The Differentiable Lens: Compound Lens Search over Glass Surfaces and Materials for Object Detection

Geoffroi Côté[1,2] Fahim Mannan[3] Simon Thibault[1] Jean-François Lalonde[1] Felix Heide[2,3]
[1]Université Laval [2]Princeton University [3]Algolux

Abstract

Most camera lens systems are designed in isolation, separately from downstream computer vision methods. Recently, joint optimization approaches that design lenses alongside other components of the image acquisition and processing pipeline—notably, downstream neural networks—have achieved improved imaging quality or better performance on vision tasks. However, these existing methods optimize only a subset of lens parameters and cannot optimize glass materials given their categorical nature. In this work, we develop a differentiable spherical lens simulation model that accurately captures geometrical aberrations. We propose an optimization strategy to address the challenges of lens design—notorious for non-convex loss function landscapes and many manufacturing constraints—that are exacerbated in joint optimization tasks. Specifically, we introduce quantized continuous glass variables to facilitate the optimization and selection of glass materials in an end-to-end design context, and couple this with carefully designed constraints to support manufacturability. In automotive object detection, we report improved detection performance over existing designs even when simplifying designs to two- or three-element lenses, despite significantly degrading the image quality.

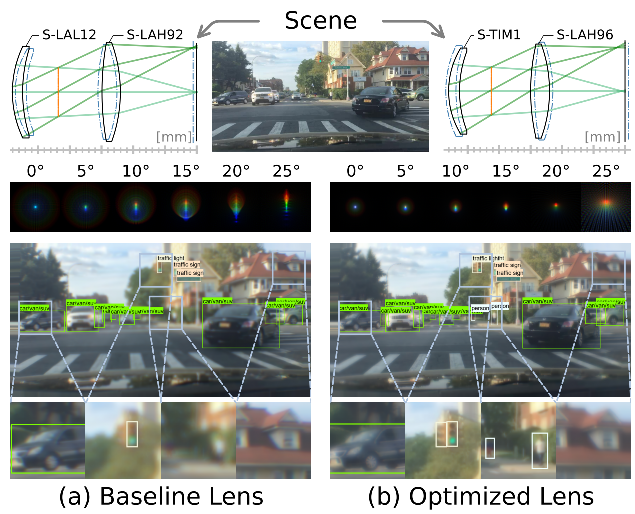

Figure 1. We introduce a differentiable lens simulation model and an optimization method to optimize compound lenses specifically for downstream computer vision tasks, and apply them to automotive object detection. Here, although the optimized two-element lens has a worse average spot size than the baseline lens (136 µm vs 80 µm), it achieves a better mean average precision (AP) on the BDD100K dataset (32.0 vs 30.3). The optimized lens sacrifices optical performance near the corners for better performance in the small and medium field values where most of the objects are located. In lens layout plots, dashed lines represent the baseline/optimized counterpart and annotations indicate the optimized glass materials.

1. Introduction

The prevailing design paradigm for typical optical systems is to conceive them in isolation by use of simplified image quality metrics such as spot size [28]. However, achieving ideal imaging properties or optimal performance on computer vision tasks generally requires a more comprehensive approach that includes the remaining parts of the image acquisition and processing chain, in particular the sensor, image signal processing, and downstream neural networks.

Over the years, many works have addressed the joint design of simple optical systems such as diffractive optical elements (DOEs) [3, 20, 27, 31]. These works approach joint optics design by simplifying the design to a single phase plate that allows for a differentiable paraxial Fourier image formation model, optimizable via stochastic gradient descent (SGD) variants. More recently, several differentiable lens simulation models have been introduced to address the more complex compound lens systems present in most commodity-type cameras. Tseng et al. [35] build such a model by training a proxy neural network, whereas other works [11, 17, 32] directly implement differentiable ray-tracing operations in automatic differentiation frameworks [1, 22], an idea also discussed in [6, 40, 41]. However, all relevant previous works [11, 17, 32, 35] optimize over only a subset of possible surface profiles and spacings, and ignore the optimization of glass materials altogether. Yet, allowing *all* lens variables to be freely optimized—that is, without predefined boundaries—provides an opportunity for increased performance on downstream tasks.

Unfortunately, lens design optimization is no trivial process. Even optimizing for traditional optical performance metrics presents significant difficulties, notably: harsh loss function landscapes with abundant local minima and saddle points [30, 36, 39], restrictive manufacturing constraints [2, 28], and risk of ray-tracing failures. Optimizing a lens jointly on vision tasks only exacerbates these pitfalls due to the noisy gradients of SGD when applied to complex vision models [35]. Moreover, joint optimization does not naturally allow external supervision from lens designers and, as such, does not necessarily result in a manufacturable lens.

In this work, we introduce a computationally efficient and differentiable pipeline for simulating and differentiating through compound spherical refractive lenses with respect to all design parameters in an end-to-end manner. Our forward model integrates exact optical ray tracing, accurate ray aiming, relative illumination, and distortion. Furthermore, we develop an optimization strategy to facilitate the end-to-end design of refractive lenses using SGD-based optimizers while strongly encouraging manufacturable outcomes. To this end, we carefully define losses to handle design constraints, and introduce *quantized continuous glass variables* to facilitate the process of selecting the best glass materials among glass catalogs that contain dozens of candidates—a challenge unmet in prior joint optimization methods.

We apply our simulation and optimization pipeline to the task of object detection (OD). We find that even simple two-element lenses such as the ones in Fig. 1 can be compelling candidates for low-cost automotive OD despite a noticeably worse image quality. Then, we validate the proposed method by demonstrating that optimizing the lens jointly with the OD model leads to consistent improvements in detection performance. We make the following contributions:

- We introduce a novel method for simulating and optimizing compound optics with respect to glass materials, surface profiles, and spacings.
- We validate the method on the end-to-end optimization of an OD downstream loss, with lenses specifically optimized for intersection over union (IoU) of bounding boxes predicted from a jointly trained detector.
- We demonstrate that the proposed method results in improved OD performance even when reducing the number of optical elements in a given lens stack.

In addition, we release our code and designs[1] in the hope of enabling further joint design applications.

Limitations In end-to-end optics design, the inherent resolution of the dataset used to represent real-world scenes—a result of the pixel count, imaging quality, and compression artifacts—needs to be discernibly superior to the modeled optics if meaningful conclusions are to be drawn. Hence, we focus on simple lenses with strong geometrical aberrations, namely refractive lenses with two to four spherical elements whose combination of aperture and field of view (FOV) exceeds the capabilities of the lens configuration. Incidentally, our method does not completely alleviate the need for human supervision; as in most lens design problems, a suitable lens design starting point is required for best performance.

