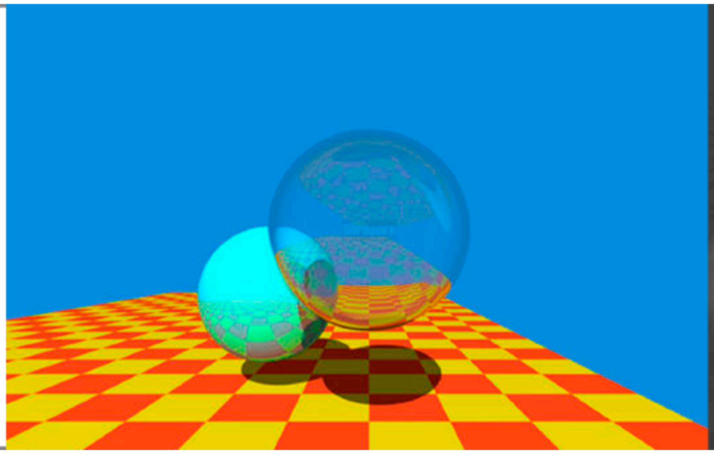
Introduction to Computer Graphics

GAMES101, Lingqi Yan, UC Santa Barbara

Lecture 6:
Rasterization 2
(Antialiasing and Z-Buffering)






http://www.cs.ucsb.edu/~lingqi/teaching/games101.html

Announcements

- Homework 1
  - Already 49 submissions so far!
  - In general, start early
- Today's topics are not easy
  - Having knowledge on Signal Processing is appreciated
  - But no worries if you don't

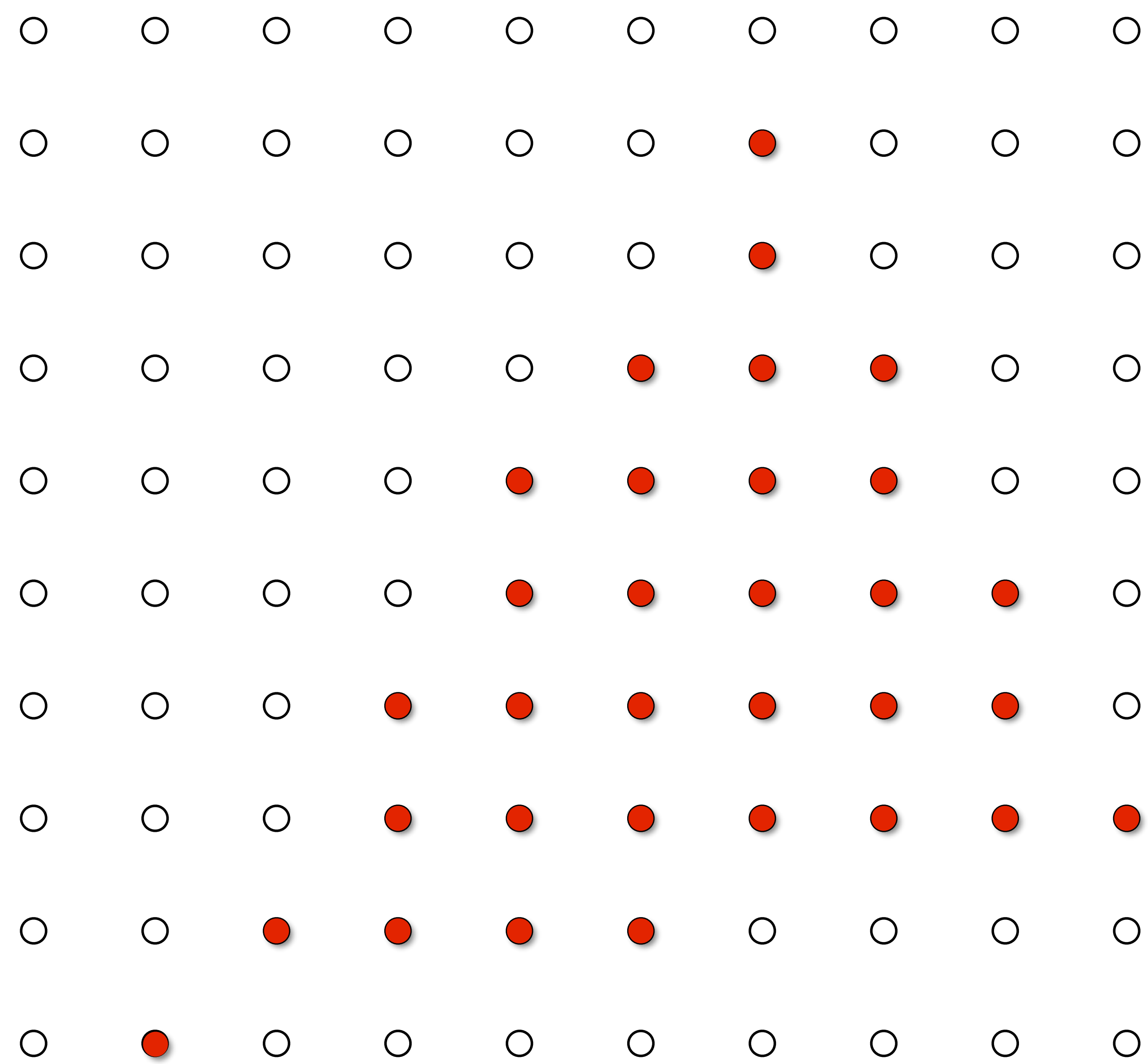
Last Lectures

- Viewing
  - View + Projection + Viewport
- Rasterizing triangles
  - Point-in-triangle test
  - Aliasing

