Компьютерная графика

Лекция 4: Камера, проекции, буфер глубины

2021

Проекции

- Мир - двухмерный/трёхмерный, произвольного размера, с произвольного ракурса

Проекции

- Мир - двухмерный/трёхмерный, произвольного размера, с произвольного ракурса
- Экран - двухмерный, $[-1, 1] \times [-1, 1]$

Проекции

- Мир - двухмерный/трёхмерный, произвольного размера, с произвольного ракурса
- Экран - двухмерный, $[-1, 1] \times [-1, 1]$
  - Координата Z тоже $[-1, 1]$
  - Про неё поговорим подробнее чуть позже

Проекции

- Мир - двухмерный/трёхмерный, произвольного размера, с произвольного ракурса
- Экран - двухмерный, $[-1, 1] \times [-1, 1]$
  - Координата Z тоже $[-1, 1]$
  - Про неё поговорим подробнее чуть позже
- За преобразование отвечает проекция

Ортографическая проекция

- Самый простой способ:

Ортографическая проекция

- Самый простой способ: игнорировать третью координату

Концептуально: $(x, y, z) \mapsto (x, y)$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ортографическая проекция

- Самый простой способ: игнорировать третью координату

$$\text{Концептуально: } (x, y, z) \mapsto (x, y)$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Именно это делает OpenGL (если `gl_Position.w = 1`)

Ортографическая проекция

- Самый простой способ: игнорировать третью координату

$$\text{Концептуально: } (x, y, z) \mapsto (x, y)$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Именно это делает OpenGL (если `gl_Position.w = 1`)
- X и Y всё ещё $[-1, 1]$

Ортографическая проекция

- Самый простой способ: игнорировать третью координату

  Концептуально: $(x, y, z) \mapsto (x, y)$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Именно это делает OpenGL (если `gl_Position.w = 1`)
- X и Y всё ещё $[-1, 1]$
- Как сделать $X \in [-W, W]$ и $Y \in [-H, H]$?

Ортографическая проекция

- Самый простой способ: игнорировать третью координату

$$\text{Концептуально: } (x, y, z) \mapsto (x, y)$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Именно это делает OpenGL (если `gl_Position.w = 1`)
- X и Y всё ещё $[-1, 1]$
- Как сделать $X \in [-W, W]$ и $Y \in [-H, H]$?

$$\begin{pmatrix} \frac{1}{W} & 0 & 0 & 0 \\ 0 & \frac{1}{H} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ортографическая проекция

- Как сделать $X \in [X_0 - W, X_0 + W]$ и $Y \in [Y_0 - H, Y_0 + H]$?

Ортографическая проекция

- Как сделать $X \in [X_0 - W, X_0 + W]$ и $Y \in [Y_0 - H, Y_0 + H]$?
  - Сдвинуть на $(-X_0, -Y_0)$, а затем применить масштабирование:

$$\begin{pmatrix} \frac{1}{W} & 0 & 0 & 0 \\ 0 & \frac{1}{H} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -X_0 \\ 0 & 1 & 0 & -Y_0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ортографическая проекция

- Как сделать $X \in [X_0 - W, X_0 + W]$ и $Y \in [Y_0 - H, Y_0 + H]$?
  - Сдвинуть на $(-X_0, -Y_0)$, а затем применить масштабирование:
$$\begin{pmatrix} \frac{1}{W} & 0 & 0 & 0 \\ 0 & \frac{1}{H} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -X_0 \\ 0 & 1 & 0 & -Y_0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$
- Если размер области W, нужно разделить на W
- Если центр в X_0, нужно сдвинуть на $-X_0$

Ортографическая проекция

- Как сделать $X \in [X_0 - W, X_0 + W]$ и $Y \in [Y_0 - H, Y_0 + H]$?
  - Сдвинуть на $(-X_0, -Y_0)$, а затем применить масштабирование:
$$\begin{pmatrix} \frac{1}{W} & 0 & 0 & 0 \\ 0 & \frac{1}{H} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -X_0 \\ 0 & 1 & 0 & -Y_0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$
- Если размер области W, нужно разделить на W
- Если центр в X_0, нужно сдвинуть на $-X_0$
- Общая идея: если камера получена каким-то преобразованием, нужно применить обратное преобразование

