УДК 004.92

ПЕРСПЕКТИВНЫЕ МАТРИЦЫ В API И ИХ РЕАЛИЗАЦИЯ ПОСРЕДСТВОМ СРЕДЫ OPENGL

Казимирчик М.А., Крячев Е.В.

Белорусский государственный университет информатики и радиоэлектроники, г. Минск, Республика Беларусь

Научный руководитель: Ролич О.Ч. – канд. техн. наук, доцент, доцент кафедры ПИКС

Аннотация. Математически обосновываются перспективные матрицы, а также рассматривается их работа и реализация в среде *OpenGL*.

Ключевые слова: перспективные матрицы, компьютерная графика, программирование

Введение. Компьютерная графика является одним из наиболее перспективных и популярных направлений в современном мире. Для эффективного использования данных отображений в задачах компьютерной графики более удобной и быстрой является матричная запись. В свою очередь использование перспективных матриц в *3D* графике позволяет реализовать модель видимости человеческого глаза.

Основная часть. Компьютерный монитор представляет собой двухмерную поверхность. Трехмерная сцена, созданная с помощью *OpenGL*, должна проецироваться на экран компьютера как двухмерное изображение. Для этого преобразования проекции используется перспективная матрица. Во-первых, он преобразует все данные вершин из координат глаза в координаты отсечения. Затем эти координаты клипа также преобразуются в нормализованные координаты устройства путем деления на компонент w координат клипа.

В перспективной проекции трехмерная точка в усеченной пирамиде (координаты глаза) отображается в куб; диапазон координат x от $[l, r]$ до [-1, 1], координаты y от $[b, t]$ до [-1, 1] и координаты z от $[-n, -f]$ до [-1, 1].

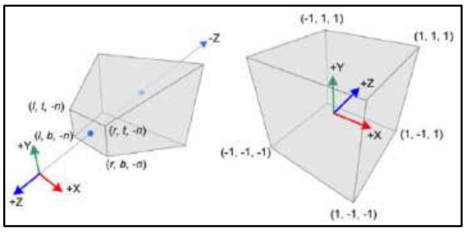

Рисунок *1* – Усеченная перспектива и нормализованные координаты устройства

В *OpenGL* трехмерная точка в пространстве глаза проецируется на ближнюю плоскость (плоскость проекции). На следующих диаграммах показано, как точка (x_e, y_e, z_e) в пространстве глаза проецируется на (x_p, y_p, z_p) на ближней плоскости.

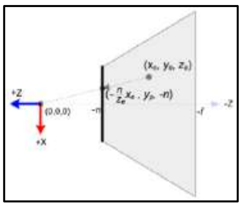

Рисунок *2* – Вид сверху на усеченный бассейн

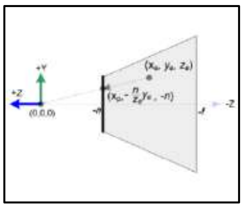

Рисунок *3* – Вид сбоку на усеченный конус

На виде сверху на усеченный конус координата x пространства глаза x_e отображается в x_p, который рассчитывается с использованием отношения подобных треугольников;

$$\frac{x_p}{x_e} = \frac{-n}{z_e},$$
$$x_p = \frac{-n \cdot x_e}{z_e} = \frac{n \cdot x_e}{-z_e}.$$

Со стороны усеченного конуса y_p также рассчитывается аналогичным образом;

$$\frac{y_p}{y_e} = \frac{-n}{z_e},$$
$$y_p = \frac{-n \cdot y_e}{z_e} = \frac{n \cdot y_e}{-z_e}.$$

После преобразования координат глаза путем умножения перспективной матрицы, координаты клипа по-прежнему являются однородными координатами. Следовательно, теперь координаты клипа становятся нормализованными координатами устройства путем деления на w-компонент координат клипа.

$$\begin{pmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} \end{pmatrix} = M_{projection} \cdot \begin{pmatrix} x_{eye} \\ y_{eye} \\ z_{eye} \\ w_{eye} \end{pmatrix}, \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_{clip}/w_{clip} \\ y_{clip}/w_{clip} \\ z_{clip}/w_{clip} \end{pmatrix}.$$

Следовательно, можно установить w-компонент координат клипа как $-z_e$. И 4-я строка матрицы становится равной (0, 0, -1, 0).

$$\begin{pmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} \end{pmatrix} = \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_{eye} \\ y_{eye} \\ z_{eye} \\ w_{eye} \end{pmatrix}, \quad w_c = -z_e$$

Затем сопоставим x_p и y_p с x_n и y_n устройства с линейной зависимостью:
$[l, r] \Rightarrow [-1, 1]$ и $[b, t] \Rightarrow [-1, 1]$.

