

ВИЗУАЛИЗАТОР ТРЁХМЕРНЫХ МОДЕЛЕЙ ДЛЯ РЕАЛИЗАЦИИ ВИРТУАЛЬНЫХ ЛАБОРАТОРНЫХ И ПРАКТИЧЕСКИХ ЗАНЯТИЙ

Савенко А. Г.

Кафедра информационных систем и технологий, Институт информационных технологий Белорусского государственного университета информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: savenko@bsuir.by

В работе описан модуль визуализации 3D моделей для реализации виртуальных лабораторных и практических занятий. Алгоритмы визуализации моделей имеют оптимизации, позволяющие выполнять визуализацию на низкопроизводительных компьютерах, а сам визуализатор является кросс-платформенным.

ВВЕДЕНИЕ

Современная сфера образования претерпевает ряд изменений, обусловленных, в том числе, общими тенденциями цифровизации общества в целом. Виртуальная и дополненная реальности становятся более приемлемым способом доставки учебного контента [1]. Также такой подход имеет ряд существенных преимуществ, связанных со стремительно развивающейся дистанционной формы обучения [2]. Отдельным достоинством виртуализации обучающего контента является игровой подход при обучении, повышающий интерес и мотивацию к процессу обучения у молодых людей [3]. Эти преимущества можно реализовать при визуализации трёхмерных моделей лабораторных макетов, стендов и т.п. Разработанный модуль позволяет визуализировать модели, созданные практически в любых современных 3D редакторах, и встраивать их в систему управления обучением.

1. АЛГОРИТМЫ ВИЗУАЛИЗАЦИИ 3D МОДЕЛЕЙ

Поскольку современные графические процессоры рассчитаны на отрисовку треугольных фрагментов, 3D модель должна состоять из треугольников. Каждый треугольный фрагмент отдельно растеризуется графическим процессором на экран. В начале, в память видеокарты загружается информация о геометрии и материалах модели (текстуры альbedo, карты нормалей, металличности, шероховатости, фонового затенения и др.). Очистка экрана происходит при помощи стандартных алгоритмов, применяемых и в других визуализаторах. В вершинном шейдере происходит преобразование вершин моделей в нужный формат для отображения на экран монитора. Для быстрого преобразования модели в пространство NDC используются матричные преобразования. Для этого в вершинном шейдере каждую вершину модели нужно умножить на соответствующую матрицу итогового преобразования. Матрица итогового преобразования, в свою очередь, представляет собой произведе-

ние матриц вида, проецирования и локального преобразования. Матрица вида содержит информацию для отрисовки модели с нужной точки в пространстве (камеры) и имеет вид представленный на рисунке 1.

$$M_{view} = M_R M_T = \begin{bmatrix} l_x & l_y & l_z & 0 \\ u_x & u_y & u_z & 0 \\ f_x & f_y & f_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -x_c \\ 0 & 1 & 0 & -y_c \\ 0 & 0 & 1 & -z_c \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} l_x & l_y & l_z & -l_x x_c - l_y y_c - l_z z_c \\ u_x & u_y & u_z & -u_x x_c - u_y y_c - u_z z_c \\ f_x & f_y & f_z & -f_x x_c - f_y y_c - f_z z_c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

где l_x, l_y, l_z – нормированный вектор от камеры влево в соответствующих координатах;
 u_x, u_y, u_z – нормированный вектор от камеры вверх в соответствующих координатах;
 f_x, f_y, f_z – нормированный вектор от камеры к объекту в соответствующих координатах;
 x_c, y_c, z_c – координаты текущей позиции смотрящего (пространства глаза).

Рис. 1 – Матрица вида

Матрица проецирования позволяет визуализировать модель в перспективе и имеет вид представленный на рисунке 2.

$$\begin{bmatrix} \frac{1}{ar \cdot \tan(\frac{\alpha}{2})} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan(\frac{\alpha}{2})} & 0 & 0 \\ 0 & 0 & \frac{-NearZ - FarZ}{NearZ - FarZ} & \frac{2 \cdot FarZ \cdot NearZ}{NearZ - FarZ} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

где ar – коэффициент между шириной и высотой прямоугольной области, на которую будет осуществляться проекция (коэффициент соотношение сторон);
 α – угол обзора глаза (камеры) по вертикали;
 $NearZ$ – позиция ближней Z-плоскости (позволяет обрезать объекты, находящиеся слишком близко к камере);
 $FarZ$ – позиция дальней Z-плоскости (позволяет обрезать объекты, находящиеся слишком далеко от камеры).

Рис. 2 – Матрица проецирования

Матрица локального преобразования хранит в себе данные о локальных трансформациях модели (угол вращения, перемещения в пространстве, масштабирования и т.д.) и, как правило, содержится в файле модели. Следующим основным этапом визуализации является фрагментный шейдер. Входными данными для него являются интерполированные данные из вершинного шейдера для каждого растеризуемого фрагмента и информация о материалах модели. Выходными данными является цвет фрагмента. Таким образом, в данном шейдере определяется цвет каждого фрагмента (пикселя) визуализированной модели. Для реалистичного отображения модели в данном шейдере просчитывается освещённость модели. Так как каждый фрагмент реагирует на свет по-разному, то необходимо принимать во внимание систему материалов (текстура альbedo, карты нормалей, метал-

