

УДК 004.92

## ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ВЫЧИСЛЕНИЙ МЕСТОПОЛОЖЕНИЯ 3D-ПРИМИТИВОВ НА СЦЕНЕ ИЗОБРАЖЕНИЙ

Е. А. Малафеев, д. т. н. В. С. Ситников

Одесский национальный политехнический университет  
Украина, г. Одесса  
jikero@mail.ru, sitnvs@mail.ru

*Рассмотрены вопросы формирования графического изображения с использованием шейдеров, а также показаны сложности, связанные с перемножением нескольких матриц. Проведенный анализ итоговой матрицы показал возможность упрощения за счет исключения модельной матрицы из преобразования примитива, что позволило упростить вычисления и уменьшить нагрузку на центральный процессор.*

*Ключевые слова:* «биллборд», примитив, шейдер, проекционная матрица, видовая матрица, модельная матрица, нормализованная система координат.

Для вывода графики на современные устройствах используют шейдеры — программы, которые выполняются графическим процессором и обеспечивают конечное изображение на экране. Самая простая программа обработки графики состоит из двух шейдеров. Первый шейдер — вершинный задает вершинам примитивов позицию на экране, второй — фрагментный отвечает за цвет фрагмента изображения. Это так называемый программируемый конвейер рендеринга. Работа первого шейдера связана с большим объемом вычислений, т. к. в 3D-графике для расчета расположения примитива на экране необходимо выполнить ряд перемножений матриц (проекционной, видовой и модельной).

Например, для вывода «биллбордов», которые представляют собой квадрат, состоящий из двух треугольников, на которых натянута текстура с изображением необходимого объекта, нужно обеспечить перпендикулярность камере (наблюдателю). Такая операция связана с перемножением на матрицу поворота.

Для достижения реалистичности изображений таких эффектов, как взрыв, огонь, дым и т. д., используется система сотен и тысяч частиц, каждая из которых представляется «биллбордом». Каждому «биллборду» из этой системы частиц необходимо задать закон поведения, который состоит из задания скорости, изменения размера, прозрачности и цвета, что приводит к необходимости перевычисления итоговой матрицы для каждой частицы, а это обуславливает большую нагрузку на вычислительные средства.

Повышение эффективности вычислений и уменьшения времени подготовки 3D-изображения возможно за счет упрощения вычислений для одной частицы. Анализ итоговой матрицы дает возможность исключить модельную матрицу из расчетов и оставить только видовую и проекционную матрицы.

Простейший вершинный шейдер преобразования координат «биллборда» представлен на листинге 1. Вершинный шейдер выполняется для каждой вершины примитива. С помощью входных параметров в него передаются атрибуты вершины и итоговая матрица `u_matModelViewProjection` как результат умножения проекционной, видовой и модельной матриц. Для всех вершин примитива модельная матрица задает поворот и перенос вершины, располагая ее в мировой системе координат. Видовая матрица поворачивает и смещает вершину, располагая ее в системе координат камеры. Проекционная матрица переводит вершину в нормализованную систему координат, что соответствует единичному кубу с вершинами в диапазоне от  $-1$  до  $1$  по всем осям.

Поворот и трансформация «биллборда» обеспечивает модельная матрица, которая обычно рассчитывается процессором и передается в шейдер или прямо перемножается на координаты вершины и передается уже в трансформированном виде. Этих операций можно избежать, если подробно изу-

чить результат перемножения на проекционную матрицу. Поскольку любые вершины примитивов будут располагаться в координатах от  $-1$  до  $1$  по всем осям, то можно рисовать примитив в плоскости XY в диапазоне этих координат, не перемножая на матрицы, а передать на выход вершинного шейдера позицию вершины. В этом случае примитив всегда будет перпендикулярен камере на экране. Однако для обеспечения примитива в перспективе и перемещения наблюдателя по сцене необходимо иметь перспективную и видовую матрицы. Таким образом, нет необходимости в модельной матрице (листинг 2).

Листинг 1. Вершинный шейдер преобразования координат «билборда»:

```
attribute vec3 a_vPosition; // координаты в модельной системе координат
attribute vec2 a_vTexCoord; //текстурные координаты
uniform mat4 u_matModelViewProjection; // произведение модельной, видовой и проектирующей
//матриц
varying vec2 v_vTexCoord;
void main() {
    gl_Position = u_matModelViewProjection *vec4(a_vPosition,1);
    v_vTexCoord = a_vTexCoord;
}
```

Листинг 2. Вершинный шейдер преобразования координат «билборда» без использования модельной матрицы:

```
attribute vec3 a_vPosition;
attribute vec2 a_vTexCoord;
uniform vec4 u_vInitPos; // центр билборда
uniform mat4 u_matViewProjection;
varying vec2 v_vTexCoord;
void main() {
    vec4 vPos = u_matViewProjection * u_vInitPos;
    gl_Position = vec4(vPos.x+ a_vPosition.x,vPos.y+ a_vPosition.y,vPos.z,vPos.w);
    v_vTexCoord = a_vTexCoord;
}
```

Применение предложенного подхода позволяет, не ухудшая качества изображения, повысить эффективность вычислений, уменьшить время преобразования, а также вызовов графического процессора.

---

Malafeev E. A., Sytnikov V. S.

### **Improving the computing efficiency of 3D-primitives location on the scene of imaging.**

The authors consider the problems of graphic images formation using shaders. The paper also shows the complications connected with the multiplication of several matrices. The analysis of the final matrix has shown the possibility of simplification by eliminating the model matrix from transformation of the primitive, which makes it possible to simplify calculations and reduce the load on the CPU.

Keywords: *"billboard", primitive, shader, projection matrix, modelview matrix, model matrix, normalized coordinate system.*