Computers & Graphics 36 (2012) 673–684

Contents lists available at SciVerse ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on CANS

# Ray prioritization using stylization and visual saliency

Markus Steinberger *, Bernhard Kainz *, Stefan Hauswiesner, Rostislav Khlebnikov, Denis Kalkofen, Dieter Schmalstieg

*Institute for Computer Graphics and Vision, Graz University of Technology, Austria*

ARTICLE INFO

*Article history:*
Received 10 October 2011
Received in revised form
20 February 2012
Accepted 20 March 2012
Available online 30 March 2012

*Keywords:*
Ray-tracing
Ray-casting
Volume rendering
Photorealistic rendering
Visual saliency

ABSTRACT

This paper presents a new method to control scene sampling in complex ray-based rendering environments. It proposes to constrain image sampling density with a combination of object features, which are known to be well perceived by the human visual system, and image space saliency, which captures effects that are not based on the object's geometry. The presented method uses Non-Photorealistic Rendering techniques for the object space feature evaluation and combines the image space saliency calculations with image warping to infer quality hints from previously generated frames. In order to map different feature types to sampling densities, we also present an evaluation of the object space and image space features' impact on the resulting image quality. In addition, we present an efficient, adaptively aligned fractal pattern that is used to reconstruct the image from sparse sampling data. Furthermore, this paper presents an algorithm which uses our method in order to guarantee a desired minimal frame rate. Our scheduling algorithm maximizes the utilization of each given time slice by rendering features in the order of visual importance values until a time constraint is reached. We demonstrate how our method can be used to boost or stabilize the rendering time in complex ray-based image generation consisting of geometric as well as volumetric data.



## 1. Introduction

A common challenge of high-quality ray-based image generation is maintaining the scene interactivity of the applications. This interactivity is normally achieved by sacrificing some of the image quality during the interaction and by progressively refining the result as soon as the interaction stops. The simplest method in this context is regular subsampling: rendering the scene in a small viewport during interaction and stretching the resulting image to the target image size using linear interpolation. This method indiscriminately discards features and results in a blurred image or block artifacts.

Adaptive sampling approaches try to assign the computational costs to regions with high image fidelity and to approximate the remaining image parts. Typically, these techniques use features that have been detected in the image plane only. These approaches obviously require the final result as an input for the optimal result, which is impossible. Hence, image regions from previously rendered frames [1] or sparsely sampled regions [2] are used.

We investigate the key element of adaptive approaches, which is the determination of which elements of an object can be coarsened and which must be preserved. Fig. 1 outlines the core idea of this work.

Much perceptually based research has been performed in this area by researchers from different communities (e.g., the Non-Photorealistic Rendering, NPR, community). However, these results have mostly been used for scene stylization and scene enhancement so far. We present a new sampling strategy for ray-based image synthesis, which uses information about the scene, which is known to support the comprehension of three-dimensional shapes [3] and visually attractive images areas in general [4]. In this paper, these techniques are used to control the reduction of ray samples and thus to achieve a higher image quality while maintaining the same level of interactivity.

Our implementation produces a feature buffer for every frame that is efficient enough to be used during the ray generation as a lookup table for the required ray density. We derive a feature priority map from the feature buffer that consists of object space features like silhouettes, suggestive contours, ridges and valleys, combined with image space features like the visual bottom-up saliency information from previously rendered frames. All of them affect the ray density differently. Because different features generate different ray densities, our method is able to support an importance-driven rendering to guarantee the minimum desired frame rate. Even though our main focus is the efficient

* Corresponding authors.
*E-mail addresses:* steinberger@icg.tugraz.at (M. Steinberger), kainz@icg.tugraz.at (B. Kainz), hauswiesner@icg.tugraz.at (S. Hauswiesner), khlebnikov@icg.tugraz.at (R. Khlebnikov), kalkofen@icg.tugraz.at (D. Kalkofen), schmalstieg@icg.tugraz.at (D. Schmalstieg).

0097-8493/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cag.2012.03.037


**Fig. 1.** This figure illustrates the basic idea of our method. Rays do not have to be equally distributed over the scene in ray-based rendering environments to get a visually pleasing result. It is sufficient to trace rays only in areas that have been proven to convey the shape of an object (left) for a good approximation of the result (right).

visualization of volumetric datasets, we also demonstrated a way to apply our method to geometric objects with highly complex materials in ray-traced scenes.

