个人总结

# 技术工作

在本次项目中，我负责的工作是vmd文件的导出模块，目标是将json文件中的人体动作数据表现在MMD中的人物模型上。在经过几周的努力后，我成功地实现了人物上半身动作的重现，而下半身动作的重现由于工作量与难度过大，我们决定放弃实现。

在这个过程中，我具体进行的工作有：vmd文件格式的解读、vmd文件的写入方式、MMD中的坐标系探索、MMD中人物动作控制方式的探索、MMD中关节点的旋转的计算、MMD中IK骨的计算探索。

## vmd文件格式的解读

vmd文件是二进制文件，里面装着不同类型的数据和不同编码的语言，包含的数据类型有uint32\_t和float，包含的编码有ASCII、shift-JIS（日文编码）。vmd的文件格式很像网络协议格式，某一位到某一位有什么含义。

vmd文件的大致格式如下：头部、骨骼关键帧记录、表情关键帧记录、镜头关键帧记录和光线关键帧记录。在项目中，我们只需要完成头部以及骨骼关键帧记录的写入即可完成一套动作的重现。

头部包含的信息有版本信息和模型名字，编码格式都是ASCII编码，其中版本信息长度为30个byte，模型名字为10个或者20个byte，依版本而定，如果版本信息或模型名字长度不够，则用’0x00’进行填充。

其次是骨骼关键帧记录，该部分可分为两个部分，一是骨骼关键帧数量，类型为uint32\_t,长度为1个byte，意义是后面跟随的骨骼关键帧记录数量；二是骨骼关键帧记录，一个记录的长度为111个byte，里面包含的信息有骨骼名称（boneName），关键帧时间（frameTime），空间坐标（position），旋转四元数（rotation）、补间曲线（curve）。其中骨骼名称的编码类型为shift-JIS，长度为15个byte，长度不够用’0x00’进行填充，关键帧时间的类型为uint32\_t，长度为1个byte，意义是当前记录对应的关键帧，空间坐标是3个float类型的数据，大小为12个byte，旋转四元数是4个float的数据，大小为16个byte，补间曲线的是16个float的数据，大小为64个byte，意义是两个关键帧之间的变化快慢。

## vmd文件的写入方式

在知道了vmd文件格式后，我们可以通过C++的fstream流来写入二进制文件，但是在写入shift-JIS编码的骨骼名称的时候遇到了问题，出于整个项目编码统一性的考虑，我们需要在Unicode的编码下写入shift-JIS编码的骨骼名称，而同一个日文的shift-JIS编码与Unicode编码并不一样，所以并不能直接写入。

解决办法如下： 先在shift-JIS编码下把需要用到的的骨骼名称写到二进制文件中，然后通过UltraEdit打开该二进制文件，找到并记录骨骼名称的shift-JIS编码对应的十六进制数据，这样觉得到了需要的骨骼点的在shift-JIS下的编码，然后在Unicode下中直接写入骨骼对应的十六进制数据即可。

## MMD中坐标系的探索

这是MMD比较坑的一点，MMD中的坐标系是左手坐标系，而我们在使用的Qt的qquaternion库来进行四元数的计算时使用的是右手坐标系，这里需要左右手坐标系的转换。但是MMD使用的是左手坐标系这一点并不容易发现，因为在MMD中对骨骼点的旋转欧拉角在X轴显示的是左手系的旋转角度，在Y和Z轴显示的是右手系的旋转角度，即虽然MMD在计算人物的角度时统一是在左手坐标系下进行计算，但在界面显示旋转角度时只有X轴的旋转角度是左手系下的旋转角度，在没有意识到这一点之前，我一度以为MMD中的坐标系是世界上的第三种坐标系。

## MMD中人物控制方式的探索

在MMD中，一共有两种关节点，一种是普通的关节点，控制骨骼的旋转，另一种是IK骨，除了有普通关节的作用外，还有“牵一发而动全身”的作用，以左足IK骨为例，左足IK骨对应的是左脚踝关节，通过设置左足IK骨的位置，可以控制整个左腿（包括左膝关节，左脚尖）的位置。第一种关节点的position参数没有意义，IK骨的position参数与rotation参数都有意义。

在MMD中，人物上半身是没有IK骨的，上半身的根节点是大概在肚子的位置，通过每个关节点的旋转来控制上半身的运动，和unity中控制人物动作的方法一致，较为简单。对于下半身，是有IK骨的（左足IK，右足IK，センター），下半身的根节点与上半身的关节点的位置相近，MMD对于下半身的控制较为复杂，大概的思路是：通过IK骨的位置和人体关节点的生理约束来确定与该IK骨有关的整段骨头的位置（比如左足IK骨控制整个左腿），然后再通过该段骨头上每个关节点的旋转角度来做微调。对于IK骨具体的机制我也没有搞清楚，考虑到项目的时间限制，我并没有在这里做深究。