[1] https://github.com/princeton-computational-imaging/joint-lens-design

	Tseng [35]	Sun [32]	Hale [11]	Li [17]	Ours
Differentiable Lens Model					
Hands-Free	✗	✓	✓	✓	✓
Efficient	✓	✗	✓	✓	✓
Accurate PSFs	✓	✓	✗	(✓)	✓
Distortion	(✓)	✓	(✓)	(✓)	✓
Aspherics	✓	✓	(✓)	✓	✗
Optimized Lens Variables					
No Boundaries	✗	✓	✓	✓	✓
Spacings	✓	✗	(✓)	✗	✓
Surface Profiles	✓	(✓)	✗	(✓)	✓
Glass Materials	✗	✗	✗	✗	✓

Table 1. Comparison of related work on the joint optimization of refractive compound optics, where each criterion is fully ✓, partially (✓), or not ✗ met. See text for explanations.

2. Related Work

We briefly review the existing literature on joint optics design based on three aspects: optics simulation, optimization, and integration with the downstream task. Tab. 1 compares our work to other approaches that focus on compound optics.

Optics Simulation Many optics simulation models consist primarily in convolving the point spread function (PSF) of the optics design over the target image. This approach is used in several works, in particular with DOEs [3, 20, 27, 31] that have a single surface where the PSF can be approximated using paraxial Fourier-based models.

In contrast, compound lenses have multiple surfaces with varying materials and surface profiles. To capture these optical systems accurately, exact ray tracing based on Snell's Law is typically used to complement paraxial optics. As many existing models [12, 16, 19, 29] are not end-to-end differentiable, recent works have introduced new differentiable lens models to enable the joint design of compound lenses. Tseng et al. [35] employ a proxy model that learns the mapping between lens variables and PSFs using pre-generated lens data; while bypassing the intricacies of ray-based rendering, it adds a cumbersome training step that needs to be repeated for every lens configuration and only works for predefined variable boundaries. Sun et al. [32] apply Monte Carlo ray tracing from every pixel of the virtual detector at the expense of computational efficiency. Halé et al. [11] and Li et al. [17] apply ray-tracing operations in a way that is reminiscent of conventional optical ray tracing [43] to compute the PSFs, with the former assuming a Gaussian shape for the PSFs and the latter assuming a square entrance

Figure 2. Overview of our joint design approach. To tackle the lens design and OD subtasks jointly rather than in isolation, we simulate realistic aberrations on the dataset images before they enter the object detector. The lens parameters ϕ_{lens} (with curvatures c, spacings s, and glass variables g) are optimized both on a loss ℓ_{lens}, which targets optical performance and geometrical constraints via exact ray tracing, and on the OD loss ℓ_{OD}. The object detector parameters ϕ_{OD} are trained concurrently to minimize ℓ_{OD} while adjusting to the lens aberrations.

pupil. The lens simulation model developed in our work is most similar to Li et al. [17] with notable additions: we implement accurate ray aiming and avoid discontinuity artifacts by addressing distortion and relative illumination in separate steps (see Tab. 1). Crucially and as opposed to our work, [17] does not optimize for computer vision tasks. We note that our work deliberately focuses on spherical lenses to enable low-cost automotive OD.

Optics Optimization Conventional lens design tasks usually seek a design of suitable complexity that fulfills a given list of specifications; these are translated into a loss function that targets optical performance criteria as well as many manufacturing constraints [2]. The lens configuration is chosen to provide sufficient degrees of freedom (DOF) and dictates the number and nature of lens variables, notably: glass materials, spacings between each optical surface, and surface profile parameters—characterized by the curvature and, for aspherics, additional polynomial coefficients [26]. The de-facto optimizer for lens optimization is the Levenberg-Marquardt algorithm [9, 42], which is also the default option [43] in common optical design software [33, 45].

In contrast, joint optics design optimizes the optics alongside the downstream neural network parameters using SGD-based optimization; as such, developing a lens optimization strategy that synergizes with SGD is a focus of our work, enabling end-to-end optimization for vision downstream tasks. Conversely, previous works circumvent these difficulties by optimizing only a subset of both spacings and surface profile parameters [11, 17, 32] or, in the case of [35], limiting the variables within predefined boundaries (see Tab. 1).

Joint Domain-Specific Optics Optimization Many downstream tasks can be grouped under the umbrella term of image reconstruction, where the goal is to retrieve the original image despite lens aberrations [17, 34] or environmental changes such as low-light imaging [35]. This includes high-dynamic-range [20, 31], large field-of-view [23], extended depth-of-field [27], and super-resolution [27] imaging. End-to-end design has also been applied to traditional vision tasks such as image classification [4], monocular depth estimation [3, 10], or OD [3, 35]. While our approach supports any downstream task that can be trained with SGD optimization, here we focus on automotive OD, relevant to self-driving vehicles and autonomous robots.

3. Differentiable Compound Lenses

In this work, we first introduce a method for the end-to-end modeling and optimization of compound lenses in computer vision tasks. We then apply the proposed method to OD as illustrated in Fig. 2. We use natural images as input to our method and as approximations of real-world scenes with the following underlying assumptions (see supp.): the objects are infinitely distant, the RGB values are proportional to the luminance, and the FOV of the simulated lens matches the scene. These approximations allow us to rely on existing image datasets to study the effect of strong geometrical aberrations such as the ones of poorly corrected optics, that is, lenses without the required DOF to correct the aberrations under the desired specifications. Indeed, sophisticated lenses (e.g., with a larger number of elements) are of limited interest in our work since they do not significantly impact image quality. Therefore, we focus on simple lenses composed of a few (2–4) spherical lens elements while noting that poorly corrected optics are *harder* to simulate accurately due to larger PSFs, distortion, and spatial variations.

The throughput of a lens is an important consideration in low-light OD. As such, we design all lenses to have similar and relatively high throughput with a fixed f-number and no any optical vignetting. Incidentally, we fix both the FOV and focal length f such that $f = {}^{d}/_{2\tan(\mathrm{FOV}/2)}$ to ensure that the corners of the virtual image sensor (with diagonal d) correspond to the maximum FOV—assuming reasonable distortion and defocus.

In the following, we describe the core components of the proposed method. In Sec. 4, we elaborate on our complete differentiable lens simulation model. In Sec. 5, we detail the lens parameters ϕ_{lens} and our joint optimization strategy. We assess the proposed method experimentally in Sec. 6.