This is an extended version of [5], which newly introduces the use of an image-based visual saliency analysis to capture the impact of lightning effects which go beyond pure geometric features and presents a thoroughly evaluation of our technique using the HDR-VDP-2 [6] visual metric for visibility and quality predictions. The main contributions of our method can be summarized as follows:

- A method that allows the optimization of the ratio between the sampling rate of the scene and its resulting perceptual quality (Section 4).
- A method to calculate the visual saliency information efficiently enough from previous frames, so that this scene information gets applicable for a perceptually guided ray setup (Section 4.1).
- A progressively refineable sampling pattern, which is used to reconstruct sparsely sampled regions of the image (Section 4.4).
- An algorithm that uses our method to guarantee frame rates while maximizing the visual quality within the available time frame (Section 5).
- An evaluation of different object space line features to categorize them based on their abilities to enhance the image quality and a comparison to image space saliency information from previous frames (Section 6).

## 2. Previous work

Previous researchers have been concerned with the real-time performance of ray-based image generation algorithms. Recent work has introduced the exploitation of modern GPUs for solving the brute-force full-resolution ray traversal interactively, while coarse adaptive and progressive sampling approaches have been discussed since ray-tracing algorithms first became available. We give a brief overview of recent GPU methods and adaptive progressive rendering methods in Section 2.1 and discuss possible scene feature computation strategies in Section 2.2. A further overview of the reconstruction techniques for sparsely sampled data is given in Section 2.3, and previous attempts at guaranteeing minimal frame rate are outlined in Section 2.4.

### 2.1. Interactive ray-based rendering

In this section, we briefly summarize the most common approaches to speed-up ray-based rendering algorithms. These approaches can roughly be divided into algorithmic improvements and the exploitation of successively available graphics hardware features.

*Adaptive progressive rendering*. Adaptive approaches, such as the one presented in this paper, aim for the best possible trade-off between interactive frame rates and the loss of image quality, instead of finding the maximum achievable frame rate for a full quality image. Finding this trade-off is still an ill-defined problem, because the perception of quality differs between human beings and between applications. However, several algorithms exist to accelerate rendering speeds through ray reduction. The simplest method is a regular subsampling with a nearest neighbor interpolation. As discussed by [7] and still used in many interactive ray-based rendering systems [8,9], this method is prone to strongly perceivable aliasing artifacts during the interaction. To deal with this problem, most related work has investigated the impact of different sampling pattern strategies in image space [10,2,11]. The sampling pattern is usually visually noticeably refined over time until a desired quality level is reached.

Levoy reformulated the front-to-back image order volume rendering algorithm to use an adaptive termination of ray-tracing [12]. The subdivision and refinement process is based on an $\epsilon$ threshold and does not consider human feature perception and temporal coherence. Later work altered the ray termination criteria [13] depending on the required rendering time or used texture-based level of detail [14], topology guided downsampling [15] or multiple resolutions of the same dataset [16,17].

*Exploiting the GPU*. Numerous rendering engines have been developed to deal with one of the most computationally expensive problems of computer graphics: ray-tracing. Besides CPU-based libraries [18,19], most recent GPU approaches reach remarkable frame rates for low- to medium-complexity scenes for rendering in full quality [20,21]. However, screen filling scenes or scenes with very high complexity are still too slow in order to meet hard real-time constraints. Furthermore, rendering algorithms that aim at achieving real-time performance for the full-quality ray-casting of volume data use empty-space skipping [22], iso-surface ray-casting [23,24], ray pre-integration [25], homogeneous region encoding [26] and many kinds of direct GPU implementations [27, Chapter 39].

### 2.2. Important image areas

The choice of a suitable sampling pattern is crucial for adaptive rendering. For non-trivial systems, the pattern refinement strategy is usually chosen depending on prominent features. In the following paragraphs, we discuss our selected methods to find those regions.

*Image space methods*. Most methods refine the image sampling pattern based on image intensity variances. Early algorithms assume that image areas with high frequencies require a denser sampling than large uniform areas do [28,2], to gain a visually acceptable result. Later systems adapt this assumption towards the limitations of the human visual system. Ramasubramanian and colleagues [29] have been one of the first who have successfully introduced an image-based perceptual threshold map which steers the sampling density of a global illumination path tracing algorithm. The Ramasubramanian system shows that it is possible to generate images with only 5–10% of the rays which have been