to manufacture mounts, tubes, spacers, and physical barrels with aluminum alloy for assembling multiple optical elements, with a measured tolerance of 20 µm. In the design space, we empirically apply the constraints on the minimum air gap of 1 mm, the minimum edge thickness of optical elements of 2 mm, and the minimum back focal distance of 20 mm. A cross-section diagram of one representative lens is presented in the supplemental document. All lenses are assembled via C-mount to a FLIR BFLY-U3-23S6C-C camera with the same 2.3 megapixel Sony IMX249 sensor that was used for the synthetic experiments in the previous section (Sec. 6). To facilitate reproducibility we will provide all Zemax files and detailed lens manufacturing instructions.

7.2 Real-world validation of optimized Cooke triplets

We use the learned optics parameters $\mathcal{P}^*_{\text{OPTIC}}$ obtained in simulation for three of the experiments shown in Sec. 6.1, namely image quality with hardware ISP, automotive object detection, and traffic light classification. After manufacturing each of the individual lens assemblies, we measure the spatially-varying PSF of the prototype lenses using a pinhole light source with a 75 µm pinhole diameter placed at 2 m from the camera. With the pinhole source at infinity focus, instead of moving the pinhole on a translation stage, we rotate the camera viewing angle while keeping the position fixed to measure the spatially-varying PSFs. We fine-tune all baseline and optimized downstream network and ISP blocks with these post-manufacturing

Table 6. Quantitative image quality evaluations using experimental measurements captured with the fabricated prototype lenses. We compare here nominal vs. our end-to-end optimized design, performed using the chart proposed in [Tseng et al. 2019]. In addition to PSNR and SSIM as conventional metrics, we also report 1 - LPIPS [Zhang et al. 2018] as a perceptual metric (higher is better).

Methods	1 - LPIPS	PSNR	SSIM
End-to-end with Hardware ISP	0.598	17.25	0.787
Nominal with Hardware ISP	0.565	12.64	0.760

Table 7. Quantitative pedestrian-vehicle and traffic light detection evaluations using experimental measurements captured with the fabricated prototype lenses. Mean Average Precision (mAP) for object detection (OD) and traffic light (TL) state detection for our fabricated end-to-end optimized system versus the expert-designed nominal lens with detectors fine-tuned on captures from the same nominal optics.

Methods	OD	TL
End-to-end with Hardware ISP and FRCNN	46	31
Nominal with Hardware ISP and FRCNN	40	27

PSFs. Although identical RGB scenes can be used as input to different optical systems for the software experiments in Sec. 6, capturing identical frames with real lens systems is challenging—especially for the automotive experiments. For our testing setup, we placed two prototype lenses side-by-side; see the capture vehicle shown in Fig. 1. We did not use a beam-splitter setup as these can cause significant flare for HDR scenes. The two camera systems are synced using a hardware trigger and use the same fixed exposure for fair comparison. The results of each experiment are described next.

Image quality with hardware ISP. Similar to what was observed in simulation, our optimized compound lens reduces the aberrations that are present in the nominal compound lens. As such, the images acquired with our jointly optimized pipeline are superior to those acquired by separately tuning the optics and the hardware ISP. Qualitative results are shown in Fig. 13 and Fig. 14 and quantitative metrics are shown in Tab. 6.

Figs. 13 and 14 show a qualitative comparison of our optimized design against the nominal. When compared against the nominal compound lens, our end-to-end optimized optics and ISP demonstrate substantially sharper image quality in the peripheries and similar performance in the center, as evidenced by the color-checker and text inset in Fig. 13 and the city insets in Fig. 14. Tab. 6 validates this quantitatively, demonstrating that our optimized lens designs yields improved quantitative quality metrics (LPIPS, PSNR, and SSIM), computed on the custom chart from [Tseng et al. 2019].