Ортографическая проекция: обратное преобразование

- Можно считать, что по умолчанию OpenGL делает ортографическую проекцию на квадрат $[-1, 1]^2$ в плоскости XY параллельно оси Z
- Камера находится в начале координат

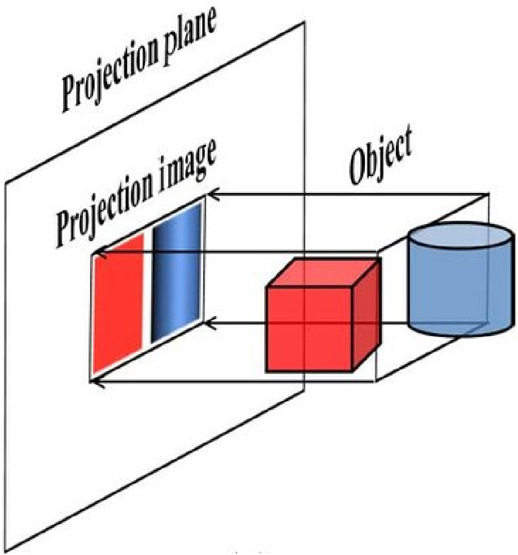
Ортографическая проекция: обратное преобразование

- Можно считать, что по умолчанию OpenGL делает ортографическую проекцию на квадрат $[-1, 1]^2$ в плоскости XY параллельно оси Z
- Камера находится в начале координат
- Если ко всему виртуальному миру (включая камеру!) применить аффинное преобразование, изображение не изменится

Ортографическая проекция: обратное преобразование

- Можно считать, что по умолчанию OpenGL делает ортографическую проекцию на квадрат $[-1, 1]^2$ в плоскости XY параллельно оси Z
- Камера находится в начале координат
- Если ко всему виртуальному миру (включая камеру!) применить аффинное преобразование, изображение не изменится
- $\Rightarrow$ Если наша камера получена аффинным преобразованием из стандартной камеры OpenGL, к объектам мира нужно применить обратное преобразование

Ортографическая проекция



Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать

Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать
  - Положением камеры $C = (C_x, C_y, C_z)$

Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать
  - Положением камеры $C = (C_x, C_y, C_z)$
  - Осями координат камеры $X = (X_x, X_y, X_z), Y = (Y_x, Y_y, Y_z), Z = (Z_x, Z_y, Z_z)$

Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать
  - Положением камеры $C = (C_x, C_y, C_z)$
  - Осями координат камеры $X = (X_x, X_y, X_z), Y = (Y_x, Y_y, Y_z), Z = (Z_x, Z_y, Z_z)$
- Делает проекцию параллельно вектору Z на параллелограмм $C \pm X \pm Y$
- Параллелограмм отождествляется с экраном

Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать
  - Положением камеры $C = (C_x, C_y, C_z)$
  - Осями координат камеры $X = (X_x, X_y, X_z), Y = (Y_x, Y_y, Y_z), Z = (Z_x, Z_y, Z_z)$
- Делает проекцию параллельно вектору Z на параллелограмм $C \pm X \pm Y$
- Параллелограмм отождествляется с экраном
- Обычно X, Y, Z взаимно ортогональны

Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать
  - Положением камеры $C = (C_x, C_y, C_z)$
  - Осями координат камеры $X = (X_x, X_y, X_z), Y = (Y_x, Y_y, Y_z), Z = (Z_x, Z_y, Z_z)$
- Делает проекцию параллельно вектору Z на параллелограмм $C \pm X \pm Y$
- Параллелограмм отождествляется с экраном
- Обычно X, Y, Z взаимно ортогональны
- Как выразить эту проекцию матрицей?

Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать
  - Положением камеры $C = (C_x, C_y, C_z)$
  - Осями координат камеры $X = (X_x, X_y, X_z), Y = (Y_x, Y_y, Y_z), Z = (Z_x, Z_y, Z_z)$
- Делает проекцию параллельно вектору Z на параллелограмм $C \pm X \pm Y$
- Параллелограмм отождествляется с экраном
- Обычно X, Y, Z взаимно ортогональны
- Как выразить эту проекцию матрицей?
- Преобразование из стандартной системы координат OpenGL $[-1, 1]^3$ в координаты области, видимой через эту камеру: смена системы координат + параллельный перенос

$$\begin{pmatrix} X_x & Y_x & Z_x & C_x \\ X_y & Y_y & Z_y & C_y \\ X_z & Y_z & Z_z & C_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ортографическая проекция: общий случай

- В общем случае ортографическую камеру можно задать
  - Положением камеры $C = (C_x, C_y, C_z)$
  - Осями координат камеры $X = (X_x, X_y, X_z), Y = (Y_x, Y_y, Y_z), Z = (Z_x, Z_y, Z_z)$
- Делает проекцию параллельно вектору Z на параллелограмм $C \pm X \pm Y$
- Параллелограмм отождествляется с экраном
- Обычно X, Y, Z взаимно ортогональны
- Как выразить эту проекцию матрицей?
- Преобразование из стандартной системы координат OpenGL $[-1, 1]^3$ в координаты области, видимой через эту камеру: смена системы координат + параллельный перенос

$$\begin{pmatrix} X_x & Y_x & Z_x & C_x \\ X_y & Y_y & Z_y & C_y \\ X_z & Y_z & Z_z & C_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Обратное преобразование (проекция) - обратная матрица

Ортографическая проекция: общий случай

- Можно представить X, Y, Z как произведение длины и нормированного вектора:

$$X = W \cdot \hat{X} \quad Y = H \cdot \hat{Y} \quad Z = D \cdot \hat{Z}$$

Ортографическая проекция: общий случай

- Можно представить X, Y, Z как произведение длины и нормированного вектора:
$$X = W \cdot \hat{X} \quad Y = H \cdot \hat{Y} \quad Z = D \cdot \hat{Z}$$
- Матрицу можно разбить на перенос, масштабирование и поворот
$$\begin{pmatrix} X_x & Y_x & Z_x & C_x \\ X_y & Y_y & Z_y & C_y \\ X_z & Y_z & Z_z & C_z \\ 0 & 0 & 0 & 1 \end{pmatrix} =$$
$$\begin{pmatrix} 1 & 0 & 0 & C_x \\ 0 & 1 & 0 & C_y \\ 0 & 0 & 1 & C_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \hat{X}_x & \hat{Y}_x & \hat{Z}_x & 0 \\ \hat{X}_y & \hat{Y}_y & \hat{Z}_y & 0 \\ \hat{X}_z & \hat{Y}_z & \hat{Z}_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} W & 0 & 0 & 0 \\ 0 & H & 0 & 0 \\ 0 & 0 & D & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ортографическая проекция: общий случай

- Тогда обратная матрица (матрица проекции):

$$\begin{pmatrix} X_x & Y_x & Z_x & C_x \\ X_y & Y_y & Z_y & C_y \\ X_z & Y_z & Z_z & C_z \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} =$$

$$\begin{pmatrix} \frac{1}{W} & 0 & 0 & 0 \\ 0 & \frac{1}{H} & 0 & 0 \\ 0 & 0 & \frac{1}{D} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \hat{X}_x & \hat{Y}_x & \hat{Z}_x & 0 \\ \hat{X}_y & \hat{Y}_y & \hat{Z}_y & 0 \\ \hat{X}_z & \hat{Y}_z & \hat{Z}_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 1 & 0 & 0 & -C_x \\ 0 & 1 & 0 & -C_y \\ 0 & 0 & 1 & -C_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ортографическая проекция: ортогональный случай