4. Optical Image Formation Model

Sampling and tracing rays from every pixel of the virtual detector [16, 32] is computationally prohibitive for our joint

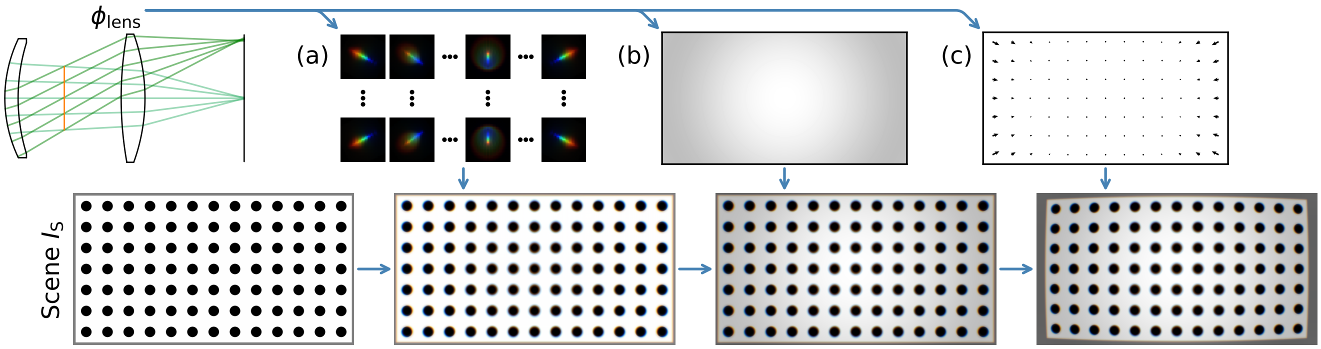

Figure 3. From the lens parameters ϕ_{lens}, our lens simulation model employs exact differentiable ray tracing to compute the spatially varying PSF grid (a), relative illumination map (b), and distortion field (c), which are successively applied to the scene image I_{S} to simulate realistic geometrical aberrations (relative illumination is 20× amplified for clarity).

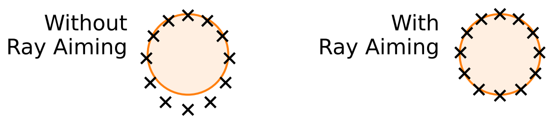

Figure 4. Rays that are targeted at the outer edge of the aperture stop (cross section is shown in orange) successfully hit the target area when using the proposed ray-aiming correction step, and badly miss otherwise. Illustrated here is for the f/2 Tessar lens used in Sec. 6 at full field of view (25°, see supp. for more examples).

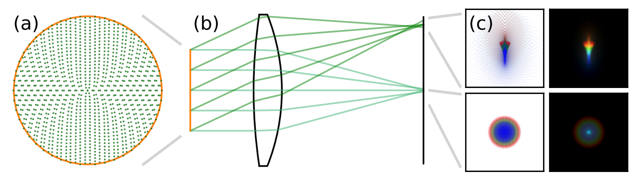

Figure 5. Computation of the geometrical PSFs used to simulate realistic aberrations. We first initialize rays at the entrance pupil (a), which in this example overlaps with the aperture stop (orange) located in object space. We propagate the rays using exact ray-tracing operations (b) to obtain the spot diagrams. Then, we apply kernel density estimation to retrieve the PSFs for all field values (c).

design approach. Instead, our differentiable lens simulation model applies a spatially varying convolution to the scene image I_{S} to generate the aberrated image

$$I_{\text{A}}(x', y') \approx \text{PSF}(x', y') * I_{\text{S}}(x', y') \,. \tag{1}$$

We model Eq. (1) by discretizing the image into a grid of patches that are convolved with their corresponding PSF. While the PSFs can theoretically take distortion and relative illumination into account (as in Tseng et al. [35]), here we simulate them in separate steps as shown in Fig. 3. This avoids discontinuity artifacts and, in the case of distortion, artificial blurring [19] as well as an increased computational burden caused by large uncentered PSFs.

Ray Tracing Exact ray tracing is achieved by alternating between two operations: 1) updating the coordinates of the rays from one interface to the next, and 2) updating the direction cosines following Snell's Law. In practice, we batch the operations over $n_{\text{r}} = n_{\text{h}} n_{\text{w}} n_{\text{p}}$ rays, where n_{h}, n_{w}, and n_{p} are the number of field values, wavelengths, and pupil coordinates. All rays are initialized at the entrance pupil. Unlike [17], we introduce a ray-aiming correction step which is critical to accurately simulate lenses with strong pupil aberrations (see Fig. 4); as in [5], the initial transverse ray coordinates are scaled by deforming the entrance pupil into a field-dependent elliptic shape (see supp.).

Geometrical PSFs Under dominant geometrical aberrations, diffraction can be safely ignored and the PSFs can be computed through the ray-counting method: setting a virtual detector on the image plane and counting the rays hitting each bin (see Fig. 5). As the PSFs of axially symmetric lenses are invariant to azimuth, here they are sampled radially at $n_{\text{h}} = 21$ equidistant field values h, then interpolated, rotated, and resized to fill the PSF grid (see Fig. 3(a)).

First, for each field, we span the entrance pupil uniformly with rays—each representing an equal pupil area and amount of energy—and trace them up to the image plane to obtain the $x \in \mathbb{R}^{n_{\text{r}}}$ and $y \in \mathbb{R}^{n_{\text{r}}}$ coordinates that compose the spot diagrams. The pupil sampling scheme ($n_{\text{p}} = 2048$) corresponds to 32 equally spaced concentric circles with jittering to properly sample the outer edge of the pupil (see Fig. 5(a)). We trace $n_{\text{w}} = 15$ wavelengths: 5 for each color channel which are selected from the quantum efficiency of a typical sensor (here, we use the Sony IMX172, see supp.).

Then, we center a square virtual grid for each field h at the spot diagram centroid $\overline{y_h} = (1/n_{\text{w}} n_{\text{p}}) \sum_{w,p} y_{h,w,p}$. We set the size of the virtual grid to 260 µm as to collect all rays throughout the full optimization process, and split it into 65 × 65 bins. Instead of naive ray counting, we employ the differentiable alternative of kernel density estimation (KDE) using a Gaussian kernel with a bandwidth half the size of a bin, which effectively spreads the energy of each ray over multiple bins. Incidentally, we reduce the computational burden by duplicating all rays and bins across the y-axis.

