Seminar topics

Advanced Computer Graphics, 2018/19

1 Introduction

The document describes the seminar topics for the Advanced Computer Graphics course given in 2018/19. The seminar work must be completed during the winter semester and defended prior to the final deadline. Students must defend the seminar at the laboratory exercises and present it at the final presentation session in front of the class.

Upon successful completion, the seminar may contribute up to 30 % to the final grade. It must be graded with more than 15 % for a positive grade. The defense of the seminar after the deadline lowers its maximum contribution: 1-week extension: 22.5 %, 2-week extension 17.5 %. Seminars can also be completed and defended prior to the given deadline.

During the semester students have to present their progress every two weeks at the laboratory exercises.

The expected outputs of each seminar are the seminar report and code repository published on GitHub under the CC-BY or MIT license. The report must be submitted through the FRI Ucilnica. It must be written as a research paper in accordance to the Eurographics LaTeX template.

2 Seminar Topics

2.1 Direct rendering of CSG geometry

Solid models defined with CSG (Constructive Solid Geometry) represent a precise description of a 3D model using Boolean operators on basic geometry primitives. CSG is often used where precise modeling is needed, e.g. for describing the geometry of detector arrays in the Large Hadron Collider in CERN. Geometry defined in such a way can be used in the final fabrication process with the best possible precision. The goal of this seminar is to implement a method for direct rendering of CSG geometry and compare it to the more standard mesh conversion and rendering approaches.

References:

[1] S. Mostajabodaveh, A. Dietrich, T. Gierlinger, F. Michel, A. "*Strok CSG Ray Tracing Revisited: Interactive Rendering of Massive Models Made of Non-planar Higher Order Primitives*", 2017

[2] D. Bogolepov, D. Ulyanov, V. Turlapov "*GPU-optimized Ray-tracing for Constructive Solid Geometry Scenes*", 2016

[3] D.Y. Ulyanov, D.K. Bogolepov, V.E. Turlapov *"Spatially Efficient Tree Layout for GPU Ray-tracing of Constructive Solid Geometry Scenes"*, 2016

[4] F. Romeiro, L. Velho, L. H. de Figueiredo *"Scalable GPU rendering of CSG models"*, 2007 & 2008

2.2 Reconstruction of segmented subcellular structures from volumetric data

In segmentation of volumetric data, individual segmented objects are represented as a set of voxels in 3D space. Such structures can represent an approximate basic shape, which is also the case for some structures in microscopy data of cellular mitochondria. For better reconstruction of 3D shapes, the goal of this seminar is to represent the segmented objects with simple 3D shapes. For a selected type of such object, one must select appropriate limitations and use them in the reconstruction process.

References:

[1] K. V. Kaltdorf, K. Schulze, F. Helmprobst, P. Kollmannsberger, T. Dandekar, C. Stigloher *"FIJI Macro 3D ART VeSElecT: 3D Automated Reconstruction Tool for Vesicle Structures of Electron Tomograms"*, 2017

[2] M. Rouhani, A. D. Sappa, E. Boyer *"Implicit B-Spline Surface Reconstruction"*, 2015.

2.3 Reconstruction of bridges, buildings or mountain overhangs in aerial LiDAR point cloud data

Capturing the world with LiDAR technology gives us an opportunity to get a more detailed insight into the landscape, without the need to walk through vast open spaces or over impassable areas. Since the data are captured from airplanes, we are limited to the representation of upper landscape layers that do not pass through the laser beams. This is the reason that sometimes we do not get a good representation of some parts of the landscape, e.g. bridges, overhangs, building walls or the details underneath the canopy. The goal of this seminar is to select one (or two) types of described objects and reconstruct and repair their representation in the point cloud data. The idea is to add new points to the point cloud which will better define the look of the selected object, e.g. points on the river surface under the bridge, points to outer walls of the buildings, points to overhanging parts of hills and mountains etc. You can use other available geodetic data for obtaining additional information for better reconstruction.

