https://doi.org/10.2352/EI.2026.38.11.HPCI-206

Lens Simulation for High-Resolution and Multispectral 3D Scenes Using Depth-Variant Lens Effects and Distributed Multi-GPU Processing

Seonghyeon Kang[1], Jeongyoung Shin[1], Sangmin Kim[1], Jeongwook Lee[1], Sung-Su Kim[1], Yitae Kim[1], Jinhee Kim[2]
[1]Samsung Electronics Co., Ltd.; Hwaseong-si, Gyeonggi-do, Republic of Korea
[2]Synopsys, Inc., Jung-gu, Seoul, Republic of Korea

Abstract

Full optical simulations of 3D scenes using commercial ray-tracing tools such as Speos provide physically accurate results but suffer from excessive computation time. To overcome this limitation, we propose a hybrid simulation pipeline that separates scene rendering and lens modeling into two stages. First, Zemax[1] is used to precompute lens characteristics including distortion maps, depth-variant relative illumination, and a library of depth-dependent point spread functions (PSFs). Second, Speos is employed to render the ideal camera scene and per-pixel depth maps without optical degradation. Finally, a distributed multi-GPU system efficiently applies the Zemax-derived optical degradations to the Speos-rendered images. This approach significantly reduces simulation time while retaining the essential physical properties of lens degradation. The proposed framework provides realistic image formation data for testing computer vision and imaging system design.

Keywords: *Lens Simulation, Depth-Variant PSF, Distributed Multi-GPU Processing*

Introduction

Accurate modeling of lens effects is a crucial component for designing modern imaging systems and for evaluating the performance of advanced computer vision algorithms. Historically, the evaluation of lens performance has been divided into two primary perspectives: the quantitative evaluation preferred by developers, and the qualitative evaluation experienced by end-users . Quantitative metrics often rely on standard 2D chart simulations, such as the TE268 and TE42 charts, which provide isolated, flat-field performance data. Conversely, qualitative evaluation requires the simulation of complex 3D scenes featuring various depths and lighting conditions, such as the widely utilized 3D Cornell box.

Commercial optical simulators, such as Ansys Speos[2], allow for physically based ray-tracing of complete 3D scenes through complex lens assemblies. However, this fully integrated computation is prohibitively slow. Rendering large, high-resolution images with full optical effects can take hours or even days to complete. This computational bottleneck severely limits practical research cycles and industrial use-cases where rapid iteration is necessary. Furthermore, modern simulation pipelines require specific capabilities: the ability to handle black-box lenses to protect proprietary designs, the generation of noise-free outputs, the incorporation of wave optics rather than relying solely on geometric optics, and a high degree of process automation.

Consequently, there is a strong and present need for a faster yet physically grounded simulation framework that can adequately handle both the geometric complexity of 3D scenes and the detailed, wave-optic characteristics of real lenses. This project aims to develop a hybrid simulation pipeline that substantially reduces computation time while maintaining physically realistic lens degradation. Specifically, our core objectives are to decouple lens modeling from scene rendering using Zemax and Speos, to build a comprehensive depth-variant optical degradation pipeline incorporating relative illumination, distortion, and PSFs, to accelerate processing via a distributed multi-GPU system, and to validate the physical accuracy of the proposed method against full Speos lens simulations.

Background and Simulation Software

Current industry-standard workflows typically rely heavily on either Zemax or Speos. Understanding the specific strengths and operational paradigms of each software is essential for recognizing the limitations of traditional simulation methods.

Zemax is a dedicated program designed specifically for dealing with lens design and detailed optical analysis. It is exceptionally powerful at extracting various intricate lens properties that dictate image quality. Key outputs include the Modulation Transfer Function (MTF), Point Spread Function (PSF), relative illumination profiles, and precise geometric distortion maps. Zemax is capable of performing raytracing based on both wave and geometric optics, allowing for a highly accurate representation of light behavior as it passes through a lens element. Furthermore, Zemax supports automation and pipeline integration via its Python-based ZOS-API[3].

