Lightweight Key Technologies for Massive Multi-player Online

WebVR Conferencing

单位信息：

作者信息：

摘要

关键字

1. **引言**

2019年底爆发的新冠病毒疫情在多个维度上改变了人们生活、工作、学习以及协作的模式，多种传统的线下行为在疫情隔离的需要之下迁移到线上环境中。此次疫情对社会运行与协作的模式的影响不仅暂时性的，同时也是持久性的，线上新的生活、工作、学习模式将在此次疫情之后得到重大的推进与普及。同时，基于疫情的影响各地政府部门也相继提出在线新经济发展模式，其归根结底是将传统的线下规划/设计/工程/审核/运维/商务/销售/管理等经济模式，经过轻量化处理后移植移动网页端，设计创新商业闭环后形成线上轻量级新经济模式，以适应用户激增后管理难度大、异地沟通成本高、大规模疫情爆发需自然隔离等社会新问题，进一步刺激并提升互联网经济的潜在活力，促进5G时代经济的新增长。

在疫情的影响之下，远程在线会议的需求激增，ZOOM会议、腾讯会议、钉钉会议等在线会议平台的使用量增幅巨大，也大大提升了相关产品公司的资本估值。但是目前的在线会议系统具有一定的限制性，主要有以下：

1. 当前的在线会议系统不具有可视化与社交特性。基于传统的音视频的方式无法生动的还原会议的实际情境，代入感较弱。
2. 当前的在线会议系统没有整合VR、AR等新型的互动与展示方式。音视频的方式主要采用传统的图像与声音的信息传播方式，沉浸感比VR、AR相比大大减弱。
3. 当前的在线会议系统不具有个性化的角色系统。采用传统头像或是真人呈现的方式，对于不同人群均具有一定的疏离感，不具有互联网社交的属性。

针对上述问题，拟针对后疫情时代下的在线远程会议需求，研发一套基于Web端的在线虚拟现实远程会议协作系统，用来填补市场上虚拟化、个性化的会议系统的空白。主要具有以下特点：

1. 采用虚拟现实的方式进行远程会议环境的呈现，给体验者带来更强的参与感 与沉浸感，使得线上与线下的距离进一步拉近；
2. 采用虚拟化身的方式进行个性化的角色重现，使体验者在虚拟空间中的互动感更加强烈，同时也对虚拟中的形象进行有效的区分，引入个性化社交性元素互远程会议的严肃应用中；
3. 采用Web+VR的方式进一步减低VR的体验与使用门槛，使系统可以基于浏览器进行跨平台、多端的一键式发布，提升用户使用的友好度与易用性；
4. **相关研究工作**

近几年来，VR社交、会议系统已经频频出现在人们的视野中，主要分为两大类第一类是基于PC客户端的VR会议或社交平台平台，如VSWork、Engage等。第二类是基于Web端的在线社交会议会展平台，如Mozilia hubs。

VSWork是国内的一款VR虚拟会议系统，用户可拥有可自定义的虚拟角色，其虚拟角色可以通过手动、照片扫描或3D扫描进行建模，用户可以通过虚拟角色在房间中进行移动或操控场景中的物体。即使对VR不熟悉的用户，也可以借助管理员的控制对场景进行浏览，而无需自己操作。同时，VSWORK的全部内容都采用云端部署，终端设备的本地负担很小，同时不同客户的内容都对应独立的服务器，以保证其文件与信息的安全性与私密性。此外，VSWORK的开发者版针对有开发能力的客户或第三方开发者开放SDK，开发者可以根据VSWORK的SDK开发满足其客户需求的技术方案。而Engage由VR教育公司IVRE开发，它支持与会者进行一对一的社交互动，并能提供高清的视觉效果及大型活动所需的管理工具，并且具有可扩展的后端来支持全球观众的实时需求。同时，其支持VR设备的同时也可在PC端使用。HTC与IVRE建立了战略合作伙伴关系，HTC将在全球范围内发行该平台，共同推广XR办公的新模式。上述两个系统全部是采用Unity进行开发，较为重量级；发布模式仍是PC端运行的可执行程序，用户使用门槛较高；支持有限度的互动与远程协助，在会议本身的体验上相对于传统的会议系统有较大的差距；