Automotive object detection. For the fabricated optimized object detection lens and the nominal lens we performed synchronized dual-camera capture in a dense urban area in North America. We manually annotated 2005 dual camera pairs for evaluation with a total of 30,264 objects falling in the pedestrian and vehicle classes as described in Sec. 6. We use an unbiased team of annotators to separately annotate the nominal and target lens captures. Fig. 1 and Fig. 10 show example captures acquired and processed with the proposed system. In low-intensity regions, the captures and processed results with the nominal lens suffer from the lower light efficiency due the larger f-number (f/4.4) compared to the end-to-end learned optical system (f/3.2). The hardware ISP is not able to recover enough signal in these low-flux regions and instead amplifies measurement noise. As a result, even with slightly larger PSF, the proposed end-to-end learned system outperforms the nominal design in object detection. As a result of the spatial distribution and the object size, the learned optics prefer uniform aberrations across all field instead, which we attribute to the fact that it is detrimental for the convolutional detector to learn spatially-varying processing. Fig. 1 and Fig. 10 validate these characteristics and shows examples where the nominal system fails to detect low-light edge boundaries between objects (e.g., parked row of cars are often missed). Object detection with the nominal lens misses pedestrians and small objects with complex background (Fig. 1 center left). We note that the object detection network architecture for our end-to-end pipeline is the same as the simulation one from Sec. 6 except with the optics and the sensor simulation layers removed for inference on captured data. The results in Tab. 7 validate that the proposed system performs significantly better in terms of 2D mean average precision.

Traffic light state detection. Using the same synchronized dual-camera setup we validate the proposed approach using our end-to-end optimized traffic light state detection lens compared to the nominal (fine-tuned) system. For the assessment, we annotated 2264 dual-camera captures with a total of 8442 traffic lights with states annotated using the same labels as described in Sec. 6. Similar to the OD lens design, the TL lens is faster (f/3.3) than the nominal design (f/4.4), resulting in substantially improved SNR, which improves detections especially in low-flux regions where the (fine-tuned) nominal ISP is not capable of recovering enough signal. As a result of the spatial distribution and size of the small traffic lights, the traffic light lens differs from the previous lens for pedestrian-vehicle detection. Specifically, the TL lens exhibits a PSF with small spot-size in the center, where small traffic lights at a distance appear, and it has a PSF with a peak component in the periphery, where closer traffic lights appear in the upper periphery of the sensor. Compared to the nominal design, this peak component results in sharper details in the periphery. Fig. 10 shows examples where improved sharpness and the lower f-number aid the detection of small traffic lights especially in challenging scenes with low ambient illumination. Here, for the nominal lens the arrows appear circular and are often too blurred to be detected. Tab. 7 validates that our end-to-end optimized optics and processing outperforms the nominal system also quantitatively on the captured validation set.

8 DISCUSSION AND CONCLUSION

Limitations. Our method does not replace human optics and software engineers in the end-to-end design process—analogously, neither do ZEMAX or TENSORFLOW/PYTORCH for optical design or machine learning design. Rather, our approach augments compartmentalized camera design tools by bridging a longstanding gap between heterogeneous sensing, compute, and algorithm design. Furthermore, end-to-end optimization is fundamentally limited by the availability of real-world data needed to simulate image formation and end-to-end task losses, e.g., an IoU detection loss. Given that larger