Speos, conversely, is tailored for broader optic design, sensor simulation, and environmental lighting analysis. It is highly adept at generating accurate depth maps of 3D environments and relies primarily on geometric optics for its raytracing calculations. Like Zemax, Speos supports programmatic control through a dedicated Python API, known as pySpeos[4], which allows developers to script scene setups and rendering tasks.

Proposed Methodology 1. Importing Zemax lens to Speos

The conventional approach to lens simulation in full 3D environments, defined in our study as method 1, involves importing the Zemax lens model directly into the Speos environment to execute the complete ray-tracing simulation. This pipeline fundamentally relies on recreating the physical lens assembly within the 3D rendering engine to calculate how light interacts with the scene and the optic system.

To perform this simulation, the lens data must undergo a rigorous conversion from Zemax to Speos. This process requires the importation of the entire physical lens module, which includes not only the optical elements but also the mechanical housing. Once imported, several physical and optical parameters must be manually or programmatically defined within the Speos environment (See Figure 1.):

- Geometric and Material Definition: The simulation mandates exact geometric information, shape structures, and material properties, including specific refractive indices for each lens element.
- Spatial Alignment: The user must define the object distance to be focused and configure the precise lens positioning within the 3D scene.
- Sensor Configuration: Accurate positioning of the sensor relative to the lens must be established to ensure the focal plane aligns correctly, which may also involve setting up auto-focusing routines.

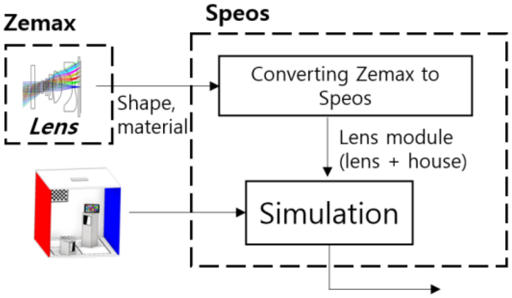


Figure 1. *Block diagram of method 1.*

Once the 3D scene and the lens module are fully configured, Speos traces the light rays through the environment and the explicit geometry of the lens to compute and output a multispectral image.

While Method 1 successfully produces a simulated scene with lens effects, the methodology is severely hampered by computational bottlenecks and functional constraints that limit its industrial viability:

- Inefficient and Heavy Computation: Simulating the physical propagation of light through complex lens geometries is highly resource-intensive. A typical modern lens system can require tracing rays through up to 6 million individual optical faces, leading to exceptionally heavy and inefficient computation. For instance, rendering our 5k x 5k Cornell box target scene required 50 hours of continuous processing time using this method.
- Restriction to Geometric Optics: Because Speos operates primarily as a geometric ray-tracer, this full simulation method yields only geometric optic results. It inherently misses nuanced wave optic phenomena, such as diffraction, which are critical for evaluating true high-resolution sensor performance.
- Incompatibility with Black-Box Lenses: One of the most significant industrial drawbacks is that Method 1 cannot accept black-box lenses. Because the simulation requires full geometric and material disclosure to trace the rays, it poses a severe security and intellectual property hurdle when evaluating proprietary lenses provided by third-party manufacturers.
- Material Constraints: This specific simulation pathway frequently restricts interactions to strictly opaque objects in the surrounding scene, complicating the accurate simulation of environments containing highly refractive or reflective elements.

Proposed Methodology 2. Depth-Variant Lens Effects (DVLE)

We proposed method 1 and found its limitation. To overcome these issues, we propose Method 2: A lens simulation system applying depth-variant lens effects (DVLE). This approach intentionally decouples the lens physics from the environmental rendering.

Stage 1: Precomputing Lens Modeling in Zemax

The first stage of the hybrid pipeline leverages Zemax solely for generating lens-specific data. Rather than tracing rays from the scene through the lens, we extract the inherent degradations the lens will cause. We generate precise distortion maps and relative illumination profiles. Crucially, we compute an extensive depth-variant PSF library. Because a lens focuses light differently depending on the distance of the subject, this PSF library is indexed meticulously by both field position and focal depth, providing significantly more realism than traditional, simplified shift-invariant blur models.