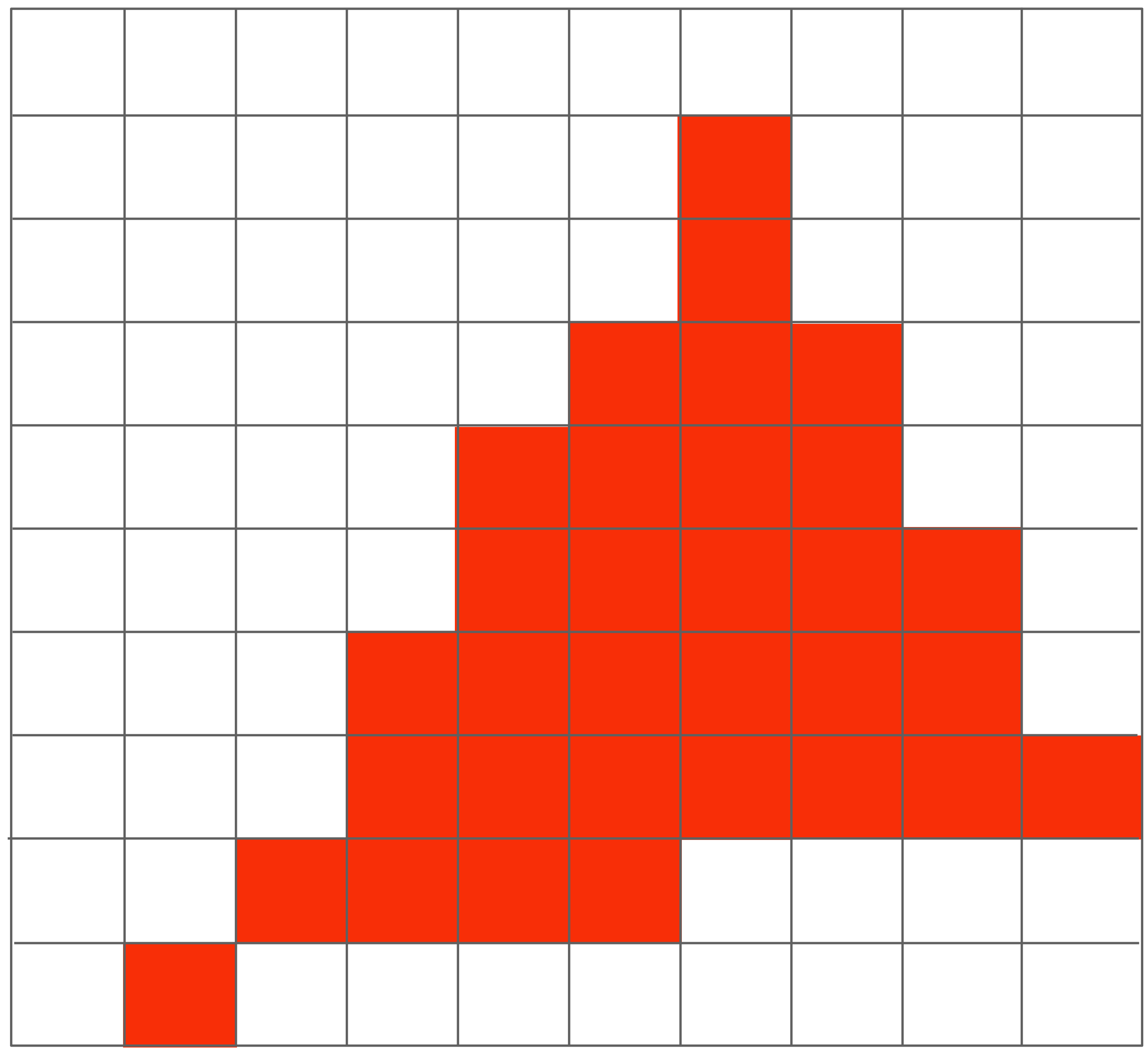
Today

- Antialiasing 抗锯齿 / 抗混叠
  - Sampling theory
  - Antialiasing in practice
- Visibility / occlusion
  - Z-buffering

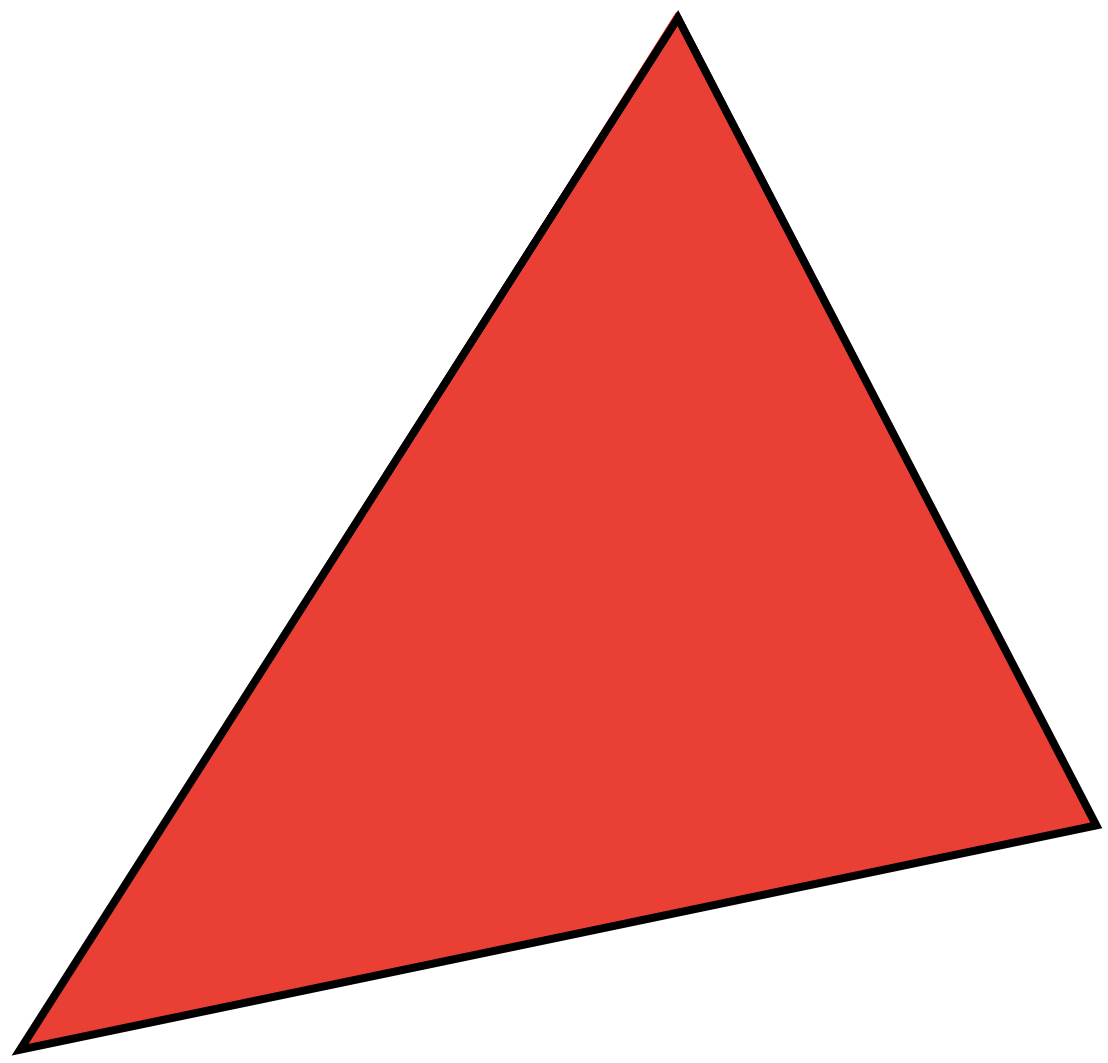
Recap: Testing in/out △ at pixels' centers