- Если X, Y, Z ортогональны, то

$$\begin{pmatrix} \hat{X}_x & \hat{Y}_x & \hat{Z}_x & 0 \\ \hat{X}_y & \hat{Y}_y & \hat{Z}_y & 0 \\ \hat{X}_z & \hat{Y}_z & \hat{Z}_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} \hat{X}_x & \hat{Y}_x & \hat{Z}_x & 0 \\ \hat{X}_y & \hat{Y}_y & \hat{Z}_y & 0 \\ \hat{X}_z & \hat{Y}_z & \hat{Z}_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{T} = \begin{pmatrix} \hat{X}_x & \hat{X}_y & \hat{X}_z & 0 \\ \hat{Y}_x & \hat{Y}_y & \hat{Y}_z & 0 \\ \hat{Z}_x & \hat{Z}_y & \hat{Z}_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ортографическая проекция: применение

Ортографическая проекция: применение

- 2D рендеринг: 2D игры, UI, карты

Ортографическая проекция: применение

- 2D рендеринг: 2D игры, UI, карты
- Проектировние моделей, зданий, деталей (вид сверху, вид сбоку, вид спереди)

Ортографическая проекция: применение

- 2D рендеринг: 2D игры, UI, карты
- Проектировние моделей, зданий, деталей (вид сверху, вид сбоку, вид спереди)
- Стилизация: 3D мир с ортографической проекцией

Ортографическая проекция: проблемы

Ортографическая проекция: проблемы

- Объекты на разном расстоянии от камеры выглядят одинаково

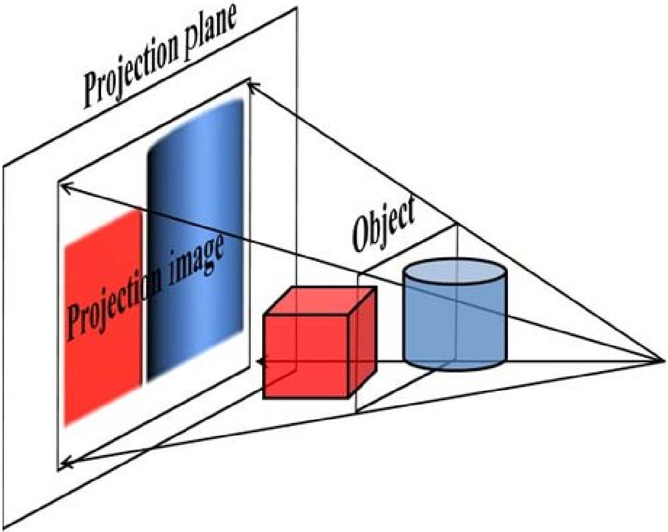
Ортографическая проекция: проблемы

- Объекты на разном расстоянии от камеры выглядят одинаково
- Нельзя оценить расстояние до объекта по его изображению

Ортографическая проекция: проблемы

- Объекты на разном расстоянии от камеры выглядят одинаково
- Нельзя оценить расстояние до объекта по его изображению
- Реальные камеры и глаза работают не так

Перспективная проекция

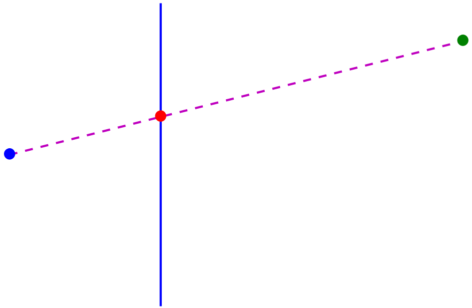


Перспективная проекция

- Есть центр проекции и плоскость проекции
- Проекция точки - пересечение прямой, проходящей через эту точку и центр проекции, с плоскостью проекции

