A Case Study on Raytracing-in-the-Loop Optimization: Focal Surface Displays

Nathan Matsuda
Oculus Research

Alexander Fix
Oculus Research

Douglas Lanman
Oculus Research

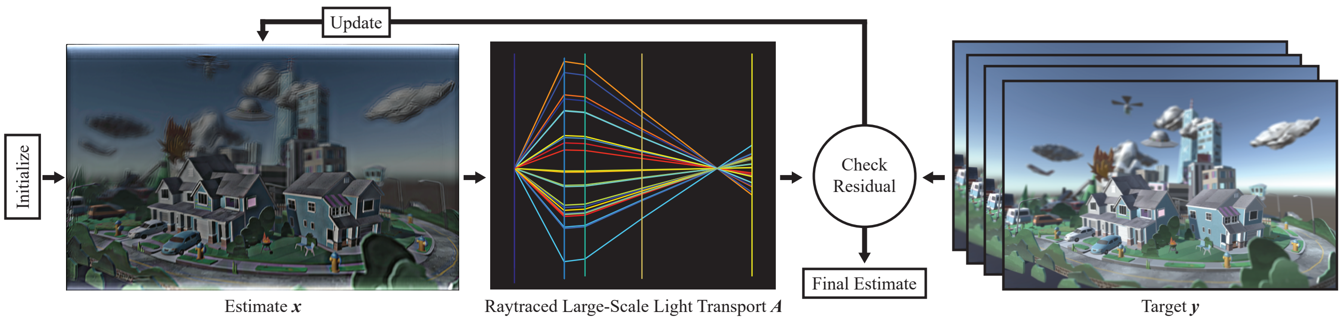


Figure 1: Computational display architectures, such as our focal surface display, often seek to approximate the focal stack y of a target 3D scene (right) using a lower-dimensional decomposition x (left) by solving a linear optimization problem of the form $y = Ax$. In such cases, problem size and model complexity frequently prevent the use of an explicit A matrix. We explore the use of a raytracing operator (center) along with iterative conjugate gradient methods to solve this type of computational display optimization.

ABSTRACT

Optimization-based design of optical systems can yield configurations that would be impractical to achieve with manual parameter adjustment. Nonetheless, most approaches are geared toward one-time, offline generation of static configurations to be fabricated physically. Recently, challenging computational imaging problems, such as seeing around corners or through scattering media, have utilized dynamically addressable optical elements to probe scene light transport. A new class of optimization techniques targeted at these dynamic applications has emerged in which stochastic raytracing replaces the fixed operators applied with conventional optimization methods. By modeling optical systems as raytracing operators, more complex non-linear phenomena and larger problem sizes can be considered.

We introduce a simple raytracing-in-the-loop optimization model for a head-mounted display (HMD) containing a spatial light modulator (SLM). Using this approach, we are able to compute color images to be displayed in concert with spatially varying SLM phase maps at a resolution that would otherwise be computationally infeasible. We also consider extensions of this model that may further enhance the performance of the target system.

CCS CONCEPTS

•**Human-centered computing** →**Displays and imagers;**

SIGGRAPH '17 Talks, Los Angeles, CA, USA

DOI: http://dx.doi.org/10.1145/3084363.3085071

KEYWORDS

head-mounted displays, multifocal displays, caustics, freeform optics, vergence-accommodation conflict, virtual reality

ACM Reference format:
Nathan Matsuda, Alexander Fix, and Douglas Lanman. 2017. A Case Study on Raytracing-in-the-Loop Optimization: Focal Surface Displays. In *Proceedings of SIGGRAPH '17 Talks, Los Angeles, CA, USA, July 30 - August 03, 2017,* 2 pages.
DOI: http://dx.doi.org/10.1145/3084363.3085071

1 OVERVIEW

A focal surface display augments conventional HMDs with a phase-only spatial light modulator (SLM) placed between the display screen and viewing optics (see Figure 2). As we have proposed [Matsuda et al. 2017], the SLM acts as a dynamic freeform lens, shaping synthesized focal surfaces to conform to the virtual scene geometry. This allows for the perception of near-correct optical blur by the viewer, potentially avoiding vergence-accommodation conflict (VAC) and allowing for higher maximum display resolutions. To generate SLM phase patterns and corresponding color images, we employ an optimization framework whose objective matches a simulated focal stack, as output from the system, to a target focal stack. Moreover, the target solution may contain multiple, time-sequential subframes. This problem size is far too large to implement the light transport model as an explicit matrix. A patch-based approach, as would typically be used for superresolution tasks or other large image-based optimizations, is not suitable here due to the blur sizes we would expect to see over the target depth of focus. Instead, we model the system via GPU-based raytracing.

This has several advantages:

- Highly parallelizable calculation.
- Simple integration of iterative optimization frameworks.
- Extensibility for other optical effects, such as diffraction.

2 RELATED WORK

Optimization involving physical systems with highly non-linear structures and large problem sizes has been explored in detail, notably in the analysis-by-synthesis models for speech recognition. This approach has long been used in vision problems, such as shape-from-shading [Nayar et al. 1991]. Recently, Gkioulekas et al. [2013] introduced a rendering-in-the-loop optimization technique to recover scattering properties of real materials. Their approach iteratively updates parameters based on measured properties, applying a gradient-domain raytraced estimate of the effect of current parameters. Klein et al. [2016] employ a similar technique to track an object hidden by a scattering occluder.