Spatially Varying Convolution We employ the spatially varying overlap-add method [14] using 9 × 9 rectangular image patches with corresponding PSFs (see Fig. 3(a)). Each

PSF in the grid is a weighted average of the sampled PSFs—the weight for a field h corresponds to the proportion of the patch that is closest to it—that is rotated to the appropriate angle, then rescaled according to the image resolution. In contrast to naive interpolation, this weighted average scheme involves the full FOV of the lens in the simulation and optimization pipeline. For smooth interpolation, we use a 2D Hann window with 25 % overlap.

Relative Illumination Assuming elliptic pupils, we can obtain a coarse approximation of the relative illumination factor R_h at a given field of interest h from the direction cosines of two meridional rays and one sagittal ray [24]. We apply the operation monochromatically (587.6 nm), then interpolate the values according to the radial coordinate of each pixel. Finally, the aberrated image is pixel-wise multiplied with the relative illumination map (see Fig. 3(b)).

Distortion To efficiently simulate distortion, we approximate the relative distortion shift D_h at each field h by comparing the mean ray height $\overline{y_h}$ at the image plane to the undistorted reference value $y_{h,\mathrm{ref}}$

$$D_h = \frac{\overline{y_h} - y_{h,\mathrm{ref}}}{y_{n_\mathrm{h},\mathrm{ref}}}\,, \tag{2}$$

where the reference values $y_{h,\mathrm{ref}}$ are the result of a monochromatic paraxial ray-tracing operation (587.6 nm). Next, the distorted (x', y') coordinates are computed by linearly interpolating and rotating the distortion shift based on the field position of each pixel. Finally, the image is warped using bicubic interpolation (see Fig. 3(c)).

5. Joint Optimization

Given the differentiable image formation model from Sec. 4, we now seek to freely optimize all lens variables on downstream tasks without compromising manufacturability, which can be facilitated by employing well-defined constraints. We note, however, that there exists no universal set of rules to assess whether a lens design is manufacturable; it notably depends on the expertise and equipment at hand.

Lens Variables We consider a compound lens as a stack of M spherical glass elements with K interfaces (including the aperture stop, but excluding the image plane) where neighboring lens elements are either air spaced or cemented together. Lens variables are denoted $\phi_\mathrm{lens} = (c', s', g)$, where $c' \in \mathbb{R}^{K-2}$ are normalized curvatures of the spherical interfaces, $s' \in \mathbb{R}^K$ normalized glass and air spacings, and $g \in \mathbb{R}^{M \times d_\mathrm{glass}}$ sets of d_glass glass variables representing the dispersion curve of each glass element. The last curvature (before the image plane) is not optimized, but algebraically solved at every training iteration to enforce a unit focal length $f' = 1$. Then, the curvatures c and spacings s are obtained by scaling their normalized counterparts to the desired focal length: $c = c'/f$; $s = s'f$. As in [7], we implement a *paraxial image solve* to help the lens remain mostly in focus throughout optimization, which requires computing the back focal length (BFL) to locate the paraxial image plane with respect to the last optical surface. Then, the last airspace $s_K = s'_K f + \mathrm{BFL}$ is retrieved from the normalized defocus s'_K, which acts as the optimized variable.

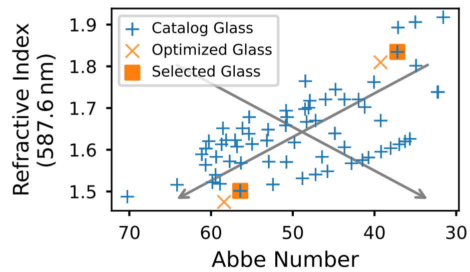

Figure 6. Quantized continuous glass variables. Using continuous glass variables enables the exploration of the solution space, but the actual variables employed to model the lens are always their closest catalog counterparts ($n_\mathrm{cat} = 65$). The arrows indicate the whitened coordinate system.

Glass Variables Our aim in optimizing glass variables is to find the best set of materials among the *catalog glasses* $(g'_1, g'_2, \dots, g'_{n_\mathrm{cat}})$. To this end, we consider $n_\mathrm{cat} = 65$ recommended glasses from the Ohara catalog [21] (see Fig. 6). We model the dispersion curve of each glass material with $d_\mathrm{glass} = 2$ variables: the refractive index at the "d" Fraunhofer line (587.6 nm) and the Abbe number. As in [32], we use the approximate dispersion model $n(\lambda) \approx A + B/\lambda^2$ to retrieve the refractive index at any wavelength λ, where A and B follow from the definition of the "d"-line refractive index and Abbe number. We obtain our normalized glass variables g by fitting a whitening transformation on the refractive indices and Abbe numbers of all catalog glasses.

Quantized Continuous Glass Variables Using continuous relaxations for glass optimization presents several issues in SGD-based end-to-end optimization. Requiring the glass variables to converge to catalog glasses while allowing them to vary significantly during training is challenging as it would require the delicate tuning of scheduled constraints.

To avoid this issue, we introduce *quantized continuous glass variables*: glass variables that only exist in discrete sets, but retain the optimizable property of continuous variables. As illustrated in Fig. 6, in the forward pass, we replace each set of variables g_m with its closest catalog glass counterpart

$$g_m^* = \arg\min_j ||g_m - g'_\mathrm{j}||_2^2\,. \tag{3}$$

As this operation is not differentiable, we approximate its gradient using the "gradient step-through" operator [38]. This operation allows glass variables to undergo meaningful optimization while ensuring that they always match available glass materials. As our approach allows large jumps in lens performance when new catalog glasses are selected from one optimization step to the other, we couple it with a *glass* *variable loss* ℓ_{GV} to help the free variables stick close to the selected glasses, therefore limiting the magnitude and frequency of such jumps. The loss minimizes the squared distance between each set of continuous glass variables g_m and the closest catalog glass

$$\ell_{\text{GV}} = \sum_{m} ||g_m - g_m^*||_2^2 \,. \quad (4)$$

We find empirically that the lenses optimized with this approach retrieve a good performance within a few steps.

Design Losses In practice, we find that the training signal due to the downstream OD loss is often noisy and can make reliable optimization challenging.