Mozilla hubs是运行在浏览器中的虚拟协作平台，无需安装其他插件，通过链接邀请他人进入自创的虚拟空间中进行相关的交互操作，并提供对VR设备的支持。可用于主持会议、在线教育等领域。其拥有的化身定制、场景定制是其非常具有特色的功能，场景定制中提供了高自由度的交互方式，支持用户自由上传模型，调整角度等操作，总的来说，相较于前面介绍的两个系统，Mozilla hubs具有更加轻量级、用户交互操作更多等特点，但是在某些场景的美观性和精度、加载速度上仍不及基于PC的系统。

LOD技术：

LOD技术即Levels of Detail的简称，意为多细节层次。LOD技术指根据物体模型的[节点](https://baike.baidu.com/item/%E8%8A%82%E7%82%B9/865052" \t "https://baike.baidu.com/item/LOD/_blank)在显示环境中所处的位置和重要度，决定物体渲染的资源分配，降低非重要物体的面数和细节度，从而获得高效率的渲染运算。

实例化渲染：

实例化（instancing）或者多实例渲染（instanced rendering）是一种连续执行多条相同的渲染命令的方法，并且每个渲染命令所产生的结果都会有轻微的差异。这是一种非常有效的，使用少量API调用来渲染大量几何体的方法。

PM 渐进式网格：

**渐进式网格**是动态[细节级别](https://en.wikipedia.org/wiki/Level_of_detail_(computer_graphics)" \o "详细程度（计算机图形学）)的技术之一。此技术由Hugues Hoppe于1996年引入，该方法使用将模型保存到结构中的渐进网格，该网格允许根据当前视图平滑选择细节级别。实际上，这意味着可以一次显示最低细节级别的整个模型，然后逐渐显示更多细节。缺点之一是相当大的内存消耗。优点是它可以实时工作。渐进式网格也可以用在计算机技术的其他领域，例如通过Internet或压缩逐步传输数据。

1. **总体技术路线**

志成和恩旸在这里补充一个技术路线图（突出几大关键技术），尝试画一下（用ppt来画图），大家一起来补充完善。

轻量化处理的核心是资源重用，我们使用该技术必须首先明白哪些资源是可重用的，并且尽可以提高资源的重用度。另外为了效果要保留丰富的多样性，参数化调整每个对象，另外可以通过对不同资源进行搭配组合来提高多样性。

首先将模型划分为多个区域（在这个问题中是划分了头部、上身、下身三个区域），每个区域可以匹配不同贴图（在这个问题中使用了男性32套贴图，女性16套贴图），这样就可以通过不同贴图搭配各种效果（男性有32^3=32768种组合，女性有16^3=4086种组合，共36864种组合）。另外，可以通过对人物对象的高矮胖瘦进行设置，还可以对对象的色调进行编辑（这个问题中主要是对下身的裤子颜色进行设置），在骨骼动画方面，可以为每个人物对象设置不同的动画播放速度。

