

PERFECT WORLD GAMES

基于GPU Driven的大量NPC绘制方案

技术支持部

王天宇

PERFECT WORLD

关于产品

PERSONA 5 THE PHANTOM X

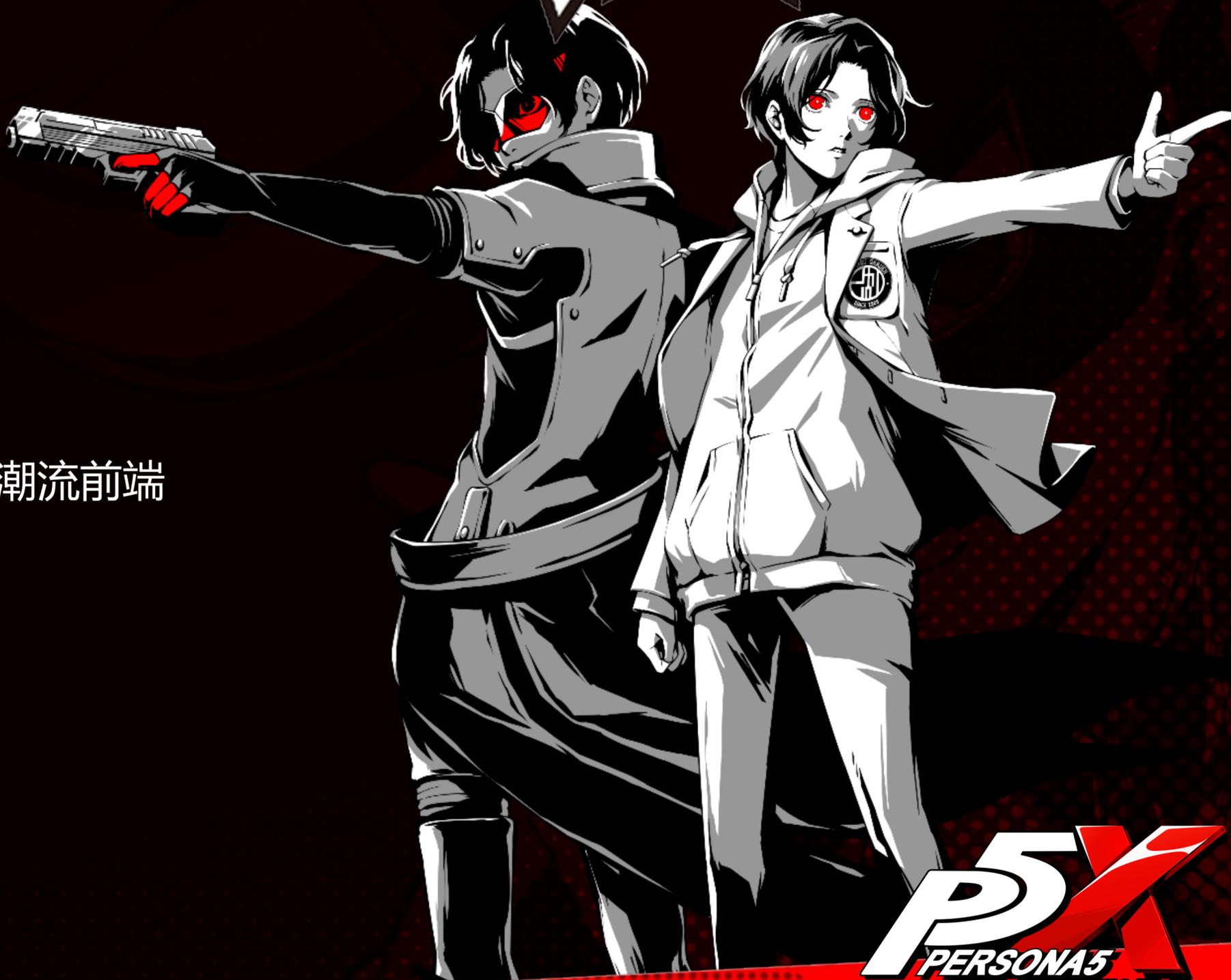


游戏名：女神异闻录：夜幕魅影

游戏类型：JRPG(Mobile)

