Model Scalar Engine

Developer: Afraz Butt

January 8, 2023

Contents

1 Background

A software renderer that takes an object file and renders it into three dimensional space. A basic walkthrough of concepts of linear Algebra is required to understand the project.

1.1 System Requirements

- Programming Language Used : Python
- Libraries required:
    1. Pygame v 2.1.2
    2. Numpy
    3. Numba Just In Time Compiler

2 Three Dimensional Planes and Homogenous System

A typical three dimensional co-ordinate system has planes in three *axes*, i.e, x, y , z . To render it on a console window, we have to *skew* it into two dimensions. This is accomplished by paradoxically *adding* a fourth dimension into the three dimensional Coordinate.

$$normal = (x, y, z), homogenoussystem = (x, y, z, w)$$

where w has value always equal to 1

3 Matrix Operations

Matrices have some elementary operations associated with them. In the Gaussian and Euclidian Spaces, these take the form of

1. Translation
2. Scaling
3. Rotation

3.1 Translation

A translation matrix simply moves an object along with one or more of the three axes. A transformation matrix representing only translations has a simple form. If we have to translate a point P (x, y, z) by T_x on the X axis, T_y on the Y axis and T_z on the Z axis.

It is given by:

$$\begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 1 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x + T_x \\ y + Ty \\ z + Tz \\ 1 \end{pmatrix}$$

To shorten this process, we insert a dummy coordinate into the matrix. Hence the need for homogenous co-ordinate system.

3.2 Scaling

A scaling matrix *skews* a given matrix by a specified ratio, so that the output matrix is an exact mirror of the input one, but contracted in size.

$$\begin{bmatrix} S_1 & 0 & 0 & 0 \\ 0 & S_2 & 0 & 0 \\ 1 & 0 & S_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x.S_1 \\ y.S_2 \\ z.S_3 \\ 1 \end{pmatrix}$$

3.3 Rotation

A Rotation matrix rotates the original matrix in a given angle over a given co-ordinate.

4 Camera Space and Projections

The new and far frustum planes provide an effective way to manage the spaces of camera. Basically, a projection *clips* the incoming lines into the planes. More details can be found on this link:
Frustum details.

5 OBJ Files

An OBJ file is a standard 3D image format that can be exported and opened by various 3D image editing programs. It contains a three-dimensional object, which includes 3D coordinates, texture maps, polygonal faces, and other object information.

- It enables users to represent complex or irregularly shaped objects by dividing their surface into small, triangular "tiles." This tessellation process makes it easier to manipulate and render the design since you can modify each tile separately from the rest.
- Another critical feature of OBJ files is their ability to specify the geometry of 3D objects and their surface properties, including texture mapping and shading. This versatility makes the OBJ file format robust for creating realistic renderings of complex three-dimensional scenes.
- OBJ files also contain information on free-form surface patches. These patches allow designers to create smooth surfaces free from distortions or seams, making them ideal for creating highly realistic textures such as skin or fabric.

6 Numpy

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently. This behavior is called locality of reference in computer science. This is the main reason why NumPy is faster than lists. Also it is optimized to work with latest CPU architectures.

Numpy documentation can be found on the website: Numpy Documentation

7 Why I Used Numba: A just in time Compiler

Numba is a powerful JIT(Just-In-Time) compiler used to accelerate the speed of large numerical calculations in Python. It uses the industry-standard LLVM library to compile the machine code at runtime for optimization. Numba enables certain numerical algorithms in Python to reach the speed of compiled languages like C or FORTRAN. It is an easy-to-use compiler that has several advantages such as:

1. **Optimizing scientific code** – Numba can be used along with NumPy to optimize the performance of mathematical calculations. For different types of numerical algorithms, arrays and layouts used, Numba generates specially optimized code for better performance.
2. **Parallelization** – Numba can be used for running NumPy on multiple cores and to write parallel GPU algorithms in Python. Python is used across a variety of disciplines such as Machine Learning, Artificial Intelligence, Data Science, etc., and across various industries such as finance, healthcare, etc. Using large data sets is the norm in such disciplines and Numba can help address the slow runtime speed due to the interpreted nature of Python.