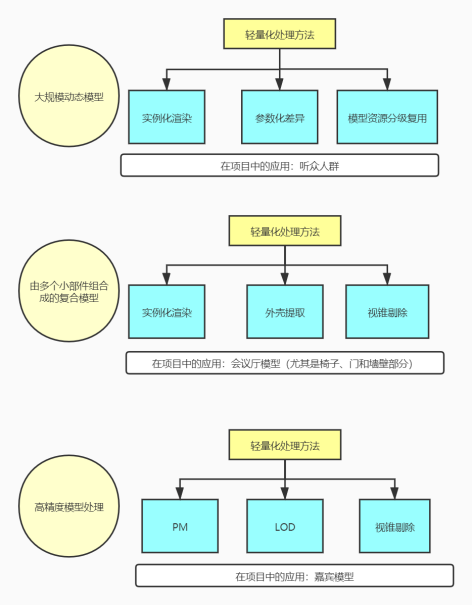


图 1 不同3D资源的轻量化处理思路

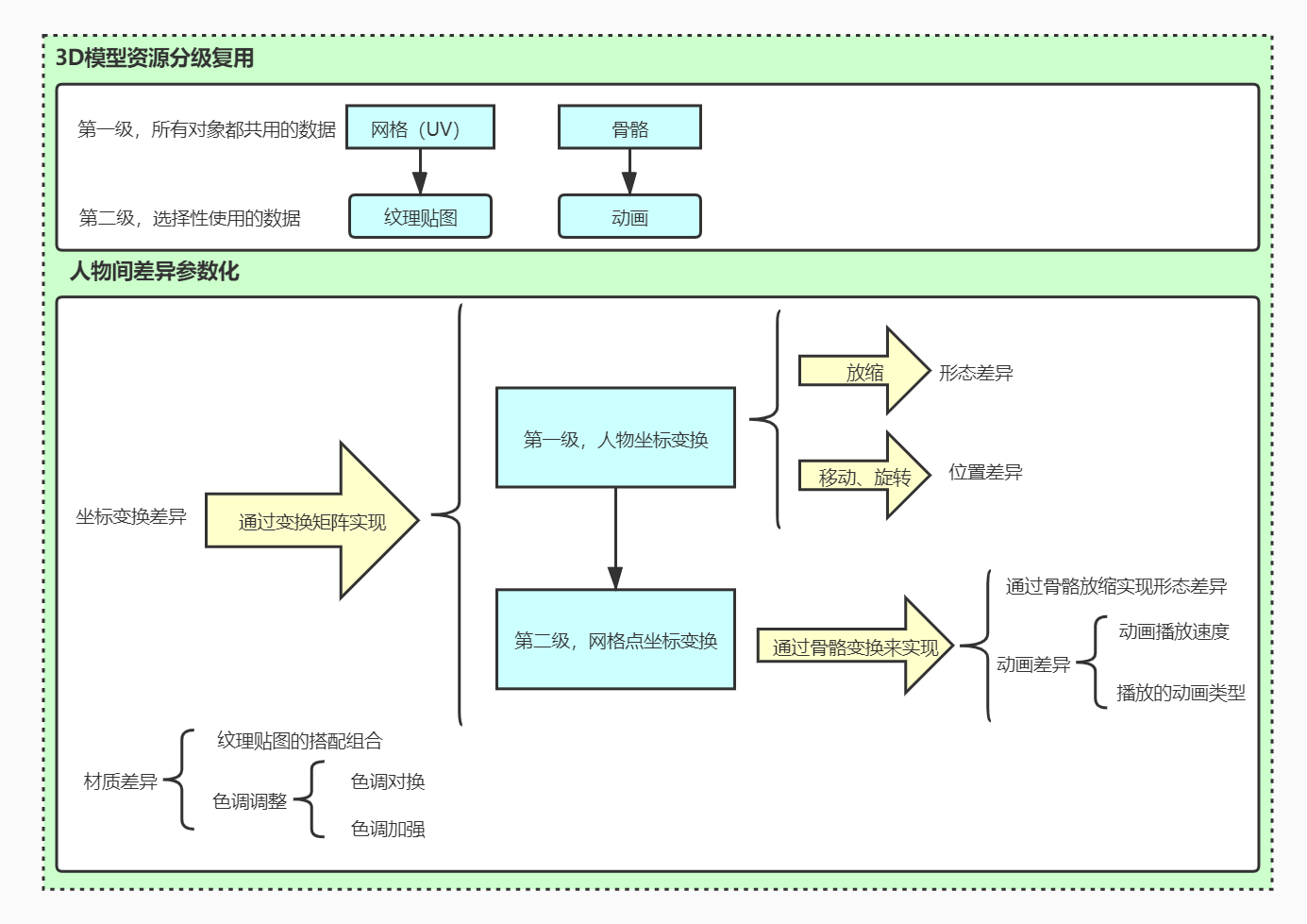


图 2 大规模人群的轻量化处理思路

1. **关键技术之一：大规模会议情境的轻量化预处理**

志成写一段引文

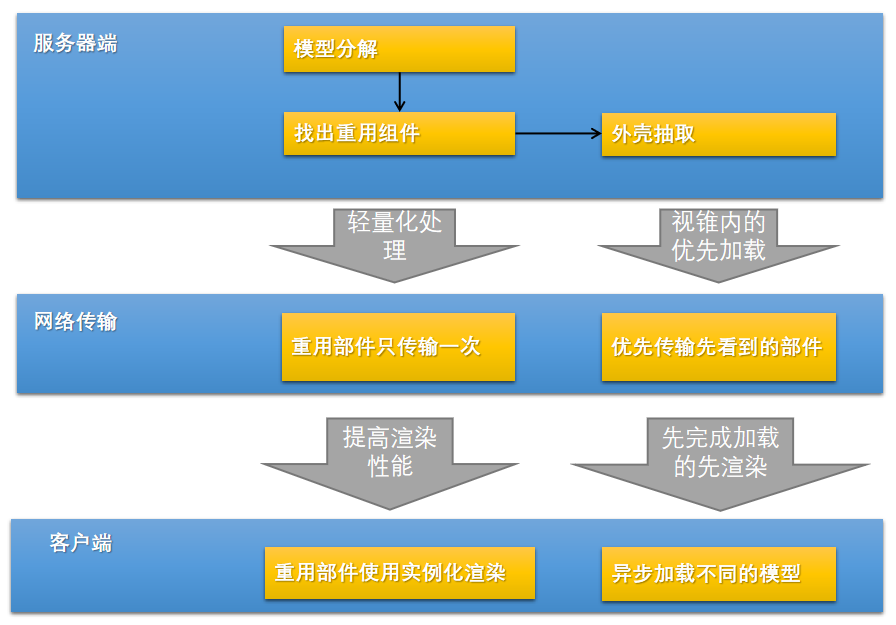


图 3 由多个小部件组合成的复合模型的轻量化处理思路

4.1**静态会场场景的轻量化预处理**

志成完善下列文字

大规模人群渲染需要存储的信息可以分为三级。第一级，所有对象都共用的数据，如模型的网格信息（网格点位置，UV等信息）；第二级，选择性使用的数据（如纹理贴图），这类数据每个对象只需要其中的一部分数据（如模型贴图有多套可供选择，但每个对象只使用其中的一套）；第二级，描述对象的参数信息（如，贴图类型，动画播放速度，高矮胖瘦，色调），这些信息每个对象都可以不同，所以每个对象的这些参数信息都要单独存放。