平台：移动双端+PC

- 一个诞生至今有25余年历史，但一直站在潮流前端的日本主机游戏IP
- 正版授权的Persona系列商业手游
- ATLUS扮演了监修和开发双重角色



P5X
PERSONA 5
THE PHANTOM X







PERFECT WORLD

— 01 —

功能与需求

涩谷全向十字路口



01 功能与需求



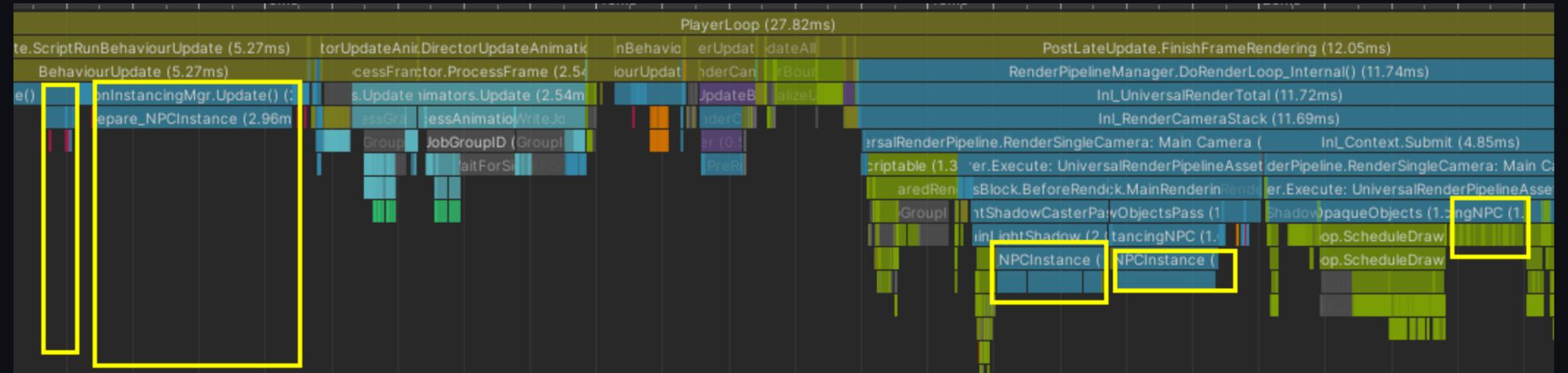
女神异闻录5中对应场景的刻画



Draw Instance方案

缺点：

- CPU消耗过高
- NPC数量达不到人群的表现
- 260个



骁龙865测试结果，绘制全流程使用了近7ms

需求整理：

- NPC的行为可以配置，如站立、走动、过马路。
- NPC持有不同的物品，如书包、雨伞。
- 同一种NPC身上的服装可以通过配置改变颜色。
- 同玩家交叠时有对应的消隐表现。
- 足够多。

设计与实现

GPU Driven Rendering :

- GPU决定哪些物体绘制。
- 没有CPU/GPU间的回读。

优点 :

