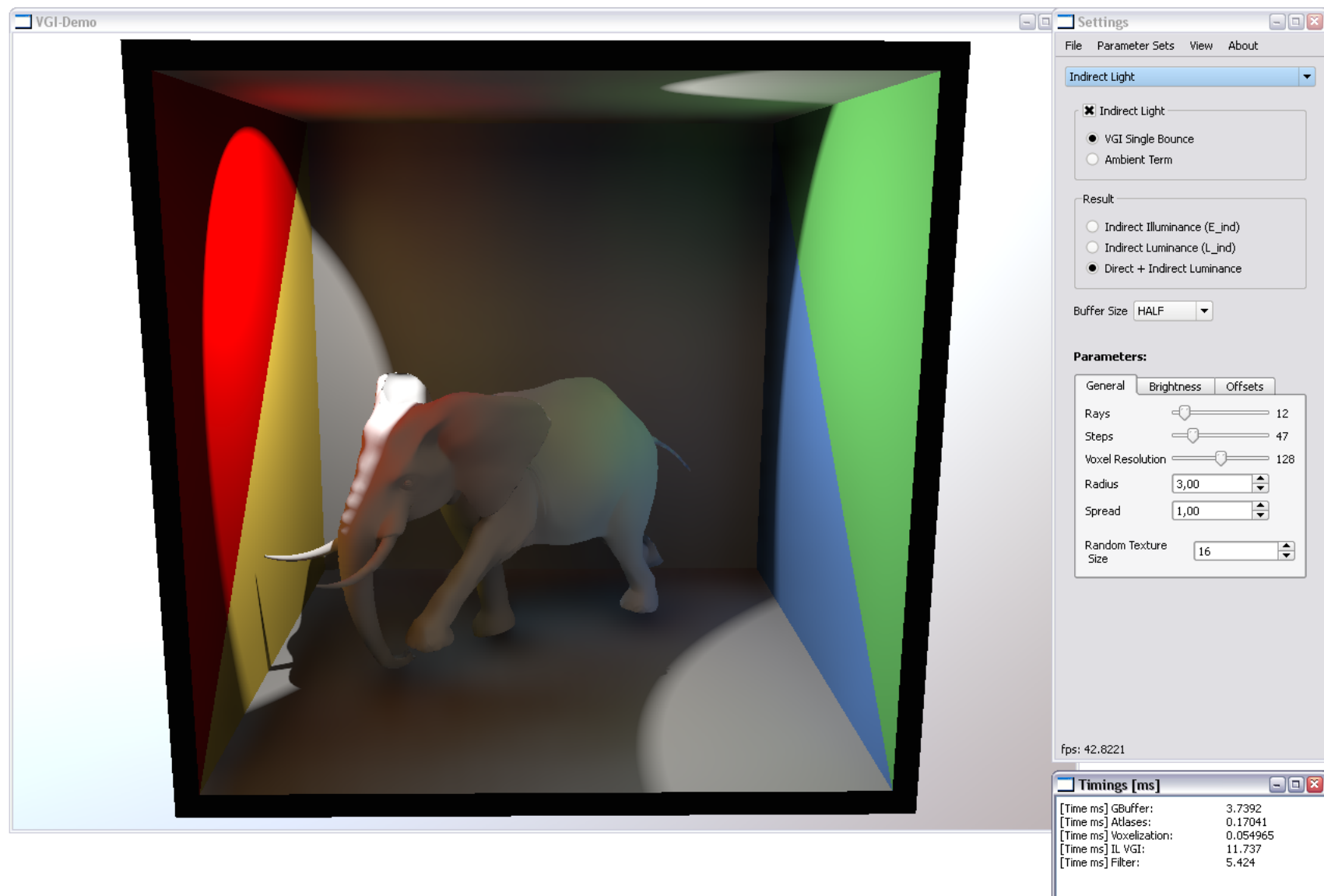# Real-time Near-Field Global Illumination based on a Voxel Model