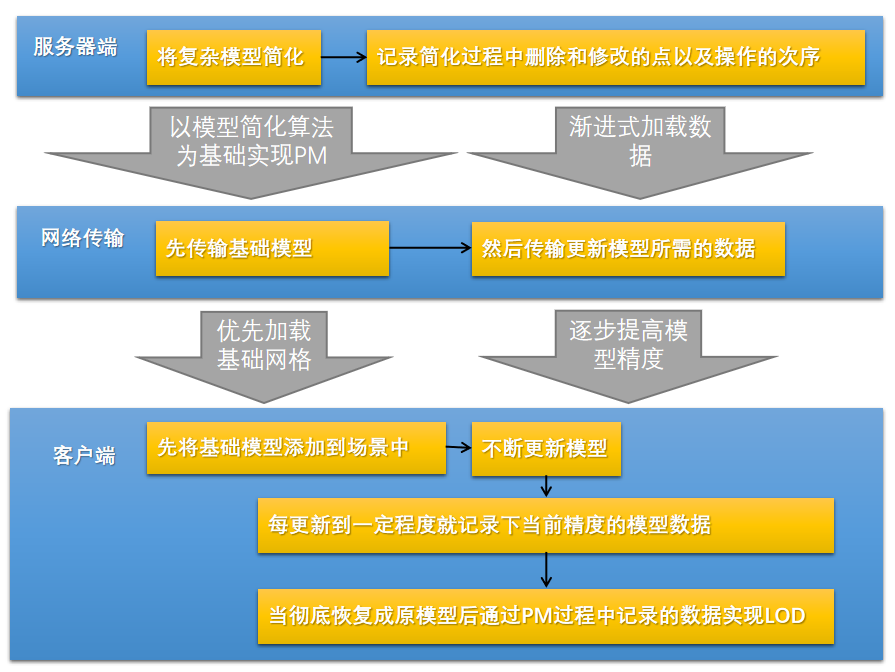


图 4 高精度模型处理（PM+LOD）

4.2**大规模参会人群的轻量化预处理（参考朴雪论文中相关内容）**

志成/恩旸完善这部分文字

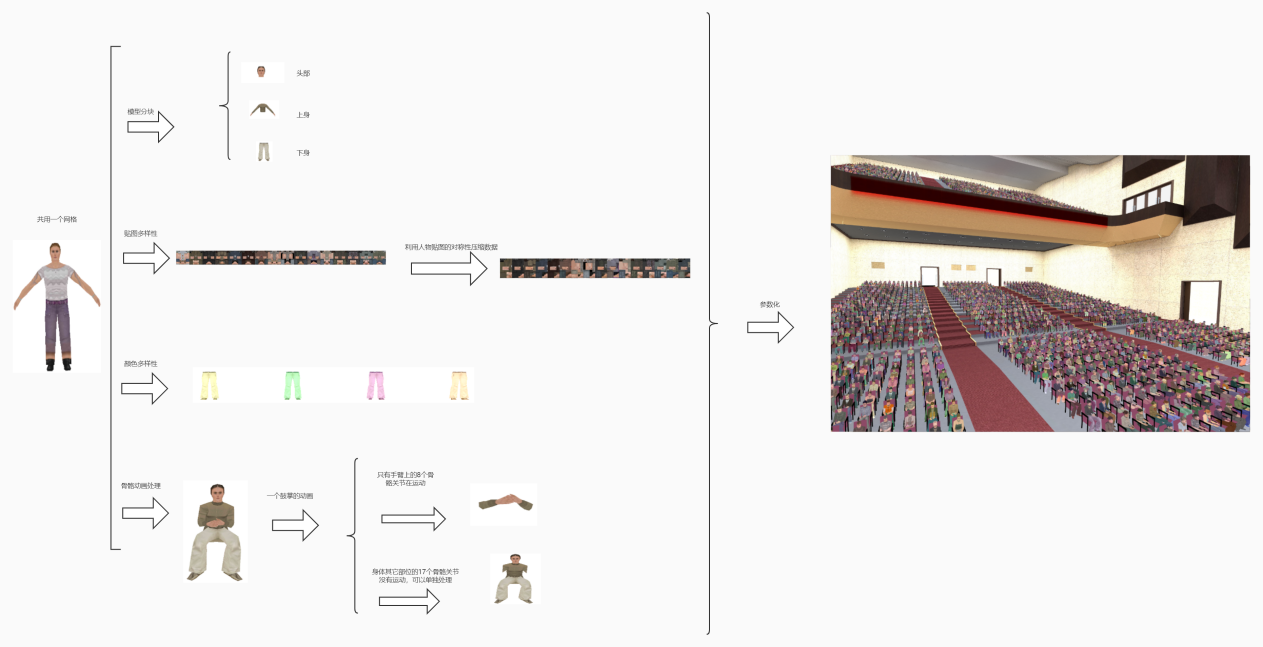


图 5 大规模人群轻量化处理技术的运用

在项目中我们只需要实现听众鼓掌的动作，这个动作只涉及到手臂的8个骨骼，其它的17个骨骼的状态没有发生变化，所以首先可以将骨骼数据分成两部分，手臂处骨骼和固定动作的骨骼。

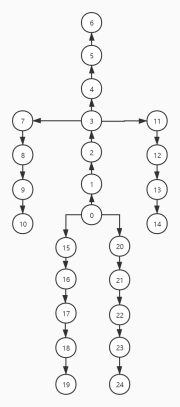


图 6 人物模型骨骼结构的分析（未完成）

每个骨骼的变换矩阵因为最后一行是固定值，所以一个变换矩阵只需要存储12个float数据。有17个骨骼在动画播放的过程中的状态没有发生变化，所以这可以提前计算好这17\*12=204个数据。涉及到手臂的8个骨骼，由于我们在项目中只需要实现鼓掌这个简单的动作，经过测试使用16帧就可以获得比较好的动画效果，这是16帧的鼓掌动画中后8帧可以看作前8帧的倒放，所以我们只需要8个骨骼8帧分别的变换矩阵，这8个骨骼需要的数据量为8\*8\*12=768。将这些数据在预处理阶段直接计算好可以减少在客户端的计算量。

