光栅化Rasterization (Antialiasing and Z-buffering)

不同的瑕疵（artifacts）

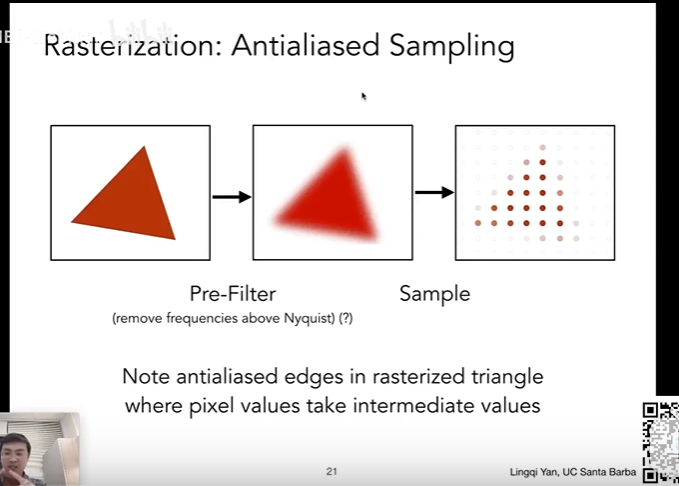
锯齿（jaggies），空间中采样带来的问题

摩尔纹（moire），图像采样不足（undersampling）

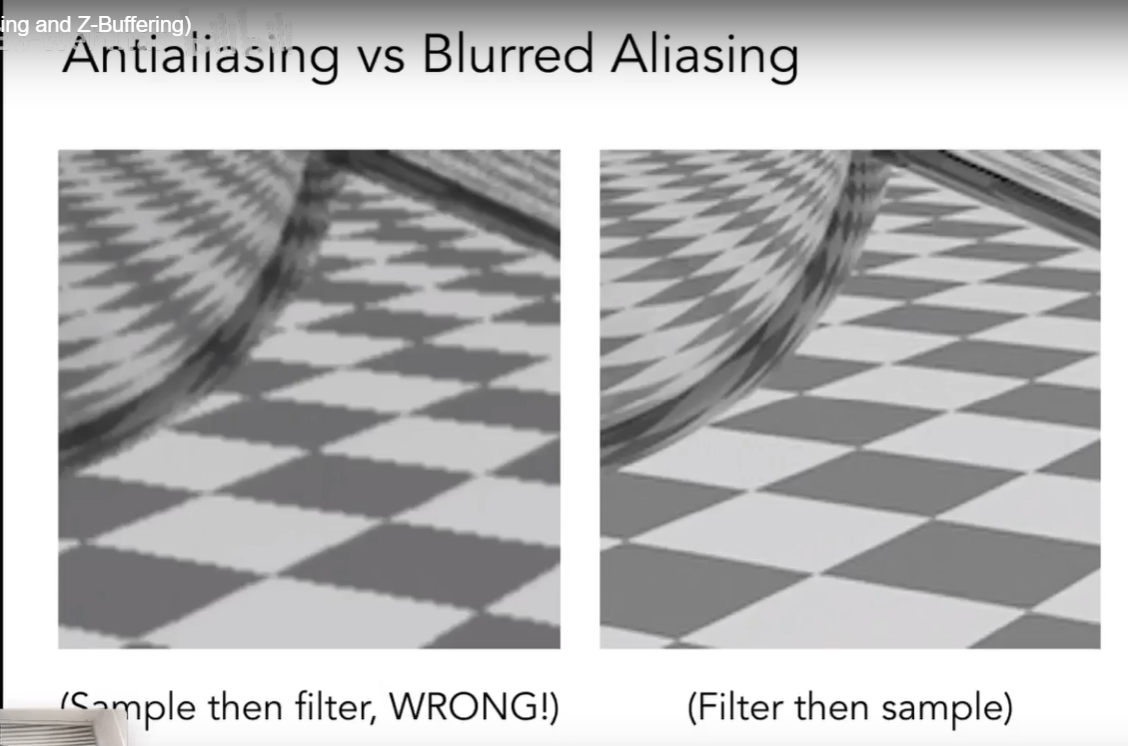
车轮效应（wagon-wheel effect），时间中采样带来的问题

* 1. Antialiasing

