Internship Report

Topic: Report on Chromatix

Author:

Ziyi Xiong

Master's Student in Photonics at ASP

Internship Period:

01-09-2024 to 28-02-2025

Supervisor:

Prof. Dr. Vladan Blahnik, Institute of Applied Physics

Industry Advisor:

Prof. Dr. Frank Wyrowski, President of LightTrans GmbH

Contents

Chapter 1

Introduction

1.1 Context of the Internship

This internship was conducted under the joint guidance of university supervisor Prof. Dr. Vladan Blahnik and industry advisor Prof. Dr. Frank Wyrowski. The core focus was the Chromatix library, an open-source Python package developed at HHMI Janelia Research Campus. Chromatix utilizes wave-optics theory and modern computational tools (particularly JAX and FLAX) to enable differentiable optical simulations.

My responsibilities included a comprehensive exploration of Chromatix, encompassing its wave-optical underpinnings, architectural structure, and practical use cases. I investigated how the library constructs and manipulates optical fields, propagates waves, and supports inverse design tasks through parameter optimization.

1.2 Goals and Objectives

The main goals of this internship were to:

1. Perform a detailed analysis of Chromatix's codebase, documenting the key classes, functions, and module structures.
2. Relate each software implementation to the corresponding physical principles in wave optics.
3. Demonstrate practical usage with various example systems such as a widefield PSF simulation, a 4f system, and a DMD-based CGH setup.
4. Evaluate the current state of the library and propose potential improvements or future directions.

1.3 Structure of This Report

This report is organized into the following chapters:

- **Chapter 1: Introduction** - Describes the objectives, context, and structure of the report.

- **Chapter 2: Core Source Code Analysis** - Analyzes Chromatix's internal components and their relationship to optical theory.
- **Chapter 3: Practical Usage Examples** - Presents detailed use cases and simulations.
- **Chapter 4: Conclusions & Future Work** - Summarizes key insights and outlines directions for further development.

Chapter 2

Chromatix in Detail

2.1 Overview

Chromatix is a differentiable wave-optics simulation library designed for modeling complex optical systems using GPU acceleration. Its key features include:

- **GPU Support**: Efficient computation on GPUs/TPUs.
- **Inverse Design**: Gradient-based optimization of optical elements.
- **Modular Design**: Clear separation into packages like `functional`, `elements`, and `systems`.
- **Multi-Wavelength and Polarization**: Full support for scalar/vector fields and spectral diversity.

2.2 Package Breakdown

Chromatix is organized into several key packages:

- `data`: Provides base shapes, images, and permittivity tensors.
- `functional`: Implements core physical transformations such as propagation and phase masking.
- `elements`: Wraps optical components as trainable FLAX modules.
- `systems`: Combines elements into full optical systems.
- `ops`: Adds general-purpose image-processing functions.
- `utils`: Offers helper tools like FFT wrappers and Zernike functions.
- `field.py`: Defines field data structures with phase/intensity computation.
- `__init__.py`: Provides initialization logic for the package.

2.3 Theoretical Foundations

Chromatix is grounded in classical and computational optics:

1. **Fourier Optics**: Uses FFT for propagation models.
2. **Jones Calculus**: Models polarization effects.
3. **Sampling Theory**: Ensures correct spatial/frequency resolution.
4. **Differentiable Programming**: Enables gradient-based optimization.

Chapter 3

Core Source Code Analysis

3.1 Field Class (`field.py`)

The `Field` class is central to representing electromagnetic fields:

- `u`: Complex-valued wave field array.
- `_dx`: Pixel spacing.
- `_spectrum`: Wavelengths in the simulation.
- `_spectral_density`: Spectral weightings.

Key Methods:

- `grid()`, `k_grid()`: Return coordinate grids.
- `phase()`, `amplitude()`, `intensity()`: Extract wave properties.
- Operator overloading for arithmetic operations.

3.1.1 Why JAX?

JAX enables automatic differentiation of optical transforms. By tracing operations like FFT and phase shifts, Chromatix supports inverse design via gradient descent. This is crucial for optimizing system parameters based on target outputs.

3.2 Functional Package

Includes key transformations:

- `propagation`: Fresnel and angular spectrum models.
- `lenses`, `polarizers`, `phase_masks`, etc.: Simulate optical components.
- `sources`: Define input wavefronts.

3.3 Elements Package

Elements wrap functions into FLAX modules:

- `FFLens`, `Polarizer`, `PhaseMask`: Trainable optical elements.

3.4 Systems Package

Combines modules into optical pipelines:

- `OpticalSystem`, `Microscopy`, `Optical4FSystemPSF`.

3.5 Performance Considerations

Performance scales with resolution and batch size. JAX with XLA enables efficient, compiled computation, particularly on GPUs. Memory usage can be significant for high-resolution or spectral simulations.

Chapter 4

Application Examples

4.1 Example 1: Optical System A

[Insert detailed description of the system, implementation steps, and results. Include figures if needed.]

4.2 Example 2: Optical System B

[Insert detailed description of the system, implementation steps, and results. Include figures if needed.]

Chapter 5

Conclusion and Future Work

5.1 Conclusion

This internship provided deep insights into differentiable wave-optics simulation. Chromatix is well-structured and powerful for modern computational optics. Understanding its architecture and theory offers a strong foundation for future research.

5.2 Future Work

Suggestions include:

- Expanding library documentation and tutorials.
- Enhancing support for non-paraxial and nonlinear elements.
- Integrating with other simulation tools or experimental data pipelines.

Chapter 6

Acknowledgments

I wish to express my sincere gratitude to my supervisor, Prof. Dr. Vladan Blahnik, for his guidance throughout this internship.

Special thanks to Prof. Dr. Frank Wyrowski, CEO of LightTrans GmbH, for providing the internship opportunity, sharing valuable insights, and offering continuous support.

Appendix A

Additional Material

[Include scripts, extended code, or supplementary material here.]