- CPU只需要进行简单的绘制调用即可。
- 如果数据不被修改则不需频繁更新Buffer数据。
- GPU能做更精细的剔除

整体思路：

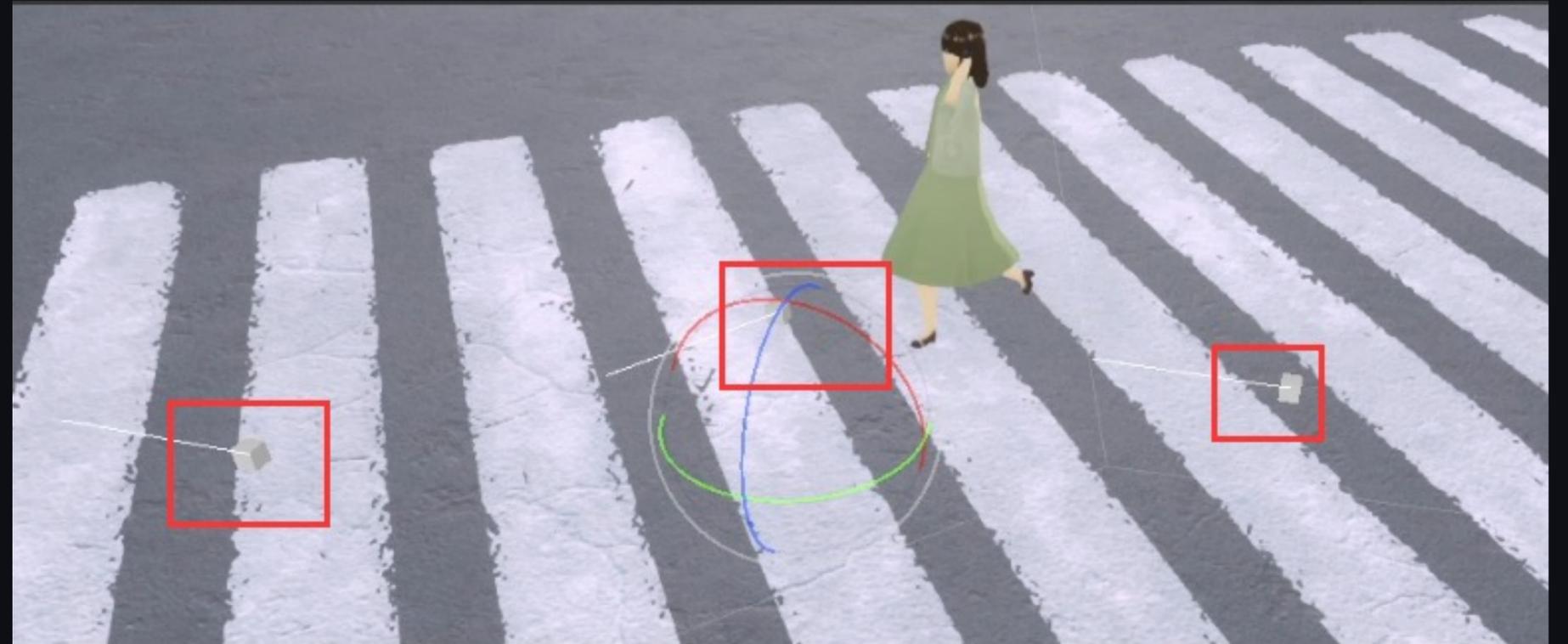
- 为了尽量减少更新NPC的数据，我们将每个NPC的行为设计为一个循环。
- 将区域的全部NPC的数据上传至GPU。
- GPU执行NPC的逻辑，在Compute Shader 中完成移动、剔除过程。
- 基于CS中的剔除结果进行绘制。
- 使用Draw Instanced Indirect 的方式为不同的NPC材质提交一次渲染指令。

