

Компьютерная графика

Лекция 8: Stencil bufer, framebuffer, renderbuffer, пост-обработка

2021

Stencil buffer (буфер трафарета)

- ▶ Особый буфер, хранящий произвольные данные, с особыми побитовыми операциями над ними

Stencil buffer (буфер трафарета)

- ▶ Особый буфер, хранящий произвольные данные, с особыми побитовыми операциями над ними
- ▶ Чем-то аналогичен буферу глубины: тоже хранит какие-то данные, такого же размера (как и цветовой буфер), тоже позволяет рисовать или не рисовать пиксель по какому-то условию (depth test)

Stencil buffer (буфер трафарета)

- ▶ Особый буфер, хранящий произвольные данные, с особыми побитовыми операциями над ними
- ▶ Чем-то аналогичен буферу глубины: тоже хранит какие-то данные, такого же размера (как и цветовой буфер), тоже позволяет рисовать или не рисовать пиксель по какому-то условию (depth test)
- ▶ У дефолтного фреймбуфера часть есть 8-битный stencil буфер (зависит от настроек контекста OpenGL)

Stencil buffer (буфер трафарета)

- ▶ Особый буфер, хранящий произвольные данные, с особыми побитовыми операциями над ними
- ▶ Чем-то аналогичен буферу глубины: тоже хранит какие-то данные, такого же размера (как и цветовой буфер), тоже позволяет рисовать или не рисовать пиксель по какому-то условию (depth test)
- ▶ У дефолтного фреймбуфера часть есть 8-битный stencil буфер (зависит от настроек контекста OpenGL)
- ▶ Можно (и нужно, если вы его используете) очищать как `glClear(GL_STENCIL_BUFFER_BIT)`

Stencil buffer (буфер трафарета)

- ▶ Особый буфер, хранящий произвольные данные, с особыми побитовыми операциями над ними
- ▶ Чем-то аналогичен буферу глубины: тоже хранит какие-то данные, такого же размера (как и цветовой буфер), тоже позволяет рисовать или не рисовать пиксель по какому-то условию (depth test)
- ▶ У дефолтного фреймбуфера часть есть 8-битный stencil буфер (зависит от настроек контекста OpenGL)
- ▶ Можно (и нужно, если вы его используете) очищать как `glClear(GL_STENCIL_BUFFER_BIT)`
- ▶ Настроить значение, которым очищается буфер: `glClearStencil`

Stencil тест

- ▶ Включить/выключить stencil тест:
`glEnable/glDisable(GL_STENCIL_TEST)`

Stencil тест

- ▶ Включить/выключить stencil тест:
`glEnable/glDisable(GL_STENCIL_TEST)`
- ▶ Настроить stencil тест: `glStencilFunc`
 - ▶ `func` - одна из констант `GL_ALWAYS`, `GL_LESS`, `GL_GREATER`, `GL_EQUAL`, ...
 - ▶ `ref` - референсное значение для теста
 - ▶ `mask` - побитовая маска для теста
- ▶ Stencil тест: `func(ref & mask, stencil & mask)`

Stencil тест

- ▶ Включить/выключить stencil тест:
`glEnable/glDisable(GL_STENCIL_TEST)`
- ▶ Настроить stencil тест: `glStencilFunc`
 - ▶ `func` - одна из констант `GL_ALWAYS`, `GL_LESS`, `GL_GREATER`, `GL_EQUAL`, ...
 - ▶ `ref` - референсное значение для теста
 - ▶ `mask` - побитовая маска для теста
- ▶ Stencil тест: `func(ref & mask, stencil & mask)`
- ▶ Так же, как с depth тестом: если stencil тест не прошёл, пиксель не будет нарисован

Stencil тест

- ▶ Как записать значение в stencil буфер?

Stencil тест

- ▶ Как записать значение в stencil буфер? `glStencilOp`
 - ▶ `sfail` - что делать, если пиксель не прошёл stencil тест
 - ▶ `dpfail` - что делать, если пиксель не прошёл depth тест
 - ▶ `dppass` - что делать, если пиксель прошёл оба теста

Stencil тест

- ▶ Как записать значение в stencil буфер? `glStencilOp`
 - ▶ `sfail` - что делать, если пиксель не прошёл stencil тест
 - ▶ `dpfail` - что делать, если пиксель не прошёл depth тест
 - ▶ `dppass` - что делать, если пиксель прошёл оба теста
- ▶ Возможные значение `sfail`, `dpfail` и `dppass`:
 - ▶ `GL_KEEP` - не менять записанное значение
 - ▶ `GL_ZERO` - записать 0
 - ▶ `GL_INVERT` - побитово обратить
 - ▶ `GL_REPLACE` - записать `ref` из функции `glStencilFunc`
 - ▶ `GL_INCR` - увеличить на 1, если значение меньше максимального
 - ▶ `GL_DECR` - уменьшить на 1, если значение больше минимального (0)
 - ▶ `GL_INCR_WRAP` - увеличить на 1 с целочисленным переполнением
 - ▶ `GL_DECR_WRAP` - уменьшить на 1 с целочисленным переполнением

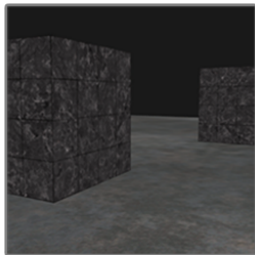
Stencil тест

- ▶ Дополнительно можно включать/выключать запись отдельных битов stencil буфера: `glStencilMask`