To account for this as well as several manufacturing constraints, we add a set of design losses to assist the optimization. First, we complement the noisy downstream detection loss with a *spot size loss* ℓ_{S} for stability. The spot size is equivalent to the RMS size of the PSF (for a given field h) and is computed from the same transversal ray coordinates x and y that compose the spot diagram (see Sec. 4). We formulate ℓ_{S} as the average spot size across all field values

$$\ell_{\text{S}} = \frac{1}{n_{\text{h}}} \sum_{h} \sqrt{\frac{1}{n_{\text{w}} n_{\text{p}}} \sum_{w,p} \left(y_{h,w,p} - \overline{y_h}\right)^2 + x_{h,w,p}^2} \,. \quad (5)$$

Next, we add two additional *ray path* and *ray angle* losses, which are defined by reusing intermediate operands from every ray r involved in the computation of the spot diagrams or spot size. The *ray path loss* ℓ_{RP} avoids overlapping surfaces, enforces sufficient center/edge thicknesses in glass elements, and imposes a sufficient image clearance (the clear space between the last element and the image sensor). It is defined using the horizontal distance $\Delta z \in \mathbb{R}^{K \times n_{\text{r}}}$ traveled by every ray r across every glass or air spacing k. We want all rays to travel a horizontal distance bounded between a lower threshold $\Delta z_{\text{min}}^{(k)}$ and an upper threshold $\Delta z_{\text{max}}^{(k)}$ that depend on the nature of the spacing. In our experiments, these are set to enforce a minimum distance of 0.01 mm in airspaces and 12 mm for image clearance, and a distance between 1–3 mm in glass. The loss is formulated as

$$\begin{aligned}\ell_{\text{RP}} = \frac{1}{n_{\text{r}}} \sum_{k,r} &\max\left(\Delta z_{\text{min}}^{(k)} - \Delta z_{k,r}, 0\right) \\ &+ \max\left(\Delta z_{k,r} - \Delta z_{\text{max}}^{(k)}, 0\right) \,. \end{aligned} \quad (6)$$

The *ray angle loss* ℓ_{RA} limits all angles of incidence θ and refraction θ' to a threshold $\theta_{\text{max}} = 60°$; this aims to avoid ray failures, stabilize the optimization process, and improve tolerancing. Tracing rays through spherical surfaces involves the computation of intermediate values $\zeta = \cos^2(\theta)$ and $\zeta' = \cos^2(\theta')$, where negative values imply ray-tracing failure in the form of missed surfaces for ζ and total internal reflection for ζ'. Similar to ℓ_{RP} (Eq. (6)), this loss is based on intermediate ray-tracing operands occurring at every interface k prior to the image plane

$$\begin{aligned}\ell_{\text{RA}} = \frac{1}{n_{\text{r}}} \sum_{k,r} &\max\left(\cos^2(\theta_{\text{max}}) - \zeta_{k,r}, 0\right) \\ &+ \max\left(\cos^2(\theta_{\text{max}}) - \zeta'_{k,r}, 0\right) \,. \end{aligned} \quad (7)$$

End-to-End Optimization All previous loss terms are combined to define a loss ℓ_{lens} that operates exclusively on the lens design parameters ϕ_{lens}

$$\ell_{\text{lens}} = \ell_{\text{S}} + \lambda_{\text{RP}} \ell_{\text{RP}} + \lambda_{\text{RA}} \ell_{\text{RA}} + \lambda_{\text{GV}} \ell_{\text{GV}} \,, \quad (8)$$

where we set $\lambda_{\text{RP}} = 100$, $\lambda_{\text{RA}} = 100$, and $\lambda_{\text{GV}} = 0.01$. Eq. (8) can be used in isolation to optimize the baseline lenses for spot size, but is also combined with object detection losses ℓ_{OD} to define the joint loss

$$\ell_{\text{joint}} = \ell_{\text{OD}} + \lambda_{\text{lens}} \ell_{\text{lens}} \,, \quad (9)$$

where λ_{lens} is set individually for each lens.

6. Experiments

In this section, we validate the proposed method on a variety of different lens design tasks, compare it to existing design methods, and confirm the effectiveness of components of the method in ablation experiments. To this end, we first introduce the dataset and training approach employed for all experiments, describe baseline lens designs in Sec. 6.1, and then discuss OD design experiments in Sec. 6.2.

Datasets We conduct our experiments with the BDD100K dataset [44] containing 80k (70k/10k for training/evaluation) all-in-focus images with moderate resolution (1280 × 720) and minimal visible aberrations even before downsampling, which makes it suitable for our experiments. We consider six aggregated classes (car/van/suv, bus/truck/tram, bike, person, traffic light, and traffic sign). We also evaluate our trained models on the Udacity autonomous driving dataset [37] which contains 14k higher-resolution images (1920 × 1200) that were annotated using the same six classes.

Sensor Simulation We consider a sensor diagonal d = 16 mm and a quantum efficiency curve that follows the Sony IMX172 sensor for representative wavelength sampling (see supp.). The lens model is applied to the unaltered dataset images which are subsequently resized (1024 × 1024) and passed to the OD model. To observe larger OD performance degradations, we also simulate a 2×-increased resolution in which the aberrations appear proportionally larger; in this setting, the dataset image occupies only the upper-left quadrant of the original virtual scene, and we simulate the aberrations accordingly as shown in Fig. 7.

Detector and Training Methodology We use the RetinaNet [18] object detector with a ResNet-50 backbone [13]

