

RenderWare Graphics

Белая книга

Матрица обзора камеры

Связаться с нами

Критерион Софтвр Лтд.

Для получения общей информации о RenderWare Graphics отправьте электронное письмо info@csl.com.

Участники

Команды разработки и документирования RenderWare Graphics.

Авторские права © 1993 - 2003 Criterion Software Ltd. Все права защищены.

Canon и RenderWare являются зарегистрированными товарными знаками Canon Inc. Nintendo является зарегистрированным товарным знаком, а NINTENDO GAMECUBE является товарным знаком Nintendo Co., Ltd. Microsoft является зарегистрированным товарным знаком, а Xbox является товарным знаком Microsoft Corporation. PlayStation является зарегистрированным товарным знаком Sony Computer Entertainment Inc. Все остальные товарные знаки, упомянутые здесь, являются собственностью соответствующих компаний.

Оглавление

| | | |
|----|-------------------------------------|---|
| 1. | Введение..... | 4 |
| 2. | Создание матрицы представлений..... | 5 |
| 3. | Реализация..... | 7 |

1. Введение

RenderWare Graphics использует матрицу, называемую Camera View Matrix. Эта матрица преобразует координаты в мировой системе координат в пространство камеры RenderWare Graphics. В этом документе описываются шаги, предпринимаемые для создания матрицы вида камеры. Следует понимать, что эта матрица пересчитывается для каждого кадра, `RwCameraBeginUpdate` функция. В терминологии RenderWare Graphics это называется синхронизацией камеры.

2. Создание матрицы представлений

RenderWare Graphics объединит несколько более простых матриц преобразования для создания матрицы вида. Они показаны в уравнениях ниже.

C=TOSPX

где

$$T = \begin{pmatrix} -\Gamma_x & T\Gamma_x & a_x & 0 \\ -\Gamma_y & T\Gamma_y & a_y & 0 \\ -\Gamma_z & T\Gamma_z & a_z & 0 \\ -\Pi_x & -\Pi_y & -\Pi_z & 1 \end{pmatrix} \quad O = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ o_x & o_y & 1 & 0 \\ o_x & -o_y & 0 & 1 \end{pmatrix}$$

Γ правый вектор
 $T\Gamma$ вверх вектор
 a в векторе
 o смещение вида
 \mathcal{J} ширина экрана
 $час$ высота экрана

$$C = \begin{pmatrix} -\frac{1}{2\mathcal{J}} & 0 & 0 & 0 \\ 0 & \frac{1}{2\mathcal{J}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \Pi = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

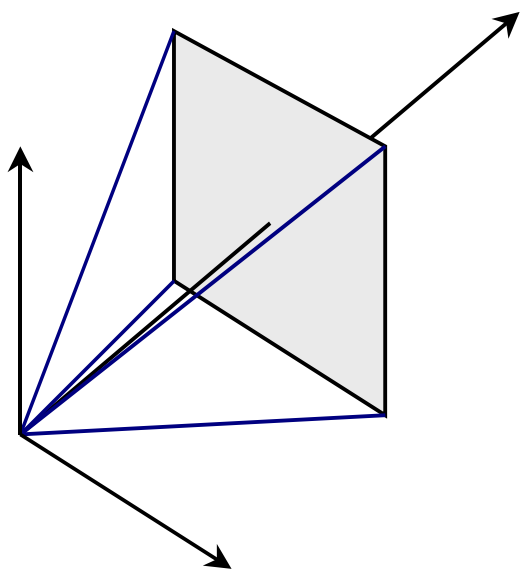
$$X = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{1}{2} & \frac{1}{2} & 0 & 1 \end{pmatrix}$$

Видно, что для вычисления матрицы вида камеры используются 5 различных матриц. Первая из них, T , является матрицей переориентации, которая преобразуется в трехмерную систему координат, ориентированную так, что камера смотрит в положительном направлении оси z и находится в начале координат.

Матрица O выполняет необходимое смещение, если приложение установило смещение вида с помощью `RwCameraSetViewOffset`. Аналогично, S изменяет размер пространства камеры, подгоняя его под окно просмотра, вызов которого `RwCameraSetViewWindow` устанавливает.

Матрица P отображается из трехмерной системы координат в усеченную пирамиду камеры и, следовательно, является проекцией. (Для параллельной модели камеры эта матрица заменяется нулевой матрицей с ведущей строкой $(1, -1, 1, 1)$).

Наконец, матрица X применяет сдвиг к координатному пространству. Усеченная пирамида камеры в RenderWare Graphics занимает только первый октант, эффективно преобразуясь из усеченной пирамиды, определяемой началом координат и единичным квадратом с центром в точке $(0,0,1)$, в пирамиду, определяемую началом координат и единичным квадратом с центром в точке $(0,5, 0,5)$.



3. Реализация

Следует отметить, что конкатенация этих матриц не выполняется так, как описано в этом документе. Реализация оптимизирована, а матрица вида вычисляется напрямую, а не с использованием умножения матриц. Лицензиаты исходного кода могут увидеть код для функций синхронизации камеры `vsrsrc\basamera.c`.