Available online at www.sciencedirect.com
ScienceDirect
Computer-Aided Design 39 (2007) 421–426
COMPUTER-AIDED DESIGN
www.elsevier.com/locate/cad

Error-guided adaptive Fourier-based surface reconstruction

Oliver Schall*, Alexander Belyaev, Hans-Peter Seidel

Computer Graphics Group, Max-Planck-Institut für Informatik, Stuhlsatzenhausweg 85, 66123 Saarbrücken, Germany

Received 14 September 2006; accepted 16 February 2007

Abstract

In this paper, we propose to combine Kazhdan's FFT-based approach to surface reconstruction from oriented points with adaptive subdivision and partition of unity blending techniques. This removes the main drawback of the FFT-based approach which is a high memory consumption for geometrically complex datasets. This allows us to achieve a higher reconstruction accuracy compared with the original global approach. Furthermore, our reconstruction process is guided by a global error control accomplished by computing the Hausdorff distance of selected input samples to intermediate reconstructions. The advantages of our surface reconstruction method also include a more robust surface restoration in regions where the surface folds back to itself.

Keywords: Surface reconstruction; Fast Fourier transform (FFT); Oriented point data; Partition of unity

1. Introduction

Many of today's applications make use of 3D models reconstructed from digitized real-world objects such as machine parts, terrain data and cultural heritage. In spite of recent progress in developing speedy and reliable methods for surface reconstruction from scattered data [1–6] (see also references therein), the quest for fast, accurate, and adaptive surface reconstruction techniques capable of processing large and noisy datasets remains a major research issue in computer graphics and geometric modeling areas [7–12].

Recently, Kazhdan introduced a novel and elegant FFT-based reconstruction technique [3]. His approach is able to reconstruct a solid, watertight model from an oriented point set. He approaches the reconstruction problem indirectly by first determining the integral of the characteristic function of the domain bounding the input point set instead of the function itself. Using Stokes' theorem this volume integral can be transformed into a surface integral which is dependent on positions on the boundary of the volume and the corresponding normal directions. As the oriented input point set can be seen as a sampling of this boundary, it can be used to approximate the surface integral and with it the integral of the characteristic function. To finally obtain the characteristic function of the dataset itself, the integration is conducted using the inverse FFT. This method allows a robust and fast reconstruction of a solid and watertight model from noisy samples. On the other hand, the approach has a high memory requirement due to its global nature. The integral of the characteristic function has to be sampled on a uniform grid for the whole volume in order to be able to apply the inverse FFT. This limits the maximal reconstruction resolution of the approach on today's computers to a level where the reconstruction of fine details of the input data is not possible. Furthermore, the approach has no global error control and its globality prevents the accurate reconstruction of regions of the input data which are close to each other but represent disconnected regions of the surface. Our work proposes a simple solution to overcome these limitations while preserving the advantages of the global approach.

The general idea of our technique is to employ an error-guided subdivision of the input data. For this, we compute the bounding box of the input and apply an octree subdivision. In order to decide whether an octree leaf cell needs to be subdivided, we compute a local characteristic function for the points inside the cell using the global FFT-based approach. This is a non-trivial task since the points inside a cell do not, in general, represent a solid. We propose a solution to that problem which avoids creating surface parts which are not represented

* Corresponding author. Tel.: +49 6819325439.
E-mail addresses: schall@mpi-inf.mpg.de (O. Schall), belyaev@mpi-inf.mpg.de (A. Belyaev), hpseidel@mpi-inf.mpg.de (H.-P. Seidel).

0010-4485/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2007.02.005

by points. If the resulting local approximation inside the cell is not accurate enough, the cell needs to be subdivided. By iterating this procedure, we compute overlapping local characteristic functions at the octree leaves for each part of the input with an user-defined accuracy. We obtain the final reconstruction by combining the local approximations using the partition of unity approach and extracting the surface using a polygonization algorithm. One advantage of our adaptive approach is that the characteristic function is only determined close to the surface and not for the whole volume. As the reconstruction accuracy is mainly limited by memory requirements, this allows us to obtain higher reconstruction resolutions. Additionally, the adaptiveness allows us a more accurate reconstruction of strongly bended regions of the input.