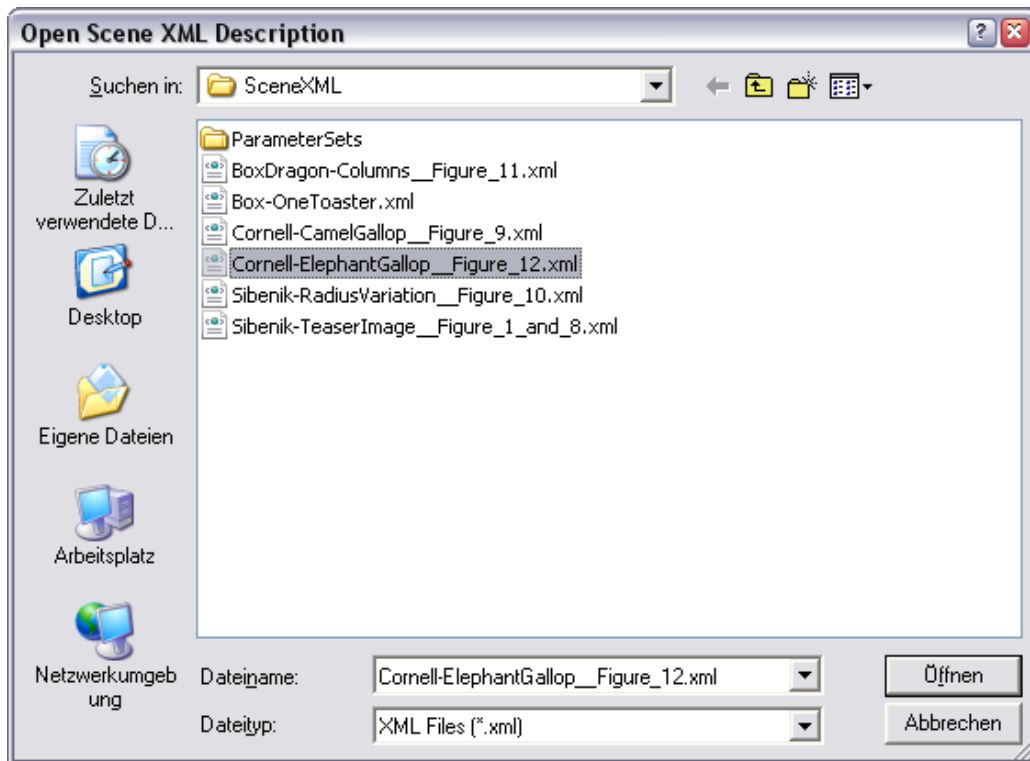Sinje Thiedemann, Niklas Henrich, Thorsten Grosch, Stefan Müller

DEMO

STARTING THE DEMO

Execute **VoxelGI.exe**

Choose a scene file (XML) from the **SceneXML** folder

The program starts and the scene is loaded from obj-files located within the **models** folder.

During startup you can view the **console** for initialization information.

NOTE:

- Models are **animated** via a **sequence of Wavefront-OBJ-files**. After loading all geometry is compiled into **display lists**. During this process (after program start) the frame rate is low. It should then stabilize.
- **Loading a new scene** requires a restart of the application.
- The file name contains the **number of the figure** in the article for which this scene has been used.
- Program has only been tested on **NVIDIA graphics cards**.
- Program has been compiled with Visual Studio 2008 on Windows XP 32 bit.

USAGE
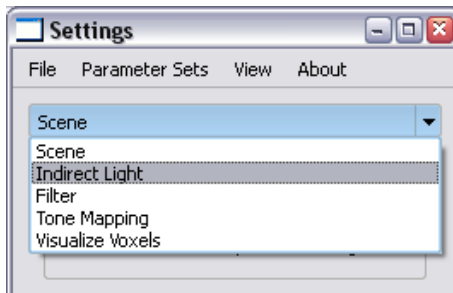
You can **switch between different control panels** with the blue combo box on the top of the **Settings window**.

In the **Indirect Light** panel you can set various parameters that change quality and computation time of the indirect light.

E.g., you can view the contents of the indirect lighting buffer before it is combined with materials and direct light (see screenshot below, Indirect Illuminance). The default is Direct + Indirect Luminance.

You can change the resolution of the indirect light buffer via Buffer Size.

Please refer to the tooltips in the GUI for more information on the parameters inside the tabs at the bottom of the panel.

- In the General Tab you can change the number and the length of rays that are traced through the voxel hierarchy, the maximum number of iteration steps during traversal and the resolution of the binary voxel texture.

