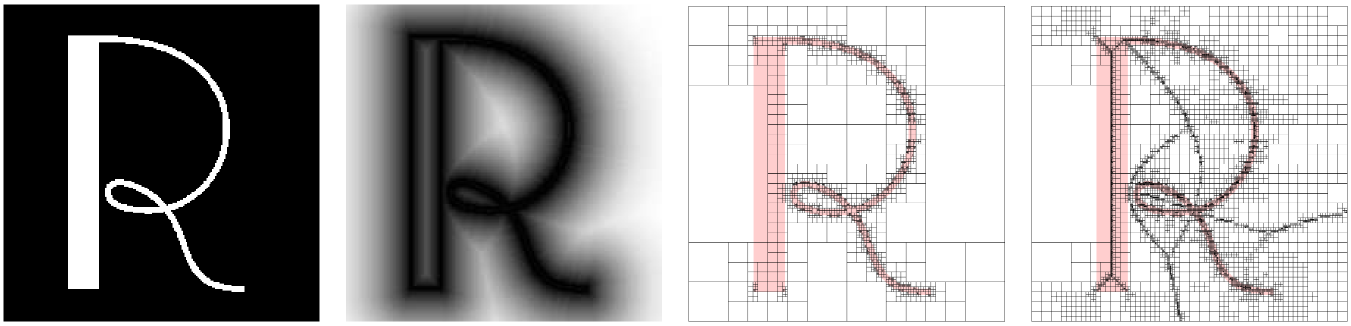
Revisiting Adaptively Sampled Distance Fields

Luiz Henrique de Figueiredo **Luiz Velho**
IMPA – Instituto de Matemática Pura e Aplicada

João Batista de Oliveira
Pontifícia Universidade Católica do Rio Grande do Sul



Introduction

Implicit surfaces are a powerful shape description for many applications in computer graphics. An implicit surface is defined by a function $f\colon \mathbf{R}^3 \to \mathbf{R}$ as the set of points satisfying $f(p) = 0$. Implicit representation becomes more effective when f is a *signed distance function*, i.e., when $|f|$ gives the distance to the closest point on the surface and f is negative inside the object and positive outside the object bounded by the surface.

The distance function to an arbitrary surface does not have a simple analytic description, and we must resort to approximations. One simple solution is to use a volumetric representation, constructed by sampling f uniformly, but such models are very large and their resolution is limited by the sampling rate. Recently, Frisken et al. [1] proposed *adaptively sampled distance fields* (ADFs) as a way to overcome these problems.

In this technical note, we revisit the ADFs and make two contributions to the original framework. First, we analyse the ADF representation and discuss some possible improvements. Second, we show how to compute ADFs more efficiently.

Representation of ADFs

ADFs are represented by a hierarchical spatial decomposition that is recursively subdivided according to an adaptation criteria on the distance function f. This representation allows the reconstruction of f within some prescribed accuracy. In [1], an octree is selected as the spatial data structure to implement ADFs. Samples of f are stored at cell vertices of the octree and trilinear interpolation is used for reconstruction and gradient estimation.

The octree-based ADF was inspired on the *3-color octree*. In a 3-color octree, cells are classified as *interior*, *exterior*, or *boundary*, according to their relation with the solid object being represented; cells that contain the boundary of the object are subdivided up to the finest level. The ADF is similarly defined, except that it uses a different cell subdivision criterion: boundary cells of ADFs are subdivided only when the distance function within a cell is not well approximated by trilinear interpolation of its corner values.

Because boundary cells in an ADF are the only cells tested for subdivision, the distance function f is only guaranteed to be reconstructed within a prescribed accuracy near the surface — interior and exterior cells may give very bad estimates of f. We believe that this is a subtle point which is not properly emphasized in the original ADF paper. To make this point more explicit, we propose to call the original ADF representation the *boundary ADF*.

As shown by Frisken et al., the boundary ADF is sufficient in some applications, such as ray tracing and surface carving, where just local boundary information is needed. On the other hand, the boundary ADF is insufficient for applications that require accurate global information about f, such as offsetting, path planning, and volume rendering.

Another important remark concerns the *skeleton*, or *medial axis*, of the object, which is the set of singular points of the distance function f. The skeleton provides a fundamental shape characterization and is used in many applications, specially in computer vision. It is clear that the boundary ADF cannot correctly capture the skeleton of the object. In fact, it is easy to see that in the cells containing the skeleton, f will not be well approximated by trilinear interpolation.

We propose a variant of the ADF representation that is guaranteed to reconstruct f with the desired accuracy at any point of its domain: simply apply the adaptation criteria to *all* cells of the octree instead of just to the boundary cells. We call this representation *global ADF*.

The figure above shows the "R" shape used by Frisken et al., its signed distance function, the boundary ADF, and the global ADF, represented as quadtrees. Comparing the two representations, we observe that the global ADF reveals all the characteristics of the distance function, capturing the skeleton very well.

We are currently investigating an implementation of the global ADF in which a piecewise linear approximation of the skeleton is explicitly represented in the octree. We hope that this will provide a compact representation that allows a precise reconstruction of f.

Computation of ADFs

There are two basic strategies to generate an ADF: bottom-up or top-down. In the bottom-up approach, first a fully subdivided octree is constructed by regularly sampling the function f; then, in a second step the octree is simplified based on the adaptation criteria. While this approach can take advantage of incremental methods to compute uniform samples of f, it has the drawback of requiring a priori specification of the finest resolution level. It also needs more intermediate storage.

In the top-down approach the octree is recursively subdivided from the root cell to the leaves according to the adaptation criteria. This approach has the advantage of being simple and efficient in time and space. On the other hand, it requires an oracle to implement the adaptation criteria. This oracle must rely on global information about f within a cell. We are currently investigating a fast an robust implementation of such an oracle, using a combination of stochastic sampling and interval techniques.

1 Sarah F. Frisken, Ronald N. Perry, Alyn P. Rockwood, and Thouis R. Jones. Adaptively sampled distance fields: A general representation of shape for computer graphics. *Proceedings of SIGGRAPH 2000*, pages 249–254, July 2000.