Image Quality Is Not All You Want: Task-Driven Lens Design

Supplementary Material

1. Implementation details

Due to the length constraint, some implementation details can not be fully described in the main paper. In addition, the design methods slightly differ for different lenses. For clarity, in the following section, we describe the implementation details for

1.1. Details for optical simulation

Optical lens. An optical lens is represented by a sequence of aspherical surfaces in 3D space. Each aspherical surface is defined by the following equation:

$$z(r) = \frac{r^2}{R\left(1+\sqrt{1-(1+\kappa)\frac{r^2}{R^2}}\right)} + \alpha_4 r^4 + \alpha_6 r^6 + \alpha_8 r^8 + \alpha_{10} r^{10}, \quad (1)$$

where R represents the radius of curvature, κ is the conic term, and α_4 to α_{10} are the even polynomial terms. The function $z(r)$ gives the displacement of the surface from the vertex at a distance $r = \sqrt{x^2+y^2}$ from the axis. An additional parameter d denotes the z-coordinate (position) of each lens element. In our application, we use the curvature $c = \frac{1}{R}$ for optimization, and we set κ to 0.

Point spread function. Optical rays originate from a spatial point and propagate through all surfaces, eventually intersecting the sensor plane. The intersection points are assigned to different sensor pixels, forming the point spread function (PSF). The PSF describes the response of the lens to a point light source at a specific position. In an ideal imaging system, the PSF is a Dirac peak, disregarding the diffraction effect, resulting in sharp and clear camera captures. However, in actual lenses, the PSF typically exhibits different shapes from the ideal Dirac peak due to the presence of optical aberrations [6, 12]. Since the lenses we aim to design have a large aperture size and optical aberrations dominate the optical properties, we neglect the diffraction effect of the lens.

Differentiable ray tracing. The differentiable ray tracing module in the lens design pipeline is built on the public **dO** engine [13], utilizing the PyTorch framework. During the forward ray tracing process, a computation graph is constructed. In the back-propagation process, the gradient of each lens parameter is computed according to PyTorch's automatic differentiation capability.

Camera sensor. We utilize a camera sensor with a diagonal distance of 4 mm and a standard 1080p resolution format corresponding to a sensor size of 1.8 μm. Due to the small size of the camera sensor, it has a short hyper-focal distance, and the optical properties do not change significantly when the object depth exceeds the hyper-focal distance. Therefore, we only consider infinitely far imaging and disregard the depth-of-field effect.

1.2. Details for dataset simulation

Image simulation. We utilize the ImageNet dataset [2] for both training and evaluation purposes. The training images have low resolution, whereas commercial image sensors typically have much higher resolution, often in the megapixel range. Additionally, in practical scenarios, it is uncommon for an object to occupy the entire full-resolution image. Taking these factors into account, we treat the training images as patches that appear at different locations within the full-resolution sensor image. As depicted in Fig. 1, we select 9 point light sources from the object space to represent the overall optical characteristics of the lens. In our experiments, each input image appears in all 9 fields of view on the sensor plane. The point light sources are positioned at a depth of 20 m to approximate infinitely far imaging. In the image simulation process, we first perform ray tracing to calculate the PSF at different fields of view. Subsequently, we convolve the input image with the PSF to simulate the camera-captured images.

Dataset size. The ImageNet dataset consists of 1,281,167 training images and 50,000 validation images. We employ the validation set to evaluate the classification accuracy of each lens. Since we consider each image occurring at 9 different fields of view, the dataset size is augmented by a factor of 9. Consequently, the training dataset length increases to 11,530,503, and we calculate the classification accuracy based on 450,000 images. During training, we use a batch size of 64. After the image simulation, a total of 576 images are forwarded to the downstream network for image classification.

Data augmentation. Data augmentation [10] is applied to the training images. Although we are designing lenses instead of training the network, data augmentation is important for two primary reasons. First, since we only consider a quarter of the object space, data augmentation techniques such as image flipping and rotation help simulate images occurring in the remaining parts of the object space. Second, the original training images are typically sharp and clear, which may result in the designed TaskLens converging to a local minimum where it can produce clear images. To address this, data augmentation methods, including sharpness adjustments and blurring, assist TaskLens in exploring other image variations beyond the dataset.