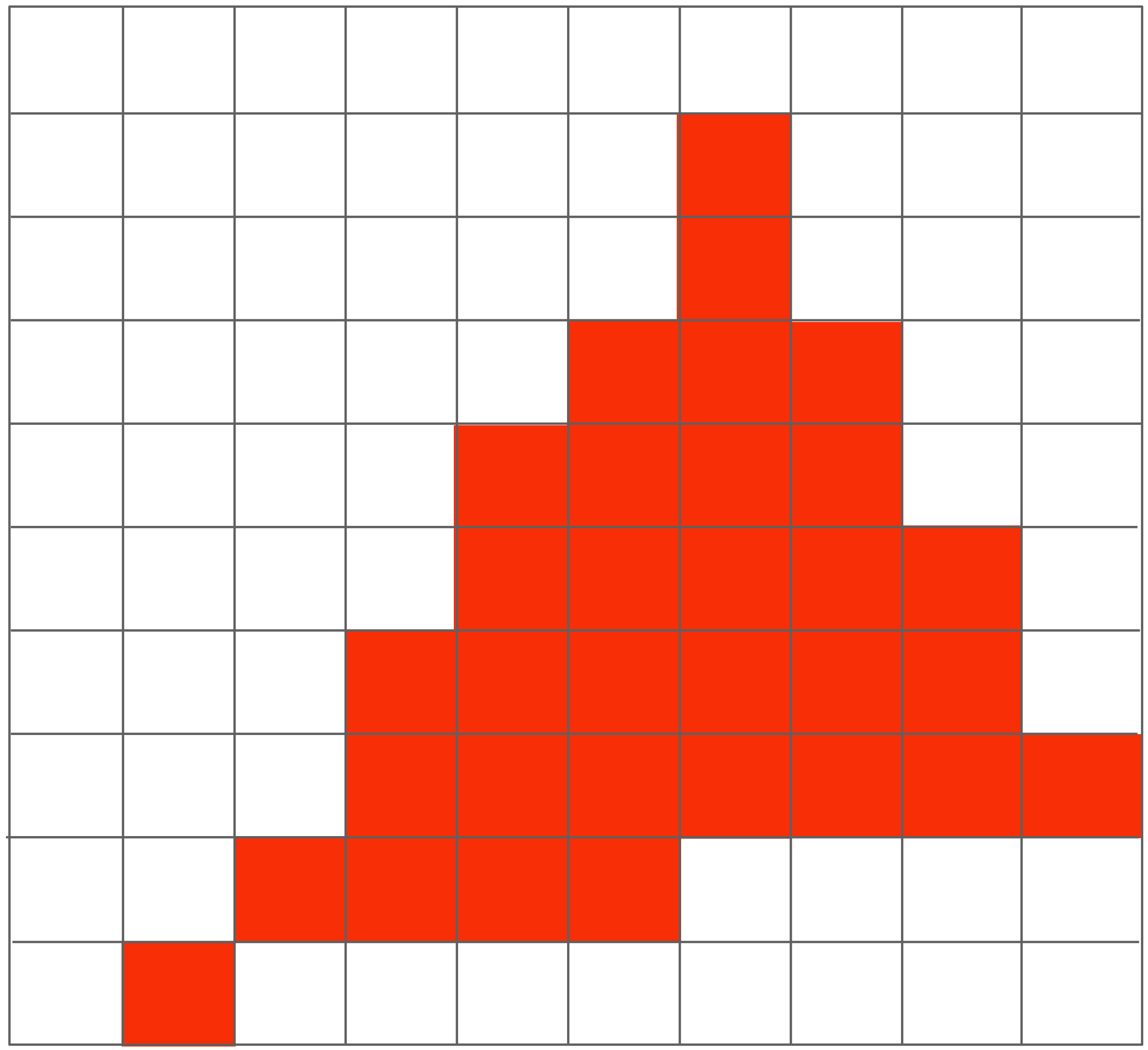
Pixels are uniformly-colored squares



Compare: The Continuous Triangle Function

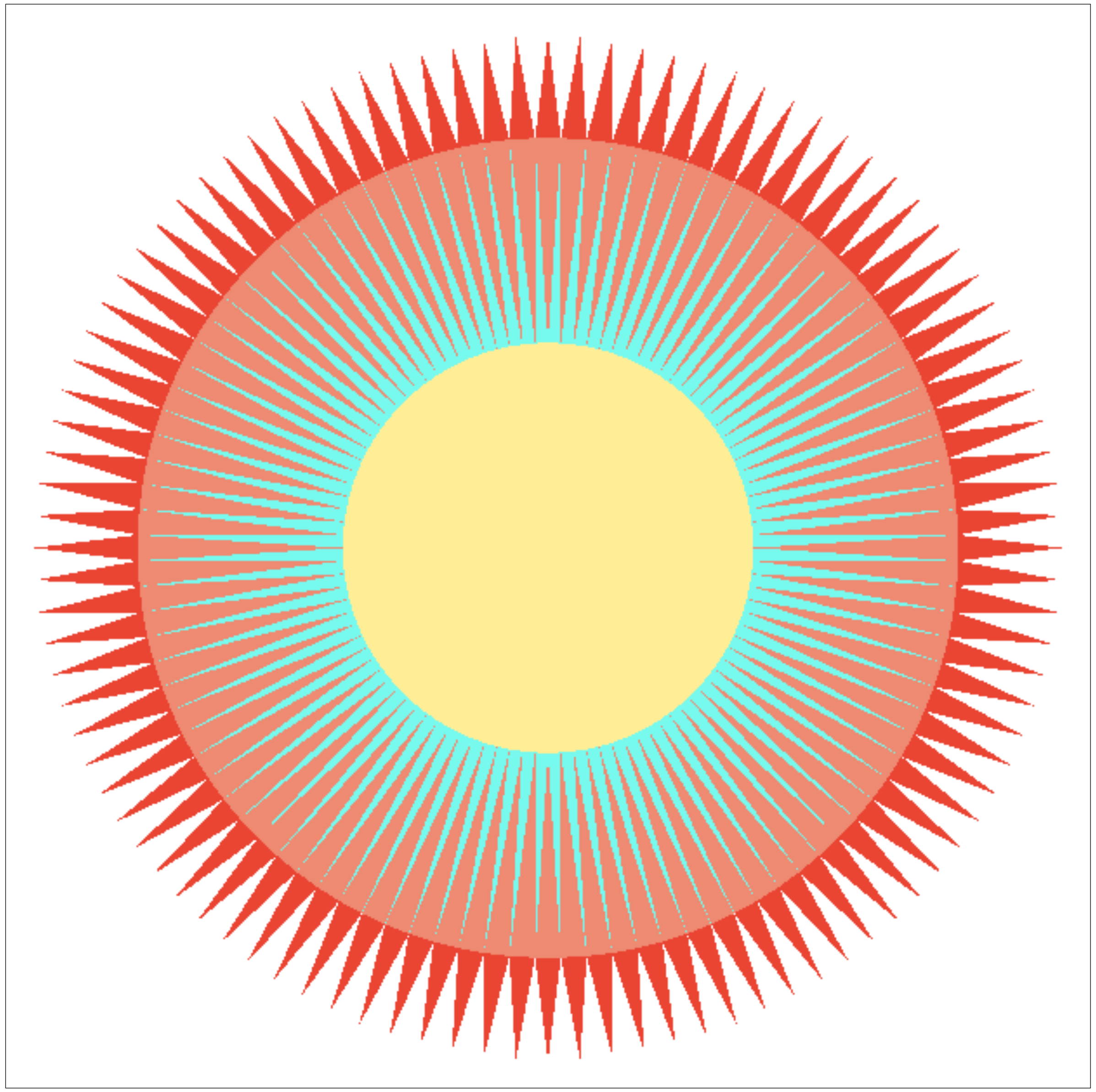


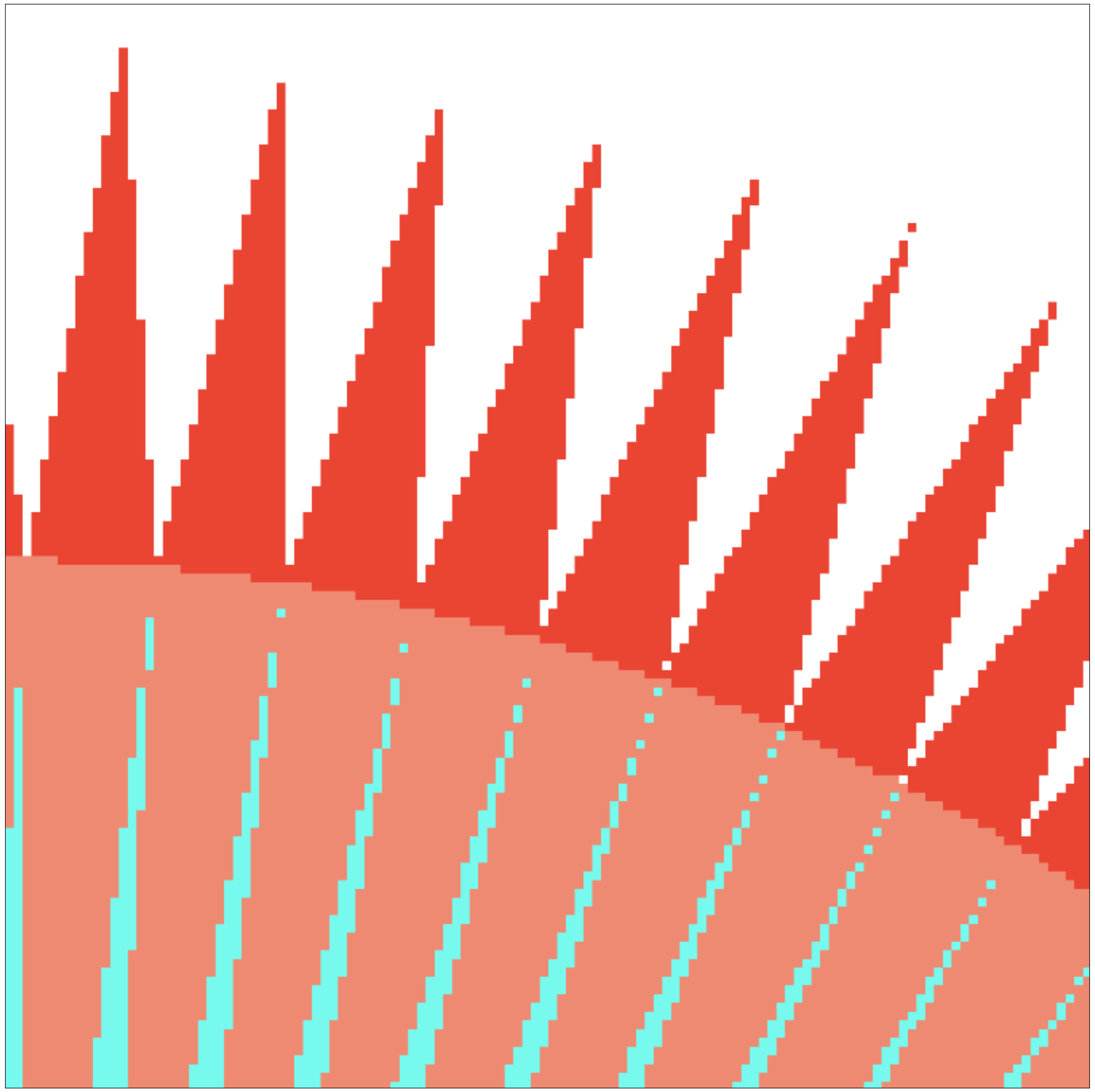