1. **关键技术之二：细粒度化渐进式传输调度（参考李柯论文相关内容）**

5.1轻量化缓存管理

**志成这一块**

将场景中需要的数据细化拆分，相同部件只传输一次。

分级传输，提高初始加载速度。人物贴图进行了多细节层次处理，先传输低像素的纹理贴图，再传输高像素的贴图。

5.2基于兴趣度的细粒度化在线打包

**志成写这一块**

5.3 带宽自适应的渐进式传输调度

志成写这一块

为了确保复用的人物对象资源不被重复存储，需要将场景中的人群统一管理，我们用JS设计了以一个对象来管理整个的人群。

为了提高渲染效率，所有的人物在渲染时作为一个整体，一次性将人群所需的全部数据输入着色器。这就需要我们管理好每个人的所有参数信息（位置，大小，贴图类型，等等），渲染时一次性将这些信息传入shader。

**6 轻量级大规模会议场景在线渲染**

志成完成这一块

骨骼动画数据的输入格式是32的float浮点数，经过测试发现骨骼动画数据实际上并不需要这么高的精度，所以我们设计了一种16位的浮点数格式，这样就使得传入着色器的骨骼数据量减少了一半。

项目中人物的贴图左右对称，所以可以只传入着色器左半部分贴图，这样传入着色器的贴图数据量就减少了一半。

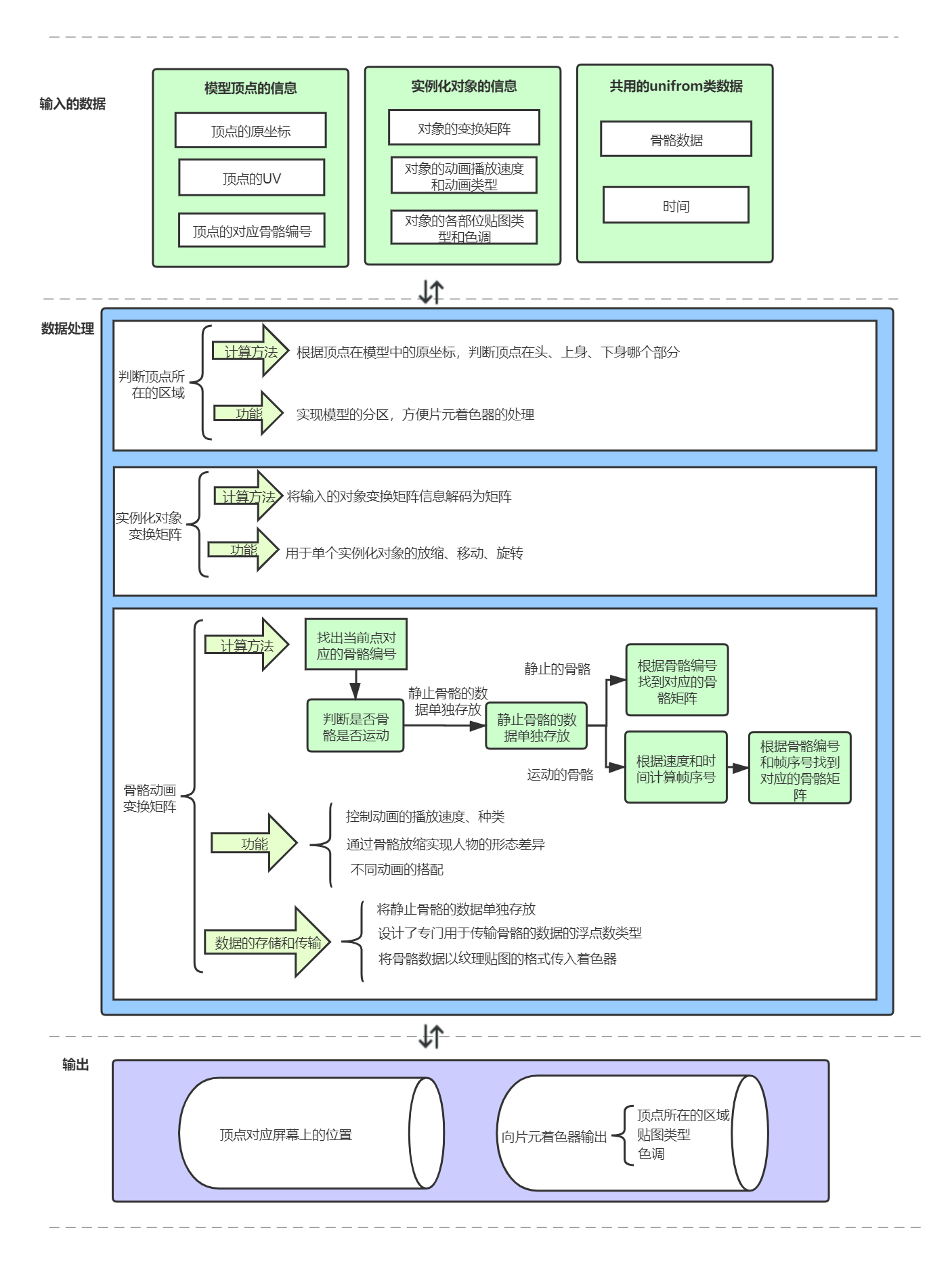


图 7 顶点着色器的优化处理

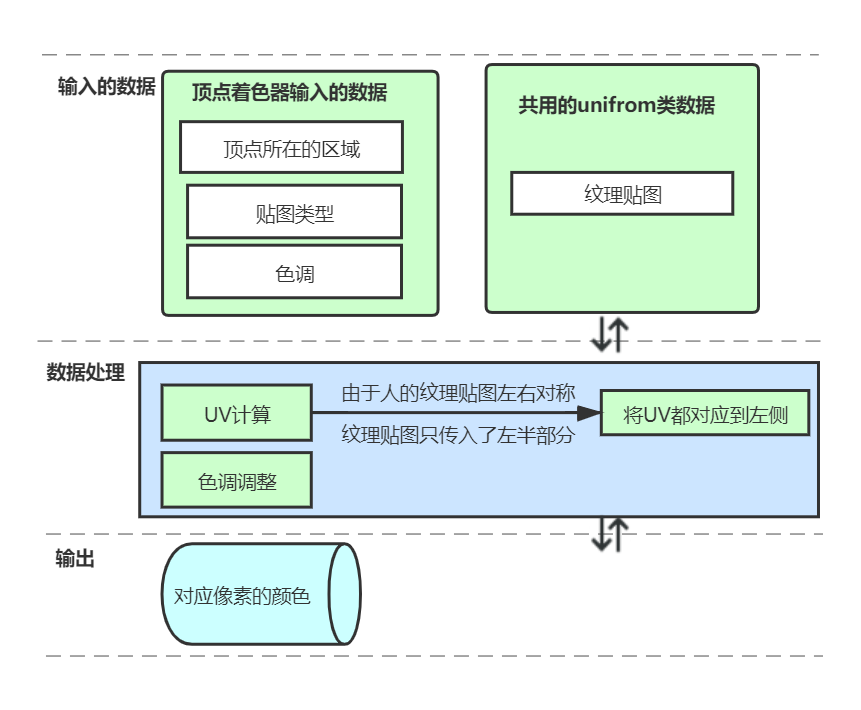


图 8 片元着色器的优化处理

5.实验结果与性能分析

志成/恩旸搞这一块

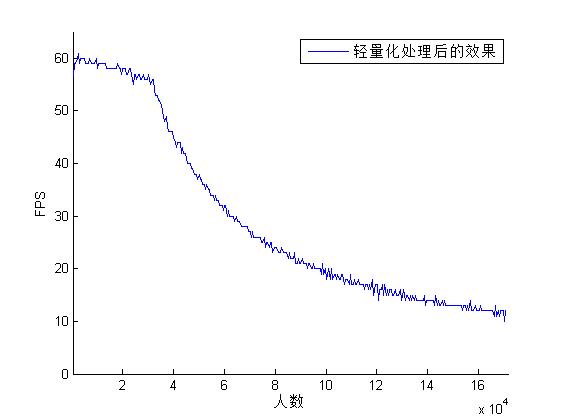


图 9 大规模人群的轻量化处理的效果

1. 结论

志成/恩旸搞这一块

1. Reference

Mozilia会议系统

Engage会议系统

VSWork会议系统