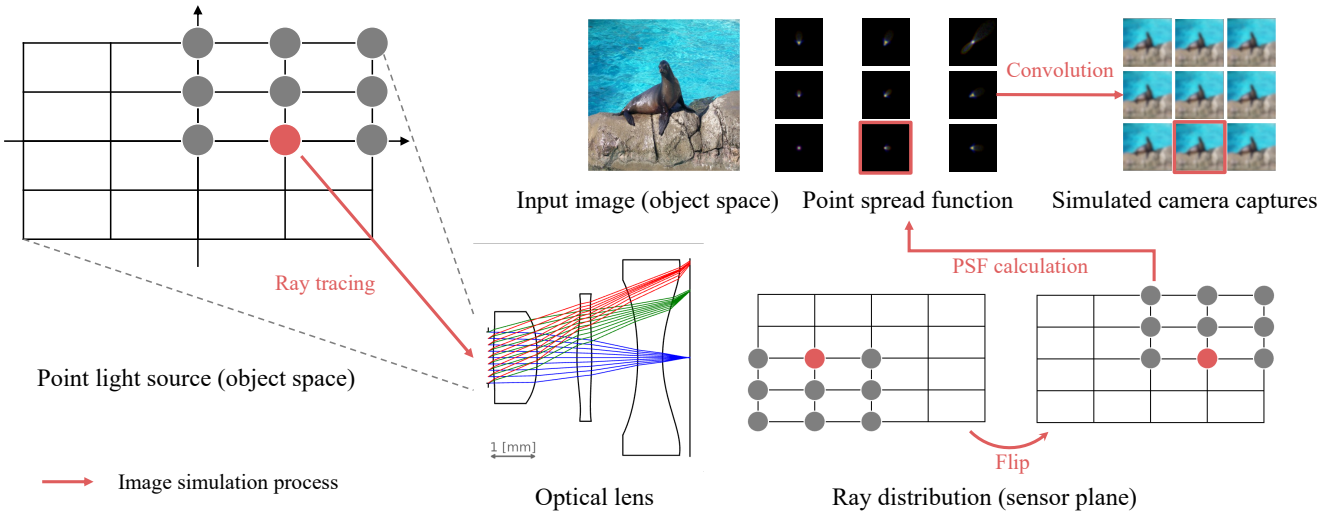

Figure 1. In the image simulation process, we first perform ray tracing to calculate PSF at different fields of view. Then we apply convolution with the input image to simulate the camera-captured image in each field of view. Since the training dataset usually has a low resolution while the commercial camera sensor commonly exhibits a much larger resolution, we treat the training image as patches that appear at different locations in the full-resolution sensor image.

Table 1. Learning rates for different lens parameters.

Curvature c	Position d	Polynomial α_4	Polynomial α_6	Polynomial α_8	Polynomial α_{10}
$1e^{-4}$	$1e^{-4}$	$1e^{-4}$	$2e^{-6}$	$4e^{-8}$	$8e^{-10}$

1.3. TaskLens design

For Task-Driven image classification lens design, we employ a well-trained ResNet50 network [4] for supervision. The network implementation utilizes the "timm" library. The lens is designed from scratch using image classification loss. The initial learning rates for each lens parameter are presented in Table 1. As we are designing lenses from scratch, we apply a regularization term to penalize the obliquity term of optical rays, following the approach in [15]. The total loss function is formulated as:

$$\mathcal{L} = \mathcal{L}_{classi}(f(g_\theta(x) - x) - 0.1\mathcal{L}_{reg}, \qquad (2)$$

where x represents the input image, g denotes the image simulation process, and f is the classification network. The first term, $\mathcal{L}_{classi}$, refers to the image classification loss, while the second term, $\mathcal{L}_{reg}$, represents the regularization term that encourages a smooth light path and prevents degenerate lens structures.

1.4. ImagingLens design

Three ImagingLenses are designed based on each TaskLens with the objective of achieving the best imaging quality. ImagingLens #1 is designed by minimizing the l_2 error of the spot diagram. We select 11×11 fields of view, and for each field of view, we trace 1024 rays to compute the spot diagram. We adopt the curriculum learning strategy described in [15] for lens optimization. Specifically, we dynamically adjust the weight of different fields of view in the loss function to overcome local minima. The optimization process employs the same optimizer and scheduler settings as the TaskLens design and runs for 10,000 iterations until convergence.

The other ImagingLenses are tweaked further by importing the three TaskLenses respectively into Zemax. We used the default merit function: RMS spot x+y referring to centroid ray, pupil integration over 16 rings and 12 arms. Individual air and glass thickness boundary constraints are set to avoid negative and impractical thicknesses. While the entrance pupil diameter is fixed according to each TaskLens, the working F-number is used to control the system's first-order feature. After the merit function is set, all the surface radii, aspheric coefficients, and thicknesses are made as variables. The optimization process typically lasts for 50 cycles until convergence.

All designed ImagingLenses are presented in Fig. 2.

1.5. End-to-End design on ImagingLens

Conventional End-to-End lens design typically begins with a well-designed imaging lens. In our paper, we also evaluate this approach by performing End-to-End training on the best ImagingLens with the corresponding fine-tuned classification network. Specifically, for the doublet lens, we utilize ImagingLens #2, and for the triplet and quadruplet lenses, we use ImagingLens #1. The lens optimiza-