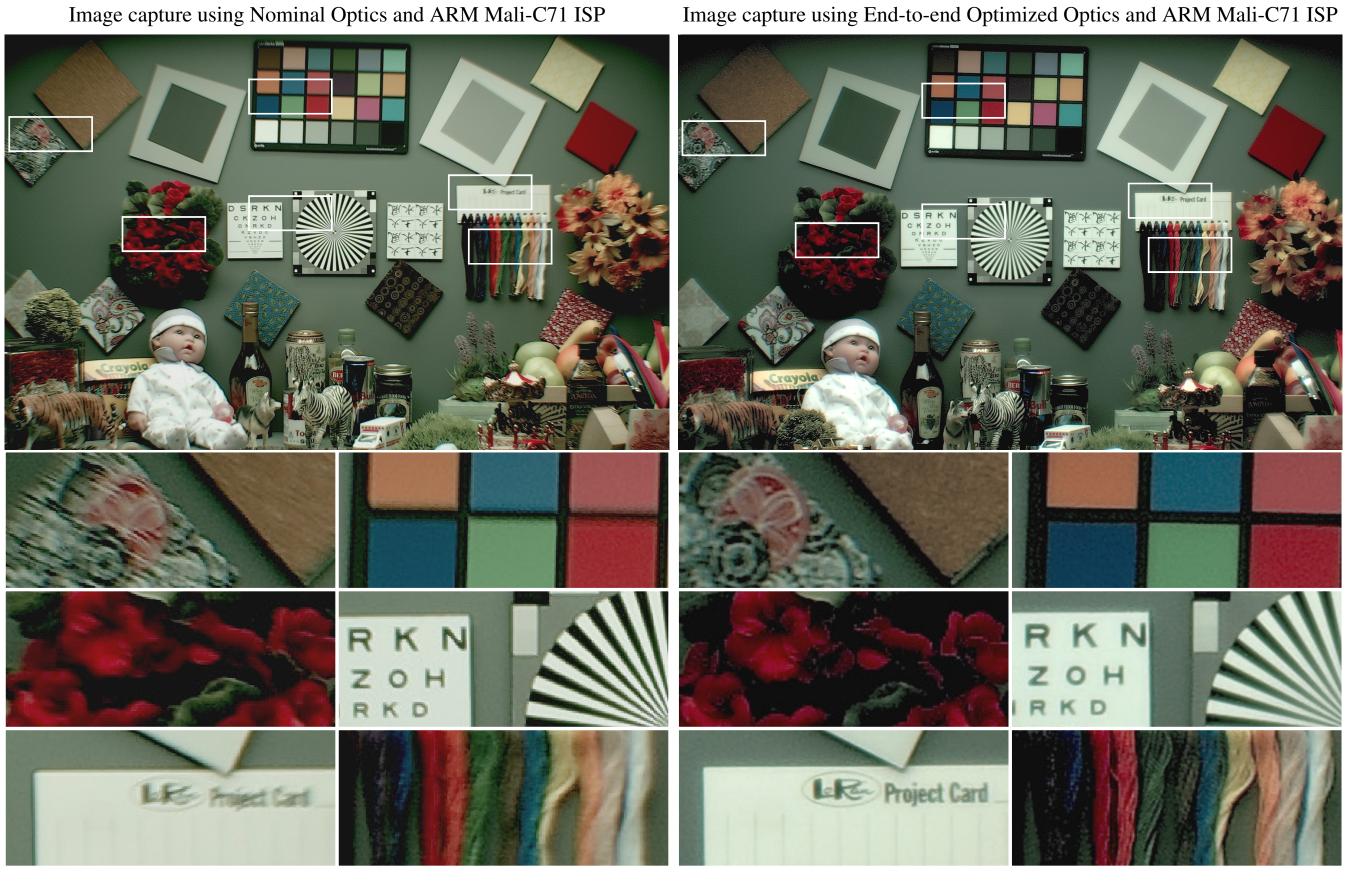


Fig. 13. Real-world prototype captures for image quality with Cooke triplet and hardware ISP. In this experimental capture we have a lightbooth set up with several objects. The image produced using the manufactured nominal optics (left) is overall blurrier than the image produced using our manufactured optimized optics (right). As discussed in the text, the optimized optics trades off slight defocus in the center of the scene for drastically improved image quality throughout the entire field of view. This trade off is a result of the limited degrees of freedom of the Cooke triplet and that a compound lens with greater degrees of freedom does not require this trade off as shown in Sec. 6.2. The six insets below the full image further highlights the differences. Note that since these images were captured sequentially, there is an unavoidable slight misalignment between the images captured using the two optics.

training corpora of realistic high-resolution multi-spectral data are not readily available, we concentrate on RGB optical designs in the optical forward model. We also assumed the scene to be at infinity, which precludes applicability on depth-sensitive applications such as mobile photography for which auto-focus is necessary.

