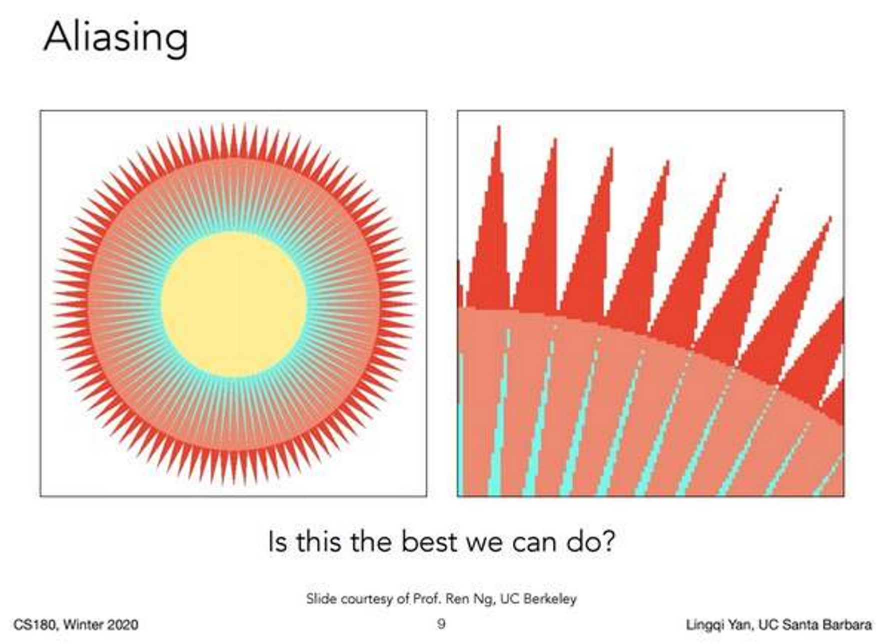
Lecture 06 Rasterization 2 (Antialiasing and Z-buffering)

2020年12月26日 17:27

走样与反走样



Wagon Wheel Illusion (False Motion)

采用上一讲的光栅化方法，会出现走样（Aliasing）现象
采样中出现的误差，错误，或不准确，称作Artifacts
例如：锯齿，摩尔纹，旋转车轮错觉等

Sampling Artifacts in Computer Graphics

Artifacts due to sampling - "Aliasing"

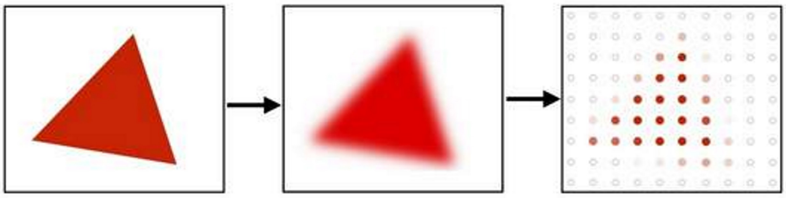
- Jaggies – sampling in space
- Moire – undersampling images
- Wagon wheel effect – sampling in time
- [Many more] …

Behind the Aliasing Artifacts

- Signals are changing too fast (high frequency), but sampled too slowly

这些artifacts背后的原因：
信号变化速度太快（高频率）
但采样速度相对太慢

Rasterization: Antialiased Sampling



Pre-Filter

(remove frequencies above Nyquist) (?)

Sample

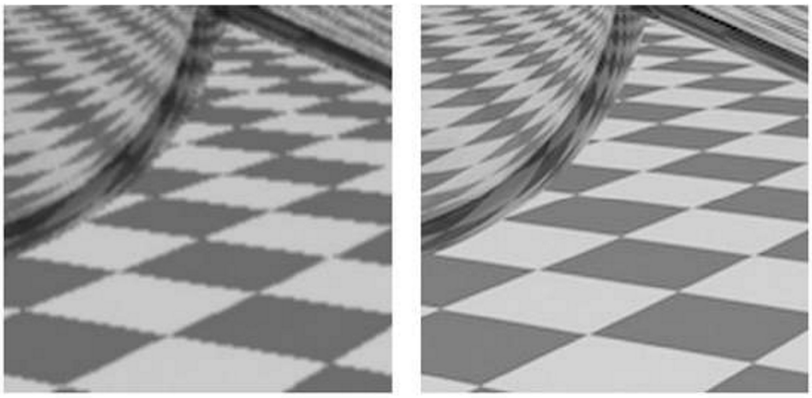
Note antialiased edges in rasterized triangle where pixel values take intermediate values