References:

[1] B. Wu, B. Yu, Q. Wu, S. Yao, F. Zhao, W. Mao, J. Wu *"A Graph-Based Approach for 3D Building Model Reconstruction from Airborne LiDAR Point Clouds"*, 2017. https://www.mdpi.com/2072-4292/9/1/92/htm

[2] M. He, Y. Cheng, Y. Nie, L. Qiu, Z. Zhao *"An Algorithm of Reconstructing LiDAR Building Models Based on Isoheight"*, 2017.

[3] M. Kadaa, L. McKinley *"3D Building Reconstruction from Lidar Based on a Cell Decomposition Approach"*, 2017.

[4] R. Cao, Y. Zhang, X. Liu, Z. Zhao *"3D building roof reconstruction from airborne LiDAR point clouds: a framework based on a spatial database"*, 2017.

[5] N. Van Sinh, T. Manh Ha, N. Tien Thanh *"Filling Holes on The Surface of 3D Point Clouds Based on Tangent Plane of Hole Boundary Points"*, 2016.

2.4 Natural neighbor interpolation of images and volumes

Interpolation is a way of reconstructing unknown values inbetween known samples. We can use it to reconstruct or compress images and volumes. A good method for interpolating scattered data (samples don't form a regular grid) is natural neighbor interpolation, which uses an underlying voronoi tessellation to keep the interpolation smooth and local. The goal of the seminar is to write an algorithm that applies the method quickly and incrementally (update the voronoi structure when additional samples are added) on a regular grid of target samples. Ideally, the interpolation would be performed for each target sample in parallel using the GPU.

References:

[1] R. Sibson, *"A brief description of natural neighbor interpolation (Chapter 2)"*, 1981.

[2] S. W. Park, L. Linsen, O. Kreylos, J. D. Owens and B. Hamann, *"Discrete Sibson interpolation"*, 2006.

[3] A. Tsidaev, *"Parallel algorithm for natural neighbor interpolation"*, 2016.

2.5 Visualizing time-based meteorological data on aerial LiDAR point cloud data

The idea of the seminar is to fuse the information from different sources. The first source are LiDAR data with added color information from Orto-photo images and the second source are the meteorological data from ARSO which offers broad range of data for weather stations across the country. The goal is to use meteorological data from the archive (such as snow cover height) and visualize it on LiDAR data. To achieve this the color information and height information of points will have to be adjusted. As a result we want to visualize how the landscape changes over the year in web-based visualization.

References:

[1] J. Kordež, C. Bohak *"Spletno upodabljanje LiDAR podatkov Slovenije z dodano barvno informacijo in senčenjem"*, 2018

[2] ARSO ARHIV - opazovani in merjeni meteorološki podatki po Sloveniji http://meteo.arso.gov.si/met/sl/archive/

2.6 Use of Dual T-Snakes for segmentation of volumetric data

Volumetric datasets consist of individual voxels which can be obtained with use of CT or from slice based data. In both cases, the final volume can be considered as slice-based data. Most of the time we are not interested in all the structures in the volume but only in the specific structures (e.g. in case of electronic microscopy data this can be certain types of structures inside the individual cells or in case of vascular volumetric data, this can be vessels).

The goal of this seminar is to implement the approach *Dual T-Snakes* [1] for the segmentation of volumetric data. More precisely on the case of electronic microscopy datasets for segmentation of cell structures. While the prototype can be developed, and evaluated in Matlab the goal is to develop the final implementation as a Slicer [2] add-on.

References:

[1] Giraldi, G. A., Strauss, E., and Oliveira, A. F., *"Dual-T-snakes model for medical imaging segmentation"*, Pattern Recognit. Lett., Vol. 24, No. 7, pp. 993–1003, 2003.

[2] Slicer. *A multi-platform, free and open source software package for visualization and medical image computing.* URL: https://www.slicer.org/

2.7 Use of T-Surfaces for segmentation of volumetric data

Volumetric datasets consist of individual voxels which can be obtained with use of CT or from slice based data. In both cases, the final volume can be considered as slice-based data. Most of the time we are not interested in all the structures in the volume but only in the specific structures (e.g. in case of electronic microscopy data this can be certain types of structures inside the individual cells or in case of vascular volumetric data, this can be vessels).