3 IMPLEMENTATION

We implemented our raytracing model using NVIDIA Optix [Parker et al. 2010]. Rays are traced from a point within the viewer's pupil, through the eyepiece, the SLM, and impinge on the display (see Figure 2). At the eyepiece surfaces, rays are refracted using the radius of curvature of the lens, its optical index, and the paraxial approximation. Light transport through the SLM, where the incident ray has direction vector $(x, y, 1)$ and wavelength λ, is modeled similar to Voeltz [2011]:

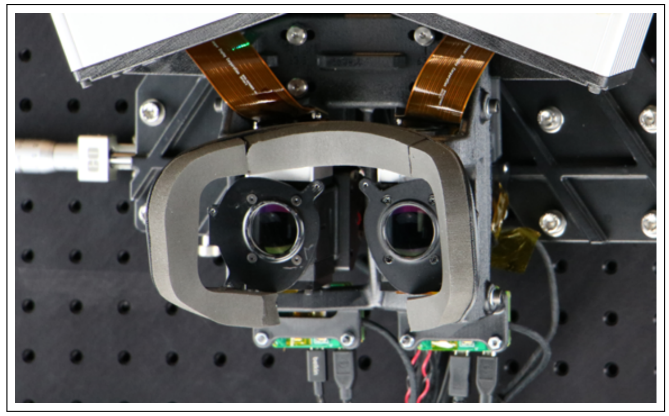
$$\left(x + \frac{\lambda}{2\pi}\frac{\partial\phi}{\partial x}(p_x, p_y), y + \frac{\lambda}{2\pi}\frac{\partial\phi}{\partial y}(p_x, p_y), 1\right) \tag{1}$$

Where (p_x, p_y), are SLM pixel centers in the discrete set Ω_p.

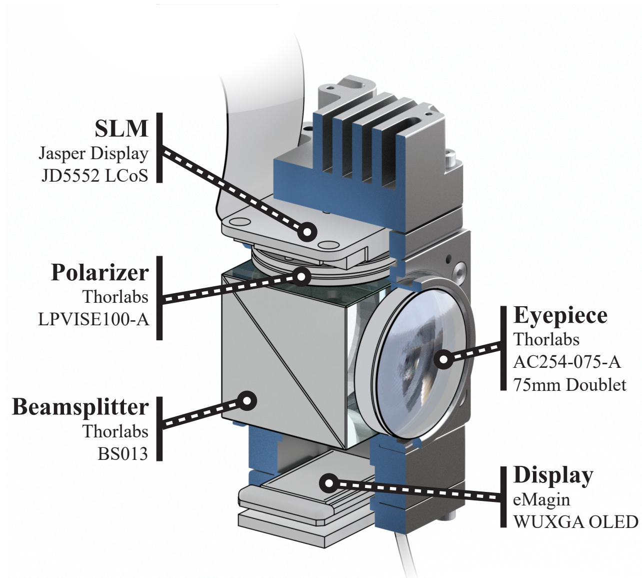
To model retinal blur, we accumulate Poisson-disk sampled rays that span the viewer's pupil. Initial ray directions are chosen to approximate an ideal lens focused at a depth z. For each chief ray (θ_x, θ_y) and depth z, we sum across a bundle of rays $R_{\theta_x,\theta_y,z}$ from the Poisson-sampled pupil. This produces an estimate of the retinal blur when focused at a depth z. We define these preceding steps as the *forward operator* $r = A_{z,\phi}(c)$, which accepts a phase function ϕ and color image c and predicts the perceived retinal image r when focused at a distance z. Its transpose, which maps retinal image samples to display pixels, can be similarly evaluated with ray tracing operations with accumulation in the color image c rather than the retinal image r. These forward and adjoint operators are applied with an iterative conjugate gradient least squares solver in Matlab.

All system parameters are matched to our physical prototype, shown in Figure 2. Using a dataset of 3D scenes rendered using Unity, we utilized our optimization approach to generate decompositions that can be displayed by the prototype. We evaluated these results relative to the corresponding ideal focal stacks using the same raytracing model to simulate system output.

In summary, emerging computational displays require precise modeling of an increasingly complex array of optical and display elements. As shown in our case study of focal surface displays, raytracing-in-the-loop optimization is becoming necessary to drive such devices. In this talk, we review related analysis-by-synthesis methods and discuss applications to computational displays.



(a) Construction of the Prototype



(b) Arrangement of the Optical Components

Figure 2: (a) Our binocular focal surface display prototype incorporates commodity optical and mechanical components. (b) A cutaway of of the prototype exposes the arrangement of the optical components.

REFERENCES

Ioannis Gkioulekas, Shuang Zhao, Kavita Bala, Todd Zickler, and Anat Levin. 2013. Inverse volume rendering with material dictionaries. *ACM Transactions on Graphics (TOG)* 32, 6 (2013), 162.

Jonathan Klein, Christoph Peters, Jaime Martín, Martin Laurenzis, and Matthias B Hullin. 2016. Tracking objects outside the line of sight using 2D intensity images. *Scientific Reports* 6 (2016).

Nathan Matsuda, Alexander Fix, and Douglas Lanman. 2017. Focal Surface Displays. *ACM Trans. Graph.* 36, 4 (2017).

Shree K Nayar, Katsushi Ikeuchi, and Takeo Kanade. 1991. Shape from interreflections. *International Journal of Computer Vision* 6, 3 (1991), 173–195.

Steven G. Parker, James Bigler, Andreas Dietrich, Heiko Friedrich, Jared Hoberock, David Luebke, David McAllister, Morgan McGuire, Keith Morley, Austin Robison, and Martin Stich. 2010. OptiX: A General Purpose Ray Tracing Engine. *ACM Transactions on Graphics* (2010).

David George Voelz. 2011. *Computational Fourier Optics: A MATLAB Tutorial.* SPIE Press.