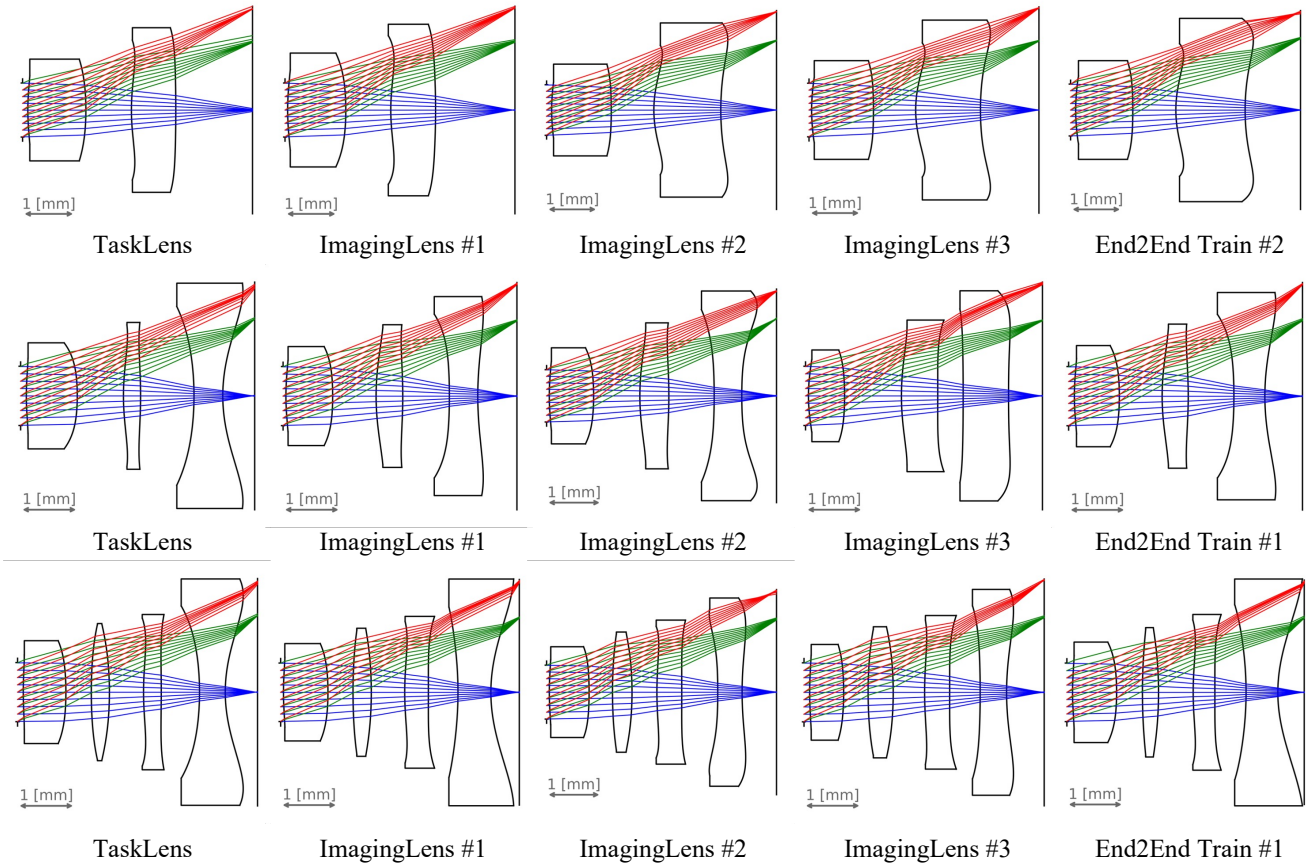

Figure 2. Lens structures and ray paths.

tion settings remain the same as those used in the previous TaskLens design, and we set the initial learning rate for the network to $1e^{-5}$. The End-to-End training process is executed for 1 epoch on the ImageNet training set which usually reaches the convergence.

As illustrated in Fig. 2, the lenses trained via the End-to-End approach exhibit similarities to their corresponding initial ImagingLenses. This observation demonstrates that End-to-End training fails to escape from local minima. Furthermore, the quantitative classification accuracy reported in the main paper indicates that the image classification lenses designed through End-to-End training do not perform as well as the TaskLens.

1.6. Image classification network fune-tuning

We conduct fine-tuning on the classification network for each designed lens to maximize its classification capability. We utilize the AdamW optimizer [9] and the CosineAnnealing scheduler [8] with a warm-up scheme for 6,000 steps. Specifically, for ResNet50 [4] and MobileNetV3/L [5], we set the initial learning rate to $1e^{-5}$, while for Swin-B [7] and ViT-L/16 [3], we use an initial learning rate of $1e^{-6}$. ViT-L/16 training is performed on 4 A100-80G GPUs, while the other models utilize 2 A100-80G GPUs.

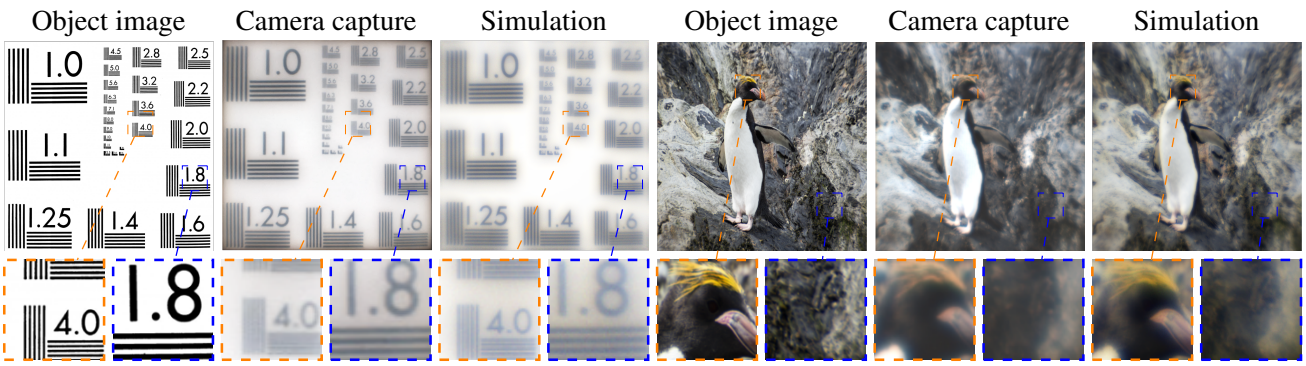

Figure 3. Image simulation evaluation using off-the-shelf optical lens. Our simulation closely resembles the camera captures.

2. Optical simulation evaluation

In our experiments, we did not manufacture the designed lenses for real-world evaluations due to budget constraints. However, the ray tracing accuracy of the simulator we used is comparable to the commercial software Zemax, which is often considered the industrial standard. Additionally, we employed an off-the-shelf optical lens to evaluate the simulation results. As shown in Fig. 4, we utilized the Thorlabs LB1471-A lens and the FLIR GS3-U3-41S8C-C camera for evaluation. A high-resolution monitor is used to display the object image, allowing us to compare the camera-captured image with the simulated image. As demonstrated in Fig. 3, the simulated images of the resolution chart and an RGB image closely resemble the camera captures, despite some contrast differences caused by the characteristics of the camera sensor. These experiments support the accuracy of our simulation, leading us to believe that the lenses designed through our task-driven approach will yield promising results in real-world scenarios.