- The Brightness Tab contains scale factors for (in)direct light, occlusion and environment map.

- The Offsets Tab controls traversal related offsets (that should prevent self-occlusion of objects) and a threshold for RSM lookup.
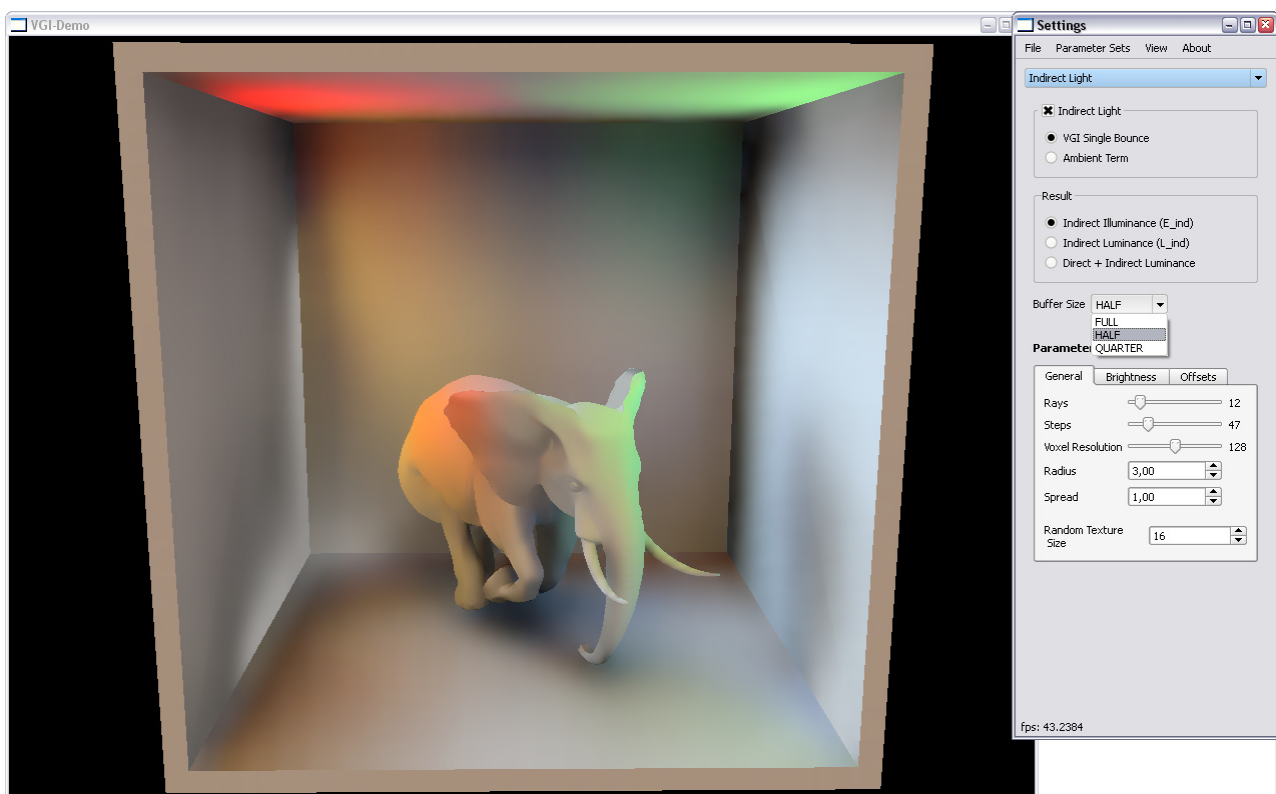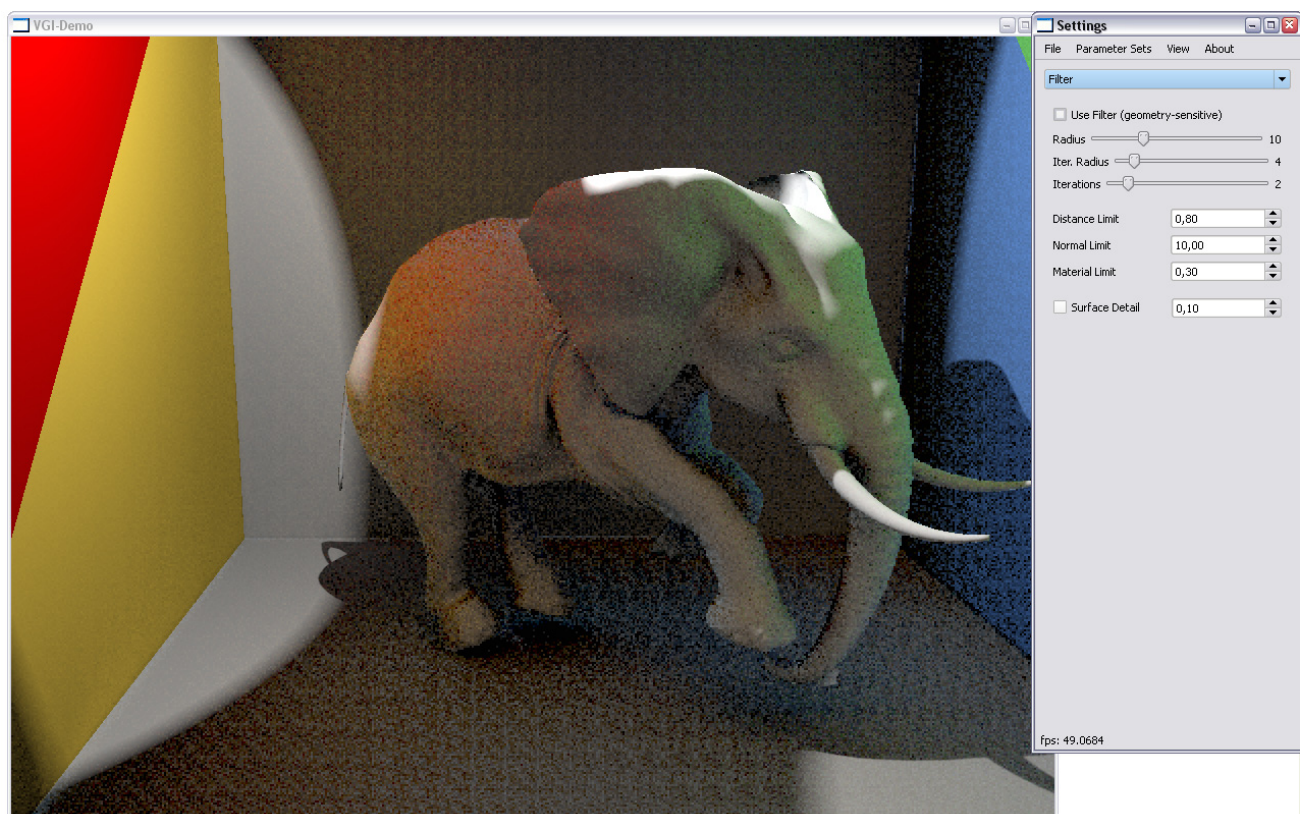
