# 9.9 小结

**1．**混合技术可以将当前的光栅化像素（也称为源像素）与后台缓冲区中的像素（也称为目标像素）混合（融合）在一起。该技术通常用来渲染半透明物体，比如水和玻璃。

**2．**混合方程为：

**C** = **C**src ⊗ **F**src ⊞ **C**dst ⊗ **F**dst

*A* = *A*src*F*src ⊞*A*dst*F*dst

注意，RGB分量与alpha分量的混合方程是分开的。二进制运算符⊞可以是**D3D11\_BLEND\_OP**枚举类型定义的任何一个运算符。

**3．F**src、**F**dst、*F*src和*F*dst称为混合系数，它们提供了一种自定义混合方程的途径。它们可以是**D3D11\_BLEND**枚举类型定义的任何一个成员。另外，以“**\_COLOR**”结尾的混合系数不能用于alpha混合方程。

**4．**源alpha信息来自于漫反射材质。在我们的框架中，漫反射材质是一个纹理贴图，纹理的alpha通道存储了alpha信息。

**5．**HLSL的内置函数**clip(x)**可以用来完全丢弃源像素，使源像素不再接受后续处理。该函数只能在像素着色器中使用，当x<0时丢弃当前像素，不再对它进行后续处理。另外，该函数非常适合于渲染那些完全不透明或者完全透明的像素（它用于丢弃完全透明的像素——这些像素的alpha值接近于0）。

**6．**雾可以用来模拟各种天气效果和大气透视，掩盖渲染过程中出现的不自然的人工痕迹，避免“蹿出”问题的发生。在我们模拟的线性雾中，我们为雾指定了一个颜色、一个相对于摄像机的起始位置和一个范围。三角形表面点的颜色等于照颜色与雾颜色的加权平均值：

*foggedColor* = *litColor* + *s* (*fogColor* − *litColor*) = (1 − *s* ) ∙ *litColor* + *s* ∙ *fogColor*

参数*s*的取值范围是从0到1，它是一个以表面点和观察点之间的距离为自变量的函数。随着表面点和观察点之间的距离增大，雾在表面点颜色中所占的比例会越来越大。