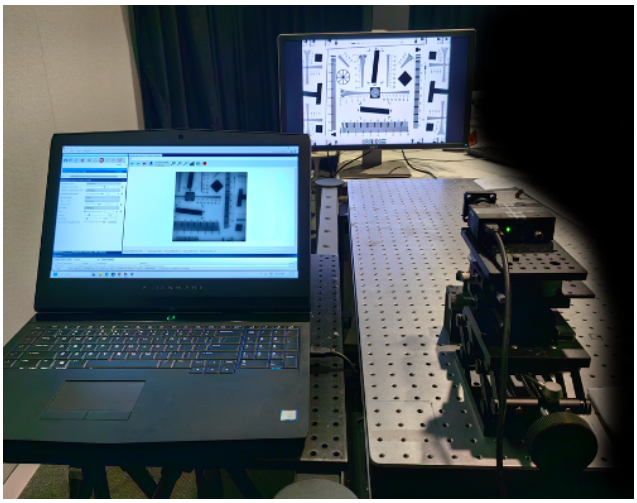

Figure 4. Experimental setup for image simulation evaluation.

Table 2. Image classification accuracy of different lenses with random manufacturing and assembling errors. From left to right: doublet TaskLens, doublet ImagingLens (#2), triplet TaskLens, triplet ImagingLens (#1), four-element TaskLens, four-element ImagingLens (#1).

Lens	TaskLens	ImagingLens (#2)	TaskLens	ImagingLens (#1)	TaskLens	ImagingLens (#1)
Designed	70.08%	68.54%	73.40%	70.04%	73.61%	72.27%
Lens #1	69.73%	67.43%	72.82%	66.24%	72.58%	71.07%
Lens #2	69.65%	67.27%	72.91%	66.36%	72.66%	70.99%
Lens #3	69.83%	67.23%	72.78%	66.21%	72.65%	70.98%
Avg (#1 $\sim$ #3)	**69.74%**	67.31%	**72.84%**	66.27%	**72.63%**	71.10%
Avg decrease	**- 0.34%**	-1.23%	**-0.56%**	-3.77%	**-0.98%**	-1.17%

Designed: designed lens without manufacturing and assembling errors.
Lens #1 $\sim$ #3: designed lens with random manufacturing and assembling errors.

3. Additional results

3.1. Tolerance analysis

Lens manufacturing and assembly errors can decrease imaging performance, subsequently affecting the efficacy of downstream high-level applications. Consequently, the tolerance of lenses is also a crucial factor in lens design. To evaluate the performance of our designed lenses under various manufacturing and assembly errors, we introduced random errors to create three "perturbed" lenses for each design. As presented in Tab. 2, the performance of each perturbed lens was evaluated using a network model trained on unperturbed designs.

By calculating the average classification accuracy, we discovered that our TaskLens consistently outperforms the classical imaging lens under different manufacturing and assembly errors. Notably, our TaskLens demonstrates greater tolerance, as indicated by a smaller performance decrease compared to the corresponding ImagingLens. Specifically, the triplet ImagingLens shows the most significant performance decrease, up to -3.77%, while the corresponding triplet TaskLens exhibits a more modest decrease of -0.56%. These results suggest that our TaskLens is more resilient to manufacturing and assembly errors than classical designs. We attribute this tolerance to the task-driven lens design approach, where perfect imaging performance is not a primary goal, and the network architecture can compensate for a certain degree of optical aberrations, thus enhancing the overall tolerance of the designed lenses.

3.2. Multi-task-driven lens design

In this paper, we focus on the image classification task, as it is the most straightforward application to demonstrate the image feature extraction capability of a neural network. This capability is fundamental to many high-level computer vision applications, such as object detection and instance segmentation. Consequently, we believe that image classification is the most direct application to showcase the desired optical characteristics for high-level computer vision tasks.

However, in real-world scenarios, cameras are typically used for multiple downstream applications, making it crucial to utilize optical lenses that excel in various computer vision tasks. In Tables. 3 and 4, we assess the performance of object detection and instance segmentation on simulated images captured through each lens. Quantitative results show that our TaskLens outperforms the corresponding ImagingLens in these applications, despite not being specifically designed for them. Qualitative results are shown in Fig. 5 and 6. This finding suggests that our TaskLens can be applied to multiple applications and is practical in real-world settings. Moreover, it supports the idea that the "long-tail" PSF inherent in TaskLens may offer broader advantages across a range of high-level computer vision applications.

It is important to note that our approach is "task-driven" rather than "task-specific". This distinction implies that a lens optimized for image classification does not necessarily lead to diminished performance in other applications. Furthermore, while other research works have explored "task-specific" designs for End-to-End computational lenses [1, 11, 14], the question of whether there is a generally optimal lens form remains open for exploration.

Figure 5. Qualitative object detection results. Left: TaskLens (triplet). Right: ImagingLens (triplet #1). We use the official pretrained "faster_rcnn_R_50_FPN_3x" model without fine-tuning for each lens. Left example: more people are detected in the TaskLens simulated image. Right example: the traffic light is successfully detected only in the TaskLens simulated image (marked by the red box). Moreover, detection boxes in the TaskLens render images usually have higher confidence.

Figure 6. Panoptic segmentation results. Left: TaskLens (triplet). Right: ImagingLens (triplet #1). We use the official pretrained "panoptic_fpn_R_101_3x" model without fine-tuning for each lens. Segmentation results on TaskLens rendered images contain fewer artifacts. The corresponding detection results have higher confidence.