NPC移动

使用路点间插值的方式实现移动和旋转。

为了压缩数据，我们只记录Y轴的旋转。

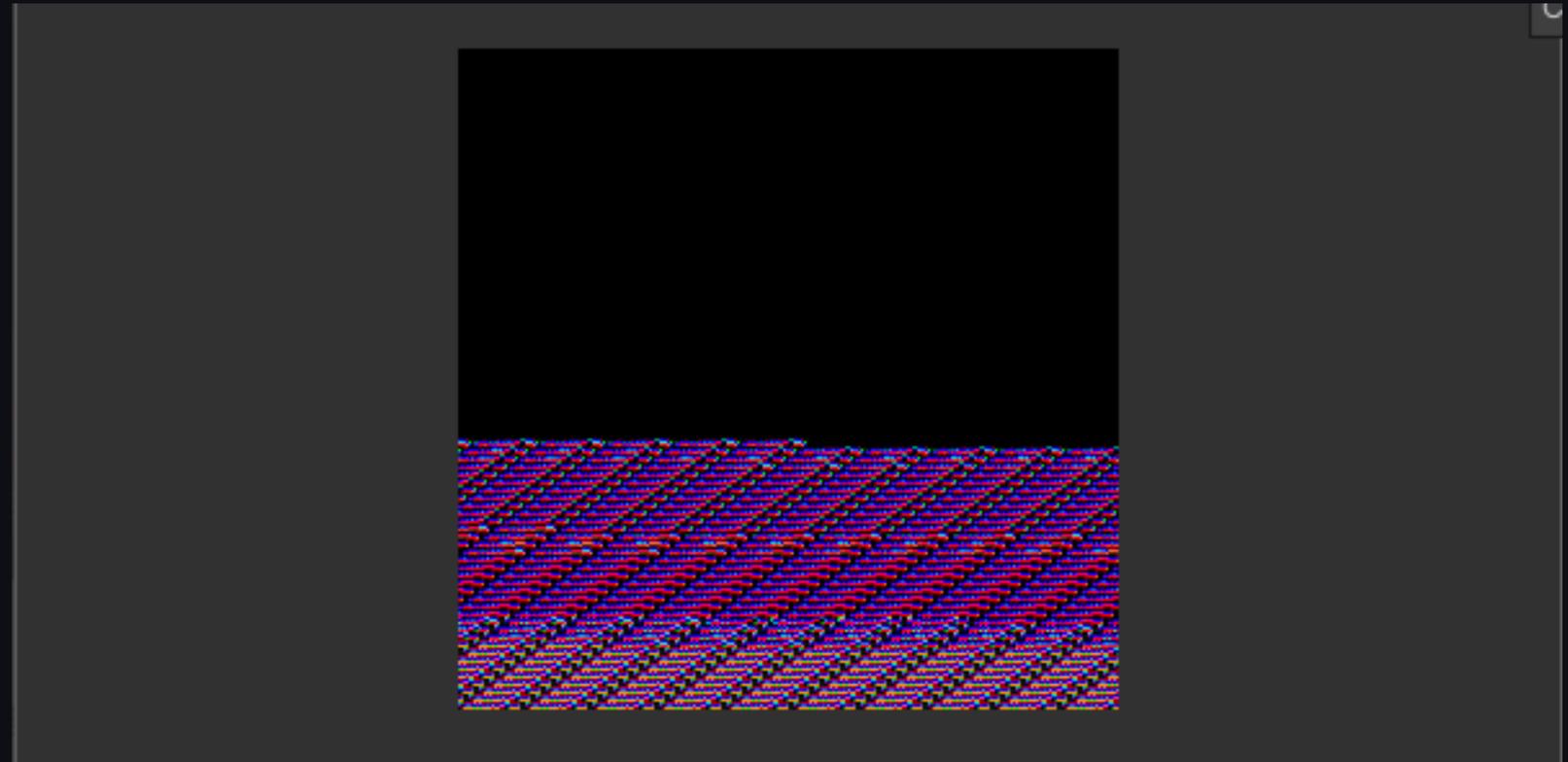
最终经过实际的摆放，10个路点基本就能满足NPC的全部行为需求了。



路点铺设

动作表现

- GPU Skin 表现动作。
- 动作的ID记录在路点信息中。



剔除

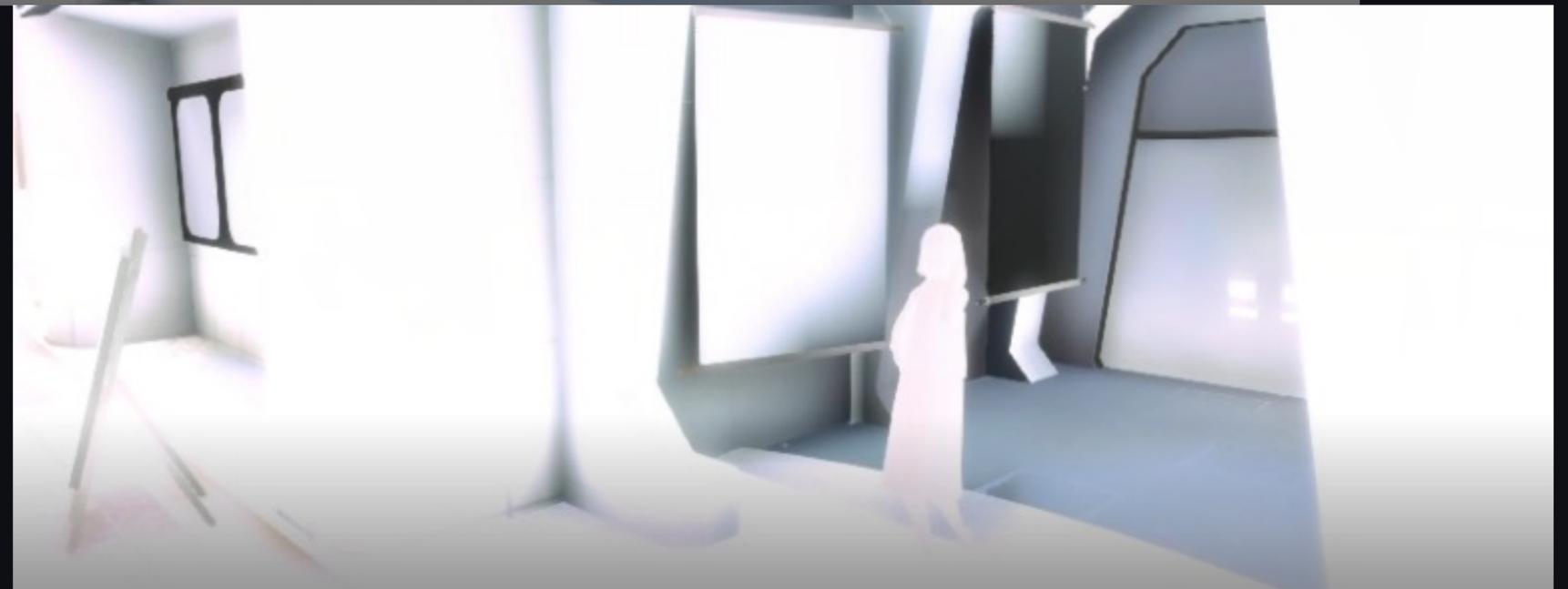
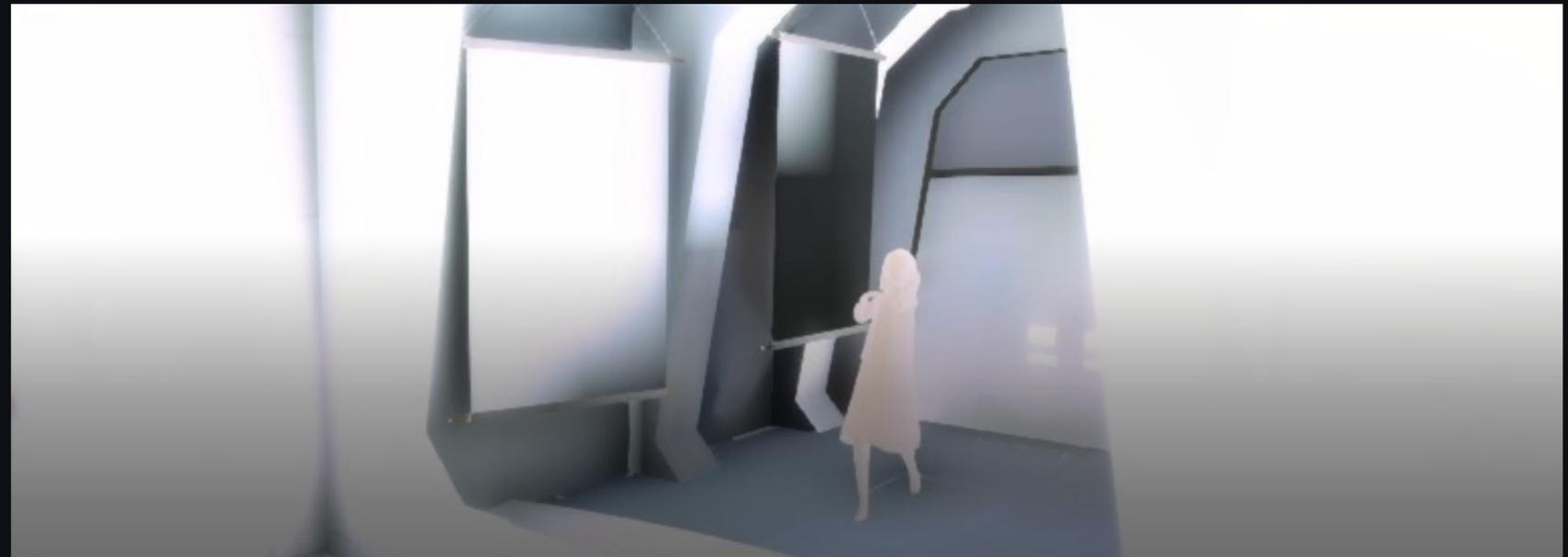
- PC平台使用HIZ
- 移动平台因为显示距离短，HIZ反而是负优化，使用视锥剔除。



HIZ

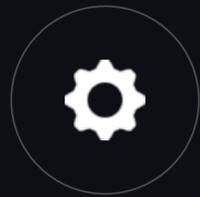
全局光照

- 路点中额外记录当前的Light Probe。
- NPC移动过程中对两个球谐函数进行插值。



全局光照

项目定制需求



大量铺设

- 策划负责配置路线，每一路线上可摆放多个不同NPC，导出时转为渲染所需要的数据格式。



等红绿灯、过马路

- 整个场景的时间统一规划，搭设NPC时，配合红绿灯切换的时机来配置动作节点。



多种手持物

- 将一个NPC的全部手持物混入与NPC本身的Mesh合并，额外添加骨骼及动画配合手持物做表现。



换色

- 采样额外的Mask图来实现。

编辑工具



氛围NPC编辑器

NPC方案

segu_day (Ambient NPC Asset) [新建] [保存] [生成]