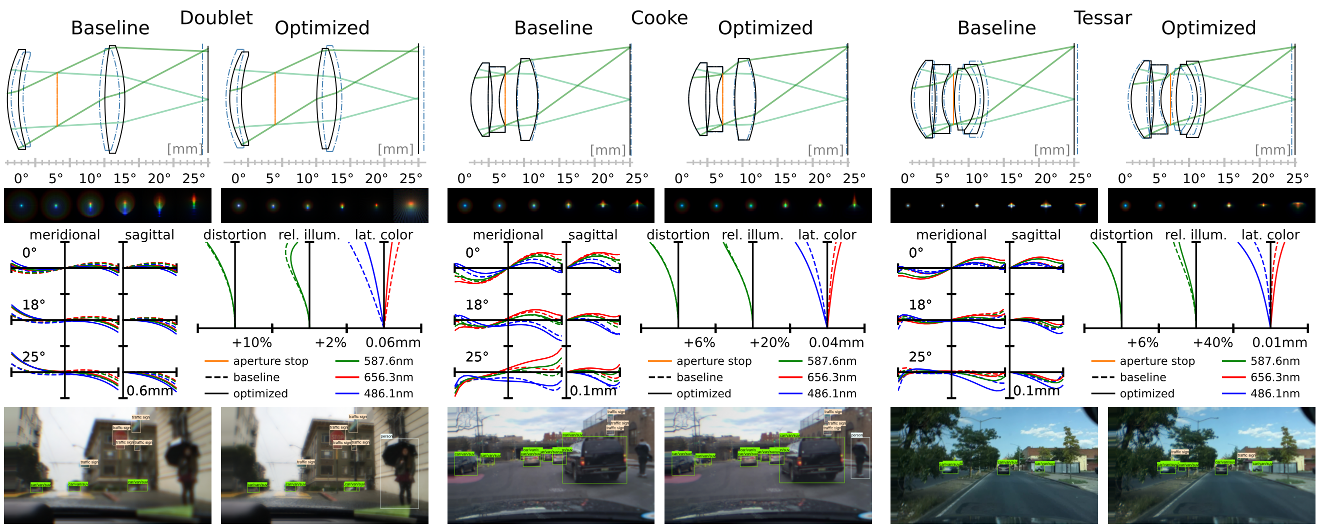

Figure 7. Comparison between lenses optimized for spot size (Baseline) or object detection (Optimized) under 2× simulated resolution, for 2- (Doublet), 3- (Cooke), and 4-element (Tessar) designs. From top to bottom, we show 1) the lens designs (dashed lines represent the baseline/optimized counterpart); 2) PSFs for different fields; 3) aberration charts (left: ray fan plots; right: field curves); and 4) images where the bottom-right corner corresponds to on-axis imaging (0°) and the top-left corner to the maximum field of view (25°). Here, in contrast to the baseline lenses, the optimized lenses detect additional persons (Doublet and Cooke) or traffic signs (Tessar).

for all experiments. We train all OD models with a batch size of 8 and we jointly optimize the lens and OD model with Adam [15]. The learning rates are set to $5{\cdot}10^{-5}$ for ϕ_{OD} and $5{\cdot}10^{-3}$ for ϕ_{lens} over 50k steps, then both are decayed to 0 over 100k subsequent steps following a half cosine cycle.

Lens Distortion in Object Detection The object-matching operation commonly used in IoU losses interferes with distortion since it moves the content associated with predefined anchors. To account for this, we use Eq. (2) to apply a correction step to all ground truth boxes when computing the OD losses, by shifting the midpoint of each bounding box segment, then drawing a new bounding box around the shifted coordinates. However, to evaluate the average precision (AP) of the OD models in an unbiased manner, we apply the correction step to the *predicted* boxes instead.

6.1. Baseline Lenses

We conduct our experiments using typical lenses with 2–4 elements as visualized in Fig. 7. In contrast to the 2-element Doublet, the 3-element Cooke triplet lens has sufficient DOF for moderate aperture and FOV imaging [28]. The 4-element Tessar lens can be seen as a modified Cooke triplet with more DOF [28]. All lenses are optimized for the same first-order specifications, namely f/2 for aperture, $\pm 25°$ for FOV, and focal length $f = 17.2$ mm. We note that even 4-element spherical lenses do not have the required DOF to adequately correct geometrical aberrations under this combination of aperture and FOV [28]. Therefore, our results can be interpreted as approximate lower bounds on OD performance; reducing the aperture or FOV would likely lead to similar or better performance in all cases.

To obtain our baseline lenses, we follow common practice and start from several starting points with various configurations—namely, different aperture stop locations or cemented interfaces—from varied sources [8, 25, 28], re-optimize each of them using Eq. (8), and select the ones that have the best average spot size according to Eq. (5).

6.2. Automotive Object Detection

We report our lens designs optimized for object detection in Tab. 2 in terms of mean average precision (AP)—averaged over the IoU thresholds (0.5, 0.55, . . . , 0.95) and all six object classes. Additionally, we report the averaged PSNR and SSIM image quality metrics to compare the images before and after simulating the blur-inducing aberrations (prior to applying relative illumination and distortion).

To provide approximate upper bounds for OD performance, which is equivalent to training and evaluating the OD models without any aberrations, we first report the AP under "perfect" optics (first row of Tab. 2). We also evaluate this trained model when simulating the effect of each baseline lens; this scenario, labeled 🔒–🔒 in Tab. 2, is akin to attempting OD on strongly aberrated images using an off-the-shelf OD model. This leads to a large decrease in AP (-19.5/-7.4/-2.9 for 2/3/4 elements on 1× res. on BDD).

Then, we fine-tune the OD model to account for the simulated aberrations (🔒–🔓), by modeling the lens using its known design. In practice, this could also be achieved by capturing a dataset using the manufactured lens and training on it. This greatly alleviates the AP drop compared to "perfect" optics (-3.8/-1.1/-0.7 for 2/3/4 elements on 1× res. on BDD) despite the significant degradation in image quality.

Optics	ϕ_{lens}	BDD100K Spot↓	BDD100K PSNR↑	BDD100K SSIM↑	ϕ_{OD}	BDD100K AP↑	Udacity AP↑
Perfect	–	–	–	–	–	34.1	28.2
Doublet (1× res.)	🔒	80.1	26.6	0.82	🔒	14.6	11.5
					🔓	30.3	23.5
	🔓	135.5	27.6	0.85	🔓	32.0(+1.7)	25.6(+2.1)
*Doublet (2× res.)	🔒	80.1	24.3	0.78	🔓	25.0	18.4
	🔓	124.5	25.5	0.81	🔓	28.1(+3.1)	21.9(+3.5)
Cooke (1× res.)	🔒	30.6	29.1	0.88	🔒	26.7	20.6
					🔓	33.0	26.3
	🔓	36.6	28.7	0.87	🔓	33.3(+0.3)	26.8(+0.5)
*Cooke (2× res.)	🔒	30.6	27.2	0.84	🔓	31.5	24.9
	🔓	36.5	26.6	0.83	🔓	31.7(+0.2)	24.8(−0.1)
Tessar (1× res.)	🔒	14.8	29.6	0.90	🔒	31.2	25.7
					🔓	33.4	26.9
	🔓	14.8	29.6	0.90	🔓	33.6(+0.2)	27.0(+0.1)
*Tessar (2× res.)	🔒	14.8	28.8	0.87	🔓	31.3	24.4
	🔓	24.7	27.9	0.85	🔓	32.2(+0.9)	26.2(+1.8)

Table 2. Mean spot size (µm, see Eq. (5)), image quality metrics (PSNR and SSIM), and final OD performance (AP) across varied experimental settings. The lens and OD model parameters ϕ_{lens} and ϕ_{OD} are either optimized 🔓 or fixed 🔒. When ϕ_{OD} is fixed, we use the same parameters as in the perfect optics baseline (first row). Settings with * are visualized in Fig. 7 (see supp. for others).