The goal of this seminar is to implement the approach *T-Surfaces* [1,2] for the segmentation of volumetric data. More precisely on the case of electronic microscopy datasets for segmentation of cell structures. While the prototype can be developed, and evaluated in Matlab the goal is to develop the final implementation as a Slicer [3] add-on.

References:

[1] Strauss, E., Jimenez, W., Giraldi, G. A., Silva, R., and Oliveira, A. F., *"A semi-automatic surface reconstruction framework based on t-surfaces and isosurface extraction methods"*. In: International Symposium on Computer Graphics, Image Processing and Vision (SIBGRAPI'2002), 2002.

[2] Tian Shen et al., 2011. *"Active volume models for 3D medical image segmentation"*. 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp.707–714.

[3] Slicer. *A multi-platform, free and open source software package for visualization and medical image computing.* URL: https://www.slicer.org/

2.8 Delaunay based Vector segmentation of volumetric data

Volumetric datasets consist of individual voxels which can be obtained with use of CT or from slice based data. In both cases, the final volume can be considered as slice-based data. Most of the time we are not interested in all the structures in the volume but only in the specific structures (e.g. in case of electronic microscopy data this can be certain types of structures inside the individual cells or in case of vascular volumetric data, this can be vessels).

The goal of this seminar is to implement the approach *Delaunay based Vector segmentation* [1] for the segmentation of volumetric data. More precisely on the case of electronic microscopy datasets for segmentation of cell structures. While the prototype can be developed, and evaluated in Matlab the goal is to develop the final implementation as a Slicer [2] add-on.

References:

[1] Spanel, M. et al., 2007. *"Delaunay-based Vector Segmentation of Volumetric Medical Images."* Proceedings of the 12th International Conference on Computer Analysis of Images and Patterns (CAIP'07), Lecture Notes in Computer Science, 4673, pp.261–269.

[2] Slicer. *A multi-platform, free and open source software package for visualization and medical image computing.* URL: https://www.slicer.org/

2.9 Flood simulation visualization

Where does all the water from the rain go? Often, we are wondering how the water from rain flows on the terrain surface. There are many approaches which accurately model the water flow on the surface as well as of the groundwater.

The main goal of this seminar is to implement a simple simulation and visualization of surface water flow on the provided terrain surface. We want to use real terrain data (which can be obtained from ARSO [1] in different formats) and simulate the rain intensity over this area. You may implement the approach described in [2] or suggest a use of the different approach. The simulation can be implemented either in C++ and OpenGL/Vulkan or in JavaScript and WebGL. The user must have an option of choosing own terrain model and the option of changing the rain intensity parameter. The simulation can be calculated offline, but visualization must be real-time.

References:

[1] LiDAR dataset, ARSO. URL: http://gis.arso.gov.si/evode/profile.aspx?id=atlas_voda_Lidar@Arso

[2] Brodtkorb, André R., Martin L. Sætra, and Mustafa Altinakar. *"Efficient shallow water simulations on GPUs: Implementation, visualization, verification, and validation."* Computers & Fluids 55 (2012): 1-12.

2.10 Sound propagation using path tracing

While path tracing is one of the algorithms used in physically based rendering for achieving realistic representations of scenes, it was recently also applied on the sound domain to get the precise simulation of sound propagation in the world. The approach has to be adapted due to the differences between light and sound wave propagation. Such approach is presented in [1]. The goal of this seminar is to implement bidirectional sound transport algorithm for calculation of sound propagation in given scene. In our case the scenes are real world scenes from New York City.

References:

[1] Bidirectional sound transport. URL: https://asa.scitation.org/doi/abs/10.1121/1.4987162

2.11 Special topic [with prior agreement]

If you have in mind a specific topic which is in line with the course syllabus and is not covered in any of above topics you may present your idea to the Professor or Teaching Assistant and discuss with them whether your idea can be defined as a seminar topic for this course. You must contact and discuss your topic prior to the deadline of topic selection and you have to get the agreement for the topic from the Professor or Teaching Assistant.