GSoC 2020 Proposal

Organization : Python Software Foundation

Sub-organization : Fury

FURY: Improve Shader Framework

ABOUT ME

Name: Lenix Lobo
College : (Pune Institute of Computer Technology) University of Pune, SPPU (Pune, India)
 Major : Information Technology
Expected Graduation 2020
Github: https://github.com/lenixlobo
Time Zone : IST
Portfolio : http://lenixlobo.github.io/

I am a fourth year student pursuing B. Engineering at the Pune institute of computer technology, India. I am interested in computer graphics, specifically realistic rendering and real time immersive visualization .

Experience:

I have worked as an intern at the Center for Development of Advanced Computing (CDAC), Pune wherein i was assigned the task of distributing the existing code base on a Nvidia-GPU using CUDA.

Throughout my undergraduate phase I have worked on multiple projects closely related to computer graphics. Some of these i have mentioned below,

O-Nect : O-Nect is an open source implementation of real time human pose estimation algorithms to create a cost effective motion capture setup using blender .

Shadie : Shadie is a OpenGL playground where I implemented model importing, and shading algorithms using GLSL.

YAPT: YAPT is a ray tracer built using C++.

Why Open source and Why Fury?

I came across FURY before the organizations were announced while looking for visualization libraries in python and immediately was interested in its capabilities. An opportunity to contribute would help me better understand the underlying technologies used in visualization and also help a very easy yet powerful library for python developers.

What work have you done so far ?

Over the past weeks I have gotten familiar with VTK Python which is the underlying library on top of which Fury is built.I have also made a few contributions which as mentioned below. Having worked with OpenGL before, learning VTK has been an easy transition so far. I have worked with shaders before and look forward to utilizing my previous knowledge to further extend the rendering capabilities of FURY.

I have attached screenshots of some of my work built using **VTK shaders**

1. Simulating virus using VTK shader:

    The project uses a slightly modified version of perlin noise to manipulate the vertices of the sphere based on normals on the sphere. The original idea was based on Nvidia vertex noise implementation using GLSL.

    http://developer.download.nvidia.com/SDK/9.5/Samples/samples.html#glsl_vnoise
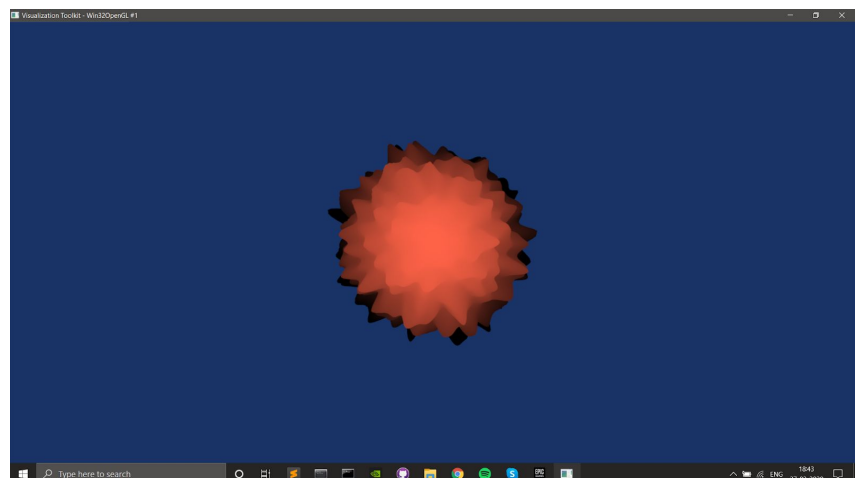
2. (Committed to repository)Tutorial : Importing Models and using shaders

