Senior Project – Computer Science – 2015

Optimizations for Rendering Realistic Lens Flares

Stephen DiIorio

Advisor – Prof. Matthew Anderson

Introduction

A lens flare is an optical effect consisting of halos and geometric shapes. They are created by light bouncing around a lens system, like one found in a camera.

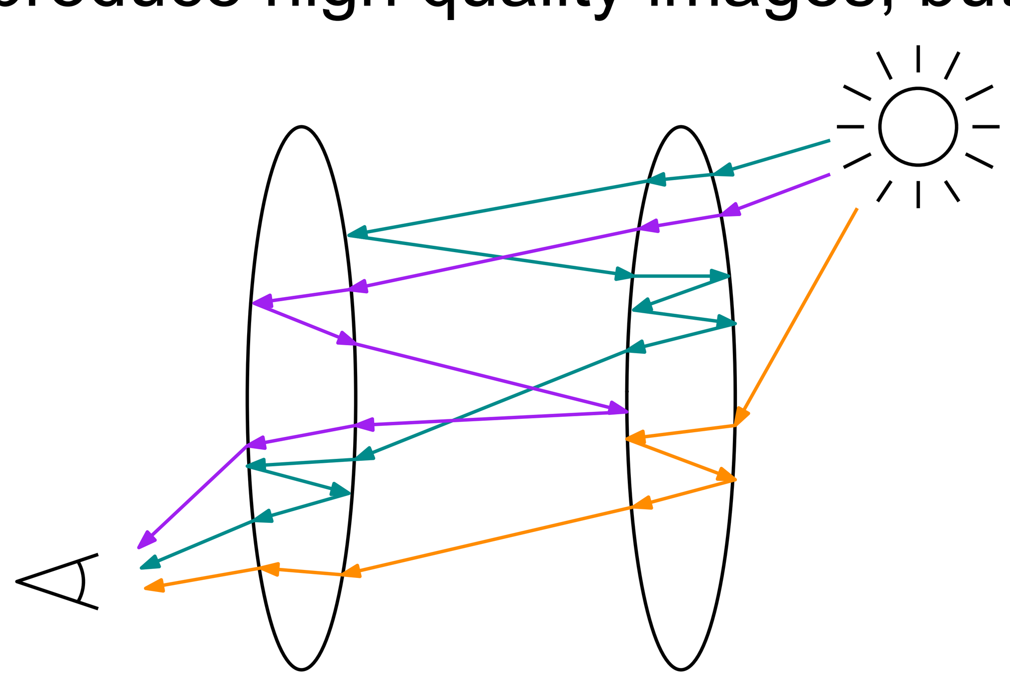
Above is an example of a lens flare, with streaks of light emanating from the sun and colored hexagons [1].

Problem

Given a lens system, can we quickly simulate high fidelity lens flares that match those found in real life?

Ray Tracing

Ray tracing can solve this problem and be used to produce high quality images, but this is a ***slow*** process.

