

备忘录_曲面细分阶段

笔记本: DirectX 12

创建时间: 2022/9/5 8:31

更新时间: 2022/9/7 9:38

作者: handsome小赞

- 使用 曲面细分的三个理由
 1. 基于GPU实现动态LOD
 2. 物理模拟与动画特效
 3. 节约内存

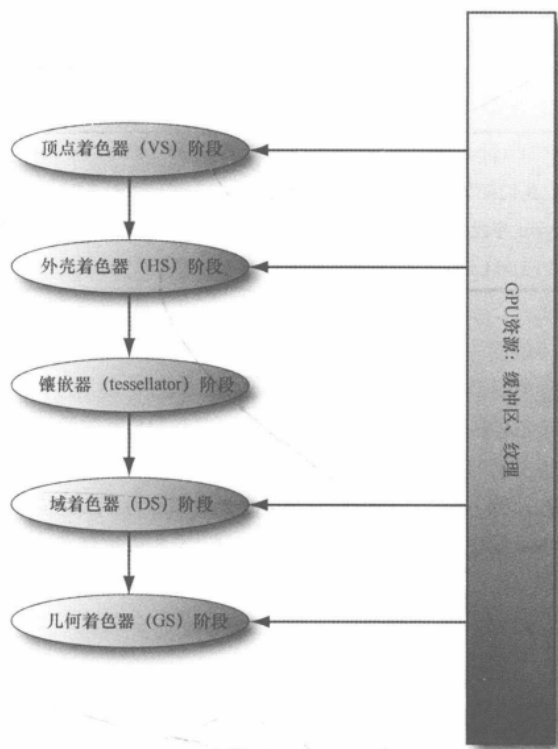


图 14.1 图中展示的渲染流水线子集便是曲面细分阶段（即处于中间位置的 3 个渲染步骤）

-
- 曲面细分的图元类型

- 进行曲面细分时，使用
`ID3D12GraphicsCommandList::IASetPrimitiveTopology` 向 IA（输入装配）阶段提交具有若干控制点的面片，Direct3D支持具有1~32个控制点的面片：
`D3D_PRIMITIVE_TOPOLOGY_1_CONTROL_POINT_PATCHLIST`
...
`D3D_PRIMITIVE_TOPOLOGY_32_CONTROL_POINT_PATCHLIST`
同时，需要设置
`D3D12_GRAPHICS_PIPELINE_STATE_DESC::PrimitiveTopologyType =`
`D3D12_PRIMITIVE_TOPOLOGY_TYPE_PATCH`

- 在开启曲面细分时，顶点着色器就彻底沦陷为“处理控制点的着色器”。动画与物理模拟的计算工作都会在对几何体进行镶嵌化处理之前的顶点着色器种以较低的频次进行。

- 外壳着色器

外壳着色器实际由常量外壳着色器和控制点外壳着色器组成

常量外壳着色器

会针对每个面片逐一进行处理（即每处理一个面片就被调用一次）

它们的任务是输出网格的曲面细分因子

```
struct PatchTess
{
    float EdgeTess[4] : SV_TessFactor;
    float InsideTess[2] : SV_InsideTessFactor;

    // 可以在下面为每个面片附加所需的额外信息
};
```

- 四边形：边缘曲面细分因子 $\times 4$ 、内部曲面细分因子 $\times 2$
 - 三角形：边缘曲面细分因子 $\times 3$ 、内部曲面细分因子 $\times 1$
 - D3D 所支持的最大曲面细分因子为 64 (D3D12_TESSELLATOR_MAX_TESSELLATION_FACTOR)，如果曲面细分因子为 0，则该面片会被后续处理阶段丢弃。这就使我们能够以每个面片为基准来实现如视锥体剔除与背面剔除这类优化。
 1. 如果面片根本没有出现在视锥体范围内，那么就能将它从后续的处理中丢弃（倘若已经对该面片进行了镶嵌化处理，那么其细分后的各三角形将在三角形剪裁期间被抛弃）
 2. 如果面片是背面朝向的，那么就能将其从后面的处理过程中丢弃（倘若该面片已经进行了镶嵌化处理，则其细分后的各三角形会在光栅化阶段的背面剔除过程中被抛弃）
 - 确定镶嵌次数的常用衡量标准：
 1. 根据与摄像机之间的距离。
 2. 根据占用屏幕的范围。
 3. 根据三角形的朝向。
 4. 根据粗糙程度。
 - 性能建议：
 1. 曲面细分因子为 1，那么就考虑在渲染此面片时不对它进行细分处理；否则，便会在曲面细分阶段白白浪费GPU资源。
 2. 考虑到性能又涉及GPU对曲面细分的具体实现，所以不要对小于8个像素的三角形进行镶嵌化处理。
 3. 使用曲面细分技术时要采用批绘制调用（即尽量将曲面细分任务集中执行）（在绘制调用之间往复开启关闭曲面细分的代价极其高昂）
- 控制点外壳着色器**
- 以大量的控制点作为输入输出，每输出一个控制点，此着色器都会被调用一次。
- 该外壳着色器的应用之一是改变曲面的表示方式，增加其控制点

```
[domain("quad")]
[partitioning("integer")]
[outputtopology("triangle_cw")]
[outputcontrolpoints(4)]
[patchconstantfunc("ConstantHS")]
[maxtessfactor(64.0f)]
```

上面的控制点外壳着色器还用到了以下几种属性。

1. domain: 面片的类型。可选用的参数有 tri (三角形面片)、quad (四边形面片) 或 isoline (等值线)。
2. partitioning: 指定了曲面细分的细分模式。
 - a. integer. 新顶点的添加或移除仅取决于曲面细分因子的整数部分, 而忽略它的小数部分。这样一来, 在网格随着曲面细分级别而改变时, 会容易发生明显的突跃 (popping) 的情况^①。
 - b. 非整型曲面细分 (fractional_even/fractional_odd)。新顶点的添加或移除取决于曲面细分因子的整数部分, 但是细微的渐变“过渡”调整就要根据因子的小数部分。当我们希望将粗糙的网格经曲面细分而平滑地过渡到具有更佳细节的网格时, 该参数就派上用场了。理解整型细分与非整型细分之间差别的最佳方式就是通过动画实际演示、比对, 因此, 本章末会有相应的练习来令读者比较两者的差异。
3. outputtopology: 通过细分所创的三角形的绕序^②。
 - a. triangle_cw: 顺时针方向的绕序。
 - b. triangle_ccw: 逆时针方向的绕序。
 - c. line: 针对线段曲面细分。
4. outputcontrolpoints: 外壳着色器执行的次数, 每次执行都输出 1 个控制点。系统值 SV_OutputControlPointID 给出的索引标明了当前正在工作的外壳着色器所输出的控制点。
5. patchconstantfunc: 指定常量外壳着色器函数名称的字符串。
6. maxtessfactor: 告知驱动程序, 用户在着色器中所用的曲面细分因子的最大值。如果硬件知道了此上限, 便可了解曲面细分所需的资源, 继而就能在后台对此进行优化。Direct3D 11 硬件支持的曲面细分因子最大值为 64。

注意 outputtopology 还有一个 point 选项

• 镶嵌器阶段

程序员无法对镶嵌器这一阶段进行任何控制。但此环节会基于常量外壳着色器其