Table 3. Object detection results on COCO 2017 validation set, using the official pretrained "faster_rcnn_R_50_FPN_3x" model. The first row represents the detection results on the original images. We then render all images from the COCO 2017 validation set with different lenses (from the second row to the bottom: doublet, triplet, and four-element lens) and evaluate detection performance on rendered images.

Lens	mAP (%)	AP50 (%)	AP75 (%)	APS (%)	APM (%)	APL (%)
Original images	40.22	61.00	43.83	24.16	43.51	51.99
TaskLens	**34.12**	**54.18**	**36.37**	**17.36**	**37.97**	**49.41**
ImagingLens (#2)	32.44	52.47	34.76	17.21	36.24	48.17
TaskLens	**35.44**	**56.05**	**37.75**	**17.39**	**38.64**	**49.71**
ImagingLens (#1)	34.08	53.64	36.58	16.66	36.87	48.81
TaskLens	**36.85**	**56.37**	**38.37**	**17.94**	**39.70**	**52.00**
ImagingLens (#1)	35.23	55.03	38.23	17.24	38.23	50.12

Table 4. Instance segmentation results on COCO 2017 validation set, using the official pretrained "mask_rcnn_R_50_FPN_3x" model. The first row represents the segmentation results on the original images. We then render all images from the COCO 2017 validation set with different lenses (from the second row to the bottom: doublet, triplet, and four-element lens) and evaluate segmentation performance on rendered images.

Lens	mAP (%)	AP50 (%)	AP75 (%)	APS (%)	APM (%)	APL (%)
Original images	37.16	58.60	39.88	18.63	39.48	53.30
TaskLens	**30.55**	**52.15**	**32.28**	**12.05**	**32.70**	**50.19**
ImagingLens (#2)	29.08	50.81	31.85	11.73	32.35	47.73
TaskLens	**31.96**	**52.91**	**33.91**	**12.67**	**33.95**	**50.71**
ImagingLens (#1)	31.27	51.33	33.10	12.42	32.73	49.74
TaskLens	**32.99**	**53.24**	**35.12**	**12.89**	**34.74**	**52.63**
ImagingLens (#1)	31.73	52.13	34.51	12.68	33.12	51.99

4. Additional data

4.1. Lens data

In this section, we provide the detailed lens data in Table 5 to Table 19. The data format is similar to the Zemax lens data format, which we think is easy to understand. The "Thickness" term represents the distance to the next surface. "End2EndLens" represents the End-to-End training results from the best ImagingLens.

4.2. Specifications of actual lens designs

Achieving uniformity in the FoV and effective imaging height across various lens designs can be challenging. To ensure that TaskLens does not exhibit superior visual task performance due to reduced design difficulties, we analyze optical specifications for each lens in Table 20 and compare the design difficulties. The lens design difficulty is correlated with FoV and sensor diagonal distance, and we compute the variations in difficulty relative to the original target parameters (FoV 68.8°, sensor diagonal distance 4 mm) using a linear approximation. Shown in Table 20, our analysis reveals that TaskLens possesses the highest design difficulty, leading us to conclude that the comparison is fair.

References

[1] Julie Chang and Gordon Wetzstein. Deep optics for monocular depth estimation and 3D object detection. In *Int. Conf. Comput. Vis.*, pages 10193–10202, 2019. 5

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255. Ieee, 2009. 1

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016. 2, 3

[5] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for MobileNetv3. In *Int. Conf. Comput. Vis.*, pages 1314–1324, 2019. 3

[6] Rudolf Kingslake and R Barry Johnson. *Lens design fundamentals*. academic press, 2009. 1

[7] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12009–12019, 2022. 3

[8] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 3

[9] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 3

[10] Samuel G Müller and Frank Hutter. TrivialAugment: Tuning-free yet state-of-the-art data augmentation. In *Int. Conf. Comput. Vis.*, pages 774–782, 2021. 1

[11] Vincent Sitzmann, Steven Diamond, Yifan Peng, Xiong Dun, Stephen Boyd, Wolfgang Heidrich, Felix Heide, and Gordon Wetzstein. End-to-end optimization of optics and image

Table 5. Lens data of doublet TaskLens.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04					
2 (Asphere)	1.108	0.2279	1.32	OKP4	-6.230556e-02	-2.692730e-03	2.011200e-04	4.264318e-06
3 (Asphere)	0.869	-0.1988	1.96	AIR	-3.943852e-02	6.418004e-03	6.530462e-04	1.290069e-05
4 (Asphere)	0.847	0.2406	2.69	PMMA	-5.051126e-02	-6.483181e-03	-2.742670e-04	-6.921774e-06
5 (Asphere)	1.463	-0.0379	3.17	AIR	3.602685e-03	-4.919978e-03	-4.744795e-05	4.207488e-06
6 (Sensor)			4.00					

Table 6. Lens data of doublet ImagingLens #1.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04					
2 (Asphere)	1.110	0.2093	1.55	OKP4	-3.557858e-02	-2.151537e-03	2.138008e-04	4.856510e-06
3 (Asphere)	0.858	-0.1934	2.19	AIR	-5.836600e-02	5.725940e-03	5.713565e-04	7.521212e-06
4 (Asphere)	0.855	0.2525	2.86	PMMA	-5.963345e-02	-6.859528e-03	-2.956688e-04	-8.457044e-06
5 (Asphere)	1.514	-0.0675	3.30	AIR	6.357471e-03	-4.605474e-03	-2.022348e-05	6.599236e-06
6 (Sensor)			4.00					

processing for achromatic extended depth of field and super-resolution imaging. *ACM Trans. Graph.*, 37(4):1–13, 2018. 5

[12] Warren J Smith. *Modern optical engineering: the design of optical systems*. McGraw-Hill Education, 2008. 1

[13] Congli Wang, Ni Chen, and Wolfgang Heidrich. dO: A differentiable engine for deep lens design of computational imaging systems. *IEEE Transactions on Computational Imaging*, 8:905–916, 2022. 1

[14] Gordon Wetzstein, Aydogan Ozcan, Sylvain Gigan, Shanhui Fan, Dirk Englund, Marin Soljačić, Cornelia Denz, David AB Miller, and Demetri Psaltis. Inference in artificial intelligence with deep optics and photonics. *Nature*, 588 (7836):39–47, 2020. 5

[15] Xinge Yang, Qiang Fu, and Wolfgang Heidrich. Curriculum learning for ab initio deep learned refractive optics. *arXiv preprint arXiv:2302.01089*, 2023. 2

Table 7. Lens data of doublet ImagingLens #2.