What's Wrong With This Picture?



Jaggies!

Aliasing





Is this the best we can do?

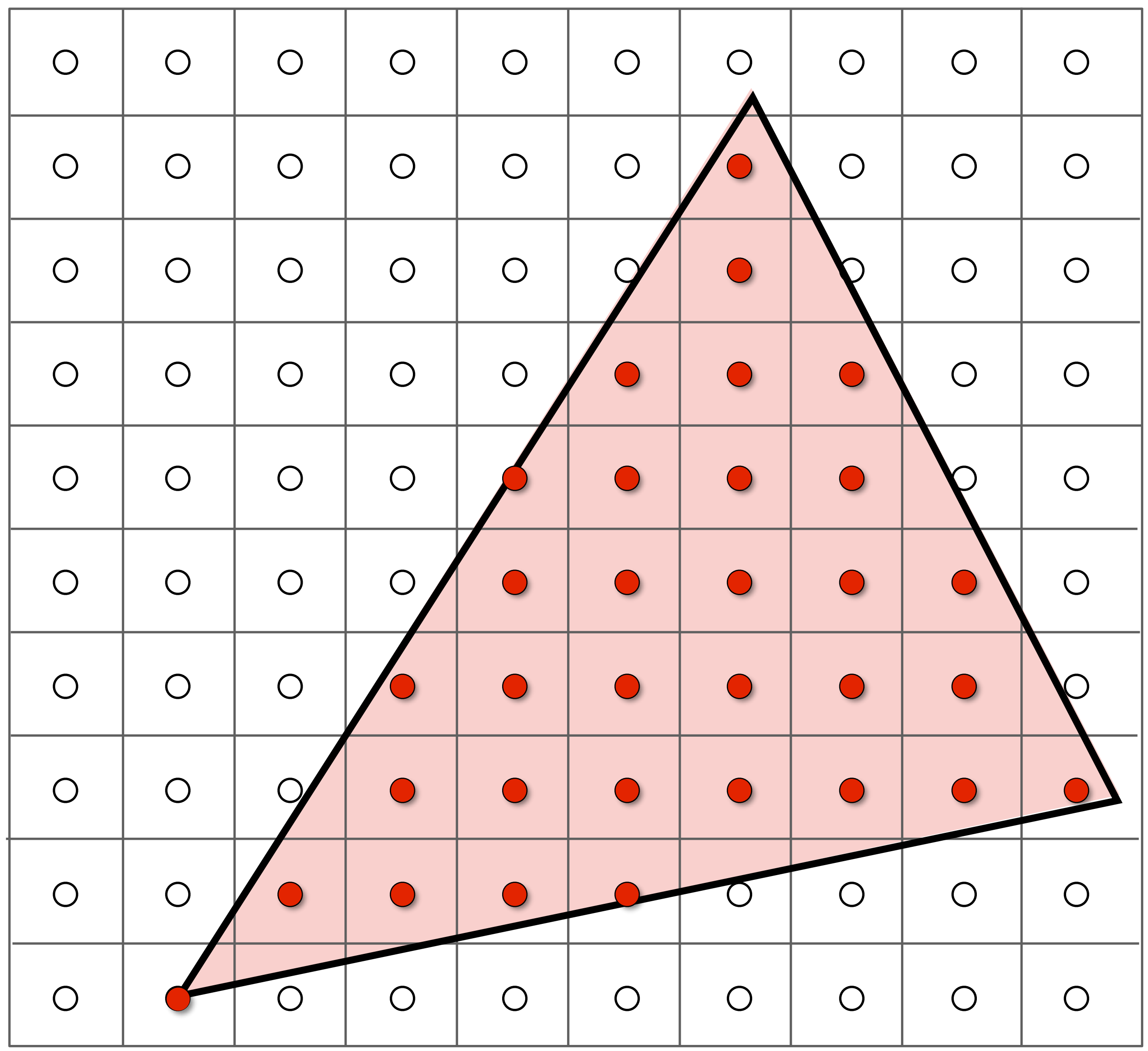
Slide courtesy of Prof. Ren Ng, UC Berkeley