Stage 2: Ideal Scene Rendering in Speos

Simultaneously, Speos is utilized to render an ideal, uncorrupted version of the 3D scene. Operating essentially as an ideal pinhole camera, Speos processes the scene free of any lens degradations[4]. The outputs of this stage are a pristine multispectral scene image and a corresponding highly accurate, pixel-wise depth map.

Stage 3: Distributed Multi-GPU Optical Degradation

The final stage is the systematic application of the Zemax-derived optical degradations to the Speos-generated ideal images. This is achieved by combining the scene and depth map from Speos with the DVLE variables (relative illumination, distortion, and PSF) extracted from Zemax.

- Distortion and Illumination: Distortion correction is applied strictly on a pixel-wise basis, while the relative illumination data is used to modulate the intensity fall-off accurately across the field of view.
- PSF Convolution: Using the depth map, weight maps are generated by depth. The depth-variant PSF convolution is then applied selectively according to the exact depth of each individual pixel[6][7][8].

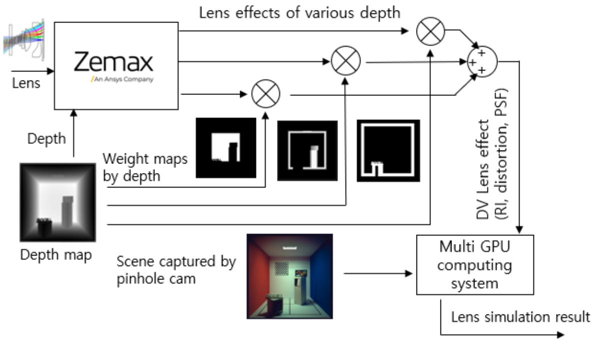


Figure 2. *Block diagram of the method 2.*

The Figure shows workflow of DVLE method. Because applying complex PSF convolutions on a pixel-by-pixel basis across large matrices is computationally intensive, a distributed multi-GPU computing system[5] is employed. This multi-GPU acceleration provides a practical, scalable solution for processing large, high-resolution images rapidly.

Experimental Setup and Results

To ensure the proposed hybrid framework preserves essential physical accuracy, we conducted a series of comparative experiments evaluating the output of the DVLE method against the traditional full Speos lens simulation (Method 1).

4.1 Edge Spread Function (ESF) Validation

The primary validation of optical accuracy was performed by comparing the Edge Spread Function (ESF). A slanted edge chart was placed in a 3D simulated space to capture the transition gradients. The ESF was tested and plotted across multiple distinct parameters. The figure 3 shows the chart for the simulation and its shape in 3D space.

- Wavelengths: Tests were independently run at 450nm, 550nm, and 650nm to account for chromatic variations.
- Field and Depth Positions: Data was collected at the center axis, designated as 0.0F (Distance D = 1m), at an intermediate field of 0.4F, and at 0.62F (tested at distances D = 1.375m and D = 0.625m).

Table 1. shows the results of the experiment. The graphical plots resulting from this experiment confirm that the DVLE method reproduces the baseline Speos results closely. The plotted curves for Zemax (both FFT and GEO) and the applied DVLE aligned tightly with the Speos reference data points across all tested wavelengths and field positions.

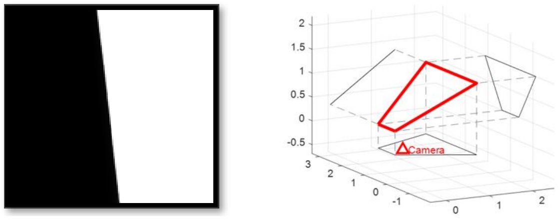


Figure 3. *Slatend edge chart (left) and the chart in 3D space and camera (right)*

*4.*2 High-Resolution Cornell Box Scene Simulation

To evaluate the pipeline's efficiency and robustness in a complex environment, a full-scale experiment was conducted utilizing a 3D Cornell box target scene.