Surface	Thickness [mm]	Curvature $[mm]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}	α_{12}
1 (Aper)	0.100		1.04						
2 (Asphere)	0.828	0.2632	1.10	OKP4	-2.941442e-02	-1.739322e-02	-5.028618e-02	1.460911e-02	7.344432e-01
3 (Asphere)	1.160	-0.1006	1.80	AIR	-7.069314e-02	4.010654e-03	-6.927734e-03	3.027622e-04	9.092269e-03
4 (Asphere)	1.303	0.4436	2.54	PMMA	-5.967423e-02	-8.591303e-03	-4.543824e-03	-9.672559e-04	-1.942622e-03
5 (Asphere)	1.082	0.1736	3.42	AIR	1.812585e-02	-1.161345e-02	-1.200210e-03	-4.974294e-05	-5.326503e-06
6 (Sensor)			3.90						

Table 8. Lens data of doublet ImagingLens #3.

Surface	Thickness [mm]	Curvature $[mm]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}	α_{12}
1 (Aper)	0.100		1.04	AIR					
2 (Asphere)	1.190	0.2718	1.27	OKP4	-3.195822e-02	-2.216966e-02	-5.040844e-02	1.460761e-02	7.344432e-01
3 (Asphere)	0.812	-0.0909	1.92	AIR	-6.991725e-02	3.324919e-03	-6.962574e-03	3.018416e-04	9.092269e-03
4 (Asphere)	1.247	0.4497	2.60	PMMA	-5.536971e-02	-7.750550e-03	-4.525531e-03	-9.670507e-04	-1.942622e-03
5 (Asphere)	1.124	0.1734	3.49	AIR	2.866083e-02	-1.051536e-02	-1.155842e-03	-4.851578e-05	-5.297208e-06
6 (Sensor)			4.00	AIR					

Table 9. Lens data of doublet End2EndLens.

Surface	Thickness [mm]	Curvature $[mm]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}	α_{12}
1 (Aper)	0.100		1.04	AIR					
2 (Asphere)	1.111	0.2665	1.33	OKP4	-3.540349e-02	-1.654536e-02	-5.024989e-02	1.460925e-02	7.344432e-01
3 (Asphere)	0.837	-0.0989	1.93	AIR	-7.081930e-02	3.693011e-03	-6.945701e-03	3.021957e-04	9.092269e-03
4 (Asphere)	1.348	0.4429	2.60	PMMA	-7.077941e-02	-8.365979e-03	-4.539080e-03	-9.672103e-04	-1.942622e-03
5 (Asphere)	1.076	0.2188	3.60	AIR	1.354988e-02	-1.232918e-02	-1.213283e-03	-4.982987e-05	-5.323246e-06
6 (Sensor)			3.90	AIR					

Table 10. Lens data of triplet TaskLens.

Surface	Thickness [mm]	Curvature $[mm]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100			AIR				
2 (Asphere)	0.887	0.2405	1.32	PMMA	-1.201920e-01	-7.420412e-03	4.264761e-05	3.585087e-07
3 (Asphere)	0.804	-0.2585	1.86	AIR	-1.485584e-01	-3.720956e-03	4.013637e-04	8.193966e-06
4 (Asphere)	0.299	0.2408	2.49	OKP4	-4.984390e-02	-2.352248e-03	-1.390607e-04	-4.099868e-06
5 (Asphere)	0.930	-0.1243	2.58	AIR	3.638798e-02	-1.723573e-03	-1.122244e-04	3.827879e-07
6 (Asphere)	0.501	-0.1916	3.14	PMMA	-9.229322e-03	-1.210485e-05	-1.093474e-04	-3.512698e-06
7 (Asphere)	0.554	0.3484	3.96	AIR	-1.864598e-02	-2.933642e-03	-1.108433e-04	6.989184e-07
8 (Sensor)			4.00	AIR				

Table 11. Lens data of triplet ImagingLens #1.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.814	0.2476	1.25	PMMA	-7.998895e-02	-4.237434e-03	1.447068e-04	2.772055e-06
3 (Asphere)	0.756	-0.2059	1.74	AIR	-1.213981e-01	-9.009774e-04	4.764367e-04	9.609429e-06
4 (Asphere)	0.465	0.2307	2.41	OKP4	-2.160468e-02	-2.337896e-03	-1.632334e-04	-4.971110e-06
5 (Asphere)	0.811	-0.1190	2.51	AIR	3.271303e-02	-9.073227e-04	-7.741294e-05	1.429375e-06
6 (Asphere)	0.495	-0.0881	2.88	PMMA	-3.529020e-02	-1.742304e-03	-1.535450e-04	-4.782808e-06
7 (Asphere)	0.683	0.1661	3.50	AIR	-1.497258e-02	-1.607410e-03	-6.544410e-05	2.114299e-06
8 (Sensor)			4.00	AIR				