The rest of this paper is organized as follows. Section 2 presents details of our surface reconstruction technique. The data partitioning step and the computation of a local characteristic function for each cell is described in Section 2.1. Their integration into a global function and the extraction of the final surface is presented in Section 2.2. We show the results of our technique in Section 3 before we conclude and describe future work in Section 4.

2. Adaptive FFT-based surface reconstruction

In this section, we present our adaptive FFT-based surface reconstruction technique (in the following denoted as AdFFT) in detail. We first describe the error-controlled subdivision of the adaptive octree structure and the computation of overlapping local surface approximations for the input points associated with the octree leaves. We then integrate the local approximations using the partition of unity approach to reconstruct the final model.

2.1. Adaptive octree subdivision

The general idea of the partition of unity approach is to divide the data domain into several pieces and to approximate the data in these domains separately. The resulting local approximations are then blended together using smooth and local weighting functions which sum up to one over the whole domain.

In order to find local characteristic functions of the domain bounding the input point cloud, we first compute the axis-aligned bounding box of the input data. We then apply an adaptive octree subdivision of this bounding box. In order to decide whether a cell needs to be subdivided, we compute the characteristic function of this cell and its vicinity with a fixed accuracy. If the surface extracted from this characteristic function approximates the points in the cell sufficiently close according to a user-defined accuracy, the cell has not to be subdivided further.

How to compute the characteristic function for a cell of the octree is not obvious, as a straightforward application of the global FFT-based method always determines a characteristic function representing a solid, whereas the points in a cell form in general non-closed surface patches. To avoid irrelevant surface parts occurring in the local characteristic function, we use the construction shown in Fig. 1. We embed the octree cell including its oriented input samples at the center of a larger cell with doubled edge lengths. In order to allow a smooth transition between adjacent local characteristic functions later in the integration step, we add points in the vicinity of the original octree cell to the construction. In our implementation, we choose all points in the octree leaf cell scaled by a constant factor c around its center for the computation of the local approximation. If the parameter c is small, few samples in the neighborhood of the octree cell are considered to compute the local approximation. This might cause reconstructions of adjacent octree cells to have no smooth transition across their common boundary. Therefore, it is important that the parameter c is sufficiently large for enough neighboring samples to be considered. According to our experiments, a constant factor of $c = 1.8$ works well for all performed tests.

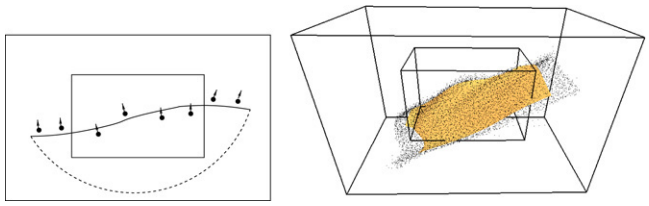

Fig. 1. Left: Local curve approximation for points inside and in the vicinity of a leaf cell (inner rectangle). The dashed line indicates the irrelevant region of the reconstructed solid. Right: Real 3D example of the sketch in the left image after pruning meaningless regions of the solid.

By using the global FFT-based method with a fixed resolution (2^5 in our implementation) on the larger volume, we then compute its characteristic function at regular grid positions. As the shape of the octree cells is not usually cubical, we transform all candidate data points and normals to fit into a cube to enable the use of the FFT. Fig. 1 sketches the idea behind this construction. The surface patch inside and in the vicinity of the octree cell is correctly reconstructed and the irrelevant surface part of the solid is outside of the inner cell. This works in the majority of cases as the irrelevant surface part has a curved shape (see Fig. 1). Additionally, adding sufficient samples in the vicinity of the octree cell increases the diameter of the shape so that the unwanted part does not cross the smaller cell. In rare cases, the crossing cannot be avoided due to very different alignment of octree cell and local surface approximation. But since the resulting unwanted surface parts are small and distant to the real surface, they can be pruned easily during the polygonization. The right image of Fig. 1 shows a real example of a local surface approximation for an octree cell and its vicinity.

To measure the accuracy of the resulting local approximation, we construct a mesh from the computed characteristic function using the Marching Cubes algorithm [13] and compute the Hausdorff distance of selected samples inside the cell to the mesh. If the average computed Hausdorff distance is above the user-defined error, the cell needs to be subdivided further. If a cell is empty, no approximation needs to be computed and we leave it untreated. In order to guarantee an efficient computation of the Hausdorff error, we use only a subset of points inside the