- Scene Parameters: The scene included specific test elements: a standard Macbeth color checker chart and a customized "ISOCELL" block consisting of both short and tall geometric cubes. These objects were positioned at variable focal depths ranging from 0.5m to 1m from the target lens (a proprietary lens provided by our company).
- Output Specifications: The simulation rendered a high-resolution 5k x 5k image with a highly detailed 1um pixel pitch, utilizing wavelengths of 450nm, 550nm, and 650nm, and applying the geometric point spread function via DVLE.

The plots in table 2 validate that the results of two experiments are nearly identical. However, the time for computing is a serious issue. The efficiency gains demonstrated in this experiment were highly significant. While Method 1 (importing the Zemax lens into Speos) required about 50 hours of continuous computational processing to complete the simulation, Method 2 (applying DVLE via the distributed multi-GPU framework) completed the exact same scene in about 20 hours.

Discussion and Conclusion

This work demonstrates a novel and practical hybrid Zemax-Speos workflow. The comparative advantages of the proposed DVLE methodology over directly importing a Zemax lens into Speos are clear.

By decoupling the processes, the DVLE method easily manages general scenes and readily accepts black-box lenses without requiring proprietary geometric exposure. It achieves vastly faster computational times while uniquely allowing for the integration of highly accurate wave optic results into a 3D environment. In contrast, the traditional method suffers from heavy computation, outputs only geometric results, fails to support black-box lens inputs, and struggles with non-opaque elements.

In conclusion, our distributed multi-GPU system achieves significant runtime reduction, rendering it orders of magnitude faster than attempting full Speos simulation for complex iterative tasks. The resulting simulated images serve as highly valuable, physically grounded test data for image restoration, denoising algorithms, and broad computer vision pipelines requiring realistic lens effects.

Future Work

The efficacy of the proposed Depth-Variant Lens Effects (DVLE) relies heavily on the acquisition of an accurate depth map. However, in scenes containing reflective or refractive surfaces such as mirrors and glass, discrepancies between the actual scene and its depth representation arise, making a precise depth definition challenging. Furthermore, the robustness of the model must be validated against highly intricate or thin structures

The ultimate objective of this study is to simulate the capture of general scenes to produce high-fidelity digital images. While this paper focuses specifically on the optical process, the full imaging pipeline encompasses subsequent stages, including pixel response, sensor characteristics, and ISP (Image Signal Processing). To validate this end-to-end pipeline, it is imperative to obtain noise-free optical images. Since Ansys SPEOS utilizes a Monte Carlo-based algorithm, achieving a theoretically noise-free result is inherently limited; even approaching such a state requires prohibitive computational resources. Consequently, a post-processing framework will be implemented to suppress noise. Addressing this noise reduction is a critical task that directly dictates the research's academic value and practical utility.