GAMES101 21 Lingqi Yan, UC Santa Barbara

一种反走样的思想：先模糊，再采样

模糊的部分，可以看到光栅化以后像素颜色取了个中间值（intermediate values）

Antialiasing vs Blurred Aliasing



(Sample then filter, WRONG!) (Filter then sample)

GAMES101 25 Lingqi Yan, UC Santa Barbara

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

But why?

1. Why undersampling introduces aliasing?
2. Why pre-filtering then sampling can do antialiasing?

Let's dig into fundamental reasons

And look at how to implement antialiased rasterization

GAMES101 26 Lingqi Yan, UC Santa Barbara

思考背后原理：

（1）为何欠采样会导致走样？

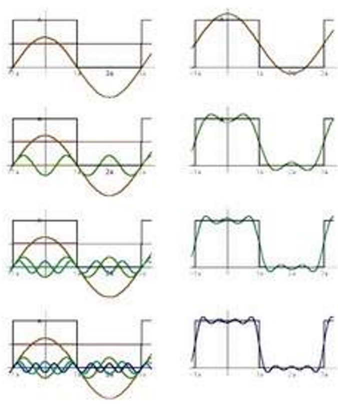
（2）为何pre-filtering（预滤波，前置滤波）可以反走样？

频域 frequency domain

Fourier Transform

Represent a function as a weighted sum of sines and cosines