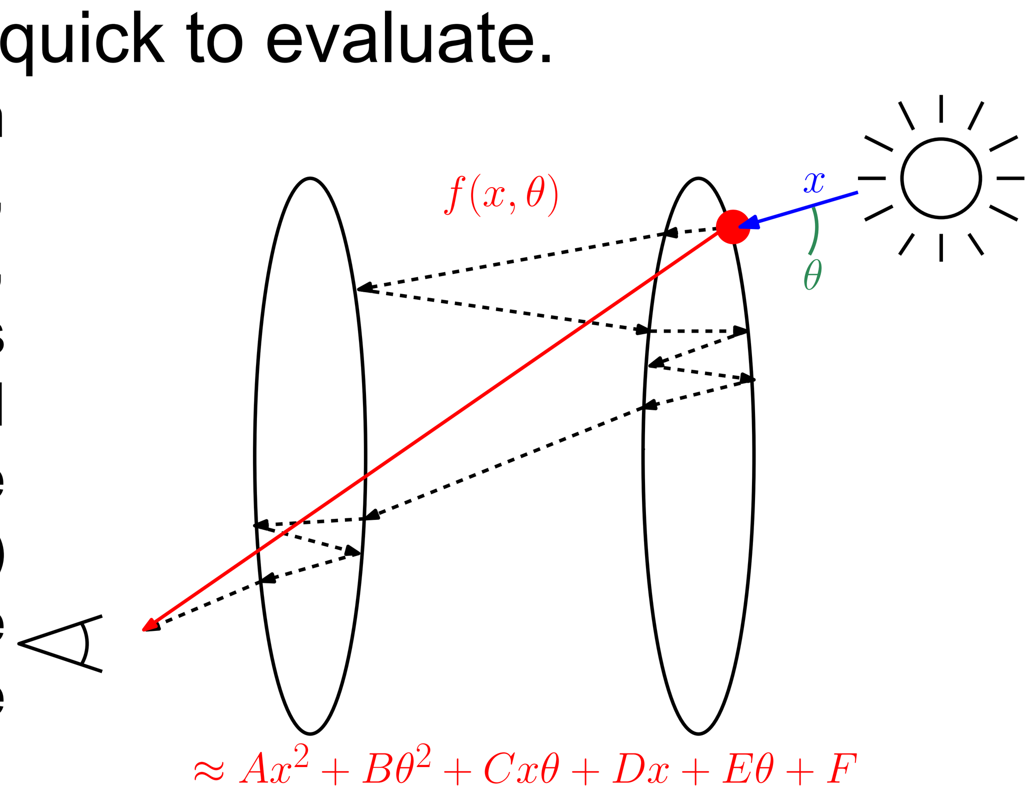


We have to sample multiple rays of light at different points and of different colors to produce a realistic image. For each ray, we have to simulate how it will behave at each lens in the system.

Polynomial Optics

Polynomial optics is a way of modeling how light moves through a lens system via a set of polynomial expressions, which are quick to evaluate.

This polynomial expression takes in initial conditions, like a position in the scene, x, and the angle light hits the lens system, θ, and tells us its approximate final position (the red path) as if we ray traced (the dotted path) through the lens system.



Project Goals

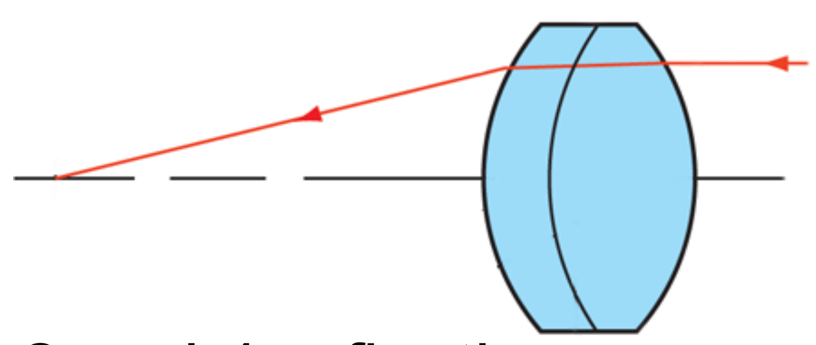
How do we quickly simulate lens systems and create polynomials that retain enough information to produce images with high fidelity?

Methods

- Some terms contribute little when evaluating a polynomial, so removing them based on a threshold should reduce the amount we have to compute.
- Memoize, or store, exponentiations of variables and reuse them to evaluate the polynomials.

Experiments

- Implemented our methods within the Polynomial Optics library [2].
- Simulated an achromatic lens using degree three polynomials [3].

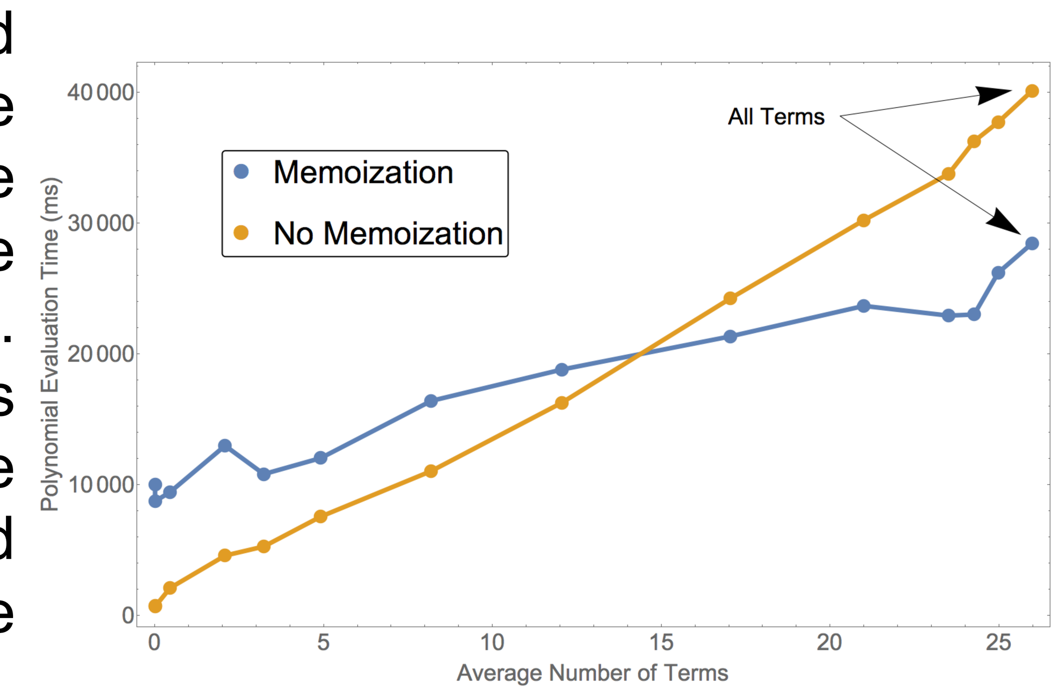


- Only generated paths with 2 and 4 reflections.
- For image evaluation, we took the largest pixel value difference from each image with respect to the one formed when keeping all the terms in the polynomial.