The tutorial is designed for users to get started with shaders and importing custom models using fury. The shaders includes templates of both vertex and fragment VKT shaders with easy to understand layout.
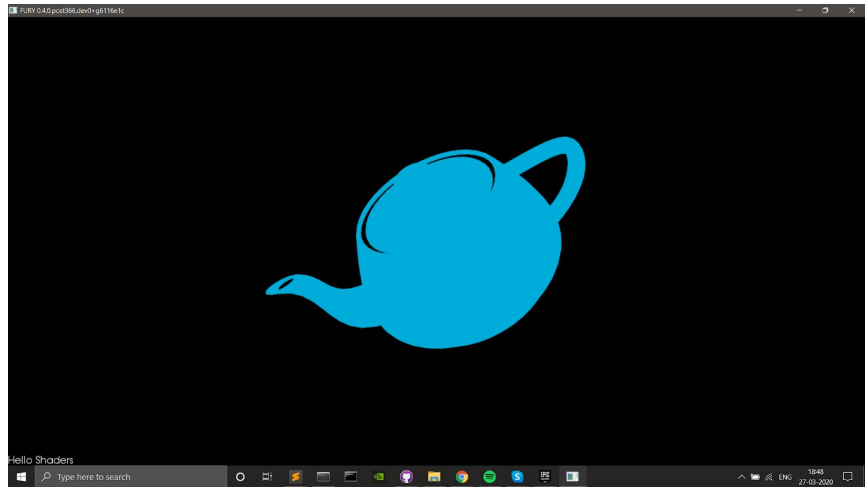
Programming languages:

I have previously worked with include C/C++, Python, OpenGL, CUDA, Assembly Language Programming, etc. I have built fury on my desktop and also made a few contributions mentioned below.

Hardware Access:

I have a Windows PC with intel i7-7700HQ, Nvidia GTX 1050Ti and 16 GB ddr4 memory. I also dual boot Linux (ubuntu) .

OVERVIEW

The current shader framework for fury is based on VTK and lacks documentation to get started which can be overwhelming for new users . The objective of this project is to enable users to be easily able to understand and use the shader framework to render stunning visual representations of data. The project involves programming vertex and fragment shaders to generate effects for more immersive visualization . Improving the animation framework can help users easily develop interactive visualizations. Apart from adding shader support and improving the animation framework, one crucial aspect is ensuring that the documentation is well defined for users to get started with interactive visualization. Along with that, adding new test cases, updating and migrating old test cases, keeping track of which issues and PRs have to be addressed is another important part of this project.

CODE CONTRIBUTIONS

Added a basic shader tutorial for new users to get started with importing model and using Shaders in Fury:

https://github.com/fury-gl/fury/pull/154

Replaced the md5 hash function with sha256 and added the new keys for the models:

https://github.com/fury-gl/fury/pull/148

Added the Utah teapot and blender Suzanne model to the fury-data repository and added the same in fury fetcher:

https://github.com/fury-gl/fury-data/pull/2

https://github.com/fury-gl/fury-data/pull/3

https://github.com/fury-gl/fury/pull/207

GOALS

Improve shader framework to facilitate the utilization by the community

Add SDFs(ray marching) based rendering capabilities to support better volumetric renderings.

Integrate PBR(Physically Based Rendering) capabilities within FURY to render realistic renderings.

Integrate texture support

TIMELINE

Week 1:

Get familiarized with the current shader framework (based on VTK) and their calls. Try to figure out what the final product should look like.

Build simple shader examples/animations based on the current framework and discuss possible improvements.

Week 2:

Discuss improvements suggested by the mentor and the community. Begin implementation of improvements .

Week 3:

Understand Shader calls specified by VTK and extend the current animation framework to add new visualizations and effects. Provide an easy framework to integrate animations in FURY.

Week 5-6:

Implement a Ray Marcher for multiple shapes using SDFs.

Week 7-8:

Refer VTK master branch to integrate Physically based Rendering capabilities within FURY to generate realistic renderings while ensuring easy usability .

Week 9-10:

Implement multiple demos showcasing complex animations with the newly added animation framework improvements.

Week 11-12:

Discuss any improvement / requirements asked by the community and work on the suggestions to accomodate more users. Work on documentation , and guides for news users to get started.

Week 13:

Additional week allocated to complete if any work is delayed.

COMMITMENTS AND AVAILABILITY

Have you applied to any other organization?

No.

Do you have any other commitments during the main GSoC time period?

During the entire GSOC period I will be available for regular developments except when my examinations are scheduled. If I am selected I will be able to contribute 40 hours a week, though I will try my best to put in more effort if the work required.

Do you have exams or classes that overlap with this period?

I have one exam scheduled during the GSoC period. The examination will be scheduled within the month of May and will not take more than a week to get done with. However, I will try my best to keep in touch with the project by maintaining frequent communication with the community during this phase.