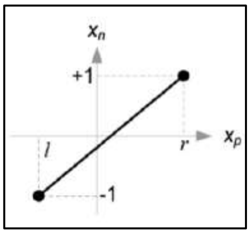

Рисунок 4 – Отображение от x_p до x_n

$$x_n = \frac{1-(-1)}{r-l} \cdot x_p + \beta,$$
$$1 = \frac{2r}{r-l} + \beta,$$
$$\beta = 1 - \frac{2r}{r-l} = \frac{r-l-2r}{r-l} = -\frac{r+l}{r-l},$$

$$x_n = \frac{2x_p}{r-l} - \frac{r+l}{r-l}.$$

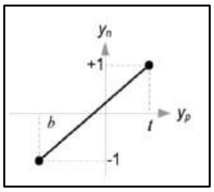

Рисунок 5 – Отображение от y_p до y_n

$$y_n = \frac{1-(-1)}{t-b} \cdot y_p + \beta,$$

$$1 = \frac{2t}{t-b} + \beta,$$

$$\beta = 1 - \frac{2t}{t-b} = \frac{t-b-2t}{t-b} = -\frac{t+b}{t-b},$$

$$y_n = \frac{2y_p}{t-b} - \frac{t+b}{t-b}.$$

Затем подставим x_p и y_p, которые найдены ранее:

$$x_n = \frac{2x_p}{r-l} - \frac{r+l}{r-l} = \frac{2 \cdot \frac{n \cdot x_e}{-z_e}}{r-l} - \frac{r+l}{r-l} = \frac{2n \cdot x_e}{(r-l)(-z_e)} - \frac{r+l}{r-l} = \frac{\frac{2n}{r-l} \cdot x_e}{-z_e} + \frac{\frac{r+l}{r-l} \cdot z_e}{-z_e}$$

$$= \left(\frac{2n}{r-l} \cdot x_e + \frac{r+l}{r-l} \cdot z_e\right) / -z_e, \text{ где } \frac{2n}{r-l} \cdot x_e + \frac{r+l}{r-l} \cdot z_e = x_c.$$

$$y_n = \frac{2y_p}{t-b} - \frac{t+b}{t-b} = \frac{2 \cdot \frac{n \cdot y_e}{-z_e}}{t-b} - \frac{t+b}{t-b} = \frac{2n \cdot y_e}{(t-b)(-z_e)} - \frac{t+b}{t-b} = \frac{\frac{2n}{t-b} \cdot y_e}{-z_e} + \frac{\frac{t+b}{t-b} \cdot z_e}{-z_e}$$

$$= \left(\frac{2n}{t-b} \cdot y_e + \frac{t+b}{t-b} \cdot z_e\right) / -z_e, \text{ где } \frac{2n}{t-b} \cdot y_e + \frac{t+b}{t-b} \cdot z_e = y_c.$$

$$\begin{pmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} \end{pmatrix} = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_{eye} \\ y_{eye} \\ z_{eye} \\ w_{eye} \end{pmatrix}$$

Теперь требуется решить только 3-ю строку матрицы. Нахождение z_n отличается от других, потому что z_e в пространстве глаза всегда проецируется на $-n$ на ближней плоскости. Поскольку известно, что z не зависит от значения x или y, используется w-компонента, чтобы найти взаимосвязь между z_n и z_e. Следовательно, можно указать 3-ю строку матрицы следующим образом:

$$z_n = \frac{z_c}{w_c} = \frac{Az_e + Bw_e}{-z_e}.$$

В глазном пространстве w_e равно 1. Следовательно, уравнение принимает вид:

$$z_n = \frac{Az_e + B}{-z_e}.$$

Чтобы найти коэффициенты A и B, используется соотношение (z_e, z_n), (-n, -1) и (-f, 1):