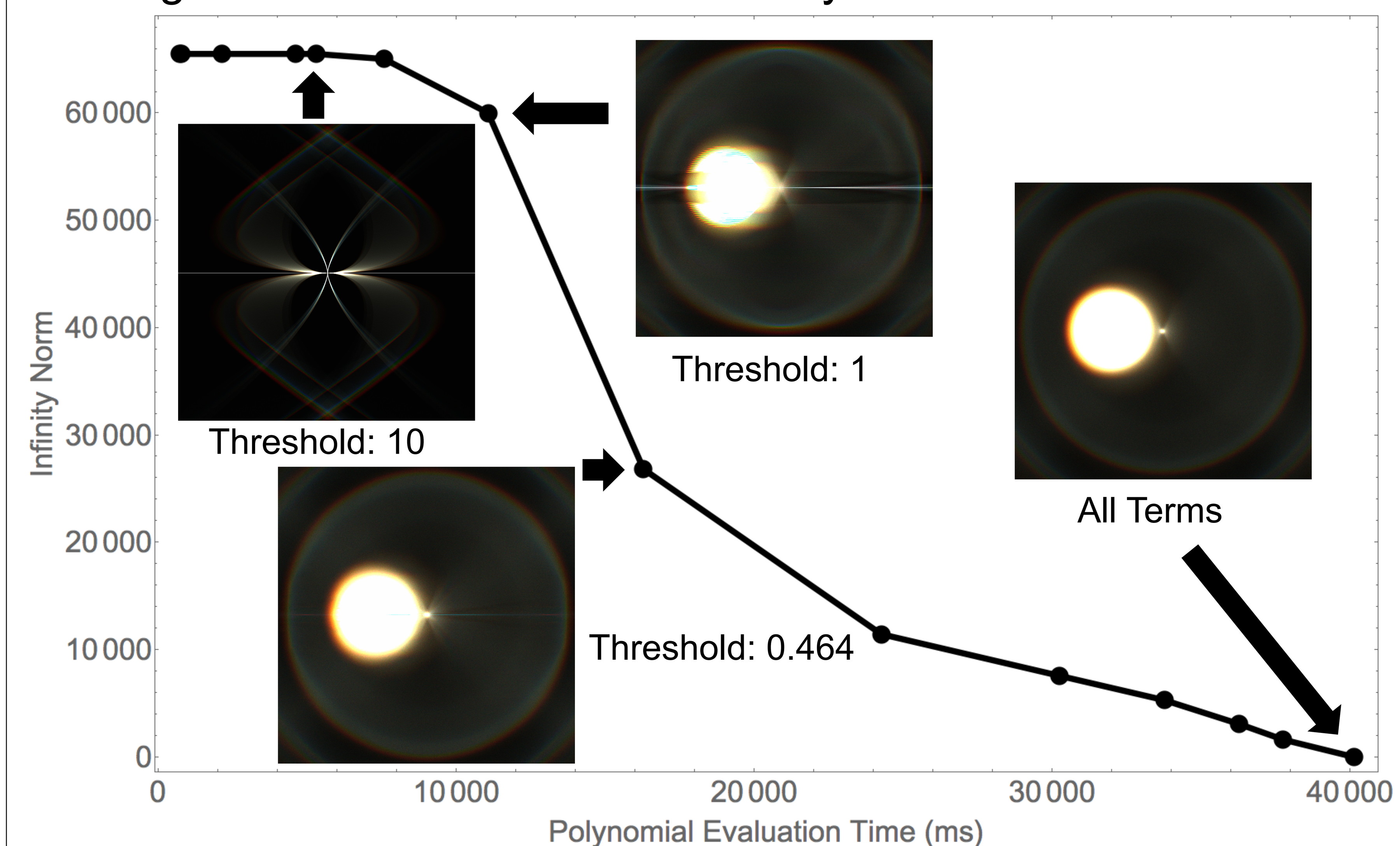
Results

Polynomial Evaluation Time versus Average Number of Terms

By removing terms based on a threshold, we decrease the average number of terms in the polynomials we evaluate. Memoizing data helps reduces the evaluation time until the overhead needed to memoize becomes more trouble than its worth.



Largest Pixel Difference versus Polynomial Evaluation Time



We would hope to have a low pixel difference and a low polynomial evaluation time, and we made advances in this regard. The images created with a threshold of 0.464 are still acceptable lens flares, and we achieve a speedup of ~2.5x in generating them compared to the images generated with all terms.

[1] Photo Credit: HWRG
[2] Hullin et al. Polynomial optics: A construction kit for efficient ray-tracing of lens systems. (2012)
[3] Edmund Optics Inc. Why use an achromatic lens?, 2014.

UNION COLLEGE CS