Diplomarbeitspräsentation

FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Coloring Meshes of Archaeological Datasets

Masterstudium:
Visual Computing

Michael Birsak

Technische Universität Wien
Institut für Computergraphik und Algorithmen
Arbeitsbereich Computergraphik
Betreuer: Associate Prof. Dipl.-Ing. Dipl.-Ing. Dr.techn. Michael Wimmer

Scope of the work

This diploma thesis presents a workflow that simplifies the texturing process of a digitized archaeological monument. The digitization is usually done using laser scanners for the geometry and digital single lense reflex cameras (DSLRs) for the color information. In order to enable a continuous mapping of the photos onto the 3D model of the monument, the gathered point cloud is converted into a triangle mesh. The mapping of the photos onto the triangles leads to the following problems that are covered in our workflow:

1) Finding an optimal mapping so that as few seams as possible remain while taking a high-resolution photo for every triangle (Labeling).
2) Adjustment of the textured model in terms of color to get rid of the remaining seams (Leveling).
3) Fine-tuning of the texture data in an image editing application to handle remaining artifacts.
4) Rendering of a 3D model with a huge amount of texture data.

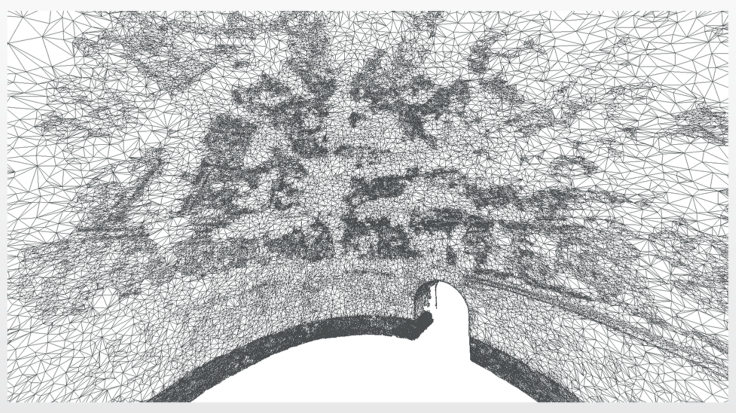
Contributions

We have implemented applications for the following tasks:

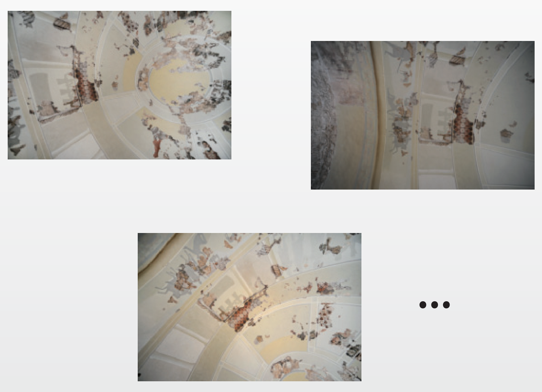
- Labeling: We improve the approach of [Lem07] by the introduction of an accurate geometry-based occlusion detection to prevent texturing of triangles with unsuitable image material.
- Leveling: We improve the approach of [Lem07] by the introduction of a further term into their least squares problem to penalize big leveling function values.
- Masks: We present an application that generates black-and-white masks for the photos. These masks signifantly simplify the manual editing steps.
- Rendering: We use virtual texturing for the visualization and present an application for the fast generation of the needed data structures *atlas* and *tile store*.

Labeling

In the labeling procedure, each triangle gets a photo assigned that is used for texturing. This is done by the minimization of an energy function.



Triangulated 3D model



Registered photos

$$\min\ E(\mathbf{M}) = \sum_{i=1}^{K} w_i^{m_i} + \lambda \sum_{\{F_i,F_j\}\in\mathcal{N}} w_{i,j}^{m_i,m_j}$$

For the occlusion detection, we use an octree to quickly filter out parts of the model that do not come into consideration for an occlusion anyway. This avoids that parts of the model are textured with photos that contain the colors of an occluder.

Labeled 3D model

Leveling

In the leveling procedure, the colors of the labeled model are adjusted to get rid of the seams that result from different lighting situations during the exposure of the photos. This is done by solving a least squares problem.