## MMD中关节点的旋转的计算

MMD使用的是左手坐标系，在进行计算时需要进行左右手坐标系的转换，其次需要注意的是每个关节点都有自己的坐标系，并且会受到上一层关节点的坐标系的影响。计算旋转角度的方法大概如下，以计算上半身躯干的旋转为例，先计算从胯部到颈部的一个方向向量，再计算从左肩到右肩的一个向量，然后两个向量叉乘得到垂直于身体向前的一个向量，然后通过该向量与方向向量确定一个三维空间下的朝向，再计算该朝向与初始朝向的旋转四元数即可。这是顶层关节点的旋转角度计算，对于下一层的关节点的旋转四元数，在进行上述计算步骤后，需要左乘上一层关节点的旋转四元数的反转来得到该关节点的坐标系下的旋转四元数。

## MMD中IK骨的计算探索

本以为由json文件中的骨骼空间坐标推导出IK骨的坐标会比较简单，因为两者都是空间坐标，但当我深入去探索之后，发现IK骨的空间坐标计算远比旋转的计算要复杂。先不考虑IK骨牵连的关节点的旋转，在仅仅考虑IK骨的空间坐标的情况下，我们就可以完成大部分的腿部动作的实现，但IK骨的空间坐标并没有想象中的那么好算。

我们不得不考虑的是原视频中人物的腿长与MMD中使用的模型的腿长的缩放系数，这是个不好确定的参数。首先MMD中人物模型的脚长是个不确定因素，因为不同人物模型的腿长会不一样，其次，在对openMMD导出的json数据进行分析后，发现即使在使用三脚架来拍摄较为稳定的视频的情况下，人物的腿长也并不是一个稳定的数值。然后我们要考虑的第二个问题是如何区别突兀的数据与抬腿等正常的数据，即使在我们取站定不动的情况下的前几十帧的平均值作为腿长的情况下，并设置在一定的波动范围内不考虑变化，仍然会有一些比较突兀的数据会让模型动起来，而我们无法仅通过一个坐标点的信息来区别突兀的数据和抬腿等正常的动作，所以我们不能简单地剔除掉突兀的点。

我有想过一些解决办法，例如计算膝盖旋转的角度来辅助判断，但这并不好实现，而且还有x轴的突兀数据，z轴的突兀数据，仅仅靠一两个关节的旋转角度并不好做出判断。于是我想到，我们可以通过整个人体的空间坐标信息来判断我们当前处于一个怎样的状态（例如走路，转身，跳，蹲等），然后给IK骨输入相对的坐标信息来实现MMD中模型的动作（即输入预先设置的坐标组合），但这需要大量的工作，并不是我一个人能在短时间内完成的。

考虑到项目开发所剩的时间已经不多，对于上半身的旋转还有优化工作要做，再考虑到下半身对IK骨有关的计算问题十分复杂，在与组长商量后，我们决定放弃。

# 项目管理

在本次项目开发中，我的身份是一名编码人员，也是团队中的攻坚士兵，在分配任务的时候，我们都没想到vmd的计算会如此复杂，所以只分配了我一个人进行工作，这也导致了最终的攻坚失败。

在平时的项目开发日程中，我的出勤率算是比较高的，仅有一个上午和一个下午的请假，平时的工作也听从组长的安排。但在进行完概要设计后，由于探索性工作花费了比较多的时间，并没有及时地上传工作文档和代码，这一点做的并不好，以后需要改进。

# 团队合作

在需求分析与概要分析阶段，我们全组都有在一起进行积极的讨论与交流。在编码阶段，由于我的工作与王恺的工作有部分重合的地方（计算关节点的旋转），所以我们在工作时有进行比较多的交流与合作，在这里也很感谢王恺对在我对MMD中的旋转计算一头雾水的时候给予我的帮助。然后在测试阶段，我又成了张熠鹏的模特，我们一起设计并完成了测试工作。在平日的工作里，我们都会进行积极的交流以确保工作的顺利的进行。

# 软件产品

最终做出的软件产品，由于我负责的vmd导出模块并没有完成定下的需求目标，只能处理并重现上半身的动作，大大地限制了用户的使用范围，在这里，我也要对我们项目组致歉，限制了我们软件产品的功能与完整性。