In the **Visualize Voxels** panel you can display the voxelized geometry by clicking the check box "Enable Voxelization" on the top.
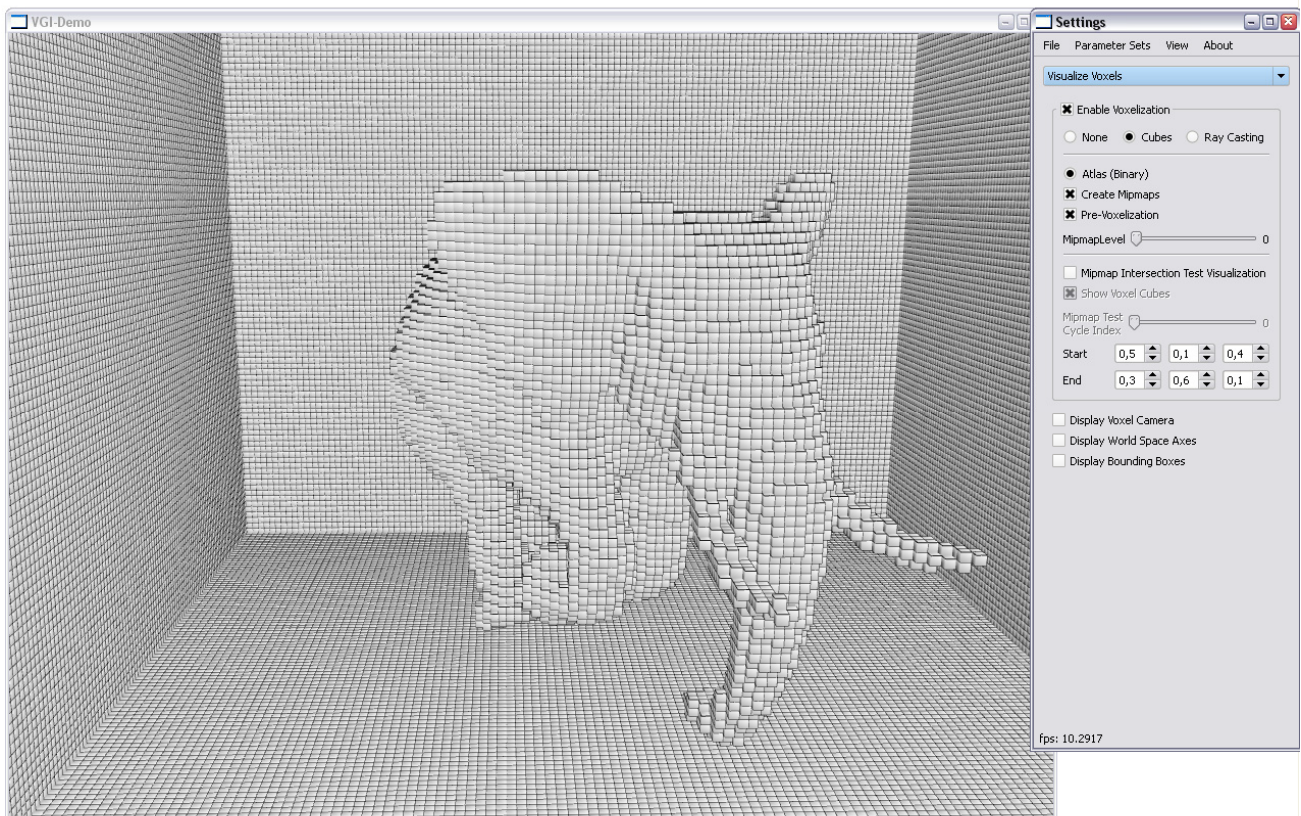
In the **Filter** panel you can de/activate and set parameters for the geometry-sensitive (upsampling) filter for the indirect light buffer.