idea：在采样之前先做一个模糊（blurring），再采样

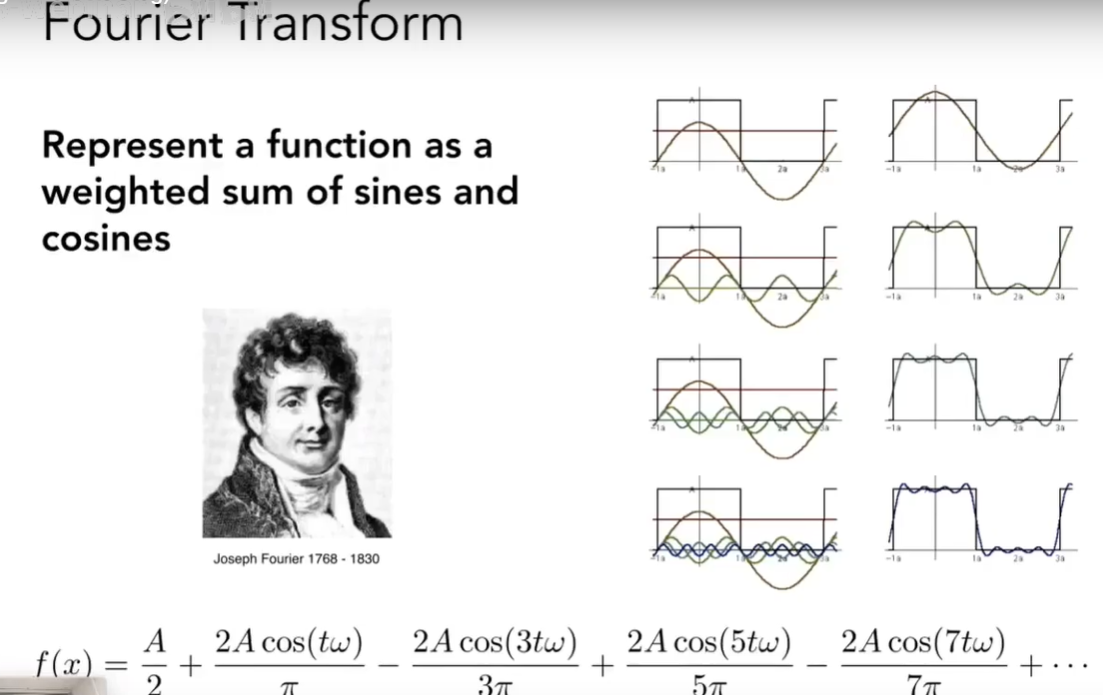


如果先采样，再模糊，不能得到反走样的结果，而是得到blurred alias

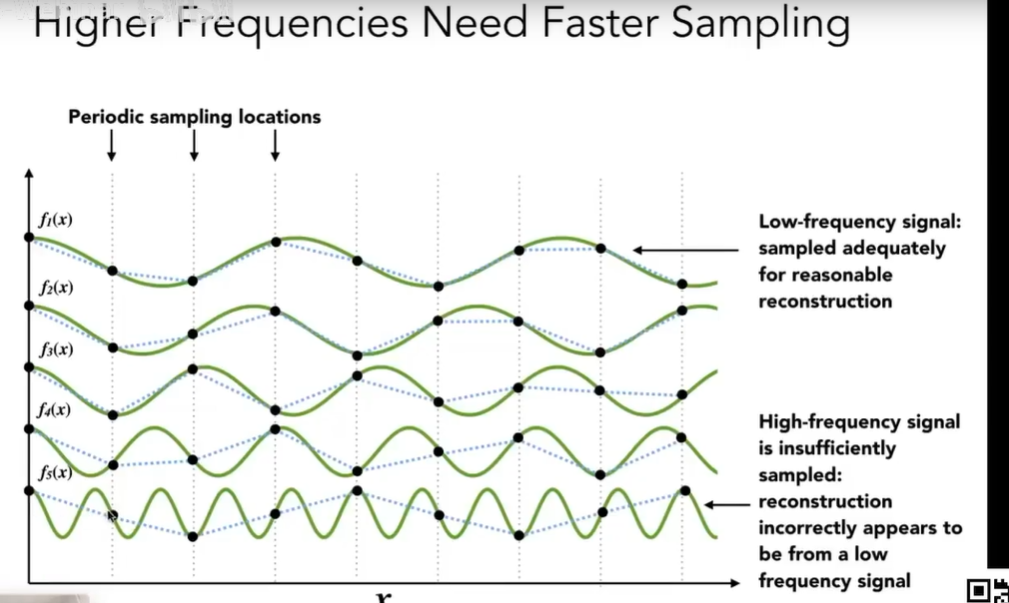


* 1. 频域 frequency domain

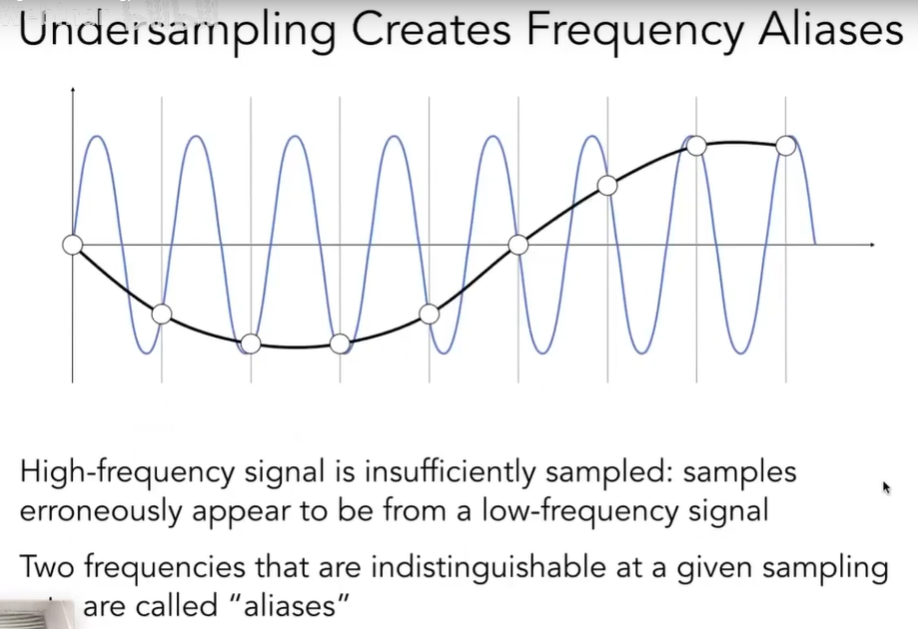
傅立叶变换：任何一个周期函数，都能用sin cos及常数项表示