NPC路径

添加 [复制] [删除]

- 斜路口1_1
- 斜路口1_2
- 斜路口1_3
- 斜路口1_4
- 斜路口1.5
- 斜路口1.6
- 斜路口1.7
- 斜路口2_1
- 斜路口2_2
- 斜路口2_3
- 斜路口2_4
- 斜路口2_5
- 站前广场_站内换乘
- 站前广场_站内换乘2
- 站前广场_出站向北
- 站前广场_出站向北2
- 站前广场_出站换线
- 斜路口3_1
- 斜路口3_2
- 斜路口3_3
- 斜路口3_4
- 斜路口4_1
- 斜路口4_2
- 斜路口4_3
- 斜路口4_4
- 斜路口5_1
- 斜路口5_2
- 斜路口5_3
- 斜路口5_4
- 斜路口5_5
- 斜路口6_1
- 斜路口6_2

路径属性

基本信息

路径名称: 斜路口1_1

路径类型: 移动且过马路

预估时间: 24.0 [使用]

时间周期: 130

路径点信息

路径点列表

[0] [坐标: (241.2, 99.6, 285.8)]

[1] [坐标: (238.7, 99.6, 290.6)] [路口]

[2] [坐标: (217.3, 99.6, 321.7)]

NPC信息

渐隐渐显:

出生间隔: 7

移动速度: 1.8

预估容量: 3 [使用]

实际数量: 10

路口组: 1

等待队列: 4,5,6,7,8,9

NPC列表

随机填充 [循环填充] [清空列表]

[0] [n_n_m_qingnian02_GPUSkin]

NPC模型: n_n_m_qingnian02_GPUSkin

行走动作: [3] 正常行走

待机动作: [2] 正常待机

配色方案: 2

颜色1: [色条]

颜色2: [色条]

颜色3: [色条]

颜色4: [色条]

[1] [n_n_f_zhongnian02_GPUSkin]

[2] [n_n_f_qingnian01_GPUSkin]

[3] [n_n_m_zhongnian01_GPUSkin]

[4] [n_n_m_zhongnian07_GPUSkin]

[5] [n_n_m_qingnian02_GPUSkin]

[6] [n_n_m_zhongnian02_GPUSkin]

[7] [n_n_m_zhongnian03_GPUSkin]

[8] [n_n_m_qingnian02_GPUSkin]

[9] [n_n_m_zhongnian04_GPUSkin]

性能与兼容性

性能特点：

- 性能消耗最大来源于顶点着色器计算GPU Skin，其次是像素着色器。
- 计算着色器的剔除几乎没什么消耗。
- 氛围NPC作为不透明物体、要求投影。还会参与PreDepth、MotionVector。

性能优化方案：

- 资源上降低顶点数量。
- 考虑缓存命中率划分数据结构。
- 剔除。
 - 随剔除结果保存的数据除NPC数据的Index外还保存了当前路点的Index，这样只需在CS中进行一次查询插值节点的遍历。
 - 所有NPC统一剔除。剔除结果根据Offset读取。
 - 阴影剔除可以缩减范围。