In the **Scene** panel you can choose whether to move the camera with the mouse (default), move the spot lights or one of the dynamic models.

Also you can change some properties of the camera, the light(s) and start or stop the animation of animated dynamic objects.

The OBJ-file-sequence is played or stopped on pushing the green Animation button.

Camera mouse control:

- Left mouse button: Move camera up/down or left/right
- Middle mouse button: Rotate camera
- Mouse wheel: Move camera forwards/backwards

Dynamic object mouse control:

- Left mouse button: Move object parallel to the ground
- Left mouse button + SHIFT: Move object up/down
- Middle mouse button: Rotate object

Spot light mouse control:

- Left mouse button: Move spot light
- Middle mouse button: Rotate spot light cone

SOURCE CODE

The Demo is implemented in C++ with OpenGL and GLSL.

- Important shader files:
  - *Atlas Rendering:*
    AtlasPosition.vert, AtlasPosition.frag

  - *Atlas Voxelization:*
    AtlasToBinaryVoxels.vert, AtlasToBinaryVoxels.frag
    MipmapVoxelTexture.vert, MipmapVoxelTexture.frag
    CreateVoxelRays.frag

  - *Voxel hierarchy traversal:*
    IntersectionTestMipmap.frag

  - *Back-projection of hit point into RSM and luminance lookup:*
    HitPointSpotLookup.frag

COMPILING

Requirements

- Visual Studio 2008 (Solution file: VoxelGI.sln)
- Libs:
  - Qt, Version 4.6.2 *(not included)*
  - SFML 2 *(included)*
  - OpenGL Mathematics Library *(included)*

Questions? Please write to: thiedemann@uni-koblenz.de