一个三角函数它有一定的频率，采样也需要有一个相应的频率，否则就容易失真。



走样：假设有两个完全不同的函数（蓝色线和黑色线），都在空心点采样，采样出来的结果结果偶都是黑色线，也就是两个不同的函数采样出来的结果是一样的，这就是“走样”。



傅里叶变换：可以把函数从时域变换到频域。时域的卷积等于频域的乘积

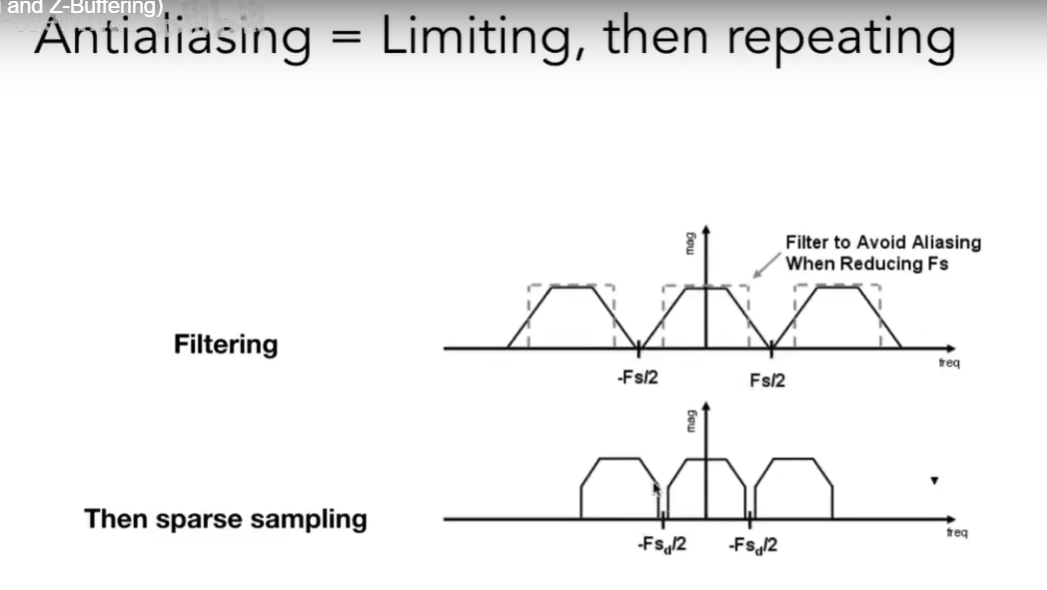
高通滤波：保留图像边界；低通滤波：模糊图像边界。（边界：指图像中颜色发生剧烈变化的地方）