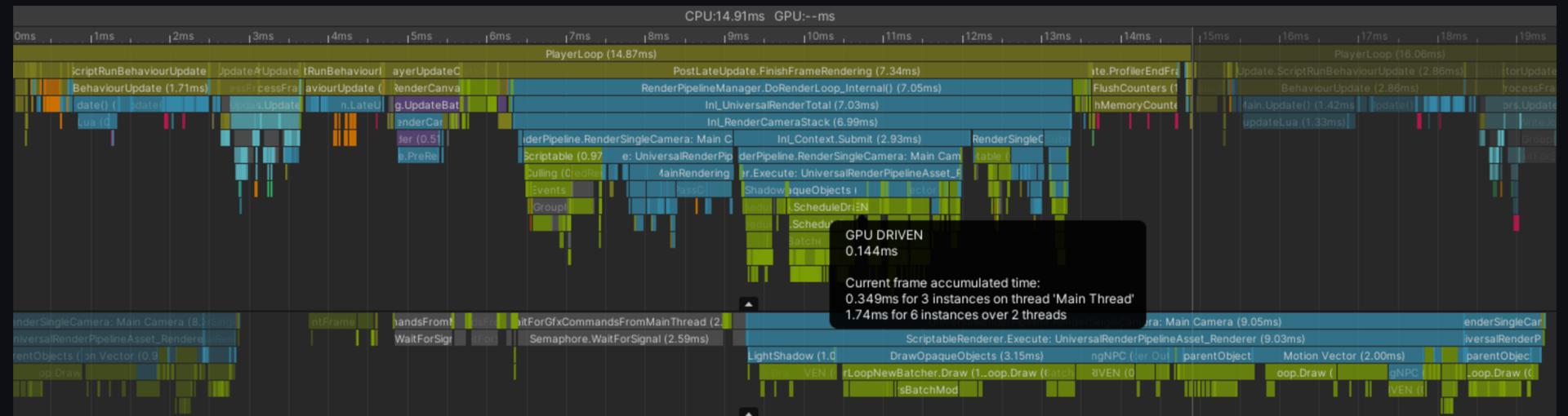
性能优化方案：

- 合并人物与手持物。



涩谷性能测试：骁龙865

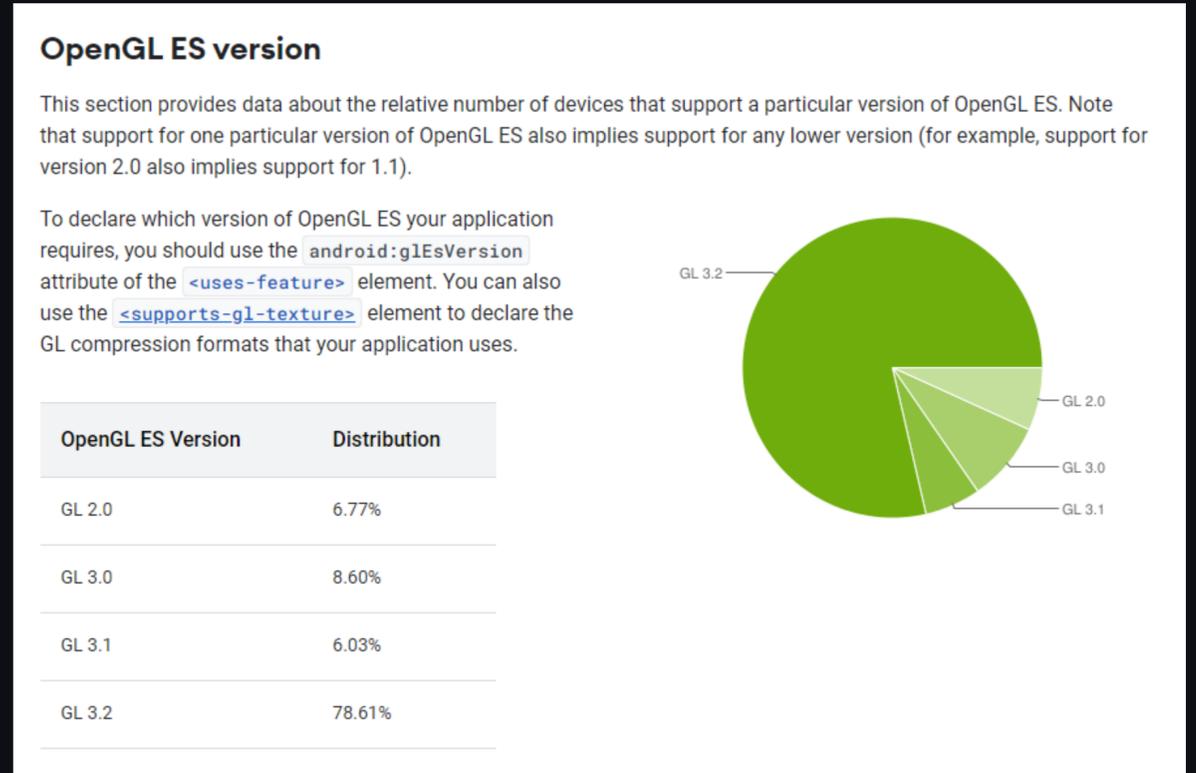
- 560个
- 21种
- 主线程0.34ms
- 渲染线程1.74ms
- GPU消耗2ms左右
- 包含早期开发没有的Motion Vector绘制消耗



