**在maya平台利用Arnold渲染器实现原神面影效果**

制作思路：

利用SDF脚本(没有脚本CSP等高线工具也可)和SP生成一张SDF距离场贴图

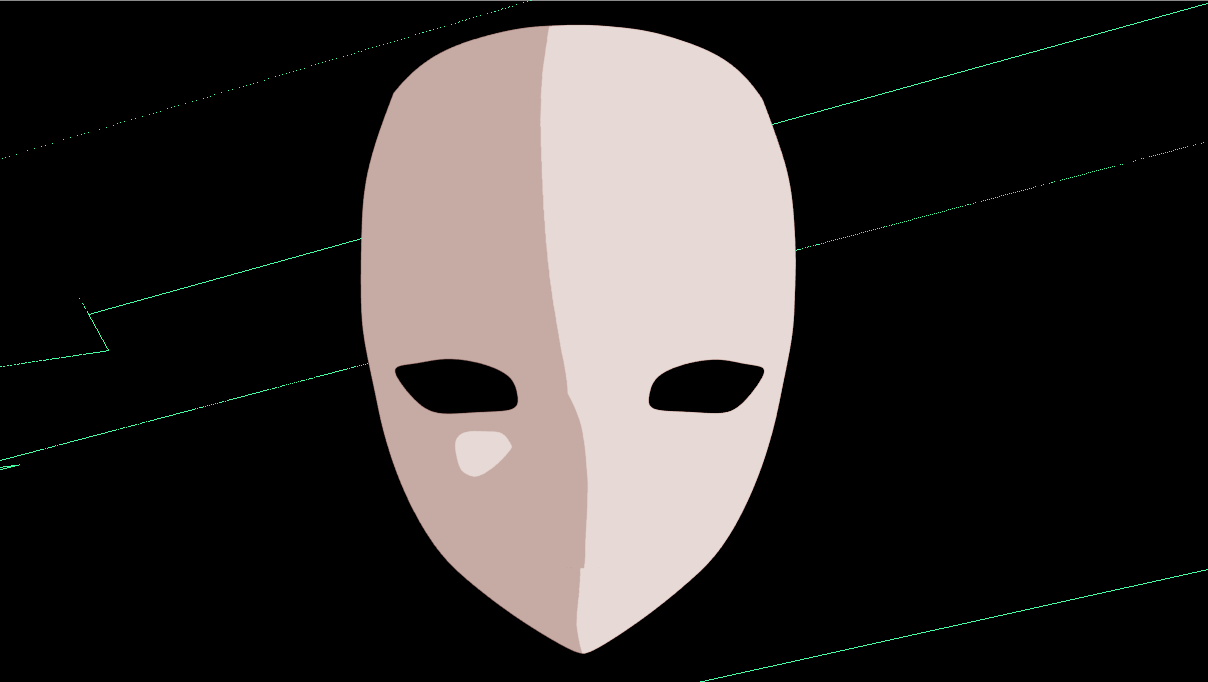
得到的SDF是一张阈值图

由此

在maya里面将N(forwardVec)和L的点乘结果映射到(1,0)区间

将点乘映射取得的值与SDF贴图的明度进行比较

大于值的区域定义为一个颜色，小于值的区域定义为另一个颜色

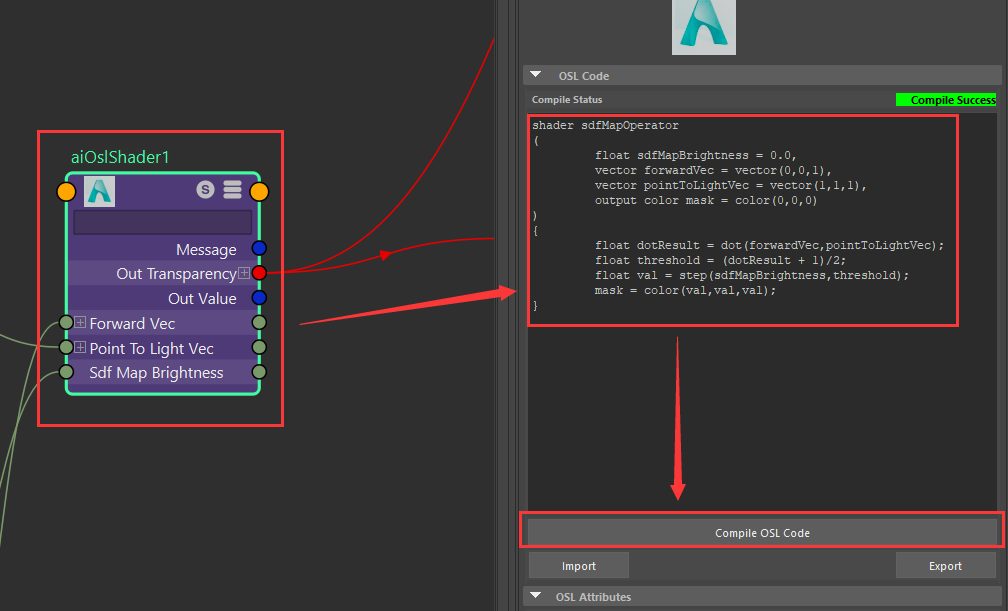


需要自己手动写一个OSL来对SDF和点乘结果进行计算

OSL(aiOSLshader)：

shader sdfMapOperator  
(  
         float sdfMapBrightness = 0.0,  
         vector forwardVec = vector(0,0,1),  
         vector pointToLightVec = vector(1,1,1),  
         output color mask = color(0,0,0)  
)  
{  
         float dotResult = dot(forwardVec,pointToLightVec);  
         float threshold = (dotResult + 1)/2;  
         float val = step(sdfMapBrightness,threshold);  
         mask = color(val);  
}

