Project Proposal

Project name	*World’s best 3d printer driver*
Abbreviation	*WB3dPD*
Supervisor	*Tobias Rittig <tobias.rittig@t-online.de>*
Consultants	*Jaroslav Křivánek <Jaroslav.Krivanek@mff.cuni.cz>*
Annotation	*UI wrapper over an existing pipeline for color texture reproduction in 3D printing, which will provide various functionality and visualizations for users.*

Motivation

Research in the Computer Graphics Group at Charles University is seeking to improve the state of the art methods for full-color 3D printing [Elek2017, Sumin 2019]. The aim is to make textured objects appear as close as possible to the user's expectations despite the printing material having significant subsurface scattering properties that blur sharp texture features and reduce the perceived contrast. An optimization algorithm recovers the appearance of printed objects using an iterative prediction+update approach. This algorithm is the only one able to accurately control the printout appearance.

As of now, the optimization is a console application that lacks interactivity with the user. It is also difficult to install since it requires numerous dependencies, which need to be installed manually. For our software project, we would like to create an application that would allow the user to access this cutting-edge research and provide useful features such as printing-table organization and interactive previewing of the results before wasting precious printing material.

Such an application would make it possible for the users to make use of the appearance optimization and allow them to print out an object that would more closely match what they had originally intended.

Project description

Our application would consist of two main parts - the existing optimization framework and a UI wrapper providing different functionality and visualizations for users.

The framework now consists of a mixture of C++ code and Python scripts. Part of our work would be to make it more consistent and define an API which would be as independent of the UI wrapper as possible, allowing future expansion of the application and also usability in a different context (for example in further scientific work).

The UI would allow the user to load a textured 3D object in different commonly used formats (i.e. obj, stl). After that, a basic sanity check would be performed to avoid problems during subsequent processing (holes in models, too thin geometry, etc.). Then the model is handed to the existing optimization pipeline, where it is voxelized using OpenVDB (as the voxels are the required input for the optimization pipeline).

The loaded model would be visualized in different modes, each of them giving different possibilities to the user. The first would be the classic interactive mode, allowing to see the actual shape of the model. The user would be able to place the model on the printing table of the device and adjust the rotation and scaling of the model.

The second mode would be a schematic view of the model voxels, also at an interactive frame rate. This view is combined with tools to see particular layers of voxels or slice through the voxel volume allowing the user to inspect the internal structure and colors assigned to voxels during the optimization process.

Additionally, the user would have the ability to see previews of the resulting object before the actual 3D printing starts, this would be possible before the optimization process is started and also the intermediate results would be visible during the process. The preview is achieved by physical simulation of light transport in the printed object, which is also the technology used in the optimization pipeline.

The output would be the assignment of colors to individual voxels, which is then used by the 3D printer.

Technologies

Frontend: a classic C++ UI framework which will be stated in the detailed specification
Backend: the real time renderers will be using Vulkan (or OpenGL); C++ base using libraries already used in the project (OpenVDB, Blosc, …) and others

Expected effort / Roadmap

Number of participants: 5 (+ 1 consultant), each working +-10 hours/week
Completion date: 9 months since the start of the project
Roadmap:

- Milestone 1 (months 1 and 2) - detailed specification
  - research of technologies, detailed plan of the functionality and its implementation
  - writing of the detailed specification
- Milestone 2 (months 3 - 6) - first working version of the application
  - working UI, rendering systems and other main points of the specification, gradually adding features and expanding connectivity to the backend
- Milestone 3 (months 7 - 9) - finalization, bug fixing, polishing
- Milestone 4 (month 9) - writing of the documentation

Team

Tobias Rittig (supervisor)
Jaroslav Křivánek (consultant)
Bohuš Brečka
Matúš Goliaš
Qingqin Hua
Vojtěch Kužel
Vojtěch Tázlar

Project Scope

Discrete models and algorithms	
	discrete mathematics and algorithms
	geometry and mathematics structures in computer science
	optimizations
Theoretical computer science	
	theoretical computer science
Software and data engineering	
x	software engineering
x	software development
	web engineering
	database systems
x	analysis and processing of large data sets
Software systems	
	system programming
	reliable systems
x	performance systems
Mathematical linguistic	
	computer and formal linguistic
	statistical methods and machine learning in computer linguistic
Artificial intelligence	
	intelligent agents

	machine learning
	robotics
Computer graphics and computer games development	
x	computer graphics
	computer games development