Master Thesis: GPU-based Computation of Projected Thickness from Various Geometry Models

X-ray imaging is an invaluable method to reveal internal structures of opaque objects because X-ray radiation is able to penetrate them. State-of-the-art X-ray imaging methods are complex tasks and often require sophisticated processing of the acquired data. For this reason, simulation of X-ray imaging experiments is going to be more and more important for modeling complex experimental techniques and evaluation of data analysis algorithms.

X-ray imaging simulation requires computation of projected thickness of a sample, which is then used for simulation of X-ray and matter interaction. In the simulation framework syris [1] developed at Karlsruhe Institute of Technology, there are two sample geometry representation possibilities, first, an analytical model based on metaballs [2], and second, 3D mesh representation. Projected thickness of a sample is computed by finding ray-sample intersections and summing up the distances between intersection pairs which constitute ray entrance and exit from an object. For realistic simulation, a large number of metaballs or polygons in a mesh are required (in order of millions), which makes the computation expensive. Moreover, finite floating point precision needs to be treated carefully in order not to produce artifacts.

Your task is to improve the current syris implementation of both geometry models in terms of numerical stability and efficiency. First, the algorithms should be improved in such a way that they would produce no artifacts and second, suitable data structures and volume-splitting strategies should be used in order to speed up the computations. Moreover, the code will be written in OpenCL [3] and executed on modern GPUs, which will further speed up the computations.

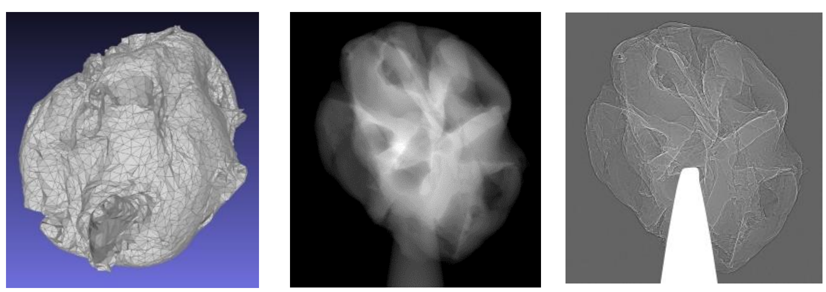

Figure: (left to right) a 3D mesh visualization, its projected thickness and X-ray simulation.

Required skills

- C and Python basics
- Computer graphics basics

[1] Farago, Tomas, et al. "syris: a flexible and efficient framework for X-ray imaging experiments simulation." *Journal of synchrotron radiation* 24.6 (2017).

[2] Blinn, James F. "A generalization of algebraic surface drawing." *ACM transactions on graphics (TOG)* 1.3 (1982): 235-256.

[3] Stone, John E., David Gohara, and Guochun Shi. "OpenCL: A parallel programming standard for heterogeneous computing systems." *Computing in science & engineering* 12.3 (2010): 66-73.

Contact: Tomas Farago, tomas.farago@kit.edu