在maya Hapershade窗口创建一个aiOslShader节点，导入上方OSL脚本，Complie OSL Code生成材质

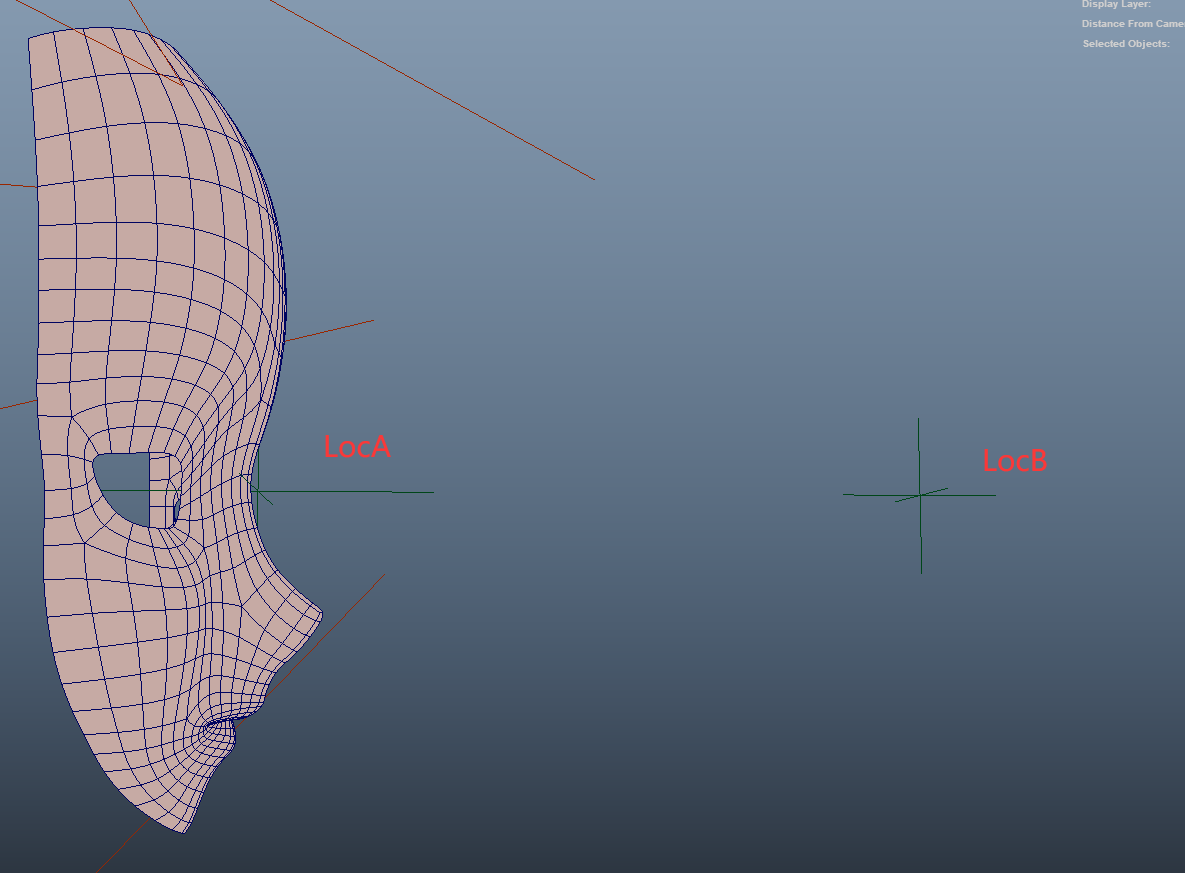


求Forward Vec：

A到B的向量等于B-A

在场景中创建两个Locator，LocA和LocB,