Setting	Optics	Spot (µm)↓	Vig. rays↓	AP↑
PM [35]	Tessar (1× res.)	176.4	13.2 %	30.8 (PM); 29.0 (RT)
Ours	Tessar (1× res.)	14.8	0.0 %	**33.6** (RT)
PM [35]	Tessar (2× res.)	104.0	7.4 %	23.9 (PM); 18.4 (RT)
Ours	Tessar (2× res.)	24.7	0.0 %	**32.2** (RT)

Table 3. Comparison with the proxy model (PM) of Tseng et al. [35] on the joint optimization of the Tessar lens. We report the AP on BDD100K, where aberrations are modeled using either the PM or exact ray tracing (RT). We also report the mean spot size and proportion of vignetted rays, where 0 % indicates that the design specifications (f-number, FOV, and no vignetting) are fulfilled.

Finally, we jointly optimize the lens alongside the OD model (🔓–🔓) using Eq. (9), which results in an increase in AP over the scenario 🔒–🔓 (e.g., +3.1/+0.2/+0.9 for 2/3/4 elements on 2× res. on BDD), validating the benefit of joint optics/OD optimization. On 1×-res. BDD, the joint optimization allows the Cooke triplet (AP of 33.3, up from 33.0) to nearly reach the performance of the baseline Tessar lens (AP of 33.4) which has one additional lens element. The improvements are not specific to the BDD100K dataset: our experiments validate that they generalize to the higher-resolution Udacity dataset. We note that the optimized lenses improve OD performance despite a mean spot size similar to or worse than the baseline lenses (see supp. for a tolerancing analysis and a comparison to other ray-tracing approaches).

Comparison to Proxy Model In Tab. 3, we perform the Tessar lens experiments using a proxy model (PM) from Tseng et al. [35]. We train the PM on the data of 10k variations of the baseline Tessar lens (see supp.) to output the PSF, relative illumination factor R_h, and distortion shift D_h for a given field value h and set of 22 lens variables. As in [35], we predetermine boundaries for all lens variables, which we optimize only on ℓ_{OD} (i.e., $\lambda_{\text{lens}} = 0$). Compared to our method, the experiments show a significant decrease in AP even when the aberrations are modeled using the PM rather than exact ray tracing during evaluation. Moreover, the lenses optimized using the PM do not fulfill the design specifications as a significant proportion of rays are vignetted, further validating the proposed method.

Setting	AP↑	Comment
Complete methodology	**32.2**	–
Continuous glass variables	25.9	Unrealistic glass ($\ell_{\text{GV}} = 0.91$)
No paraxial image solve	26.5	Last airspace is $s_K = s'_K f$
No ray path loss $\lambda_{\text{RP}} = 0$	15.4	Unfeasible design (ray failures)
No ray angle loss $\lambda_{\text{RA}} = 0$	24.2	Unfeasible design (ray failures)
No spot size loss $\lambda_{\text{S}} = 0$	11.1	Spot size of 142 um (↑9.6×)

Table 4. Ablation study on the joint optimization of the Tessar lens (2× res.), where we report the AP on BDD100K.

Ablations In Tab. 4, we report ablation experiments on the joint design of the Tessar lens for 2× resolution. The experiments validate that each component of the proposed method is required to avoid instability; in this setting, any component removal leads to a drop in OD performance and, in some cases, in manufacturability issues (see supp.). In particular, using continuously relaxed glass variables not only leads to unrealistic glass materials but also adds instability that can result in poorly behaved designs. This ablation experiment validates the role of glass material optimization in lens design for downstream detection tasks.

7. Conclusion

Where previous works in joint optics design attempt to optimize compound lenses over only a subset of possible surface profiles and spacings, here we establish a novel differentiable lens model and optimization method to enable the free optimization of all lens variables; notably, *quantized continuous glass variables* circumvent issues due to the categorical nature of glass materials. On automotive OD, we consistently observe *improvements in detection even when reducing the number of elements* in a given lens stack. Along with the release of code, we hope that this work will enable exciting future research directions such as combining different lens components (e.g., aspherics or diffractive optical elements), modeling scenes with high-resolution multispectral data, or enabling depth-sensitive downstream tasks.

Acknowledgments This research was supported by the Sentinel North program of Université Laval and NSERC. Felix Heide was supported by an NSF CAREER Award (2047359), a Packard Foundation Fellowship, a Sloan Research Fellowship, a Sony Young Faculty Award, a Project X Innovation Award, and an Amazon Science Research Award.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems. *arXiv:1603.04467*, 2016. 1

[2] J. Bentley and C. Olson. Field guide to lens design. Society of Photo-Optical Instrumentation Engineers (SPIE), 2012. 2, 3

[3] J. Chang and G. Wetzstein. Deep Optics for Monocular Depth Estimation and 3D Object Detection. In *Proceedings of IEEE International Conference on Computer Vision*, pages 10192–10201. IEEE, 2019. doi: 10.1109/ICCV.2019.01029. 1, 2, 3

[4] J. Chang, V. Sitzmann, X. Dun, W. Heidrich, and G. Wetzstein. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification. *Scientific reports*, 8(1):1–10, 2018. 3

[5] G. Côté, J.-F. Lalonde, and S. Thibault. Deep learning-enabled framework for automatic lens design starting point generation. *Opt. Express*, 29(3):3841–3854, Feb 2021. doi: 10.1364/OE.401590. 4

[6] G. Côté, J.-F. Lalonde, and S. Thibault. On the use of deep learning for lens design. *Proc. SPIE*, 12078:230 – 236, 2021. doi: 10.1117/12.2603656. 1

[7] G. Côté, Y. Zhang, C. Menke, J.-F. Lalonde, and S. Thibault. Inferring the solution space of microscope objective lenses using deep learning. *Optics Express*, 30(5):6531–6545, Feb. 2022. ISSN 1094-4087. doi: 10.1364/OE.451327. 5

[8] G. Côté. LensNet: lens design starting point generator, 2021. https://lvsn.github.io/lensnet. 7

[9] A. Girard. Excerpt from Revue d'optique théorique et instrumentale. *Rev. Opt*, 37:225–241, 1958. 3

[10] H. Haim, S. Elmalem, R. Giryes, A. M. Bronstein, and E. Marom. Depth Estimation From a Single Image Using Deep Learned Phase Coded Mask. *IEEE Transactions on Computational Imaging*, 4(3):298–310, Sept. 2018. ISSN 2333-9403. doi: 10.1109/TCI.2018.2849326. 3

[11] A. Halé, P. Trouvé-Peloux, and J.-B. Volatier. End-to-end sensor and neural network design using differential ray tracing. *Optics express*, 29(21):34748–34761, 2021. 1, 2, 3

