

物理正确的反射涉及跟踪光线的路径，因为他们反弹的表面。此过程考虑场景中的几何体、材质属性和光源以及视角。然而，这是一个非常计算密集型的过程，通常对实时渲染要求太高，特别是在复杂场景或功能不太强大的硬件上。为了实现视觉质量和性能之间的平衡，可以使用称为屏幕空间反射（SSR）的近似技术。SSR是一种通过重用已渲染到屏幕的数据来近似反射的方法。通过利用屏幕空间变体，可以显著降低与物理正确反射相关的繁重计算成本，使其成为实时渲染的可行技术。在这个配方中，我们将解释如何使用从上一节中导出的缓冲区来计算反射，例如法线和深度缓冲区。

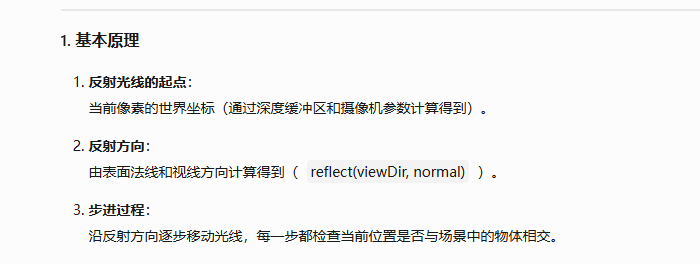


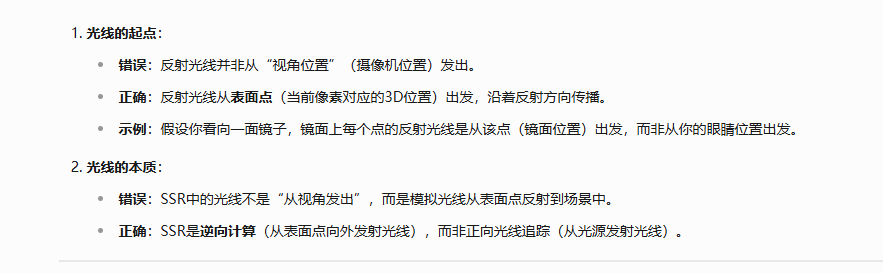






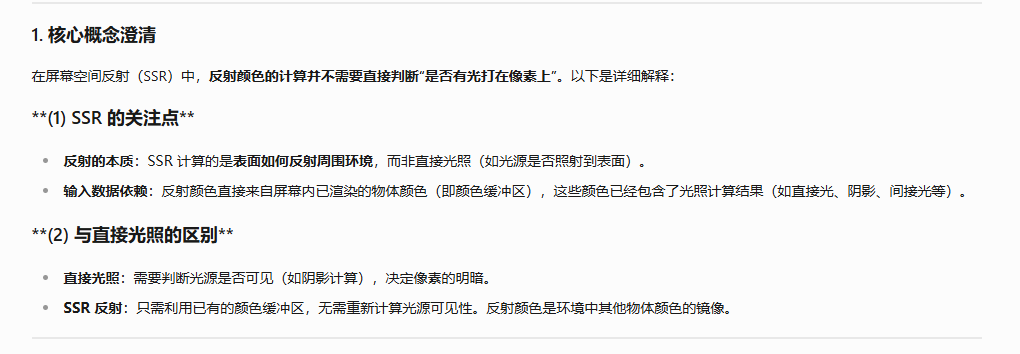


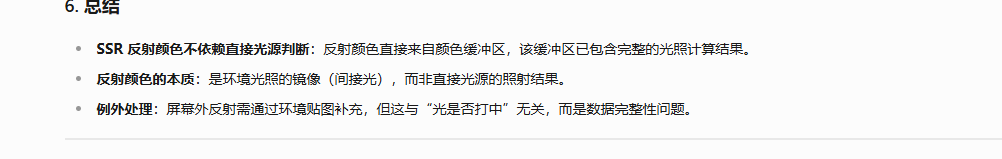


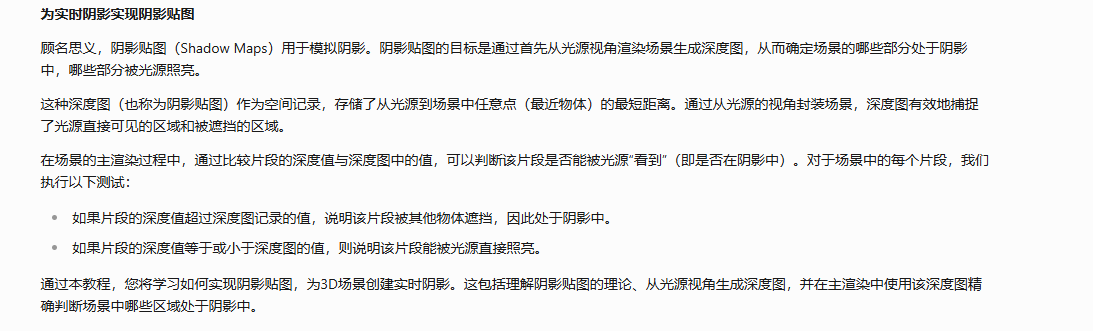