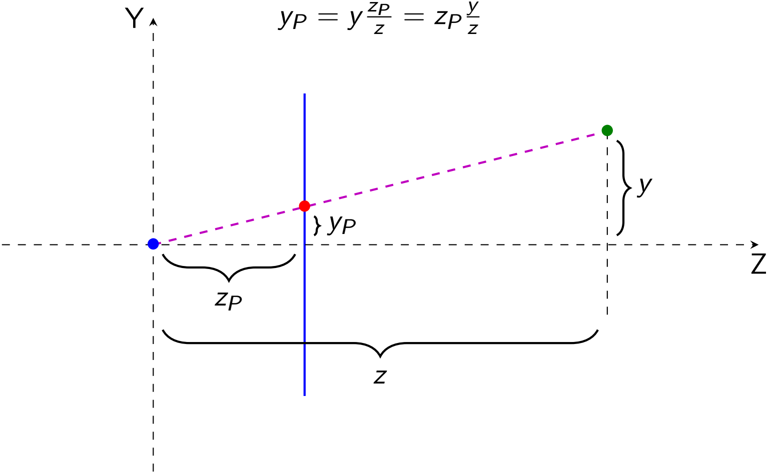
Перспективная проекция

- Есть центр проекции и плоскость проекции
- Проекция точки - пересечение прямой, проходящей через эту точку и центр проекции, с плоскостью проекции



Перспективная проекция

$$y_P = y\frac{z_P}{z} = z_P\frac{y}{z}$$



Перспективная проекция

- Чтобы вычислить перспективную проекцию с центром в начале координат и плоскостью проеции $Z = z_P$, нужно разделить на Z-координату:
$$(x, y, z) \mapsto \left(z_P \frac{x}{z}, z_P \frac{y}{z}\right)$$

Перспективная проекция

- Чтобы вычислить перспективную проекцию с центром в начале координат и плоскостью проеции $Z = z_P$, нужно разделить на Z-координату:
$$(x, y, z) \mapsto \left(z_P \frac{x}{z}, z_P \frac{y}{z}\right)$$
- Как выразить эту проекцию матрицей?

Перспективная проекция

- Чтобы вычислить перспективную проекцию с центром в начале координат и плоскостью проеции $Z = z_P$, нужно разделить на Z-координату:
$$(x, y, z) \mapsto \left(z_P \frac{x}{z}, z_P \frac{y}{z}\right)$$
- Как выразить эту проекцию матрицей? Никак: матрицы не умеют делить одни координаты на другие

Perspective divide

- Чтобы поддержать перспективную проекцию, в графическом конвейере (после вершинного шейдера, перед переводом в экранные координаты) есть специальный обязательный шаг: perspective divide

Perspective divide

- Чтобы поддержать перспективную проекцию, в графическом конвейере (после вершинного шейдера, перед переводом в экранные координаты) есть специальный обязательный шаг: perspective divide
- `gl_Position` делится на последнюю координату `gl_Position.w`

$$(x, y, z, w) \mapsto \left(\frac{x}{w}, \frac{y}{w}, \frac{z}{w}\right)$$

Perspective divide

- Чтобы поддержать перспективную проекцию, в графическом конвейере (после вершинного шейдера, перед переводом в экранные координаты) есть специальный обязательный шаг: perspective divide
- `gl_Position` делится на последнюю координату `gl_Position.w`

$$(x, y, z, w) \mapsto \left(\frac{x}{w}, \frac{y}{w}, \frac{z}{w}\right)$$

- Если $w = 1$, ничего не меняется (ортографическая проекция)

Perspective divide

- Чтобы поддержать перспективную проекцию, в графическом конвейере (после вершинного шейдера, перед переводом в экранные координаты) есть специальный обязательный шаг: perspective divide
- `gl_Position` делится на последнюю координату `gl_Position.w`

$$(x, y, z, w) \mapsto \left(\frac{x}{w}, \frac{y}{w}, \frac{z}{w}\right)$$

- Если $w = 1$, ничего не меняется (ортографическая проекция)
- Если $w =$ расстояние до камеры, получается перспективная проекция

Перспективная проекция

- Как выразить перспективную проекцию матрицей с последующим perspective divide?

Перспективная проекция

- Как выразить перспективную проекцию матрицей с последующим perspective divide?

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ z \end{pmatrix} \mapsto \begin{pmatrix} x/z \\ y/z \\ 1 \end{pmatrix}$$

Перспективная проекция

- Как выразить перспективную проекцию матрицей с последующим perspective divide?

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ z \end{pmatrix} \mapsto \begin{pmatrix} x/z \\ y/z \\ 1 \end{pmatrix}$$

- Обычно матрица перспективной проекции выглядит сложнее