Table 12. Lens data of triplet ImagingLens #2.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.804	0.2249	1.27	PMMA	-5.094405e-02	-6.908586e-03	1.420362e-04	1.420362e-04
3 (Asphere)	0.839	-0.2117	1.76	AIR	-1.059854e-01	-4.538634e-03	5.032599e-04	5.032599e-04
4 (Asphere)	0.527	0.2259	2.52	OKP4	-2.178941e-02	-4.585173e-03	-1.803447e-04	-1.803447e-04
5 (Asphere)	0.822	-0.1137	2.67	AIR	2.632563e-02	1.512580e-03	-6.432167e-05	-6.432167e-05
6 (Asphere)	0.501	-0.0733	3.16	PMMA	-3.297533e-02	5.560323e-03	-1.589946e-04	-1.589946e-04
7 (Asphere)	0.629	0.1155	3.84	AIR	6.755690e-02	-1.823045e-02	-6.960762e-05	-6.960762e-05
8 (Sensor)			3.91	AIR				

Table 13. Lens data of triplet ImagingLens #3.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.629	0.3452	1.28	PMMA	-7.871836e-02	-5.350309e-03	2.320887e-04	1.445756e-04
3 (Asphere)	0.990	-0.1742	1.61	AIR	-9.910008e-02	-1.127885e-02	3.269837e-04	4.994923e-04
4 (Asphere)	0.633	0.2240	2.42	OKP4	-2.066278e-02	-7.454170e-03	-2.438883e-04	-1.814870e-04
5 (Asphere)	0.442	-0.0203	2.66	AIR	3.662718e-02	1.895413e-03	2.899968e-05	-5.955982e-06
6 (Asphere)	0.790	-0.0176	3.05	PMMA	-3.742982e-03	2.071768e-04	-2.769423e-04	-1.612489e-04
7 (Asphere)	0.591	0.0352	3.69	AIR	1.604678e-02	-1.050925e-02	9.399757e-05	-6.632384e-05
8 (Sensor)			4.00	AIR				

Table 14. Lens data of triplet End2EndLens.

Surface	Thickness [mm]	Curvature $[mm]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.783	0.2446	1.32	PMMA	-7.827187e-02	-3.493498e-03	1.676159e-04	3.248085e-06
3 (Asphere)	0.745	-0.2121	1.78	AIR	-1.202044e-01	-1.351351e-03	4.594656e-04	9.180380e-06
4 (Asphere)	0.455	0.2390	2.42	OKP4	-2.908486e-02	-2.039731e-03	-1.408897e-04	-4.239835e-06
5 (Asphere)	0.795	-0.1285	2.53	AIR	3.206206e-02	-1.436517e-03	-1.019131e-04	6.927270e-07
6 (Asphere)	0.539	-0.1120	2.88	PMMA	-3.670126e-02	-1.170376e-03	-1.327894e-04	-4.211608e-06
7 (Asphere)	0.656	0.2392	3.65	AIR	-1.337205e-02	-2.322827e-03	-8.910213e-05	1.504667e-06
8 (Sensor)			4.00	AIR				

Table 15. Lens data of quadruplet TaskLens.

Surface	Thickness [mm]	Curvature $[mm]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.742	0.1673	1.31	PMMA	-9.882198e-02	-5.059574e-03	1.236521e-04	2.401968e-06
3 (Asphere)	0.443	-0.1912	1.81	AIR	-8.171525e-02	-2.236586e-03	4.394940e-04	9.022501e-06
4 (Asphere)	0.315	0.1119	2.37	OKP4	1.488245e-02	-1.473840e-03	2.082270e-04	4.016732e-06
5 (Asphere)	0.595	-0.1655	2.41	AIR	-9.517191e-03	1.925957e-03	5.143318e-04	1.002425e-05
6 (Asphere)	0.308	0.1844	2.60	OKP4	-5.583153e-02	-3.878689e-03	-1.800252e-04	-4.731216e-06
7 (Asphere)	0.666	-0.0452	2.73	AIR	2.723789e-02	-6.684430e-04	-8.248209e-05	8.828483e-07
8 (Asphere)	0.448	-0.1958	3.03	PMMA	-1.434444e-02	-7.493469e-04	-1.249907e-04	-3.923112e-06
9 (Asphere)	0.564	0.3755	3.98	AIR	-2.438926e-02	-3.631569e-03	-1.352864e-04	2.475713e-07
10 (Sensor)			4.00	AIR				

Table 16. Lens data of quadruplet ImagingLens #1.

Surface	Thickness [mm]	Curvature $[mm]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.742	0.1673	1.31	PMMA	-9.882198e-02	-5.059574e-03	1.236521e-04	2.401968e-06
3 (Asphere)	0.443	-0.1912	1.81	AIR	-8.171525e-02	-2.236586e-03	4.394940e-04	9.022501e-06
4 (Asphere)	0.315	0.1119	2.37	OKP4	1.488245e-02	-1.473840e-03	2.082270e-04	4.016732e-06
5 (Asphere)	0.595	-0.1655	2.41	AIR	-9.517191e-03	1.925957e-03	5.143318e-04	1.002425e-05
6 (Asphere)	0.308	0.1844	2.60	OKP4	-5.583153e-02	-3.878689e-03	-1.800252e-04	-4.731216e-06
7 (Asphere)	0.666	-0.0452	2.73	AIR	2.723789e-02	-6.684430e-04	-8.248209e-05	8.828483e-07
8 (Asphere)	0.448	-0.1958	3.03	PMMA	-1.434444e-02	-7.493469e-04	-1.249907e-04	-3.923112e-06
9 (Asphere)	0.564	0.3755	3.98	AIR	-2.438926e-02	-3.631569e-03	-1.352864e-04	2.475713e-07
10 (Sensor)			4.00	AIR				