Sampling is Ubiquitous in

普遍存在的

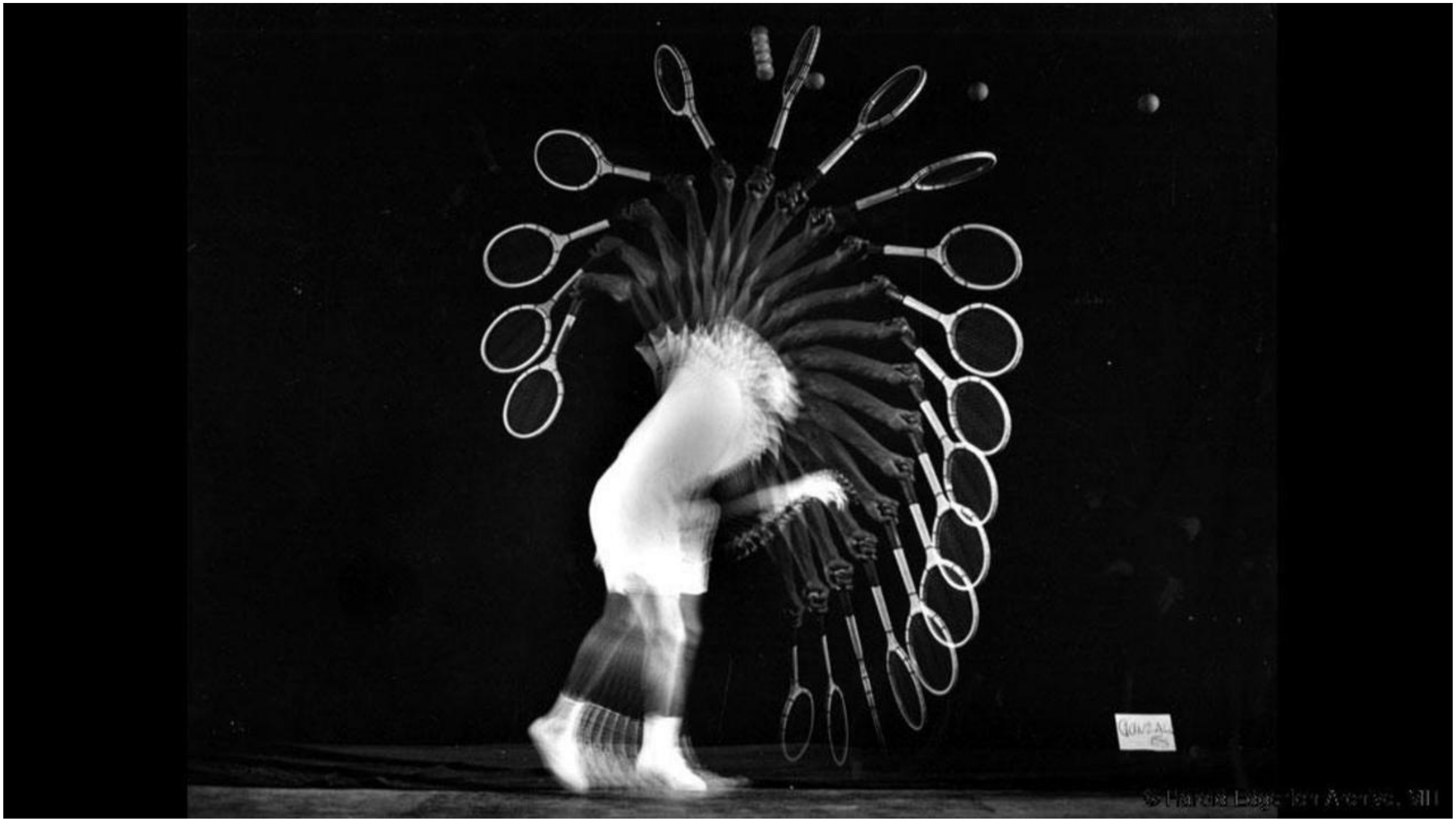
Computer Graphics

Rasterization = Sample 2D Positions



Photograph = Sample Image Sensor Plane

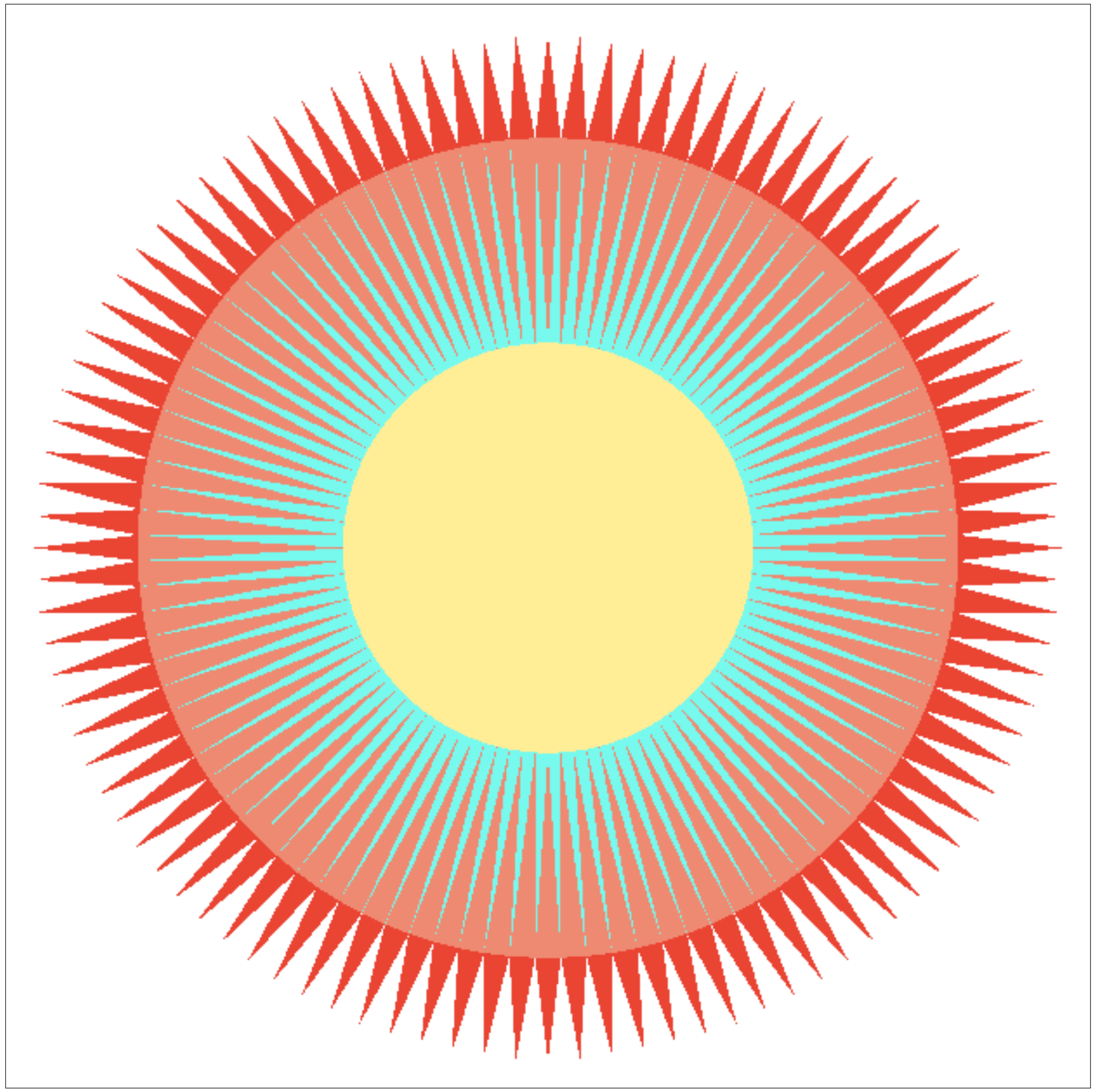
Video = Sample Time

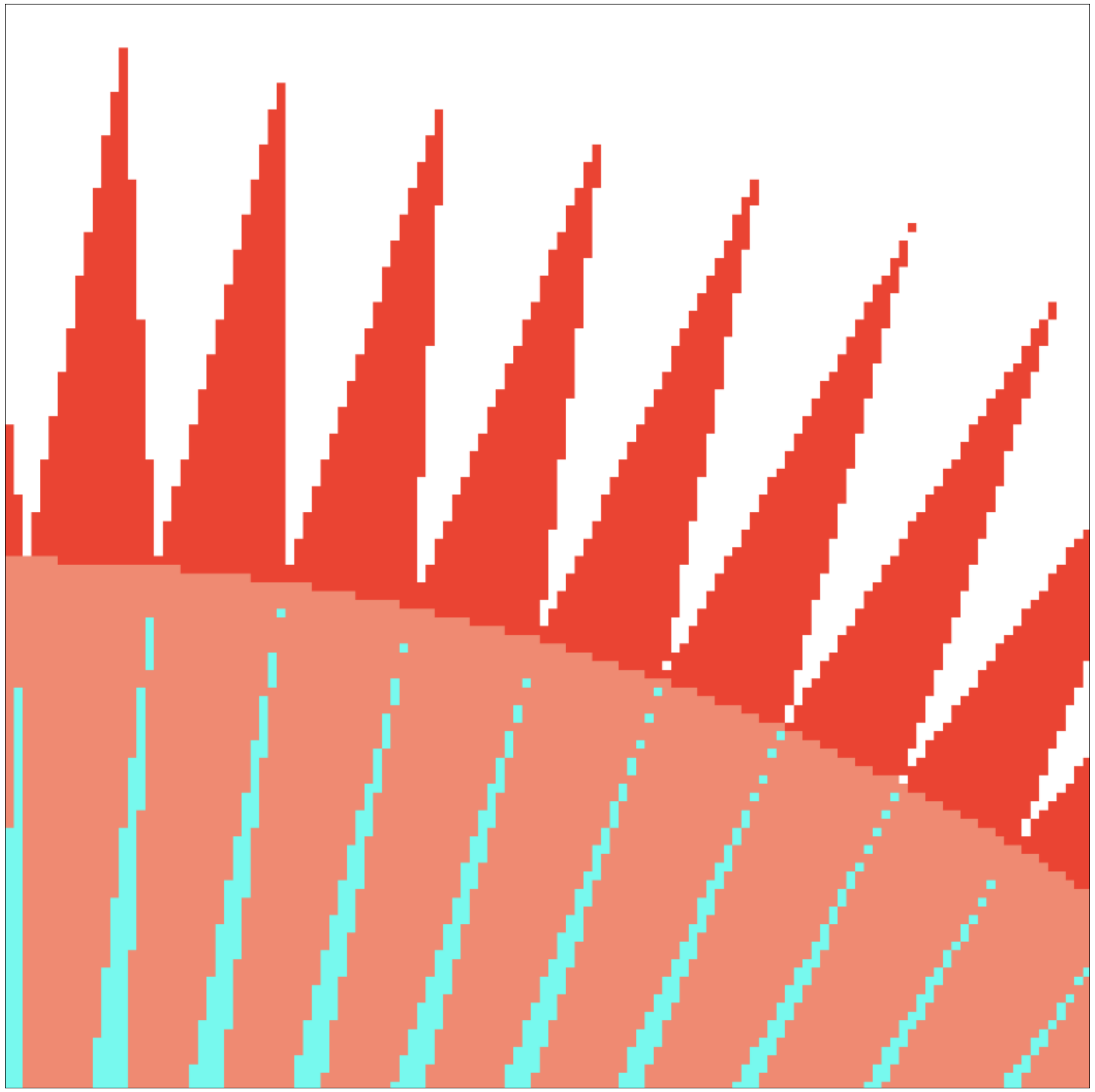


Harold Edgerton Archive, MIT

Sampling Artifacts
