Algorithm 1: component matrices computing

Input:

vertexInf={ $position$， $skinIndex$,$coordinateUV$ }

avatarParameter={

$affineMatrix$,

$animationType$,

$animationSpeed$,

$headTextureType$,

$upperBodyTextureType$,

$trousersTextureType$,

$neckHeight$,

$waistHeight$,

}

BonesAffineMatrixs ;

animationPlayTime ;

projectionMatrix ; viewMatrix ; modelMatrix ;

textureMapping ;

Output:

vextexScenePosition ;

fragmentColor ;

---

if *this bone have animation* **then**

  $frameIndex = (animationPlayTime * animationSpeed)\ mod\ frameIndexMax$;

else

  $frameIndex = 0$;

$boneMatrix \leftarrow BonesAffineMatrixs[animationType][frameIndex]$;

$vextexScenePosition =$

  $affineMatrix * boneMatrix * position$;

if $vertexInf.position < waistHeight$ **then**

  $textureType = headTextureType$;

else if $vertexInf.position < neckHeight$ **then**

  $textureType = upperBodyTextureType$;

else

  $textureType = trousersBodyTextureType$;

$fragmentColor \leftarrow textureMapping[textureType][coordinateUV]$;

return $vextexScenePosition$, $fragmentColor$;