GPU Driven NPC

兼容性问题：

- Compute Shader (GLES3.1)
- Buffer选择
 - UBO内存限制。
 - Mali设备在顶点着色器中对 SSBO 的读取存在问题。
 - 依赖Texture Buffer Object (GLES3.2 或 GLES3.1支持 GL_EXT_texture_buffer拓展)。

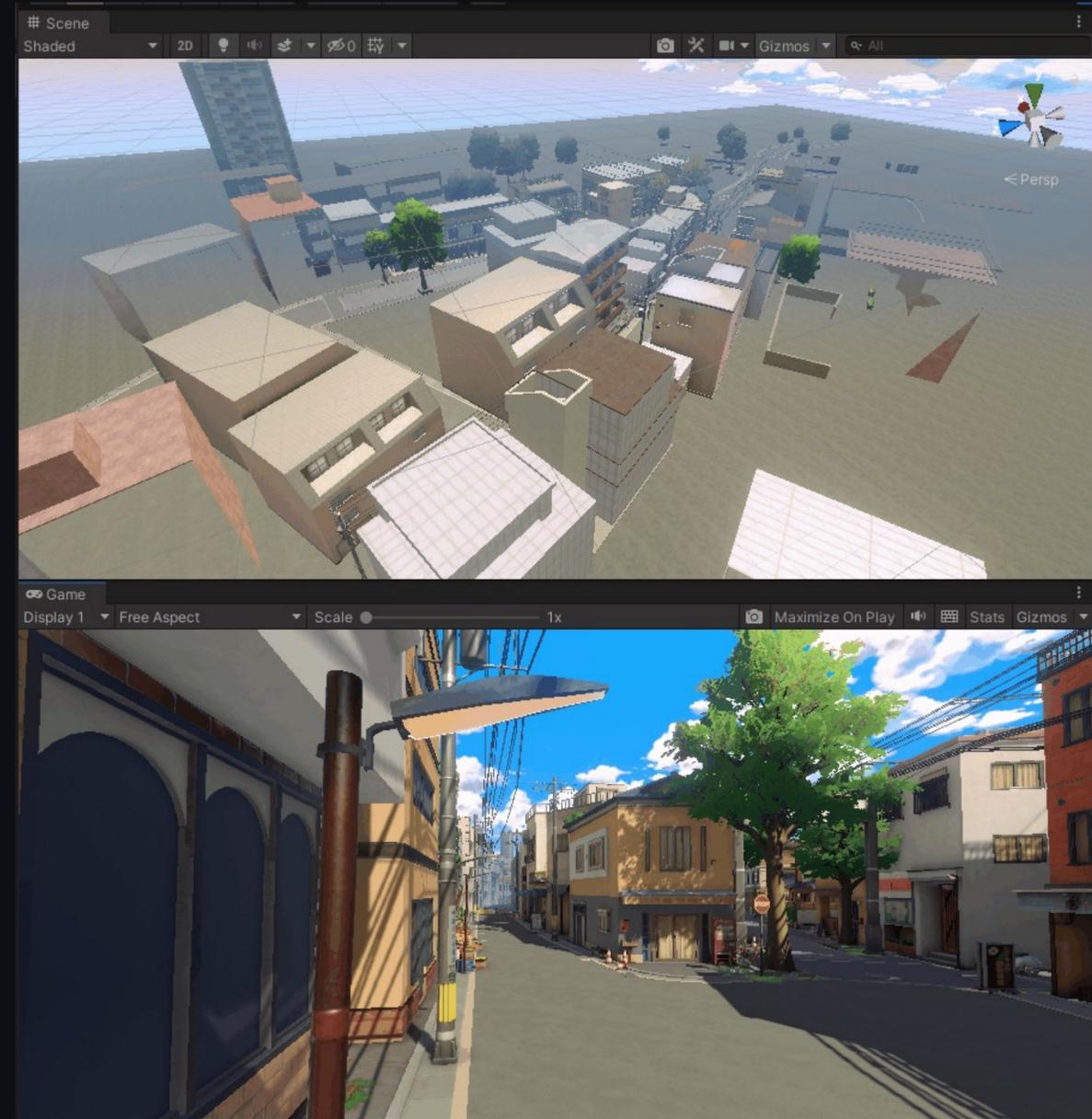


- Adreno 4xx和更高版本
- Mali T6xx和更高版本
- PowerVR Rogue G6100

更多应用

04 更多应用

- GPU Scene





THANKS