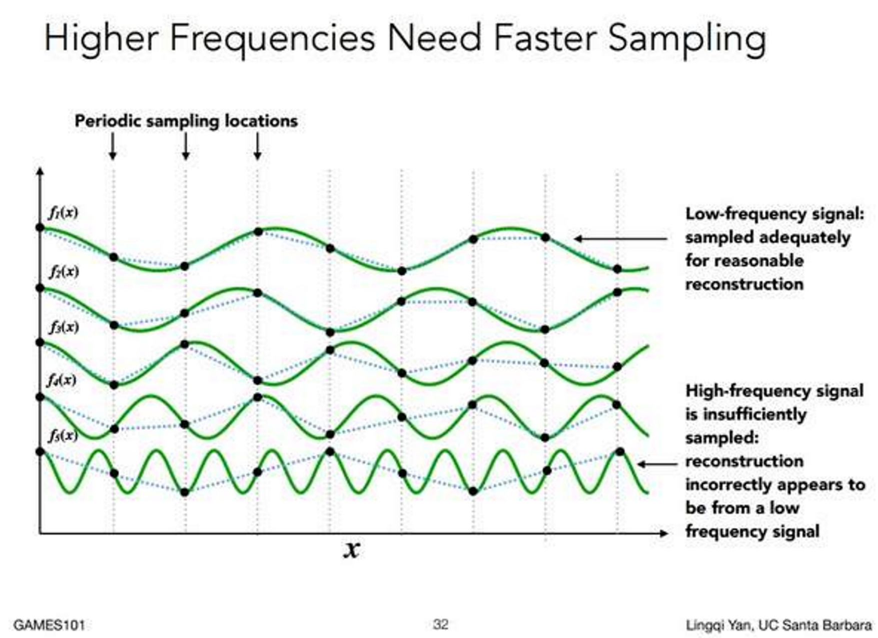
Joseph Fourier 1768 - 1830



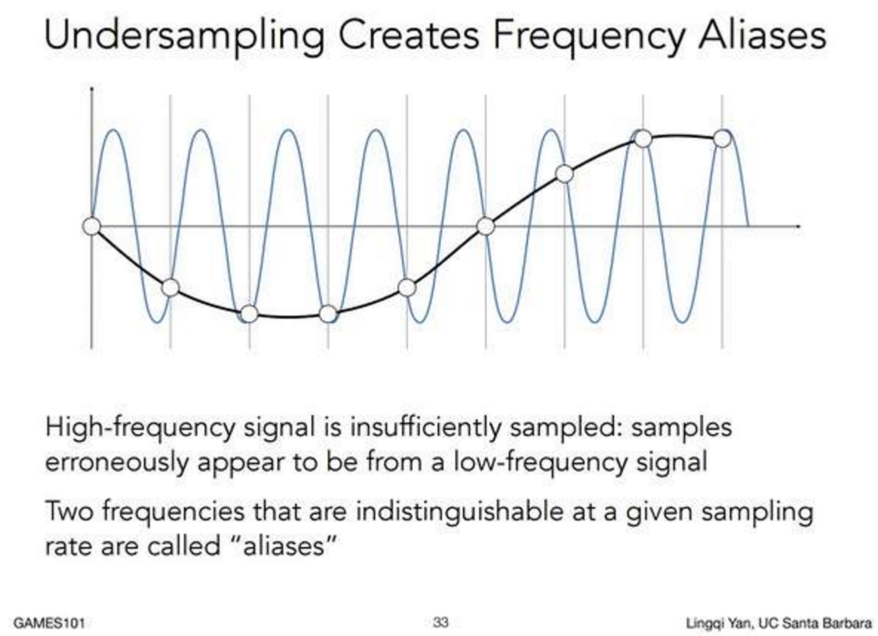
$$f(x) = \frac{A}{2} + \frac{2A\cos(t\omega)}{\pi} - \frac{2A\cos(3t\omega)}{3\pi} + \frac{2A\cos(5t\omega)}{5\pi} - \frac{2A\cos(7t\omega)}{7\pi} + \cdots$$

GAMES101 30 Lingqi Yan, UC Santa Barbara

傅立叶变换：任何一个周期函数，都能用sin， cos及常数项表示



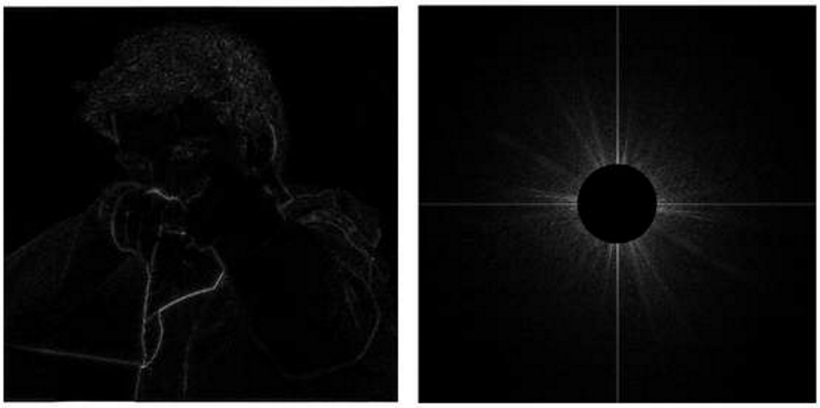
一个三角函数它有一定的频率，采样也需要有一个相应的频率，否则就容易失真。



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

Filter Out Low Frequencies Only (Edges)



High-pass filter

GAMES101 36 Lingqi Yan, UC Santa Barbara

高通滤波：保留图像边界；

Filter Out High Frequencies (Blur)

Low-pass filter

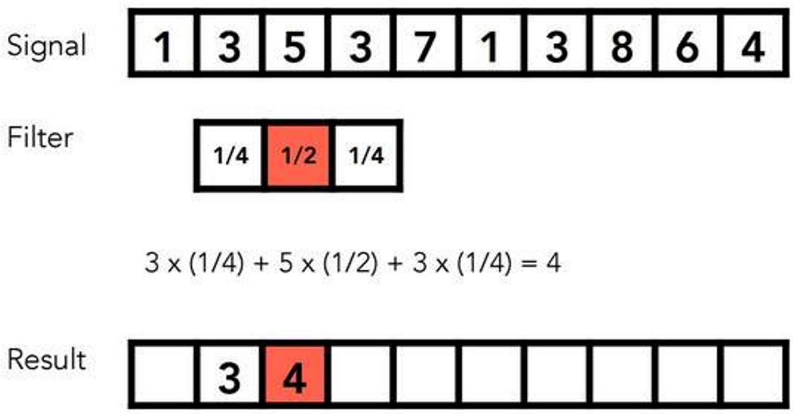
GAMES101 37 Lingqi Yan, UC Santa Barbara