Conclusion. This paper introduces a framework for joint end-to-end optimization of a compound lens model together with a realistic sensor model, hardware (or software) ISP, and downstream CNN computer vision module. We jointly train all parameters and hyperparameters of this heterogeneous camera pipeline for a domain-specific loss. The proposed framework builds on traditional tolerance analysis and seamlessly integrates with traditional optics design methods. Based on the optimized optical parameters obtained from our fully-differentiable imaging pipeline, we build five prototype compound lens designs and assess them on real-world driving data for automotive camera design. We validate the proposed method on alternative optics and post-processing for human viewing in challenging outdoor and low-light scenarios. We also validate the method for automotive camera optics together with hardware ISP post-processing and detection, beating state-of-the-art self-driving vehicle camera designs. In all applications, the approach outperforms existing compartmentalized design or fine-tuning qualitatively and quantitatively on *all* domain-specific applications tested. Furthermore, we have demonstrated that producing high-quality images for human viewing is not a necessary or even desirable constraint for machine vision applications such as automotive object detection.

Possible future directions include incorporating automatic neural architecture search [Elsken et al. 2019] and lens design search to circumvent the requirement that the number of lens elements must be specified a priori. Finally, automating sensor design, active illumination, and fusion with different sensors, e.g. in multi-camera acquisition systems, are exciting avenues for future research that this work makes first step towards.

Image capture using Nominal Optics and ARM Mali-C71 ISP | Image capture using End-to-end Optimized Optics and ARM Mali-C71 ISP

Fig. 14. Real-world prototype captures for image quality with Cooke triplet and hardware ISP. In this experimental capture we use a screen to display natural images. The image produced using the manufactured nominal optics (left) is blurrier than the image produced using our manufactured optimized optics (right). As discussed in the text, the optimized optics trades off slight defocus in the center of the scene for drastically improved image quality throughout the entire field of view. This trade off is a result of the limited degrees of freedom of the Cooke triplet and that a compound lens with greater degrees of freedom does not require this trade off as shown in Sec. 6.2. The six insets below the full image further highlights the differences. Note that since these images were captured sequentially, there is an unavoidable slight misalignment between the images captured using the two optics.

9 ACKNOWLEDGMENTS

We thank our reviewers for their invaluable comments. We thank Adam Finkelstein, Szymon Rusinkiewicz, and Hatem Ben Zakour for helpful discussions. Felix Heide was supported by an NSF CAREER Award (2047359) and a Sony Faculty Innovation Award.

REFERENCES

ISO 12233:2017. 2017. Photography - Electronic still picture imaging – Resolution and spatial frequency responses. (January 2017), 1–49. https://www.iso.org/standard/71696.html

IEEE Std 1858-2016. 2017. IEEE Standard for Camera Phone Image Quality. *IEEE Std 1858-2016 (Incorporating IEEE Std 1858-2016/Cor 1-2017)* (May 2017), 1–146. https://doi.org/10.1109/IEEESTD.2017.7921676

Donald Baxter, Frederic Cao, Henrik Eliasson, and Jonathan Phillips. 2012. Development of the I3A CPIQ spatial metrics. *Proc.SPIE* 8293. https://doi.org/10.1117/12.905752

Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. 2011. Learning Photographic Global Tonal Adjustment with a Database of Input / Output Image Pairs. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Julie Chang, Vincent Sitzmann, Xiong Dun, Wolfgang Heidrich, and Gordon Wetzstein. 2018. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification. *Scientific reports* 8, 1 (2018), 12324.