личности, шероховатости, фонового затенения и кубические карты), приближенную к реальному миру и отражающую как каждый текстел нагруженный на модель текстуры реагирует на свет. Для имитации отражения и поглощения света от реального мира используется кубические карты, которые содержат вид вокруг какой-либо точки пространства модели. Однако для влияния пространства вокруг модели на какой-либо фрагмент необходимо вычислять интегралы по полусфере окружения пикселя в режиме реального времени, что является ресурсозатратной операцией (аналитический метод). Поэтому для ускорения учета влияния окружения на просчитываемый фрагмент заранее рассчитываются некоторые возможные варианты влияния. Суть такой оптимизации заключается в том, чтобы не высчитывать интеграл для каждого пикселя визуализации (для зеркального и диффузного освещения) в реальном времени, а брать уже готовые заранее подсчитанные значения для пикселей модели с определённым шагом. Данная оптимизация позволяет очень быстро рассчитывать фотореалистичное освещение даже на низкопроизводительных компьютерах (по сравнению с другими визуализаторами). Первая кубическая карта будет содержать в себе только диффузное освещение, где каждый вектор нормали (с шагом 1/255) будет содержать в себе диффузный свет в данном направлении. Остальные кубические карты будут содержать в себе зеркальное освещение, где каждый вектор нормали (с шагом 1/255) будет содержать в себе зеркальный свет (отражение) в данном направлении. Таких кубических карт зеркального освещения будет несколько, где каждая карта содержит значения для нужной шероховатости. Все промежуточные значения освещённости рассчитываются с помощью интерполяции. Конечная формула для расчета освещения представлена на рисунке 3.

$$L_o(p) = E_d(N) \cdot k_d + E_s(R, r) \cdot (k_s \cdot BF(NV, r)_r + BF(NV, r)_g) ,$$

где N – нормаль поверхности;

V – вектор камеры;

R – вектор отражения векторов V и N ;

r – коэффициент шероховатости от 0 до 1;

E_d – является выборкой из кубической текстуры диффузного освещения с координатами N ;

E_s – является выборкой из кубической текстуры зеркального освещения с уровнем r

и координатами N ;

k_d – диффузный коэффициент материала;

k_s – зеркальный коэффициент материала;

BF – выборка из текстуры с координатами (NV, r) .

Рис. 3 – Формула расчета освещенности

В свою очередь, диффузный коэффициент материала определяется по формуле, представленной на рисунке 4.

$$k_d = albedo \cdot (1 - f_0) \cdot (1 - metallic).$$

где $albedo$, $metallic$ – параметры материала;

f_0 – константа, принятая за 0,04.

Рис. 4 – Формула расчета диффузного коэффициента материала

Зеркальный коэффициент материала определяется по формуле, представленной на рисунке 5.

$$k_s = mix(f_0, baseColor, metallic),$$

где $baseColor$, $metallic$ – параметры материала;
 mix – функция интерполяции.

Рис. 5 – Формула расчета зеркального коэффициента материала

Следующим этапом расчёта освещения является просчёт прямого освещения от направленного источника света. Для этого используется карта теней. Для определения затенённости каждого пикселя фрагмента модели выполняется тест глубины (в два прохода), который и определяет, находится ли пиксель в тени или нет. Первый проход – это дополнительная визуализация с позиции источника света. Пиксели расположенные дальше от источника света и имеющие перекрывающие их пиксели не проходят тест глубины. Значение глубины ближайшего пикселя визуализируется в отдельный буфер глубины. При втором проходе сцена визуализируется с позиции смотрящего. Буфер глубины привязывается к фрагментному шейдеру. Для каждого пикселя получается соответствующее значение глубины из буфера. Если эти значения глубины равны, то его цвет находится стандартным алгоритмом. В противном случае делается выборка из карты глубины вокруг текущего пикселя и берётся их среднее значение. Для имитации влияния ярких участков изображения сначала размываются яркие участки изображения алгоритмом Гаусса, а затем смешиваются с основным изображением. На заключительном этапе визуализации применяются все рассчитанные эффекты для визуализируемой модели.

II. ЗАКЛЮЧЕНИЕ

Разработан модуль визуализации трёхмерных моделей, создаваемых практически в любых 3D редакторах. Полученный модуль имеет ряд преимуществ по сравнению с существующими аналогами, таких как: кросс-платформенность, невысокие системные требования к компьютеру за счёт оптимизированных алгоритмов вершинного и фрагментного шейдеров, возможность запуска в веб-браузере в режиме онлайн.

1. Савенко, А. Г. Виртуальная реальность, как способ получения и доставки учебного контента / А. Г. Савенко, Н. А. Кукалев, А. Г. Савенко // Высшее техническое образование : проблемы и пути развития : материалы IX Международной научно-методической конференции / редкол. : В. А. Богущ [и др.]. – Минск : БГУИР, 2018. – С. 394 – 397.
2. Савенко, А. Г. Преимущества и перспективы использования виртуальной и дополненной реальности в дистанционном образовательном процессе / А. Г. Савенко // Дистанционное обучение – образовательная среда XXI века : материалы X международной научно-методической конференции. – Минск : БГУИР, 2017. – С. 119.
3. Савенко, А. Г. Игровой подход в обучении программированию детей и подростков / А. Г. Савенко // Информационные технологии в технических, политических и социально-экономических системах : материалы Международной научно-технической конференции. – Минск : БНТУ, 2018. – С. 30.