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）

Filtering = Convolution
(= Averaging)

滤波=卷积（=求平均）

Convolution



GAMES101 43 Lingqi Yan, UC Santa Barbara

卷积
（不严谨的说法）按照一定的卷积核（权重）求某个点位周围带权的均值

Convolution Theorem

Convolution in the spatial domain is equal to multiplication in the frequency domain, and vice versa

Option 1:

- Filter by convolution in the spatial domain

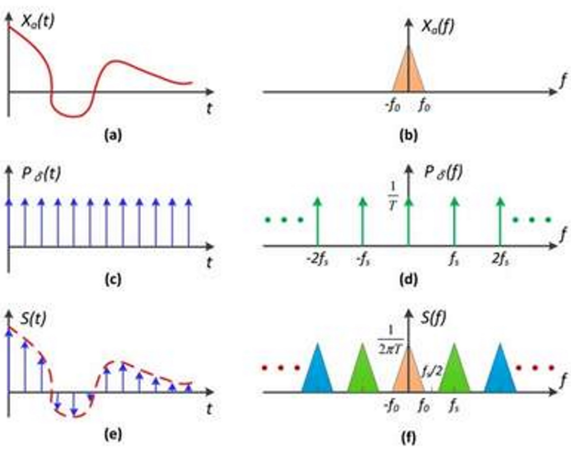
Option 2:

- Transform to frequency domain (Fourier transform)
- Multiply by Fourier transform of convolution kernel
- Transform back to spatial domain (inverse Fourier)

GAMES101 44 Lingqi Yan, UC Santa Barbara

空间域的卷积相当于频域的乘积，反之亦然

Sampling = Repeating Frequency Contents

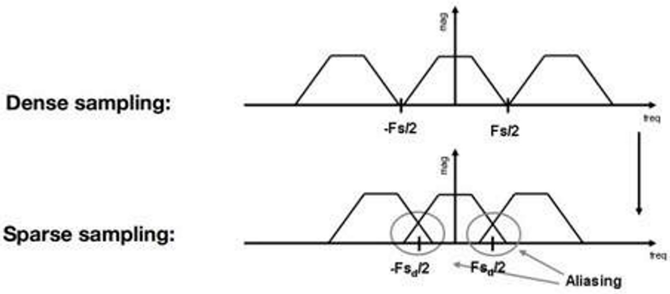


https://www.researchgate.net/figure/The-evolution-of-sampling-theorem-a-The-time-domain-of-the-band-limited-signal-and-b_fig5_301556095

GAMES101 50 Lingqi Yan, UC Santa Barbara

采样=重复频域的内容

Aliasing = Mixed Frequency Contents



走样=弄混了频率内容

How Can We Reduce Aliasing Error?

Option 1: Increase sampling rate

- Essentially increasing the distance between replicas in the Fourier domain
- Higher resolution displays, sensors, framebuffers…
- But: costly & may need very high resolution

Option 2: Antialiasing

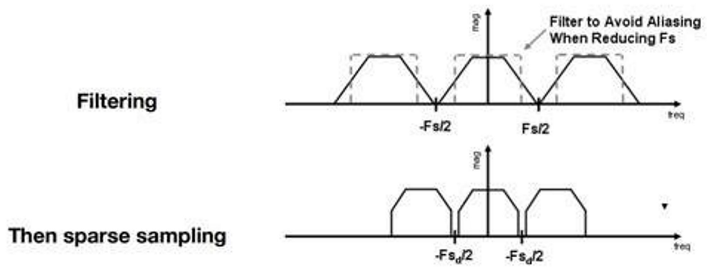
- Making Fourier contents "narrower" before repeating
- i.e. Filtering out high frequencies before sampling

如何减少走样误差?