位置摆放如图：



使用LocB减LocA取得forwardVec：

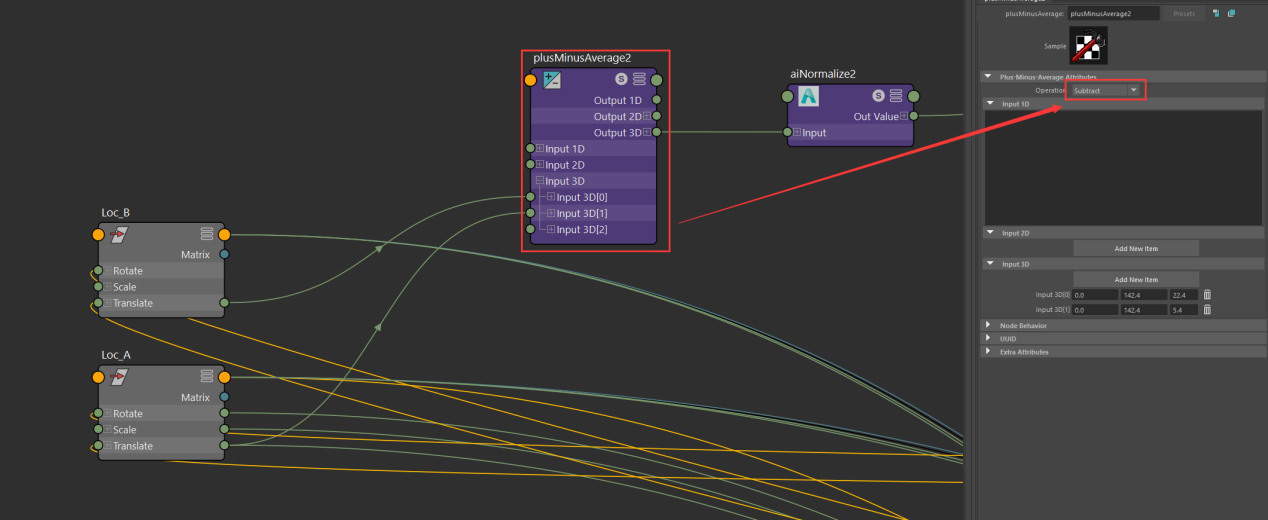
在Hypershade窗口创建一个PlusMinusAverage节点

LocB/TransLate链接plusMinusAverage/Input 3D[0]