(Errors / Mistakes / Inaccuracies) in
Computer Graphics

Jaggies (Staircase Pattern)

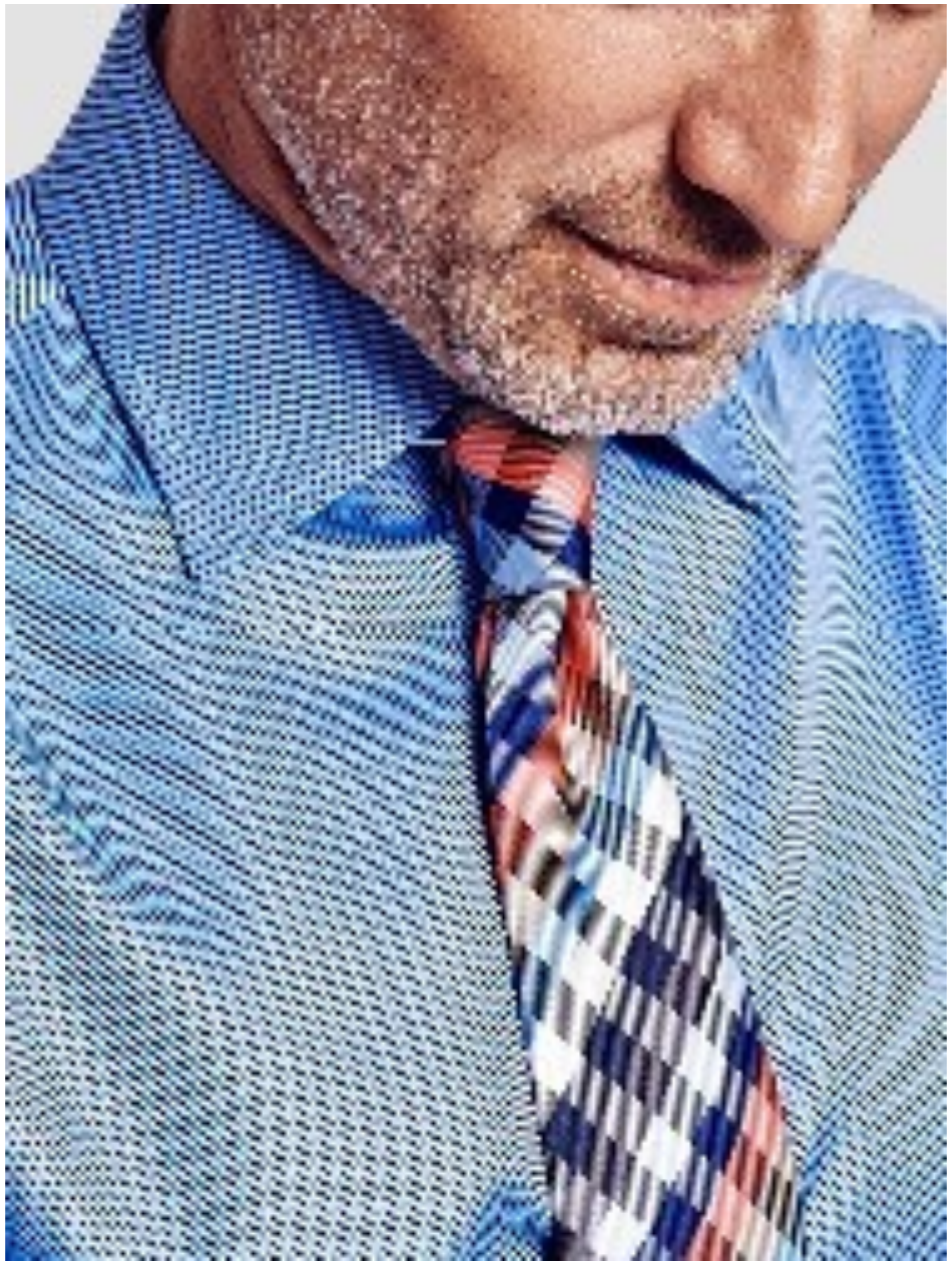




This is also an example of “aliasing” – a sampling error

Moiré Patterns in Imaging

[mwɑ:]



lystit.com

Skip odd rows and columns

隔行隔列采样的结果

Wagon Wheel Illusion (False Motion)

