

---

# Depth of Field in OpenGL

---

Anthony Walker

---

# Article

---

## *Algorithms for Rendering Depth of Field Effects in Computer Graphics*

by Brian A. Barksy and Todd J. Kosloff

---

# Overview

---

- ❖ Depth Perception and Optics
- ❖ Object Space / Pipeline Application
- ❖ Image Space / “Postprocess Methods”
- ❖ Implementation
- ❖ Example



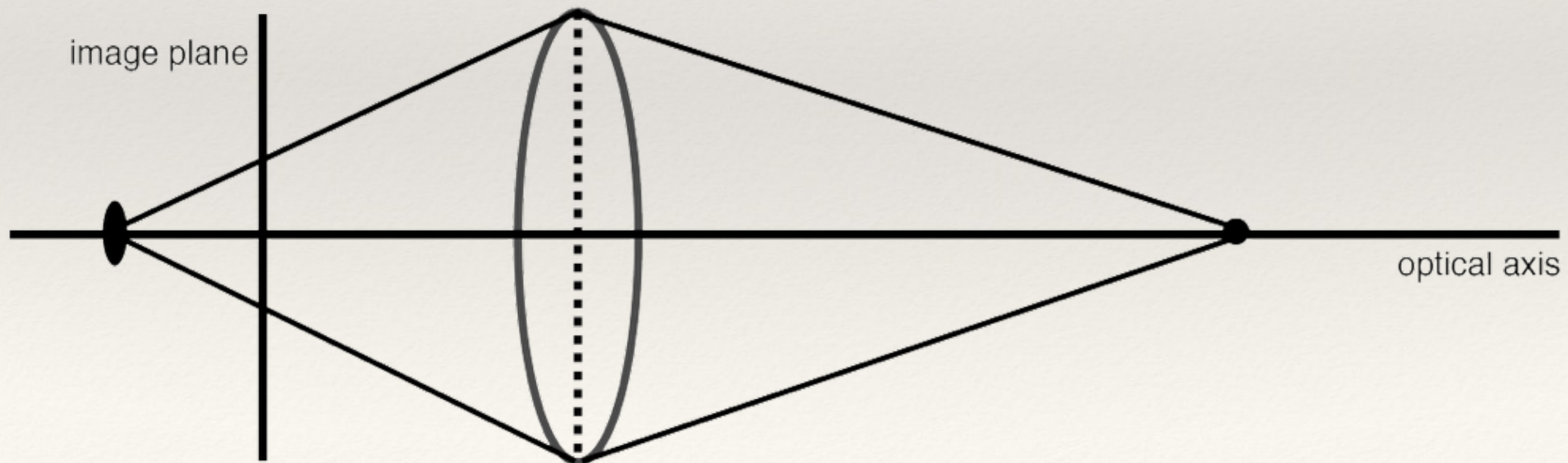


---

# Optics

---

- ❖ Human Perception of Depth
- ❖ Focal Points





---

# How are these Optics portrayed?

---

- ❖ Sleight of Hand (FPS)
- ❖ Artificially by a focus / blur effect combination

---

# Object Space

---

- ❖ Distributed Ray Tracing, “The Golden Standard”
- ❖ Accumulation Buffer
- ❖ Wave-Propogation, Splatting, Realistic Camera Models, Analytical Visibility, and many more...







---

# Image Space

---

- ❖ Qualifications / Methods:
  - ❖ Per-pixel blur level control, lack of depth discontinuity, high performance, ...
- ❖ Linear Filter

$$B(x, y) = \sum_i \sum_j psf(x, y, i, j) S(i, j)$$













---

# Implementation

---

```
-(void)setFocus:(Point) pt
{
    // get a reference to the front image
    CImage *front_image = ...;

    // is the touch within the bounds of the image
    if(CRectangleContainsPoint([front_image boundingBox], pt))
    {
        // If the point is to the left of ernies face set the front_image
        // that contains the sharp version of focus
        // 255 (opaque)
        if (pt.x < 280) { ...
        // we need to go from opaque to transparent hence
        // the adjustment
        front_image.opacity = 255 - (GLubyte)fOpacity;} ...
    }
}
```

Event Handler and Update for Future Mouse Clicks



Application