Stencil тест

- ▶ Дополнительно можно включать/выключать запись отдельных битов stencil буфера: `glStencilMask`
- ▶ Все параметры stencil теста можно настраивать отдельно для front и back граней функциями `glStencilFuncSeparate`, `glStencilOpSeparate`, `glStencilMaskSeparate`

Stencil test



Color buffer

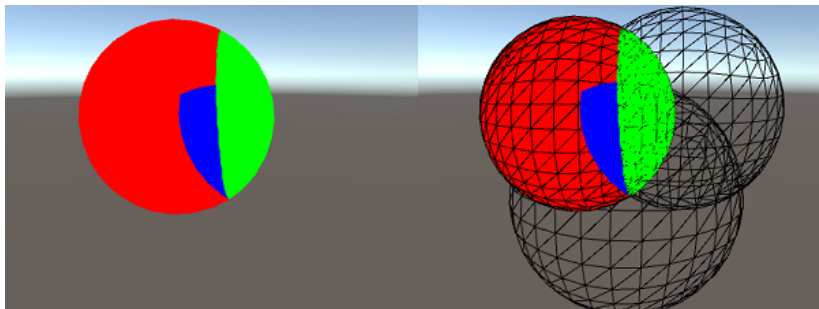


Stencil buffer



After stencil test

Stencil test



Stencil test

Always pass, value is unused
`glStencilFunc(GL_ALWAYS, 0, 0xFF);`
Unused(never fails), increment on pass, increment on z-fail
`glStencilOp(GL_KEEP, GL_INCR, GL_INCR);`
Draw 3 quads on top of each other

0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	2	2	2	1	1	0	0
0	1	2	3	3	3	1	1	0	0
0	0	1	2	2	2	1	1	0	0
0	0	1	1	1	1	0	0	0	0
0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Pass only when stencil value is ≥ 2
`glStencilFunc(GL_GEQUAL, 2, 0xFF);`
Don't modify (or do?)
`glStencilOp(GL_KEEP, GL_KEEP, GL_KEEP);`
Draw a fullscreen quad (only bright pixels are shaded)

0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	2	2	2	1	1	0	0
0	1	2	3	3	3	1	1	0	0
0	0	1	2	2	2	1	1	0	0
0	0	1	1	1	1	0	0	0	0
0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Stencil буфер: применение

- ▶ Некоторые алгоритмы рисования теней (shadow volumes)

Shadow volumes (stencil shadows)



Stencil буфер: применение

- ▶ Некоторые алгоритмы рисования теней (shadow volumes)
- ▶ Любая ситуация, в которой нужно ограничить рисование определённых пикселей:

Stencil буфер: применение

- ▶ Некоторые алгоритмы рисования теней (shadow volumes)
- ▶ Любая ситуация, в которой нужно ограничить рисование определённых пикселей:
 - ▶ Симулятор самолёта: сначала рисуется внутренность самолёта, затем - окружающий мир, только там, где не был нарисован самолёт \Rightarrow можно избежать проблем с точностью буфера глубины

Microsoft flight simulator



Stencil буфер: применение

- ▶ Некоторые алгоритмы рисования теней (shadow volumes)
- ▶ Любая ситуация, в которой нужно ограничить рисование определённых пикселей:
 - ▶ Симулятор самолёта: сначала рисуется внутренность самолёта, затем - окружающий мир, только там, где не был нарисован самолёт \Rightarrow можно избежать проблем с точностью буфера глубины
 - ▶ UI, который нужно нарисовать в какой-то ограниченной области экрана (например, scroll)

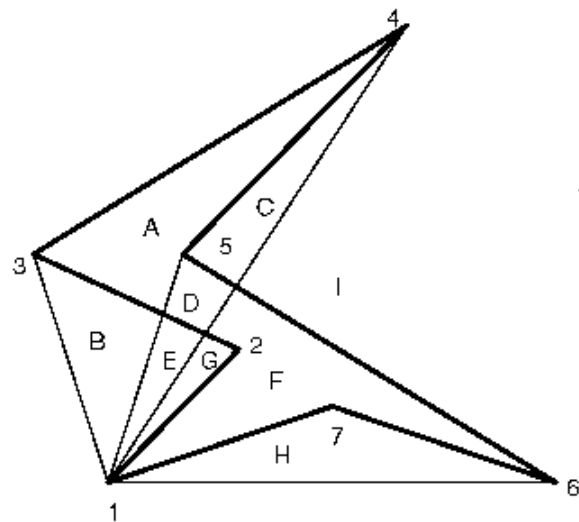
Scroll View

A Scroll Rect is usually used to scroll a large image or panel of

Stencil буфер: применение

- ▶ Некоторые алгоритмы рисования теней (shadow volumes)
- ▶ Любая ситуация, в которой нужно ограничить рисование определённых пикселей:
 - ▶ Симулятор самолёта: сначала рисуется внутренность самолёта, затем - окружающий мир, только там, где не был нарисован самолёт \Rightarrow можно избежать проблем с точностью буфера глубины
 - ▶ UI, который нужно нарисовать в какой-то ограниченной области экрана (например, scroll)
 - ▶ Рисование невыпуклых полигонов (odd-even rule)

Невыпуклый полигон



^AA: 134
B: 123 134
C: 134 145
^{*}D: 134 145 156
E: 123 134 145 156
^AF: 156
G: 123 156
H: 156 167
I: (none)

Stencil буфер: ссылки

- ▶ [khronos.org/opengl/wiki/Stencil_Test](https://www.khronos.org/opengl/wiki/Stencil_Test)
- ▶ learnopengl.com/Advanced-OpenGL/Stencil-testing
- ▶ open.gl/depthstencils
- ▶ en.wikibooks.org/wiki/OpenGL_Programming/Stencil_buffer