$$\begin{cases} \dfrac{-An + B}{n} = -1 \\ \dfrac{-Af + B}{f} = 1 \end{cases},$$

$$\begin{cases} -An + B = -n \\ -Af + B = f \end{cases},$$

$$B = An - n,$$

$$-Af + (An - n) = f,$$

$$-(f - n)A = f + n,$$

$$A = -\frac{f + n}{f - n},$$

$$\left(\frac{f + n}{f - n}\right) n + B = -n,$$

$$B = -n - \left(\frac{f + n}{f - n}\right) n = -\left(1 + \frac{f + n}{f - n}\right) n = -\left(\frac{f - n + f + n}{f - n}\right) n = -\frac{2fn}{f - n},$$

$$z_n = \frac{-\dfrac{f + n}{f - n} z_e - \dfrac{2fn}{f - n}}{-z_e}.$$

Полная матрица проекции:

$$\begin{pmatrix} \dfrac{2n}{r - l} & 0 & \dfrac{r + l}{r - l} & 0 \\ 0 & \dfrac{2n}{t - b} & \dfrac{t + b}{t - b} & 0 \\ 0 & 0 & -\dfrac{f + n}{f - n} & -\dfrac{2fn}{f - n} \\ 0 & 0 & -1 & 0 \end{pmatrix}.$$

Далее, опираясь на математическое обоснование и используя библиотеку *GLUT* в *C++* реализуется изображение *3D* куба, координаты которого заданы перспективной матрицей.

```
Matrix4f Projection(f,    0.0f, 0.0f, 0.0f,
                    0.0f, f,    0.0f, 0.0f,
                    0.0f, 0.0f, 1.0f, 0.0f,
                    0.0f, 0.0f, 1.0f, 0.0f);
```

Рисунок 6 – Параметры куба для глубины зрения изображения 1

```
Matrix4f Projection(d,    0.0f, 0.0f, 0.0f,
                    0.0f, d,    0.0f, 0.0f,
                    0.0f, 0.0f, 1.0f, 0.0f,
                    0.0f, 0.0f, 1.0f, 0.0f);
```

Рисунок 7 – Параметры куба для глубины зрения изображения 2

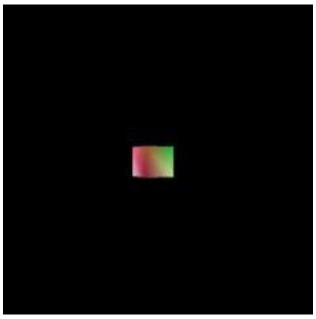

Рисунок 8 – Результат выполнение кода 1

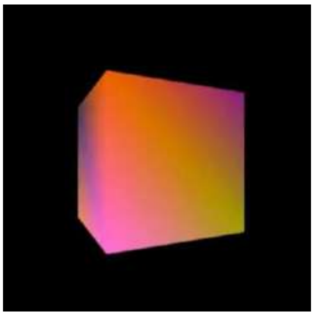

Рисунок 9 – Результат выполнение кода 2

Заключение. Создавая *3D* изображения в компьютерной графике, для реализации реалистичных размеров изображений в зависимости от расстояния до изображения, используются перспективные проекции данных изображений, которые описываются перспективными матрицами. В свою очередь рассматриваемый *Opengl* имеет в свободном доступе большое количество библиотек для их программной реализации.

Список литературы

1 *Segal M., Akeley K. The OpenGL Graphics System: A Specification (Version 3.3 (Core Profile) — March 11, 2010). – Режим доступа: https://www.opengl.org/registry/doc/glspec33.core.20100311.withchanges.pdf– Дата доступа: 02.04.2022.*

2. *Lindeman R.W. CS 543 — Computer Graphics: Projection. – Режим доступа: https://inlnk.ru/G6Ywdm – Дата доступа : 01.04.2022.*

3. *Song H.A. OpenGL Projection Matrix – Режим доступа: http://www.songho.ca/opengl/gl_projectionmatrix.html – Дата доступа: 01.04.2022.*

UDC 004.92

PERSPECTIVE PROJECTION MATRICES IN AN API AND THEIR IMPLEMENTATION THROUGH THE OPENGL ENVIRONMENT

Kazimirchik M.A., Kryachev E.V.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Rolich O.Ch. – PhD, assistant professor, associate professor of the department of ICSD

Annotation. Prospective projection matrices are mathematically substantiated, and their work and implementation in the *Opengl* environment are considered.

Keywords: perspective projection matrices, computer graphics, programming.