FAKULTÄT FÜR !INFORMATIK

Diplomarbeitspräsentationen der Fakultät für Informatik

Cell-Based Object Representation

Magisterstudium
Computergraphik & Digitale Bildverarbeitung

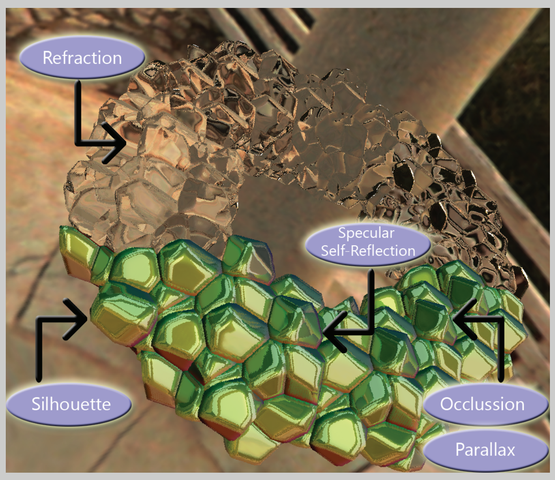
Marcel Nürnberg

Technische Universität Wien
Institut für Computergraphik und Algorithmen
Arbeitsbereich Computergraphik
BetreuerIn: DI Dr. Stefan Jeschke
Associate Prof. DI DI. Dr. Micheal Wimmer

Aim and Scope of the Work

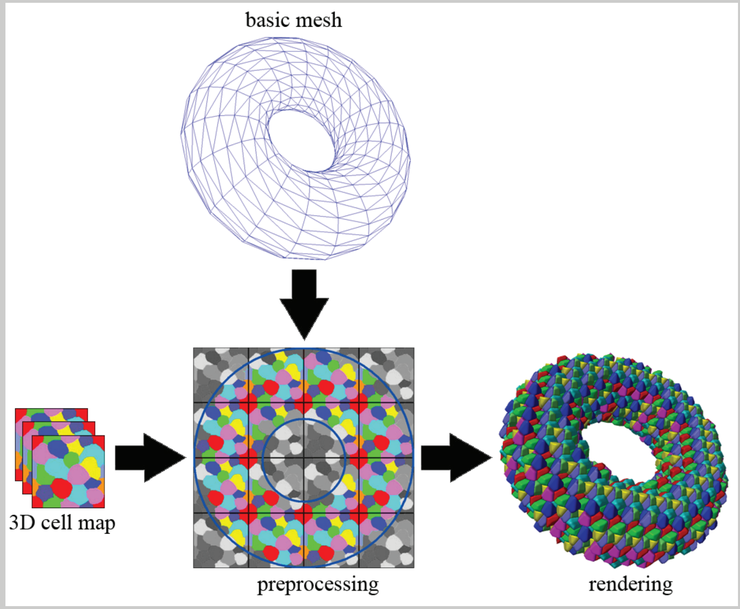
Today's real-time applications, such as computer games or virtual environments, need to display more and more geometrically complex surfaces. Texturing mapping is insufficient to produce such high geometric complexity.

This thesis proposes a cell-based approach to model and render repetitive fine scaled details with a high visual quality providing local parallax, correct occlusions and convincing silhouettes. Since cell-based objects are displayed with a ray tracing sophisticated effects such as specular self-reflection and refraction are also possible to render.



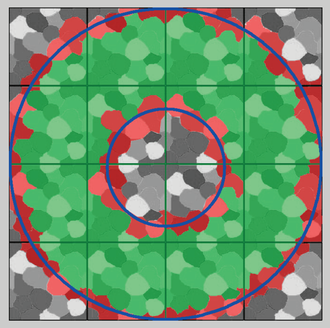
Precomputation

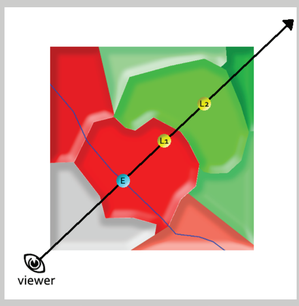
Decomposition of the object into a low frequent geometry (the rough representation called the “basic mesh”) and high frequent surface details (stored in a 3D texture map called the “cell map”). A cell map contains voxel-based components the so-called “cells”. The idea is to tile these cells regularly over the 3D object space. Consequently the set of all inside cells make up the final cell-based object.



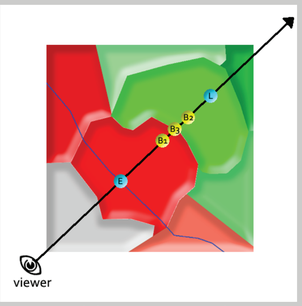
Rendering

The precomputed cell-based objects are displayed with ray tracing. By rendering the basic mesh, the object’s surface is the entry point for the ray tracing algorithm. Rays are shot in eye direction to hit an inside cell’s boundary by performing a linear search with a subsequent binary search.