[12] J. Hanika and C. Dachsbacher. Efficient monte carlo rendering with realistic lenses. In *Computer Graphics Forum*, volume 33, pages 323–332. Wiley Online Library, 2014. 2

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[14] M. Hirsch, S. Sra, B. Schölkopf, and S. Harmeling. Efficient filter flow for space-variant multiframe blind deconvolution. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 607–614. IEEE, 2010. 4

[15] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *Proccedings of the 3rd International Conference on Learning Representations*, 2015. 7

[16] C. Kolb, D. Mitchell, and P. Hanrahan. A realistic camera model for computer graphics. In *Proceedings of the 22nd annual conference on computer graphics and interactive techniques*, pages 317–324, 1995. 2, 3

[17] Z. Li, Q. Hou, Z. Wang, F. Tan, J. Liu, and W. Zhang. End-to-end learned single lens design using fast differentiable ray tracing. *Optics Letters*, 46(21):5453–5456, 2021. 1, 2, 3, 4

[18] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 6

[19] P. Y. Maeda, P. B. Catrysse, and B. A. Wandell. Integrating lens design with digital camera simulation. In *Digital Photography*, volume 5678, pages 48–58. SPIE, Feb. 2005. doi: 10.1117/12.588153. 2, 4

[20] C. A. Metzler, H. Ikoma, Y. Peng, and G. Wetzstein. Deep Optics for Single-Shot High-Dynamic-Range Imaging. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1375–1385. IEEE, 2020. 1, 2, 3

[21] Ohara Corporation. Optical Glass Catalog, 2019. 5

[22] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, volume 32, pages 8024–8035, 2019. 1

[23] Y. Peng, Q. Sun, X. Dun, G. Wetzstein, W. Heidrich, and F. Heide. Learned large field-of-view imaging with thin-plate optics. *ACM Trans. Graph.*, 38(6):219–1, 2019. 3

[24] M. P. Rimmer. Relative Illumination Calculations. In R. E. Fischer and P. J. Rogers, editors, *Optical System Design, Analysis, Production for Advanced Technology Systems*, volume 0655, pages 99 – 104. International Society for Optics and Photonics, SPIE, 1986. doi: 10.1117/12.938414. 5

[25] Schott Corporation. Optical Glass Catalog, 2019. 7

[26] R. Schuhmann. Description of aspheric surfaces. *Advanced Optical Technologies*, 8(3-4):267–278, 2019. 3

[27] V. Sitzmann, S. Diamond, Y. Peng, X. Dun, S. Boyd, W. Heidrich, F. Heide, and G. Wetzstein. End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging. *ACM Transactions on Graphics*, 37(4):1–13, Aug. 2018. ISSN 0730-0301, 1557-7368. doi: 10.1145/3197517.3201333. 1, 2, 3

[28] W. J. Smith. *Modern Lens Design*. McGraw Hill Professional, Nov. 2004. ISBN 978-0-07-177726-1. 1, 2, 7

[29] B. Steinert, H. Dammertz, J. Hanika, and H. P. Lensch. General spectral camera lens simulation. In *Computer Graphics Forum*, volume 30, pages 1643–1654. Wiley Online Library, 2011. 2

[30] D. Sturlesi and D. C. O'Shea. Future of global optimization in optical design. In *1990 Intl Lens Design Conf*, volume 1354, pages 54–69. International Society for Optics and Photonics, Jan. 1991. doi: 10.1117/12.47876. 2

[31] Q. Sun, E. Tseng, Q. Fu, W. Heidrich, and F. Heide. Learning Rank-1 Diffractive Optics for Single-Shot High Dynamic Range Imaging. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1396. IEEE, 2020. 1, 2, 3

[32] Q. Sun, C. Wang, Q. Fu, X. Dun, and W. Heidrich. End-to-end complex lens design with differentiate ray tracing. *ACM Trans. Graph.*, 40(4):1–13, jul 2021. ISSN 0730-0301. doi: 10.1145/3450626.3459674. 1, 2, 3, 5

[33] Synopsys. Code V 11.2 Documentation Library. Technical report, Synopsys, 2018. 3

[34] E. Tseng, S. Colburn, J. Whitehead, L. Huang, S.-H. Baek, A. Majumdar, and F. Heide. Neural nano-optics for high-quality thin lens imaging. *Nature communications*, 12(1):1–7, 2021. 3

[35] E. Tseng, A. Mosleh, F. Mannan, K. St-Arnaud, A. Sharma, Y. Peng, A. Braun, D. Nowrouzezahrai, J.-F. Lalonde, and F. Heide. Differentiable compound optics and processing pipeline optimization for end-to-end camera design. *ACM Trans. Graph.*, 40(2):1–19, jun 2021. ISSN 0730-0301. doi: 10.1145/3446791. 1, 2, 3, 4, 8

[36] M. v. Turnhout and F. Bociort. Instabilities and fractal basins of attraction in optical system optimization. *Optics Express*, 17(1):314–328, Jan. 2009. ISSN 1094-4087. doi: 10.1364/OE.17.000314. 2

[37] Udacity. Annotated driving dataset, 2022. URL `https://github.com/udacity/self-driving-car/tree/master/annotations`. 6

[38] A. Van Den Oord, O. Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 5

[39] M. van Turnhout and F. Bociort. Chaotic behavior in an algorithm to escape from poor local minima in lens design. *Optics Express*, 17(8):6436–6450, Apr. 2009. ISSN 1094-4087. doi: 10.1364/OE.17.006436. 2

[40] J.-B. Volatier, Á. Menduiña-Fernández, and M. Erhard. Generalization of differential ray tracing by automatic differentiation of computational graphs. *Journal of the Optical Society of America A*, 34(7):1146, July 2017. ISSN 1084-7529, 1520-8532. doi: 10.1364/JOSAA.34.001146. 1

[41] C. Wang, N. Chen, and W. Heidrich. Lens design optimization by back-propagation. In *International Optical Design Conference 2021*, volume 12078, pages 312–318. SPIE, Nov. 2021. doi: 10.1117/12.2603675. 1

[42] C. G. Wynne. Lens Designing by Electronic Digital Computer: I. *Proceedings of the Physical Society*, 73(5):777–787, May 1959. ISSN 0370-1328. doi: 10.1088/0370-1328/73/5/310. 3

[43] A. Yabe. *Optimization in Lens Design*. SPIE Press, 2018. 2, 3

[44] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020. 6

[45] Zemax. Zemax OpticStudio 19.8 User Manual. Technical report, Zemax, 2019. 3