Julie Chang and Gordon Wetzstein. 2019. Deep Optics for Monocular Depth Estimation and 3D Object Detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. 2018. Learning to See in the Dark. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Qifeng Chen, Jia Xu, and Vladlen Koltun. 2017. Fast Image Processing With Fully-Convolutional Networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Jung-Min Choi, Sung-Joon Jang, Sang-Seol Lee, Youngbae Hwang, and Byeong Ho Choi. 2014. Memory optimization of bilateral filter and its hardware implementation. In *The 18th IEEE International Symposium on Consumer Electronics (ISCE 2014)*. 1–2.

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. 2007. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Processing* 16, 8 (2007).

Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. 2019. Neural Architecture Search: A Survey. *Journal of Machine Learning Research* 20, 55 (2019), 1–21.

EMVA 1288 2016. EMVA 1288 Standard for Characterization of Image Sensors and Cameras. (December 2016). https://www.emva.org/wp-content/uploads/EMVA1288-3.1a.pdf

Qingnan Fan, Jiaolong Yang, David Wipf, Baoquan Chen, and Xin Tong. 2018. Image Smoothing via Unsupervised Learning. *ACM Trans. Graph. (SIGGRAPH Asia)* 37, 6 (2018).

Fengzhou Fang, Xiaodong Zhang, Albert Weckenmann, Guoxiong Zhang, and Chris Evans. 2013. Manufacturing and measurement of freeform optics. *CIRP Annals* 62, 2 (2013), 823–846.

Grant R. Fowles. 1989. *Introduction to modern optics.* Courier Corporation.

Andreas Fregin, Julian Müller, Ulrich Kreβel, and Klaus Dietmayer. 2018. The DriveU traffic light dataset: Introduction and comparison with existing datasets. In *IEEE International Conference on Robotics and Automation (ICRA)*.

Galvoptics. 2020. Photopic Eye Response Filter. https://www.galvoptics.co.uk/optical-components/optical-filters/photopic-eye-response-filter/. (2020).

Kenneth Garrard, Thomas Bruegge, Jeff Hoffman, Thomas Dow, and Alex Sohn. 2005. Design tools for freeform optics. In *Current Developments in Lens Design and Optical Engineering VI*, Vol. 5874. International Society for Optics and Photonics, 58740A.

Carl Friedrich Gauss. 1843. *Dioptrische Untersuchungen von CF Gauss.* in der Dieterichschen Buchhandlung.

Joseph M. Geary. 2002. *Introduction to lens design: with practical ZEMAX examples.* Willmann-Bell Richmond.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research* 32, 11 (2013), 1231–1237.

Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand. 2016. Deep joint demosaicking and denoising. *ACM Trans. Graph.* 35, 6 (2016), 191.

Michaël Gharbi, Jiawen Chen, Jon Barron, Samuel W. Hasinoff, and Frédo Durand. 2017. Deep Bilateral Learning for Real-Time Image Enhancement. *ACM Trans. Graph. (SIGGRAPH)* (2017).

Radek Grzeszczuk, Demetri Terzopoulos, and Geoffrey Hinton. 1998. NeuroAnimator: Fast Neural Network Emulation and Control of Physics-based Models. In *Proc. of the 25th Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH).* ACM.

Johannes Hanika and Carsten Dachsbacher. 2014. Efficient Monte Carlo rendering with realistic lenses. In *Computer Graphics Forum*, Vol. 33. Wiley Online Library, 323–332.

James E. Harvey, Ryan G. Irvin, and Richard N. Pfisterer. 2015. Modeling physical optics phenomena by complex ray tracing. *Optical Engineering* 54, 3 (2015), 035105.

Samuel W. Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jon Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. 2016. Burst Photography for High Dynamic Range and Low-light Imaging on Mobile Cameras. *ACM Trans. Graph.* 35, 6, Article 192 (2016), 12 pages.

