Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Graphical Models 67 (2005) 584–599
Graphical Models
www.elsevier.com/locate/gmod

Elimination of artifacts due to occlusion and discretization problems in image space blurring techniques ☆

Brian A. Barsky [a,b,*], Michael J. Tobias [a], Derrick P. Chu [a,c], Daniel R. Horn [a]

[a] *Computer Science Division, University of California, Berkeley, CA 94720-1776, USA*
[b] *School of Optometry, University of California, Berkeley, CA 94720-2020, USA*
[c] *Computer Science Department, University of California, Los Angeles, CA 90095-1596, USA*

Received 30 September 2003; received in revised form 6 June 2004; accepted 10 January 2005
Available online 21 July 2005

Abstract

Traditional computer graphics methods render images that appear sharp at all depths. Adding blur can add realism to a scene, provide a sense of scale, and draw a viewer's attention to a particular region of a scene. Our image-based blur algorithm needs to distinguish whether a portion of an image is either from a single object or is part of more than one object. This motivates two approaches to identify objects after an image has been rendered. We illustrate how these techniques can be used in conjunction with our image space method to add blur to a scene.

PACS: 42.30.Va; 42.66.Si

Keywords: Image forming and processing; Visual perception

☆ All requests for permission to use the figures in this article should be addressed to B.A. Barsky.
* Corresponding author.
E-mail address: barsky@cs.berkeley.edu (B.A. Barsky).
URL: http://www.cs.berkeley.edu/~barsky/ (B.A. Barsky).

1524-0703/$ - see front matter
doi:10.1016/j.gmod.2005.01.009

1. Introduction

Images rendered with traditional computer graphics techniques appear completely in focus. However, images formed by optical systems have depth of field; that is, some regions of the image are in focus while others are blurred. This is useful to draw a viewer's attention to a specific region of the image and can help to indicate the scale of a particular scene. The rendering of a 3D scene requires transforming from *object space* to *image space*. This transformation is parameterized by a camera angle and location. There are algorithms that can be used to apply blur in object space [3] where the entire geometry of the scene is present. Yet, after the transformation to image space, there are a wider variety of methods for applying depth of field [4].

2. Object space methods for depth of field

Since scene geometry information is readily available in object space, this is the most straightforward and ideal space in which to compute blur. Distributed ray tracing, first proposed by Cook et al. [7], sends multiple rays from the aperture of the lens, through each pixel on the image plane, and into the scene. Kolb et al. [12] proposed a more realistic algorithm for computing object space blur. For each pixel in the image, their approach emits multiple rays from the film plane, through a system of lens elements, and computes where these rays intersect the scene. Using this technique, the final color of each pixel is the average of the colors at all the nearest ray–object intersections for the rays emerging from this pixel.

Inherent in the distributed technique are questions of the density and spatial distribution of samples across the lens; these issues have been addressed by Cook [6], Dippe and Wold [8], Lee et al. [13], and Kajiya [11]. Further details are provided in our survey paper on object space techniques [3]. Additionally, simulating depth of field using these distributed ray tracing approaches is more computationally expensive than standard ray tracing. The increase in computation cost over standard ray tracing is proportional to the number of rays per pixel.

3. An image space method for depth of field

Image space techniques have been developed to avoid this increase in computation time. Potmesil and Chakravarty [14] were the first to propose an image space technique that smears colors over a region (the *circle of confusion*) for each pixel. The efficiency of the technique was improved by Rokita [15]. In 1994, Shinya [16] solved occlusion issues present in Potmesil's solution by using a sub-pixel buffer for each pixel to track all rays entering that pixel from a lens.

There has also been work regarding depth of field with respect to light fields, which results in realtime [9] and non-realtime [10] techniques. For a more detailed discussion of light fields and blur, the reader is referred to our survey paper on image space techniques [4].