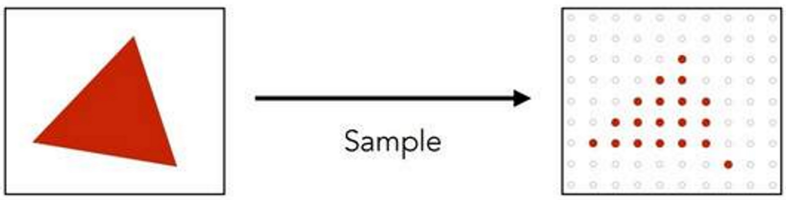
增加采样率：终极方法，但开销很大

反走样：先做模糊（低通滤波），再做采样

Antialiasing = Limiting, then repeating



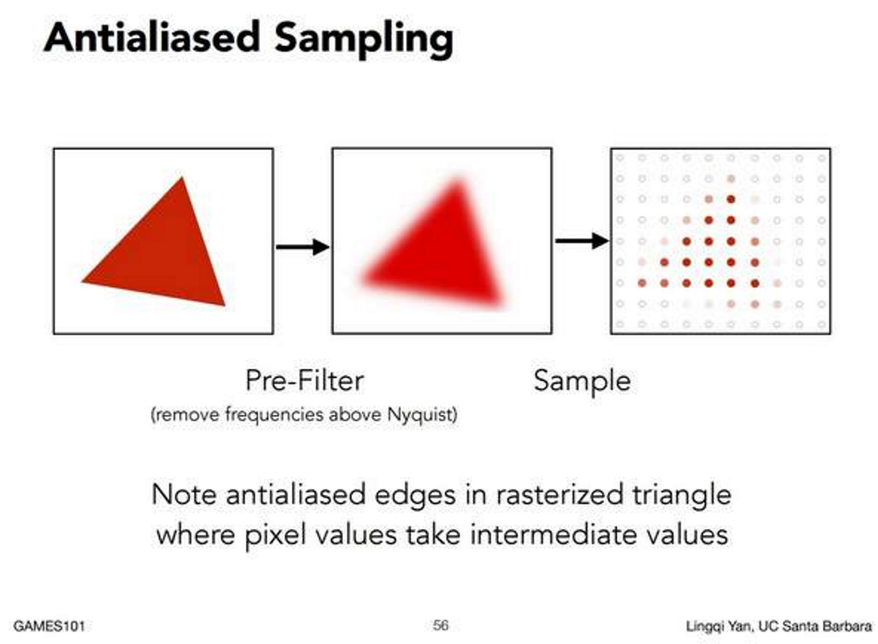
Regular Sampling



Note jaggies in rasterized triangle
where pixel values are pure red or white

普通采样：

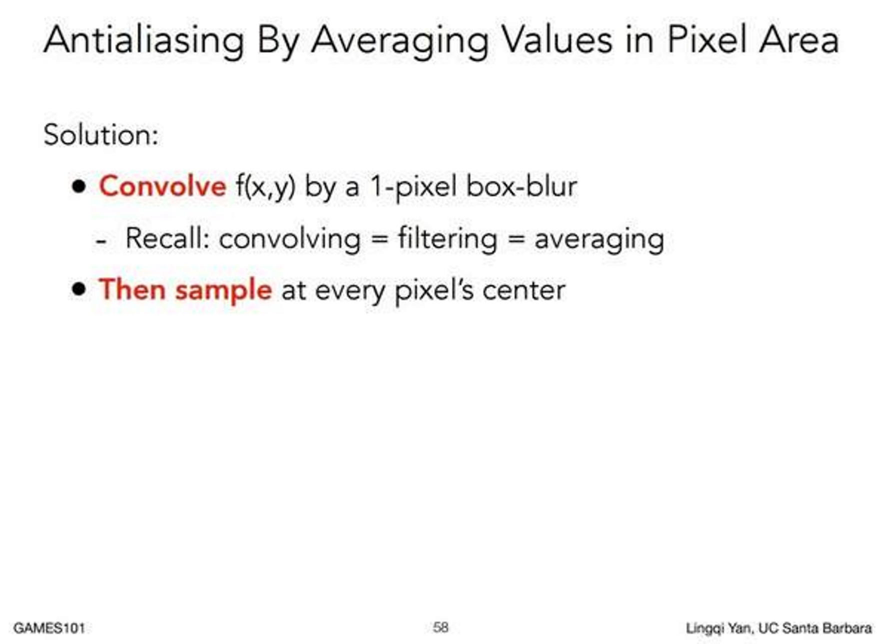
像素点非红即白，因此造成锯齿



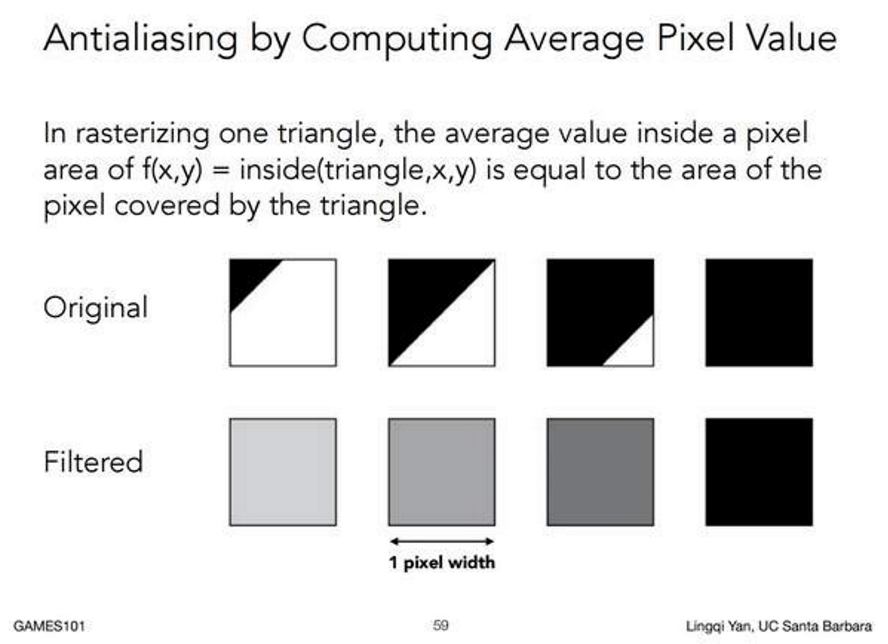
反走样后的采样：