James Hegarty, John Brunhaver, Zachary DeVito, Jonathan Ragan-Kelley, Noy Cohen, Steven Bell, Artem Vasilyev, Mark Horowitz, and Pat Hanrahan. 2014. Darkroom: Compiling High-level Image Processing Code into Hardware Pipelines. *ACM Trans. Graph. (SIGGRAPH)* 33, 4 (2014).

Felix Heide, Markus Steinberger, Yun-Ta Tsai, Mushfiqur Rouf, Dawid Pająk, Dikpal Reddy, Orazio Gallo, Jing Liu, Wolfgang Heidrich, Karen Egiazarian, Jan Kautz, and Kari Pulli. 2014. FlexISP: A Flexible Camera Image Processing Framework. *ACM Trans. Graph. (SIGGRAPH Asia)* 33, 6 (2014).

Michael Hirsch, Suvrit Sra, Bernhard Schölkopf, and Stefan Harmeling. 2010. Efficient filter flow for space-variant multiframe blind deconvolution. In *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 607–614.

Michael J. Kidger. 2002. *Fundamental Optical Design.* SPIE Press.

Rudolf Kingslake and Roger B. Johnson. 2009. *Lens design fundamentals.* Academic Press.

Craig E. Kolb, Don P. Mitchell, and Pat Hanrahan. 1995. A realistic camera model for computer graphics. In *SIGGRAPH '95*.

Kowa. 2020. LM6NCL. https://lenses.kowa-usa.com/ncl-series/490-lm6ncl.html. (2020).

Tzu-Mao Li, Michaël Gharbi, Andrew Adams, Frédo Durand, and Jonathan Ragan-Kelley. 2018. Differentiable programming for image processing and deep learning in Halide. *ACM Trans. Graph. (SIGGRAPH)* 37, 4 (2018), 139:1–139:13.

Daniel Malacara-Hernández and Zacarías Malacara-Hernández. 2016. *Handbook of optical design.* CRC Press.

Christopher A. Metzler, Hayato Ikoma, Yifan Peng, and Gordon Wetzstein. 2020. Deep Optics for Single-shot High-dynamic-range Imaging. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Ali Mosleh, Avinash Sharma, Emmanuel Onzon, Fahim Mannan, Nicolas Robidoux, and Felix Heide. 2020. Hardware-in-the-loop End-to-end Optimization of Camera Image Processing Pipelines. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jun Nishimura, Timo Gerasimow, Rao Sushma, Alexsandar Sutic, Chyuan-Tyng Wu, and Gilad Michael. 2018. Automatic ISP Image Quality Tuning Using Nonlinear Optimization. In *International Conference on Image Processing (ICIP)*.

ON Semi MT9P001. 2017. MT9P001: 1/2.5-Inch 5 Mp CMOS Digital Image Sensor. https://www.onsemi.com/pdf/datasheet/mt9p001-d.pdf. (2017).

Yifan Peng, Qilin Sun, Xiong Dun, Gordon Wetzstein, Wolfgang Heidrich, and Felix Heide. 2019. Learned large field-of-view imaging with thin-plate optics. *ACM Trans. Graph. (TOG)* 38, 6 (2019), 219.

Jonathan B. Phillips and Henrik Eliasson. 2018. *Camera Image Quality Benchmarking* (1st ed.). Wiley Publishing.

Rajeev Ramanath, Wesley E. Snyder, Youngjun Yoo, and Mark S. Drew. 2005. Color image processing pipeline. *IEEE Signal Processing Magazine* 22, 1 (2005), 34–43.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems.*

Max-Gerd Retzlaff, Johannes Hanika, Jürgen Beyerer, and Carsten Dachsbacher. 2016. Potential and challenges of using computer graphics for the simulation of optical measurement systems. 18. *GMA/ITG Fachtagung: Sensoren und Messsysteme* (2016), 322–329.

Emanuel Schrade, Johannes Hanika, and Carsten Dachsbacher. 2016. Sparse high-degree polynomials for wide-angle lenses. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 89–97.