Table 17. Lens data of quadruplet ImagingLens #2.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.778	0.1983	1.24	PMMA	-8.836898e-02	-4.262349e-03	1.490898e-04	3.101509e-06
3 (Asphere)	0.439	-0.2125	1.73	AIR	-9.584536e-02	-3.297418e-03	3.935776e-04	7.663079e-06
4 (Asphere)	0.320	0.0947	2.21	OKP4	7.747480e-03	-1.332338e-03	2.316622e-04	4.985073e-06
5 (Asphere)	0.585	-0.1425	2.27	AIR	-6.778305e-03	1.922866e-03	5.005212e-04	9.308818e-06
6 (Asphere)	0.414	0.1844	2.51	OKP4	-4.904995e-02	-4.319484e-03	-1.946336e-04	-4.974748e-06
7 (Asphere)	0.518	-0.0156	2.69	AIR	3.748472e-02	-1.111389e-05	-6.598805e-05	1.124154e-06
8 (Asphere)	0.575	-0.1522	3.02	PMMA	7.492220e-03	-1.415259e-03	-1.507766e-04	-4.519364e-06
9 (Asphere)	0.500	0.3507	4.00	AIR	-2.200943e-02	-8.533297e-04	-5.919880e-05	1.949792e-06
10 (Sensor)			3.80	AIR				

Table 18. Lens data of quadruplet ImagingLens #3.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.592	0.1764	1.31	PMMA	-6.478445e-02	-3.352266e-03	1.649022e-04	3.217212e-06
3 (Asphere)	0.426	-0.1740	1.69	AIR	-6.794760e-02	-2.302133e-04	4.845259e-04	9.795038e-06
4 (Asphere)	0.437	0.1230	2.22	OKP4	1.165871e-02	-2.376378e-03	1.686029e-04	3.062514e-06
5 (Asphere)	0.547	-0.1491	2.31	AIR	-1.377027e-02	2.317677e-03	5.479770e-04	1.098924e-05
6 (Asphere)	0.436	0.1478	2.55	OKP4	-3.458765e-02	-3.592606e-03	-1.968449e-04	-5.557803e-06
7 (Asphere)	0.531	-0.0095	2.71	AIR	3.698580e-02	-4.438855e-05	-5.078399e-05	1.950553e-06
8 (Asphere)	0.460	-0.0474	3.05	PMMA	-1.109705e-02	-1.052585e-03	-1.413057e-04	-4.595032e-06
9 (Asphere)	0.654	0.1982	3.63	AIR	-1.286063e-02	-3.620622e-03	-1.336623e-04	5.103128e-07
10 (Sensor)			4.00	AIR				

Table 19. Lens data of quadruplet End2EndLens.

Surface	Thickness [mm]	Curvature $[\text{mm}]^{-1}$	Diameter [mm]	Material	α_4	α_6	α_8	α_{10}
1 (Aper)	0.100		1.04	AIR				
2 (Asphere)	0.776	0.2094	1.31	PMMA	-9.001061e-02	-4.458418e-03	1.466841e-04	3.094893e-06
3 (Asphere)	0.455	-0.2221	1.79	AIR	-1.007577e-01	-3.324089e-03	3.923575e-04	7.613106e-06
4 (Asphere)	0.455	0.0908	2.25	OKP4	1.579886e-03	-1.668447e-03	2.300933e-04	5.049782e-06
5 (Asphere)	0.610	-0.1367	2.30	AIR	-4.271065e-03	2.299120e-03	5.036045e-04	9.268433e-06
6 (Asphere)	0.382	0.1769	2.58	OKP4	-4.762138e-02	-4.734586e-03	-2.010100e-04	-4.986154e-06
7 (Asphere)	0.549	-0.0095	2.77	AIR	3.225713e-02	4.351055e-05	-6.839795e-05	9.540291e-07
8 (Asphere)	0.559	-0.1621	3.12	PMMA	5.123787e-03	-1.587822e-03	-1.526139e-04	-4.454318e-06
9 (Asphere)	0.466	0.3675	4.04	AIR	-2.638335e-02	-2.740633e-04	-3.539184e-05	2.557922e-06
10 (Sensor)			4.00	AIR				

Table 20. Optical specifications for each lens. TaskLens has the highest design difficulty with the same number of lens elements, ensuring a fair comparison.

		FoV	Sensor Diag [mm]	Difficulty
Doublet	TaskLens	69.5° (↑ 0.7%)	4.0	↑ 1.0%
	ImagingLens #1	69.1° (↑ 0.4%)	4.0	↑ 0.4%
	ImagingLens #2	70.6° (↑ 2.6%)	3.9 (↓ 2.5%)	↓ 0.1%
	ImagingLens #3	69.5° (↑ 1.0%)	4.0	↑ 1.0%
Triplet	TaskLens	70.7° (↑ 2.8%)	4.0	↑ 2.8%
	ImagingLens #1	69.2° (↑ 0.6%)	4.0	↑ 0.6%
	ImagingLens #2	69.1° (↑ 0.4%)	3.9 (↓ 2.5%)	↓ 2.1%
	ImagingLens #3	69.1° (↑ 0.4%)	4.0	↑ 0.4%
Quadruplet	TaskLens	70.0° (↑ 1.7%)	4.0	↑ 1.7%
	ImagingLens #1	68.7° (↓ 0.2%)	4.0	↓ 0.2%
	ImagingLens #2	70.6° ↑ 2.6%	3.8 (↓ 5.0%)	↓ 2.4%
	ImagingLens #3	69.3° (↑ 0.7%)	4.0	↑ 0.7%