LocA/TransLate链接plusMinusAverage/Input 3D[1]

plusMinusAverage/Operation改为Subtract

求出的值需要添加一个aiNormalize节点



LocA和LocB需要绑定到角色面部跟随运动

使用Rigging/Constrain/Parent工具对LocB、LocA、面部模型依次做父子关系约束

取灯光向量(pointToLightVec)

在Hypershade窗口创建一个directionalLight节点

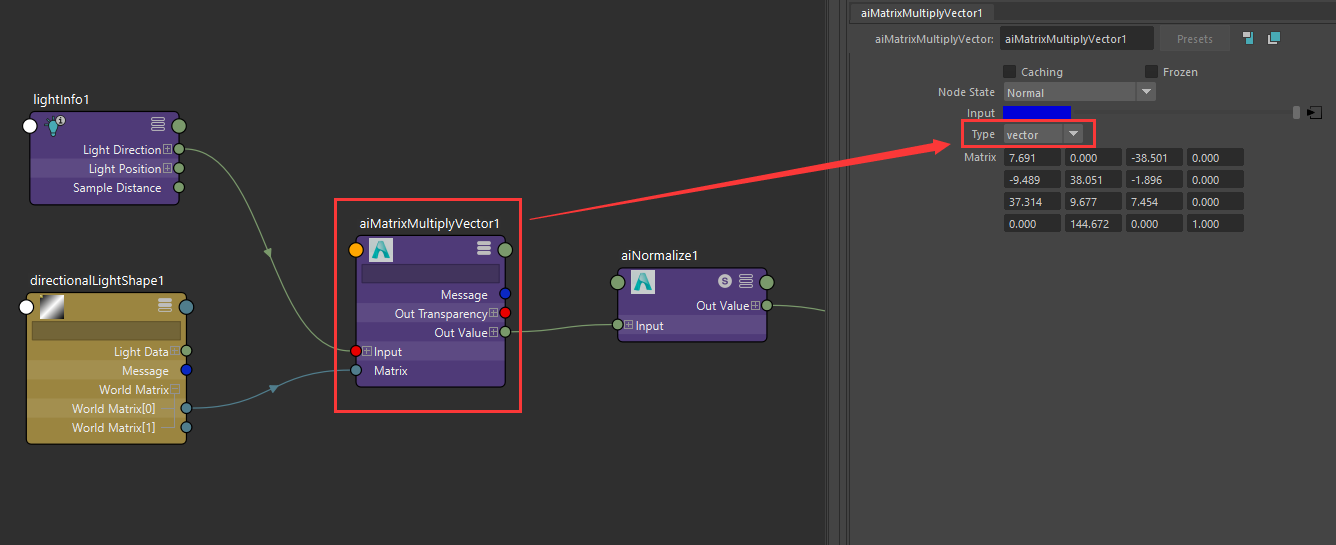
继续创建一个light Info节点

再创建一个aiMatrixMultiplyVector节点

aiMatrixMultiplyVector/Type改为vector