Ling Shao, Ruomei Yan, Xuelong Li, and Yan Liu. 2014. From Heuristic Optimization to Dictionary Learning: A Review and Comprehensive Comparison of Image Denoising Algorithms. *IEEE Transactions on Cybernetics* 44, 7 (2014), 1001–1013.

Yichang Shih, Brian Guenter, and Neel Joshi. 2012. Image enhancement using calibrated lens simulations. In *European Conference on Computer Vision.*

Vincent Sitzmann, Steven Diamond, Yifan Peng, Xiong Dun, Stephen Boyd, Wolfgang Heidrich, Felix Heide, and Gordon Wetzstein. 2018. End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging. *ACM Trans. Graph. (TOG)* 37, 4 (2018), 114.

Georgii Georgievich Sliusarev. 1984. Aberration and optical design theory. *Bristol, England, Adam Hilger, Ltd., 1984, 672 p. Translation.* (1984).

Warren J. Smith. 2005. Modern lens design. (2005).

Benjamin Steinert, Holger Dammertz, Johannes Hanika, and Hendrik PA Lensch. 2011. General spectral camera lens simulation. In *Computer Graphics Forum*, Vol. 30. Wiley Online Library, 1643–1654.

David G. Stork and Patrick R. Gill. 2014. Optical, mathematical, and computational foundations of lensless ultra-miniature diffractive imagers and sensors. *International Journal on Advances in Systems and Measurements* 7, 3 (2014), 4.

Haiyin Sun. 2016. *Lens design: a practical guide.* Crc Press.

Libin Sun, Neel Joshi, Brian Guenter, and James Hays. 2015. Lens Factory: Automatic Lens Generation Using Off-the-shelf Components. *ArXiv* abs/1506.08956 (2015).

Qilin Sun, Ethan Tseng, Qiang Fu, Wolfgang Heidrich, and Felix Heide. 2020. Learning Rank-1 Diffractive Optics for Single-shot High Dynamic Range Imaging. In *IEEE Conference on Computer Vision and Pattern Recognition.*

Qilin Sun, Congli Wang, Qiang Fu, Xiong Dun, and Wolfgang Heidrich. 2021. End-to-End Complex Lens Design with Differentiable Ray Tracing. *ACM Transactions on Graphics (TOG)* 40, 4 (2021).

Carlo Tomasi and Roberto Manduchi. 1998. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on.* IEEE, 839–846.

Ethan Tseng, Felix Yu, Yuting Yang, Fahim Mannan, Karl ST Arnaud, Derek Nowrouzezahrai, Jean-François Lalonde, and Felix Heide. 2019. Hyperparameter optimization in black-box image processing using differentiable proxies. *ACM Trans. Graph. (TOG)* 38, 4 (2019), 27.

Bruce H. Walker. 2008. *Optical engineering fundamentals.* Vol. 82. Spie Press Bellingham.

Li Xu, Jimmy Ren, Qiong Yan, Renjie Liao, and Jiaya Jia. 2015. Deep Edge-Aware Filters. In *International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Francis Bach and David Blei (Eds.), Vol. 37. PMLR, Lille, France, 1669–1678.

Yangyang Xu and Wotao Yin. 2013. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences* 6, 3 (2013), 1758–1789.

Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. 2020. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In *IEEE Conference on Computer Vision and Pattern Recognition.*

Hao Zhang, Wenjiang Liu, Ruolin Wang, Tao Liu, and Mengtian Rong. 2016. Hardware architecture design of block-matching and 3D-filtering denoising algorithm. *Journal of Shanghai Jiaotong University (Science)* 21, 2 (2016), 173–183.

Lei Zhang, Xiaolin Wu, Antoni Buades, and Xin Li. 2011. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic Imaging* 20, 2 (2011).

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE International Conference on Computer Vision and Pattern Recognition.*