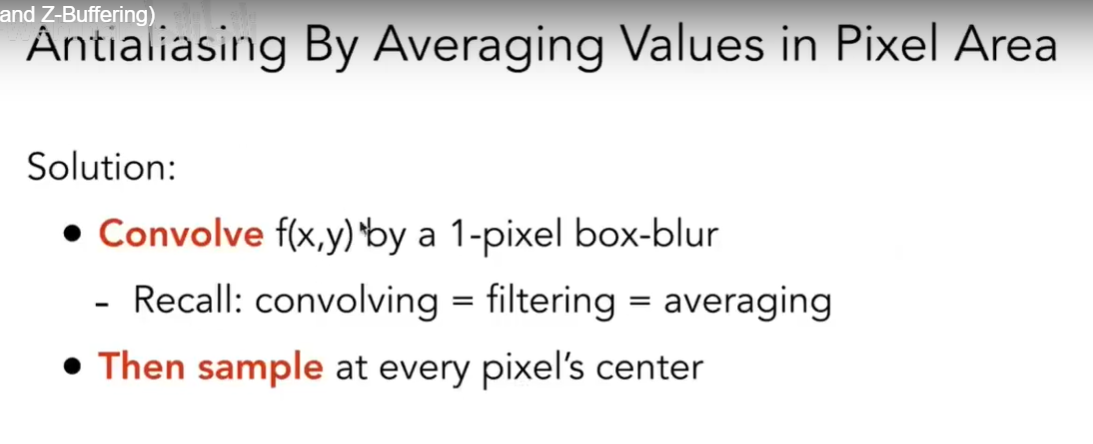
滤波=卷积（=求平均）

采样=重复频域的内容

如何减少走样误差？how to reduce aliasing error?