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

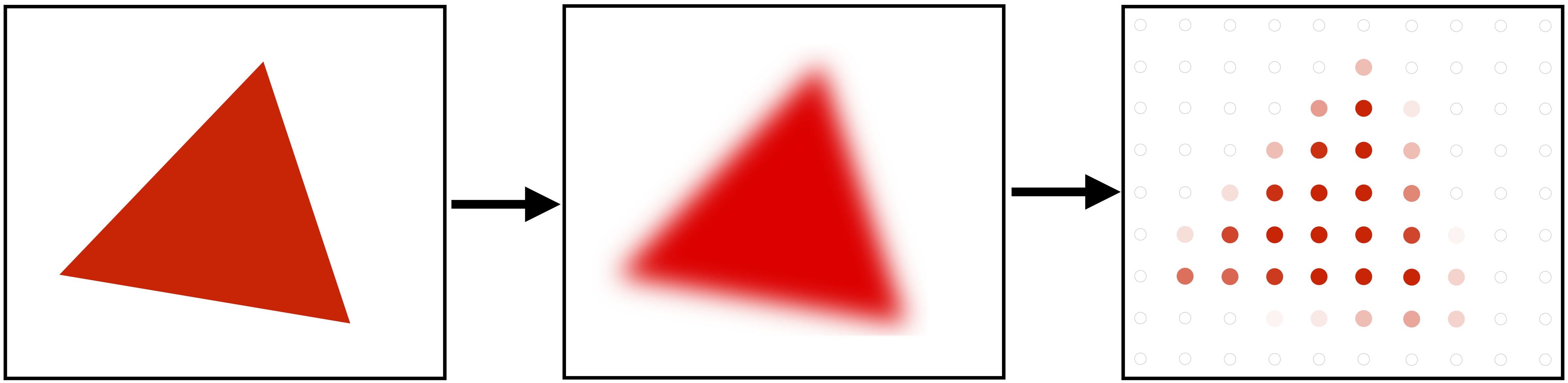
这些artifacts背后的原因: 信号变化速度太快(高频率)， 但采样速度相对太慢

Antialiasing Idea:
Blurring (Pre-Filtering) Before Sampling

Rasterization: Point Sampling in Space

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

Rasterization: Antialiased Sampling



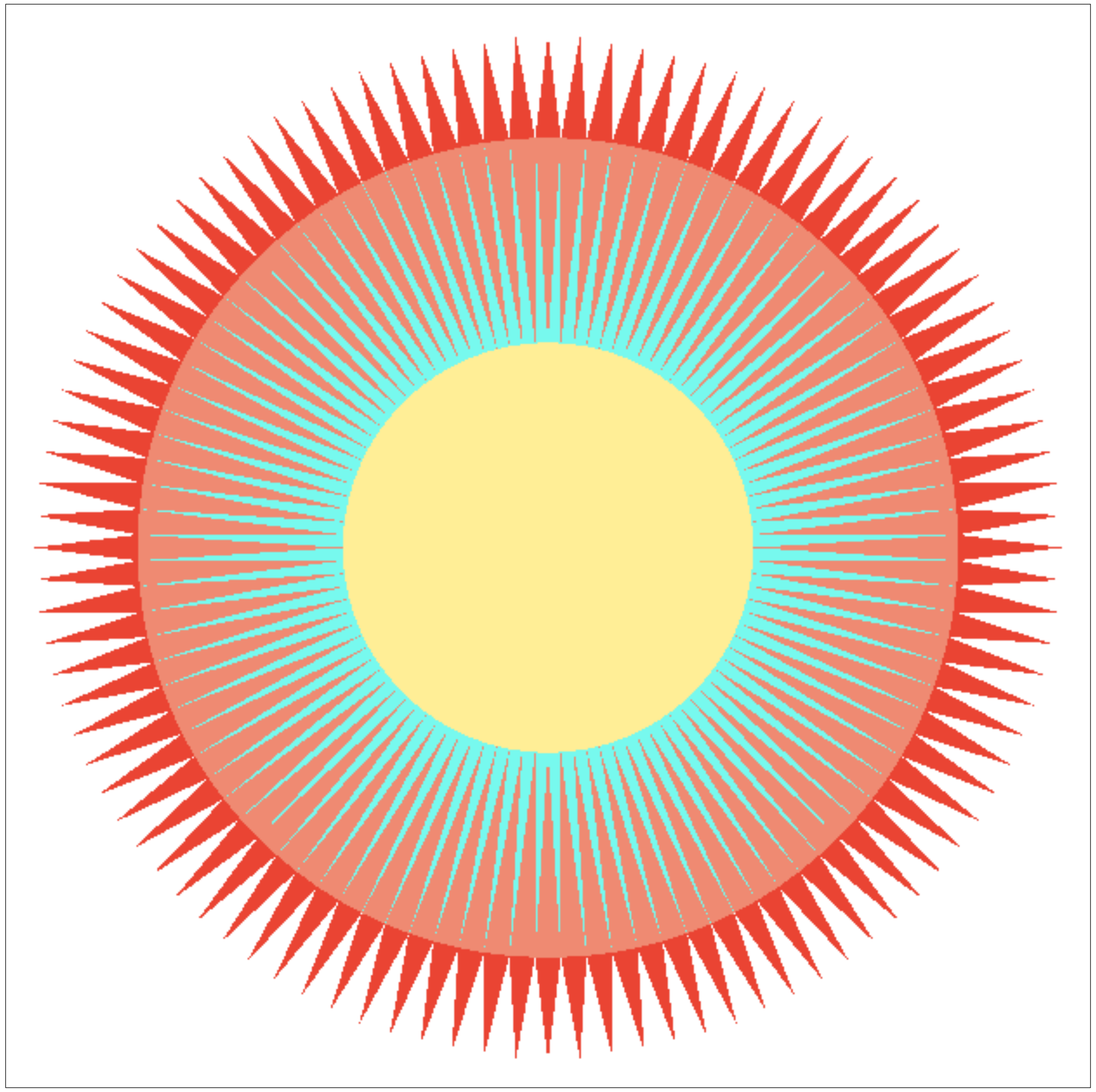
Pre-Filter
(remove frequencies above Nyquist) (?)