Table 1. ESF comparing results of DVLE method

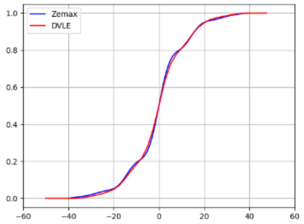
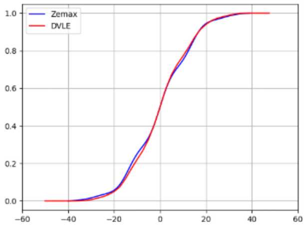
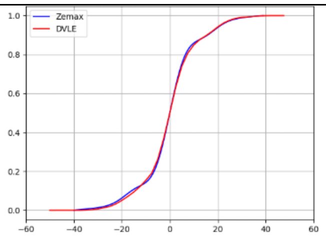
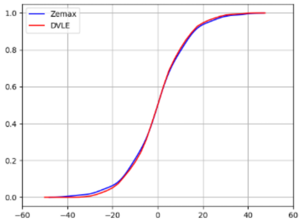
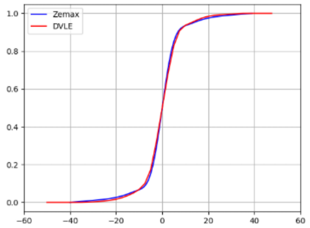
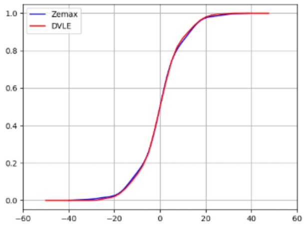
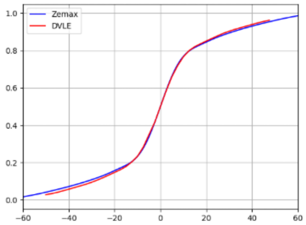
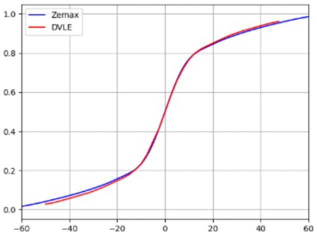
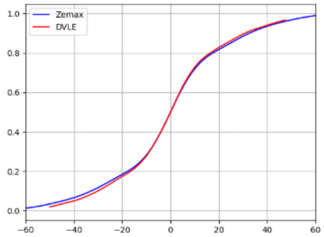
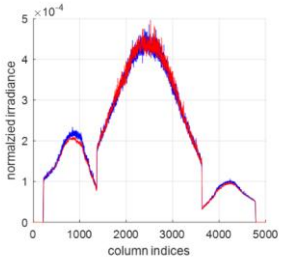
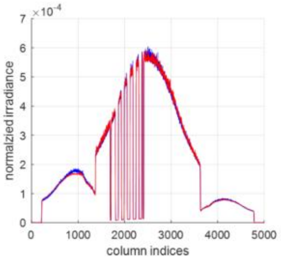
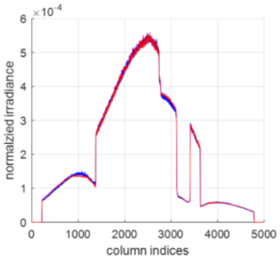
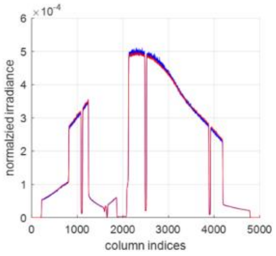
	450nm	550nm	650nm
D= 1.375m (0.62F)			
D =1m (0.0F)			
D =1m (0.0F)			

Table 2. Cross sections from the simulation methods. Red line is from Method 1, and blue line are from DVLE method

Reference

[1] Zemax Opticstudio, https://www.ansys.com/products/optics/ansys-zemax-opticstudio.

[2] Ansys Speos, https://www.ansys.com/products/optics/ansys-speos

[3] pySpeos, https://github.com/ansys/pyspeos

[4] R. Hartley and A. Zisserman, "Camera geometry and single view geometry" in Multiple View Geometry in Computer Vision, Cambridge, U.K.:Cambridge Univ. Press, pp. 151-236, 2004

[5] S. Kang et al., " Lens Simulation for High Resolution and Multispectral Image using Distributed Multi-GPU Computing System," Electronic Imaging, vol. 37, pp. 1-5, 2025

[6] Maeda, P. Y., Catrysse, P. B., and Wandell, B. A., “Integrating lens design with digital camera simulation,” Digital Photography, vol. 5678, pp. 48-58, 2005.

[7] Chen, J., Venkataraman, K., Bakin, D., Rodricks, B., Gravelle, R., Rao, P., and Ni, Y., “Digital camera imaging system simulation,” IEEE Transactions on Electron Devices, vol. 56, issue 11, pp. 2496-2505, 2009.

[8] ISetcam, https://github.com/ISET/isetcam.

JOIN US AT THE NEXT EI!

electronic IMAGING

Imaging across applications . . . Where industry and academia meet!

• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •

• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

ADVANCING IMAGING SCIENCE
75
1947 IS&T 2022
imaging.org