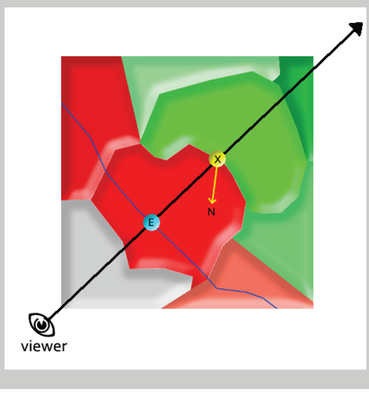
linear search



binary search

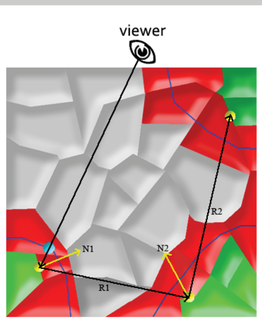
Implementation 1: Silhouettes, Parallax and Self-Occlusions

A first hit rendering is performed to get the inside cell’s boundary to finally calculate a local illumination, such as phong shading.



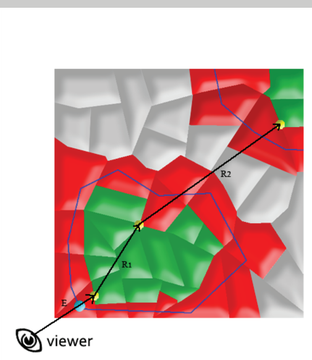
Implementation 2: Specular Self-Reflection

Higher order rays are traversed to calculate specular self-reflection by using the law of reflection.



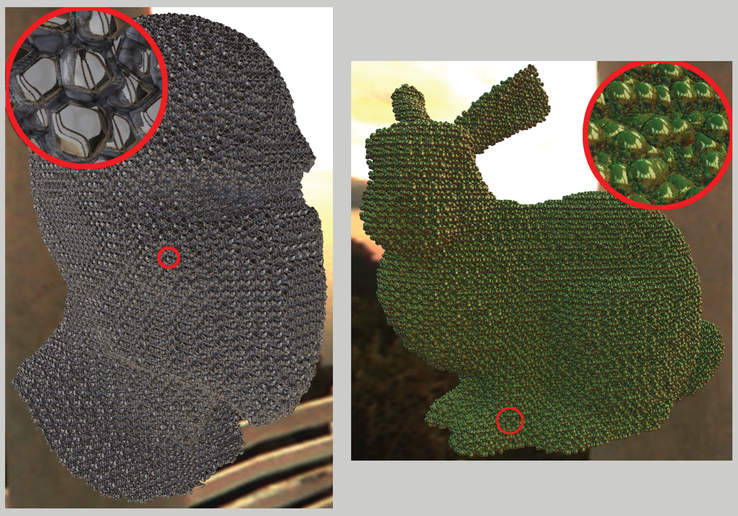
Implementation 3: Refraction

Refraction is calculated by traversing higher order rays based on snell’s law.

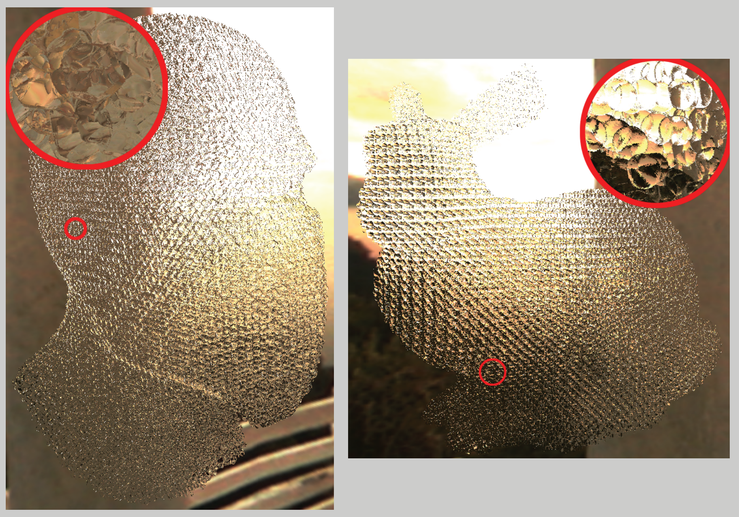


Result 1*:

Result 2*:



Result 3*:



Conclusion

The new cell-based approach in combination with ray tracing opens new ways for rendering high frequent, repetitive, fine scaled details with a high visual quality. Further sophisticated effects such as self-shadowing, caustics, translucency, ambient occlusion and many more can be easily intergrated into the cell-based ray tracer.

*Content: 3D textures taken from http://johanneskopf.de/puplications/solid/textures/index.html

Contact

Marcel Nürnberg (marcel.nuernberg@chello.at)
Stefan Jeschke (jeschke@cg.tuwien.ac.at)
Michael Wimmer (wimmer@cg.tuwien.ac.at)