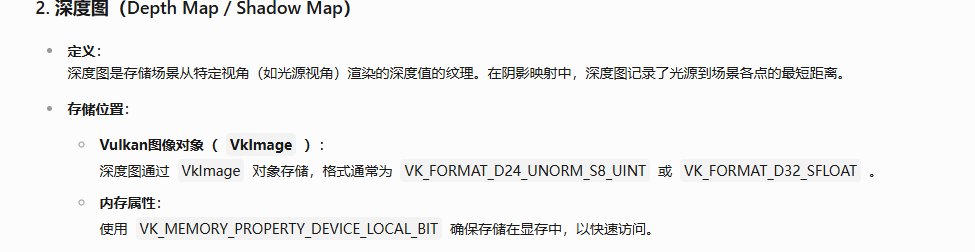












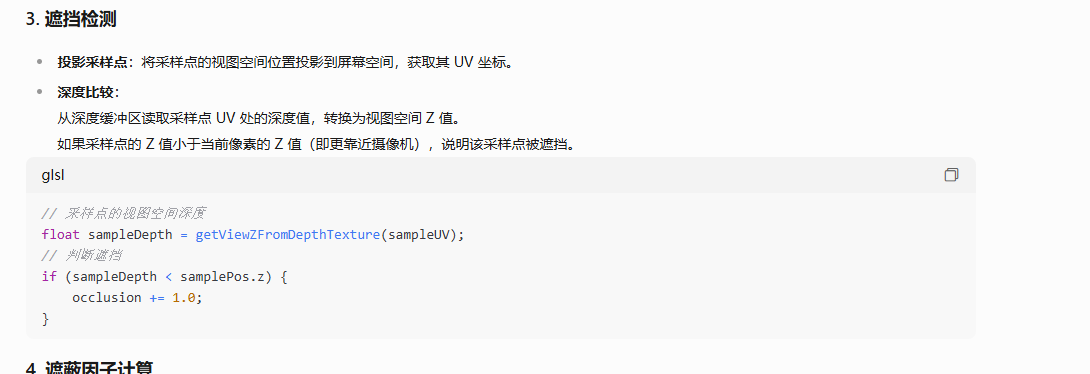


屏幕空间环境光遮挡（SSAO）可以用来近似真实的时间环境光遮挡的效果。环境光遮挡是一种着色和渲染方法，用于计算场景中每个点暴露于环境光的程度。此技术可在两个曲面或对象相交处，或对象阻挡光线到达另一个对象的位置添加更逼真的阴影。在本食谱中，您将学习如何实现SSAO以真实的时间真实地估计环境光遮挡。您将掌握如何使用这种着色和渲染技术来计算场景中每个点对环境光的曝光。

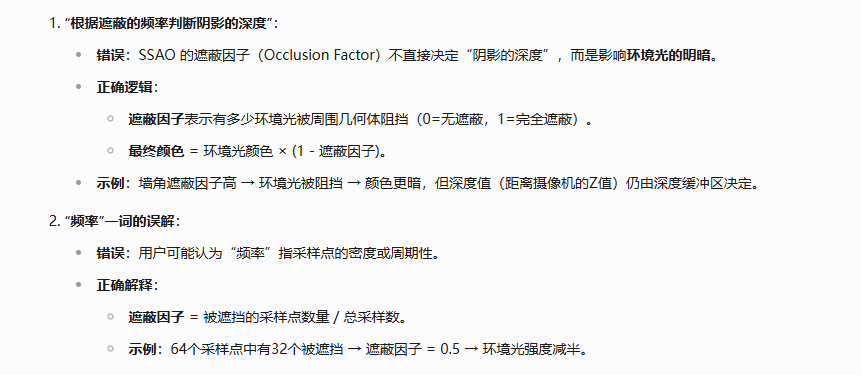


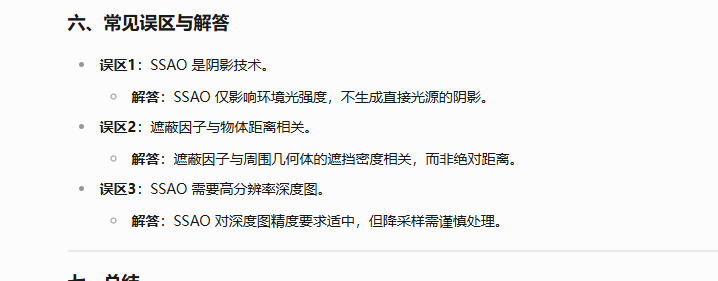












对于场景中的每个灯光，我们绘制一个体积（对于点光源，这将是一个球体;对于定向光源，这将是一个全屏四边形;对于聚光灯，我们将绘制一个圆锥体），对于该体积中的每个像素，我们从G缓冲区获取数据并计算该灯光对像素的照明贡献。然后，通常将结果加在一起（混合）到最终渲染目标以获得最终图像。