先经过预过滤，再采样

边缘取了周围的像素点颜色的中间值（intermediate values）



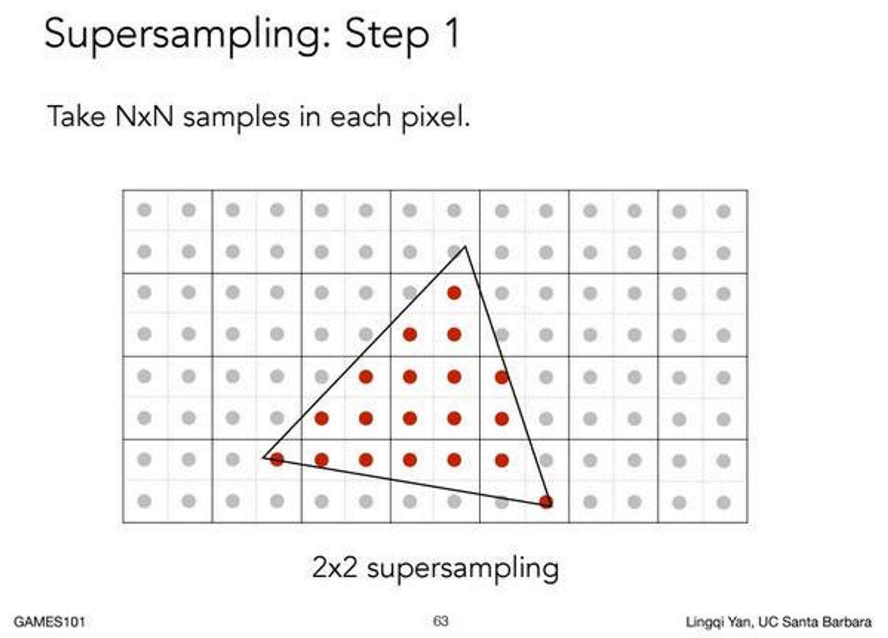
先做模糊操作（卷积），再在每个像素中心采样



通过计算平均像素点值的方法实现反走样

在对三角形光栅化时，像素点区域内部的颜色平均值等于该像素点被三角形覆盖的面积

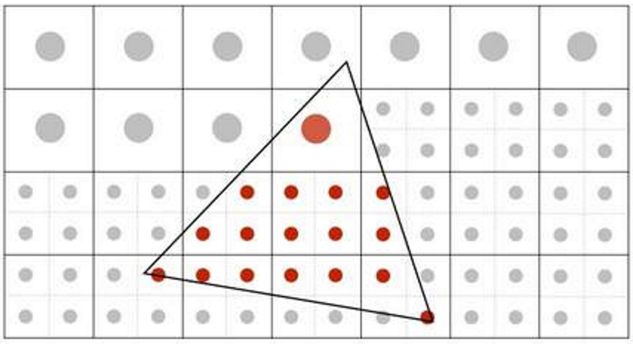
通过多重采样进行反走样（MSAA）



首先，在每个像素点内，取N*N个小的“像素”（采样点），对每个小采样点判断它们是不是在图形内

Supersampling: Step 2

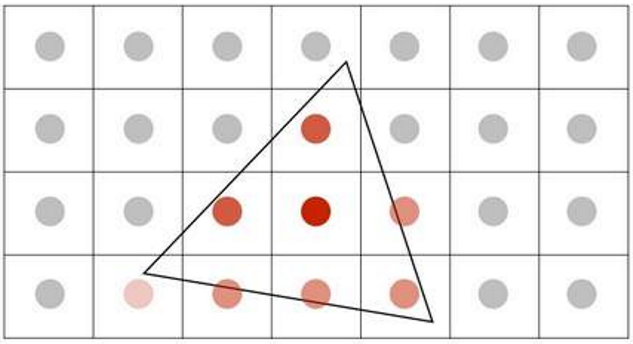
Average the NxN samples "inside" each pixel.