1. 增加采样率，increase sampling rate，终极方法。但开销很大
2. 反走样，Antialiasing。先做模糊（低通滤波），再做采样。

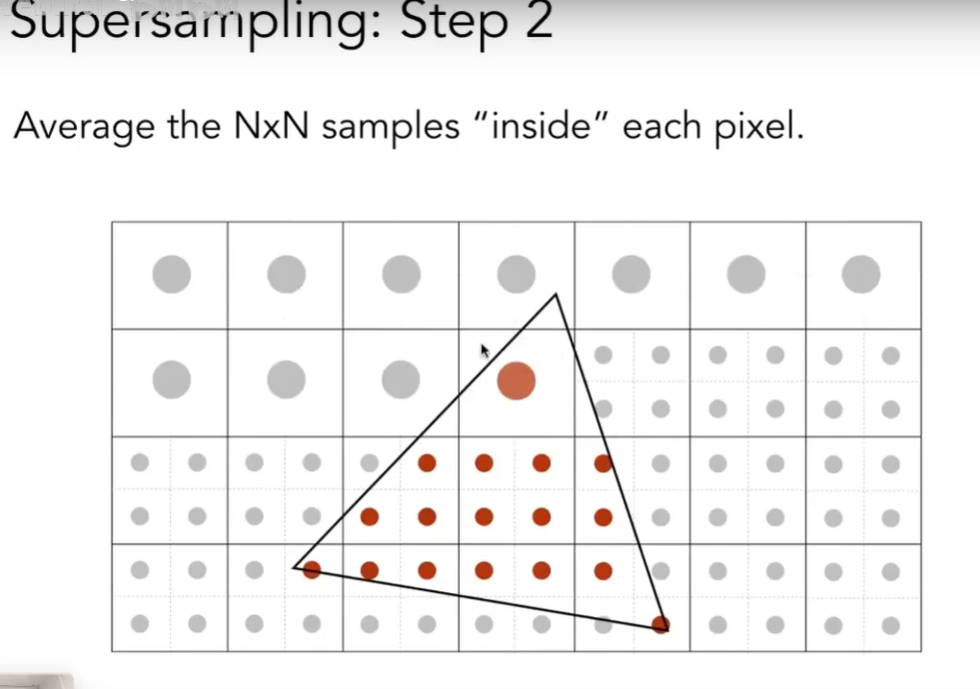


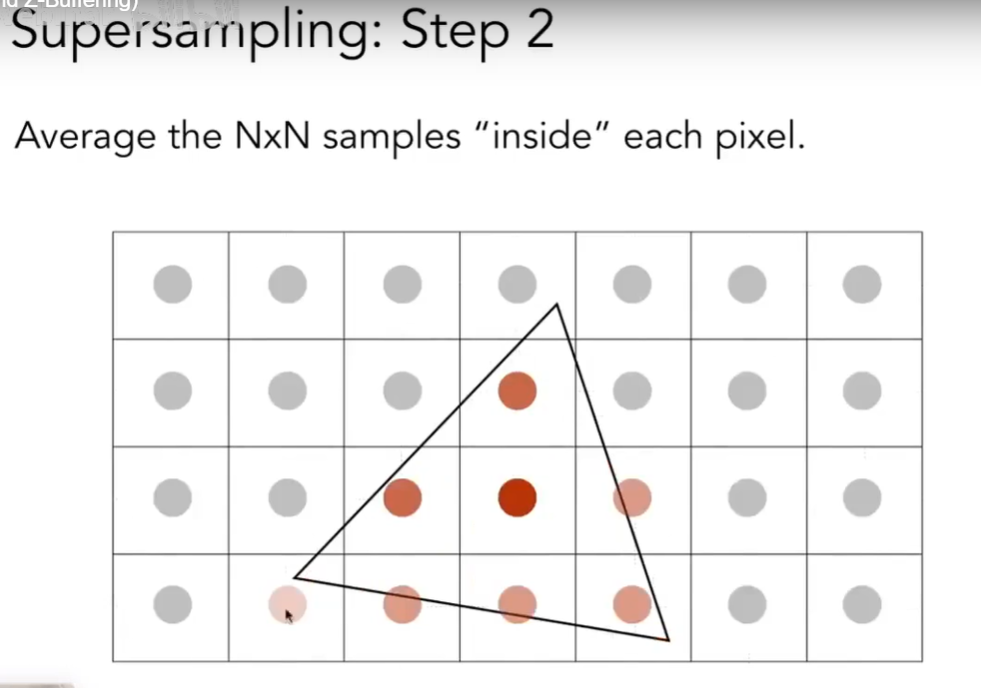


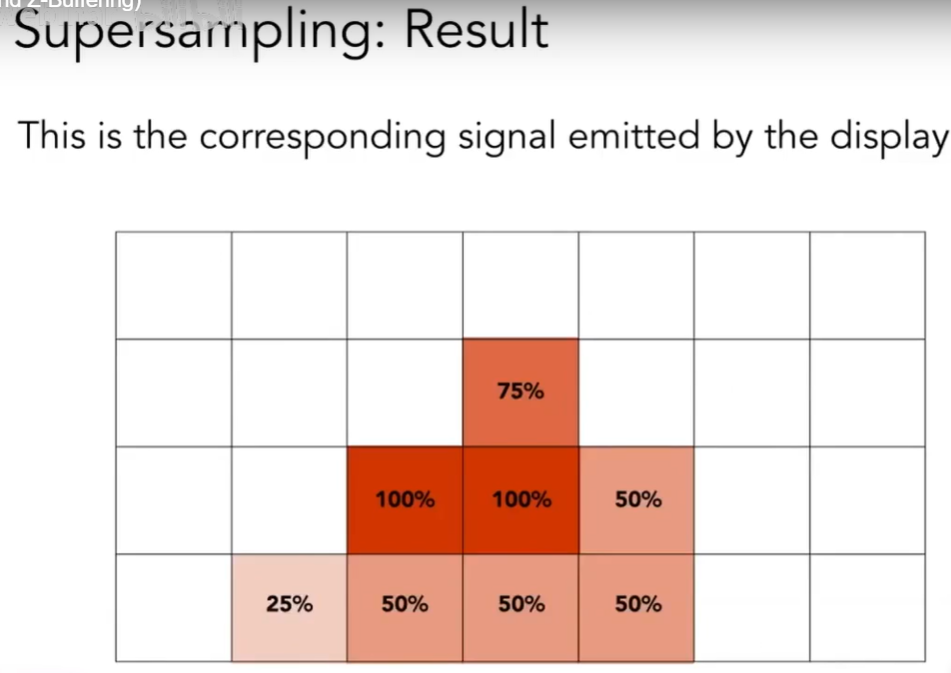
先做模糊操作（卷积），再在每个像素中心采样。

通过多重采样进行反走样（MSAA）：

假设对每个像素，将其分成许多小的像素，对每个小像素判断它们是不是在图形内，求每个小像素判断结果的平均值







MSAA并不是提升了屏幕分辨率解决了锯齿问题，而是增加采样点而得到近似的三角形覆盖

开销：增加了计算量

其他AA方法：

FXAA（Fast Approximate AA）：后期做图像处理，把有锯齿的边界找到，换成没有锯齿的边界

TAA（Temporal AA）：找上一帧的信息。对相邻两帧，找某个像素内不同位置的点，判断像素是否在三角形内

超采样（DLSS）:Deep Learning Super Sampling（小图拉大，用深度学习“猜”出缺失的细节）