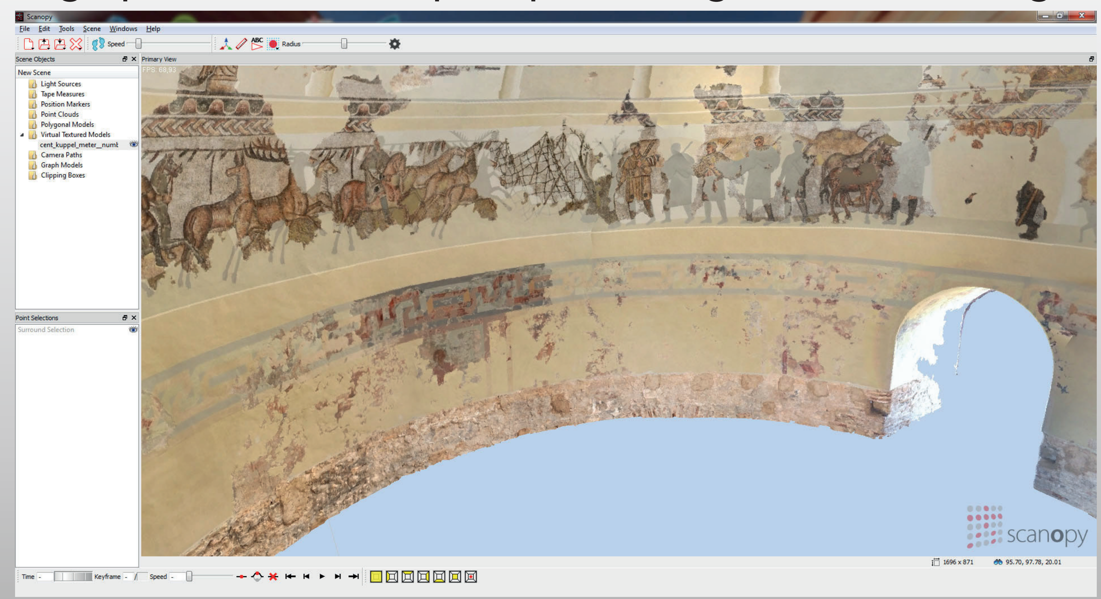
$$\min \sum_{\substack{(i_1,j)\in\mathcal{M}\\(i_2,j)\in\mathcal{M}\\(i_1,i_2)\in\mathcal{L}}} (g_{i_1}^j - g_{i_2}^j)^2 + \lambda \sum_{\substack{(i,j_1)\in\mathcal{M}\\(i,j_2)\in\mathcal{M}}} (g_i^{j_1} - g_i^{j_2} - (f_i^{j_2} - f_i^{j_1}))^2 + \mu \underbrace{\sum_{(i,j)\in\mathcal{M}} (g_i^j)^2}_{\text{our new term}}$$

Our new term helps to keep the leveled color values in the valid range. Without this term, a disadvantageous original texture function can lead to a divergence of the color values and a high loss of contrast.

Leveled 3D model

Results

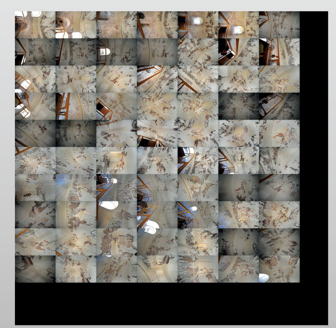
We integrated all the functionalities into Scanopy, an application developed at the Vienna University of Technology and at the Imagination. In future, our implemented tools will assist a graphic artist in the post-processing of an archaeological monument.



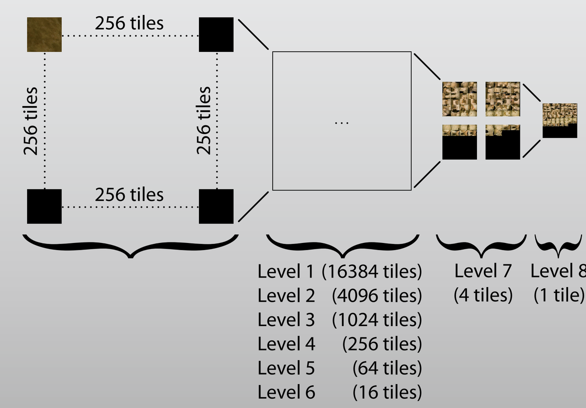
Visualization of the final textured model using Scanopy

Rendering

Our application for the generation of the data structures needed for virtual texturing is at least **4 times faster** than existing Python scripts concerning atlas generation, and at least **10 times faster** concerning tile store generation. Further, we implemented an update function so that the atlas and tile store can be updated when one or more photos have changed.



atlas

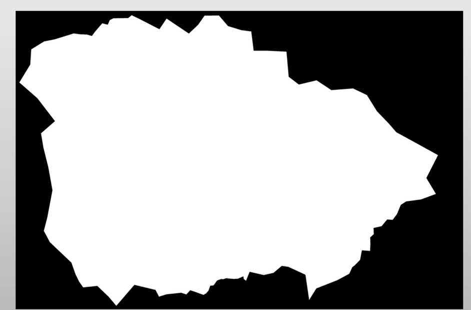


corresponding tile store

Masks

In practice, at least some manual editing steps are needed to get a perfect texturing result. The masks that are produced by our application show where the main focus of these manual editing steps has to be.

photo used for texturing



corresponding mask

References

[Lem07] V. Lempitsky and D. Ivanov. Seamless mosaicing of image-based texture maps. In Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on, pages 1–6, june 2007.

Kontakt: michael.birsak@gmx.at | wimmer@cg.tuwien.ac.at