
algorithm 1 vertexfragment

InputData:

```

1: vertexInf={ position, UV, skinIndex }//Information about the current vertex
2: instanceObjInf={ //Information about the instantiated objects
3:   matrix,
4:   animationStyle,//The type and speed of animation
5:   textureStyle,//Types and tones of maps
6: }
7: uniformInf={ //Data shared by all instantiated objects
8:   skelontonDate,//Bone data
9:   time,//Time is used to calculate the current frame number
10: }
11:
12: if bone(vertexInf.skinIndex)have animation then
13:   frameIndex = time * speed mod(frameIndexMax + 1);
14:   address0 = addressGet1(skinIndex,type,frameIndex);
15: else
16:   address0 = addressGet2(skinIndex,type);
17: end if
18: matrix1  $\leftarrow$  skelontonDate at position address0;
19: matrix2  $\leftarrow$  instanceObjInf.matrix;
20: glPosition = modelViewProjectionMatrix * matrix2 * matrix1 * position;
21: judgeArea(); //Determine which part of the body the current vertex is in

```

OutData:

```

22: glPosition //The current vertex corresponds to the position on the screen
23: sendFragmentShader={ //Information sent to fragmentshader
24:   areaType,//The area where the current point is located
25:   UV,   textureType,//Map type
26:   color,//For tone adjustment
27: }

```
