

骨骼动画

骨骼动画可以使得人物的动作更加活泼生动，当然这需要更为复杂的计算才能得到。下面我们就来详细分析下骨骼动画的原理，首先我们简要描述下骨骼动画的运行过程。

我们在创建 Mesh 的时候会在其中加入部分的骨骼顶点信息，这些骨骼顶点如同人体的骨骼一样，相互牵连，一块的运动会导致相连部分的运动。如下图 1 所示，这是人的手臂的骨骼示意图，肩部骨骼节点的运动会导致肘部一下部位都发生移动(当然没有人体那么微妙，肩部运动，手没有发生移动也可能发生)。

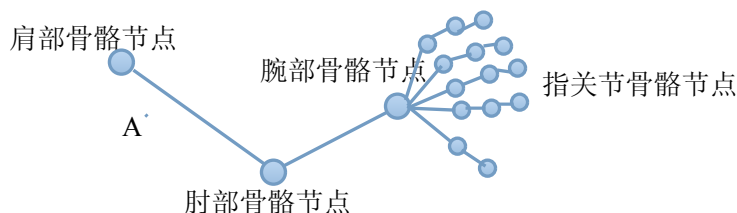


图 1 骨骼节点示意图

骨骼动画的第一步，就是要计算出当前时间每个骨骼顶点的变换矩阵，这个变换矩阵就是这个骨骼顶点相对于其原始位置的运动。例如肩部骨骼节点的变换矩阵是 $M1$ ，肘部是 $M2$ ，腕部是 $M3$ 。

第二步，将骨骼的运动反映到皮肤上。每个皮肤节点都和几个骨骼节点有一定联系，即一个骨骼节点的变换会导致相应的皮肤顶点发生变换，通常都有一个比重。例如肱骨上的某一点皮肤顶点 A，距离肩部骨骼节点为 0.3，距离肘部骨骼节点为 0.7，则该皮肤顶点受肩部骨骼节点的影响因子就是 0.7，受肘部骨骼节点的影响因子就是 0.3，因此顶点 A 在骨骼动画中的变换矩阵即为 $0.7 * M1 + 0.3 * M2$ 。

第三步，每个顶点都进行这样的计算，得出骨骼动画的变换矩阵，然后在局部坐标中进行变换即可得到骨骼动画的效果。

从上面的分析可以看出，为了正确的做出骨骼动画，我们需要采用适当的数据结构来表示骨骼动画中相关的数据结构用来计算出每个骨骼顶点的变换；其次我们需要在 mesh 顶点和骨骼顶点之间建立一个表来反映多少个骨骼节点对其有影响，并影响因子是多少；最后我们要创建一个动画类来驱动上述的数据。

描述骨骼顶点信息，我们采用了 Bone 这个数据结构来表示，在它下面可以挂接任意的子 Bone 节点。然后，我们创建了一个 Skeleton 的类来表示，每个 Skeleton 有一个根节点，同时在 Skeleton 中对 Bone 进行了编号，这些编号用于和皮肤节点之间建立联系。每个编号的骨骼在 Entity 中建立一个 Matrix 表示其变换的结果。

在读取 Mesh 时，我们就对 Bone 进行了编号，同时跟每个皮肤顶点绑定了起来，这个绑定信息反映在 SubMesh 中，我们规定一个皮肤顶点最多和四个骨骼节点发生联系。

最后是骨骼动画的描述。骨骼动画和节点动画的思路非常相似，每个动画都是由很多的 track 组成，每个 track 影响一个骨骼节点，每个 track 是由很多的关键帧组成的，做动画的过程就是从众多的 track 当中选取两个 track 进行插值得到最后的变换结果。这里涉及到 SkeletonAnimaton 类和 SkeletonNodeTrack 类。

还有一点值得提示，就是骨骼动画是在着色器程序中实现的，每个 Entity 都先要将每个骨骼节点的变换矩阵发送给 shader，shader 中根据皮肤顶点相关的骨骼节点的编号来获取其变换矩阵进行计算。

具体的 mesh 文件分析和骨骼动画计算过程见代码。