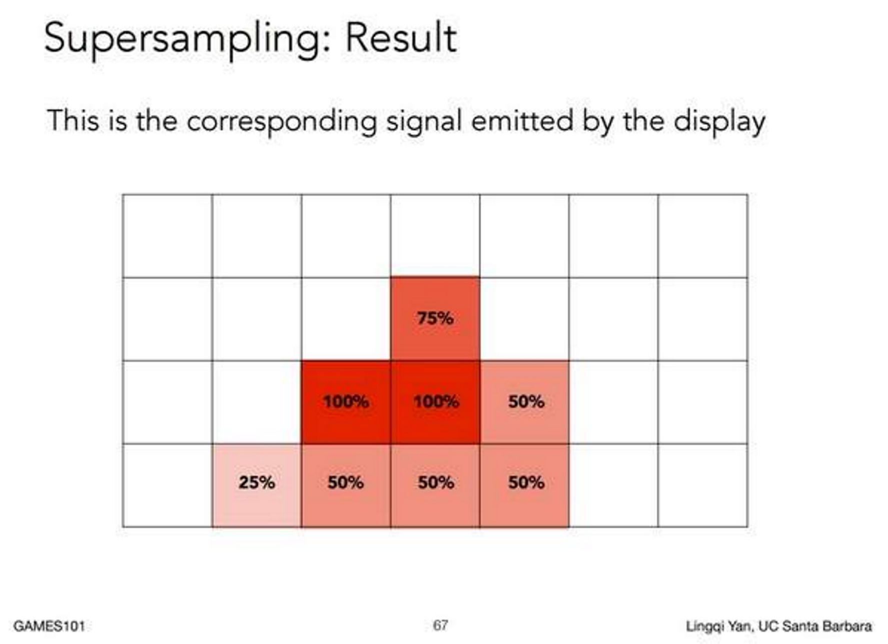


Averaging down

Supersampling: Step 2

Average the NxN samples "inside" each pixel.





求每个小像素判断结果的平均值，在图形内的小像素越多，则颜色越深，否则越浅。

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）（配置要求低，效果一般）
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）（配置要求高，效果好）
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