LightInfo/LightDirection链接aiMatrixMultiplyVector/Input

directionalLight/WorldMatrix链接aiMatrixMultiplyVector/Matrix

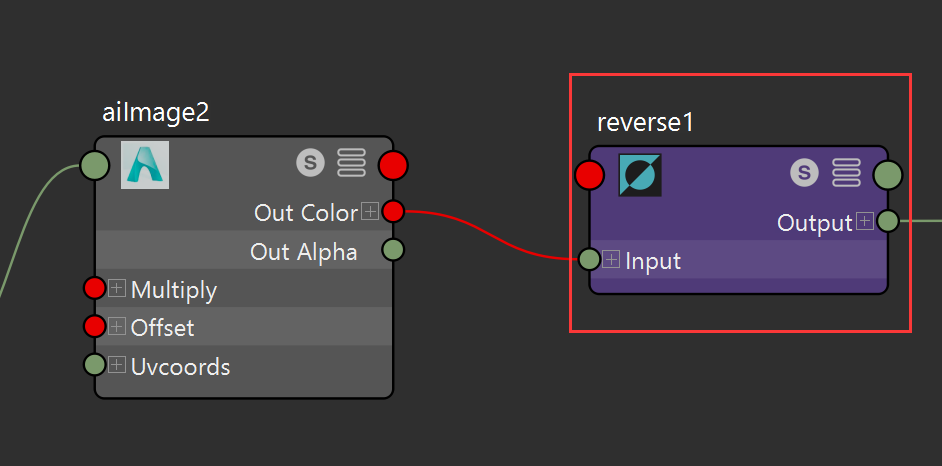


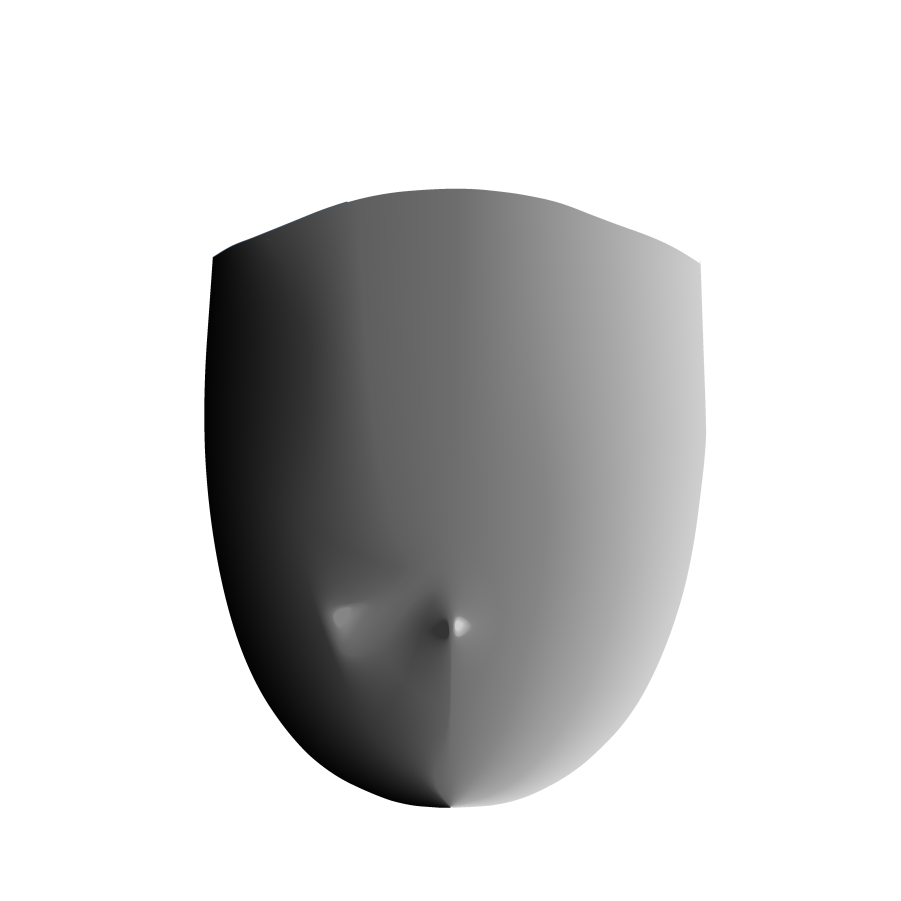
取得的值需要aiNormalize归一化一下

导入SDF贴图

最好使用aiImage导入

根据具体情况判断是否反转贴图





连接接向量和贴图到aiOslShader

灯光向量(pointToLightVec)链接aiOslShader/point To Light Vec

Locator向量（Forward Vec）链接aiOslShader/Forward Vec

SDF贴图链接aiOslShader/sdf Map Brightness