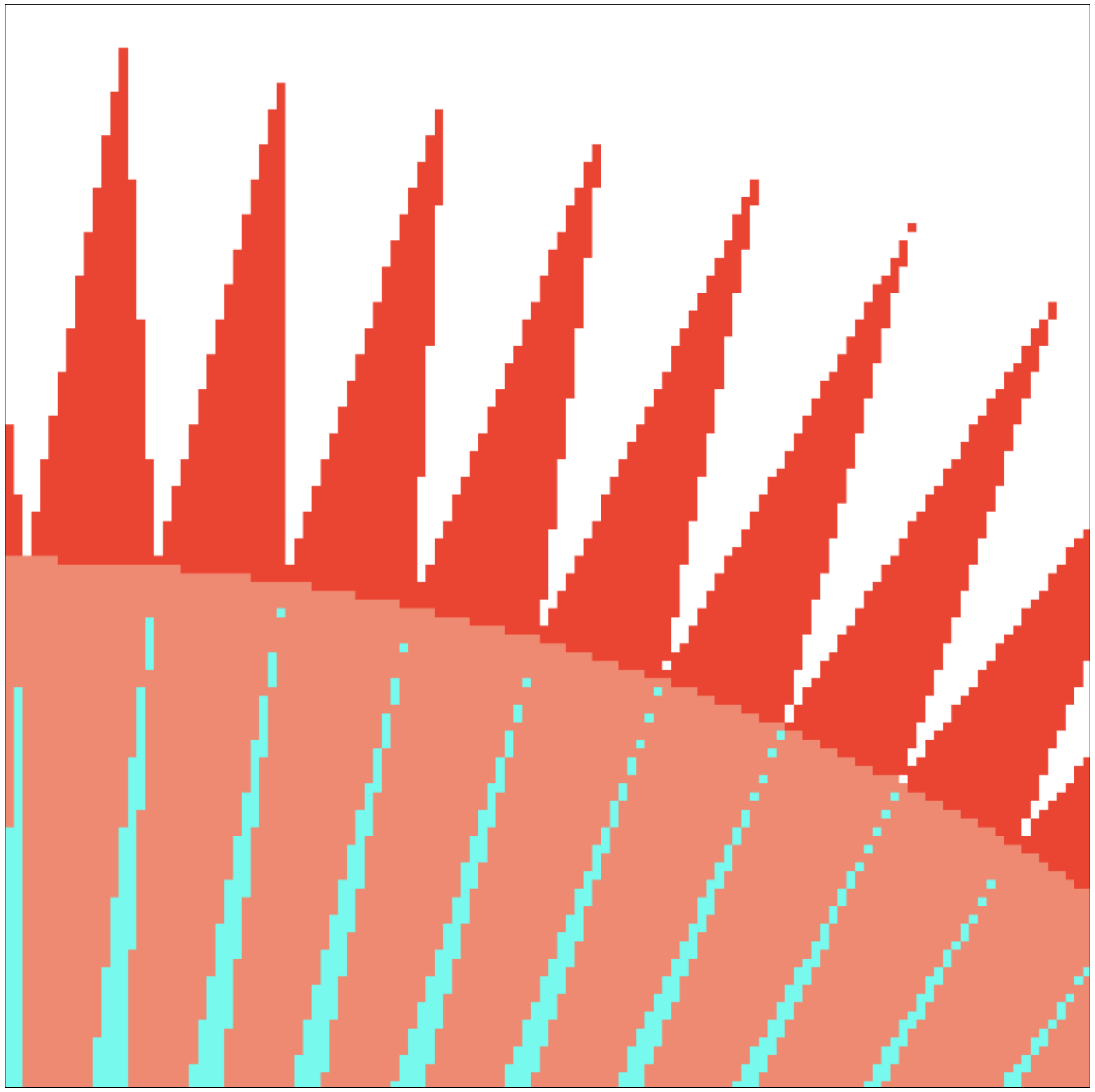
Sample

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

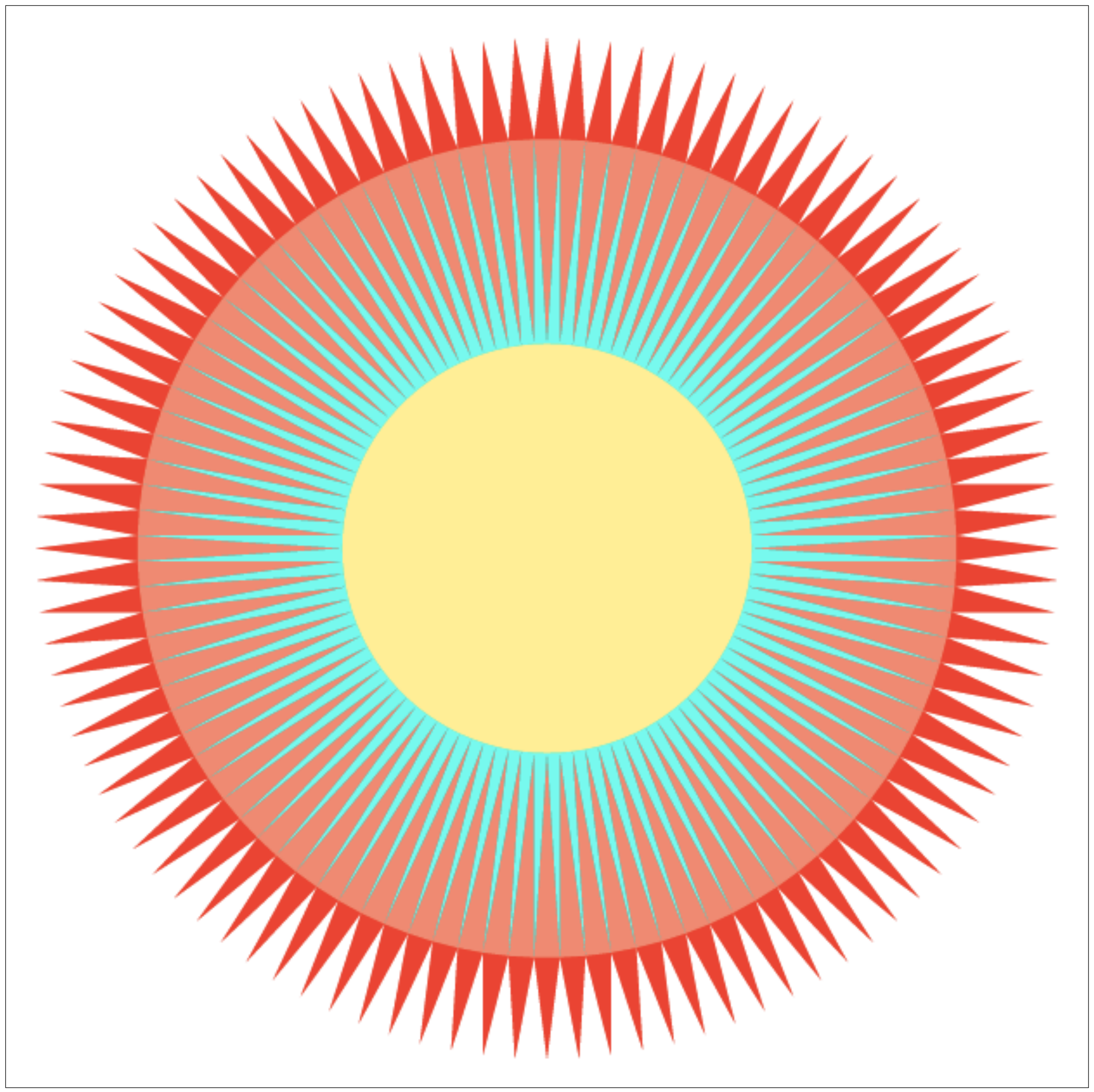
一种反走样的思想:先模糊，再采样： 模糊的部分，可以看到光栅化以后像素颜色取了个中间值(intermediate values)

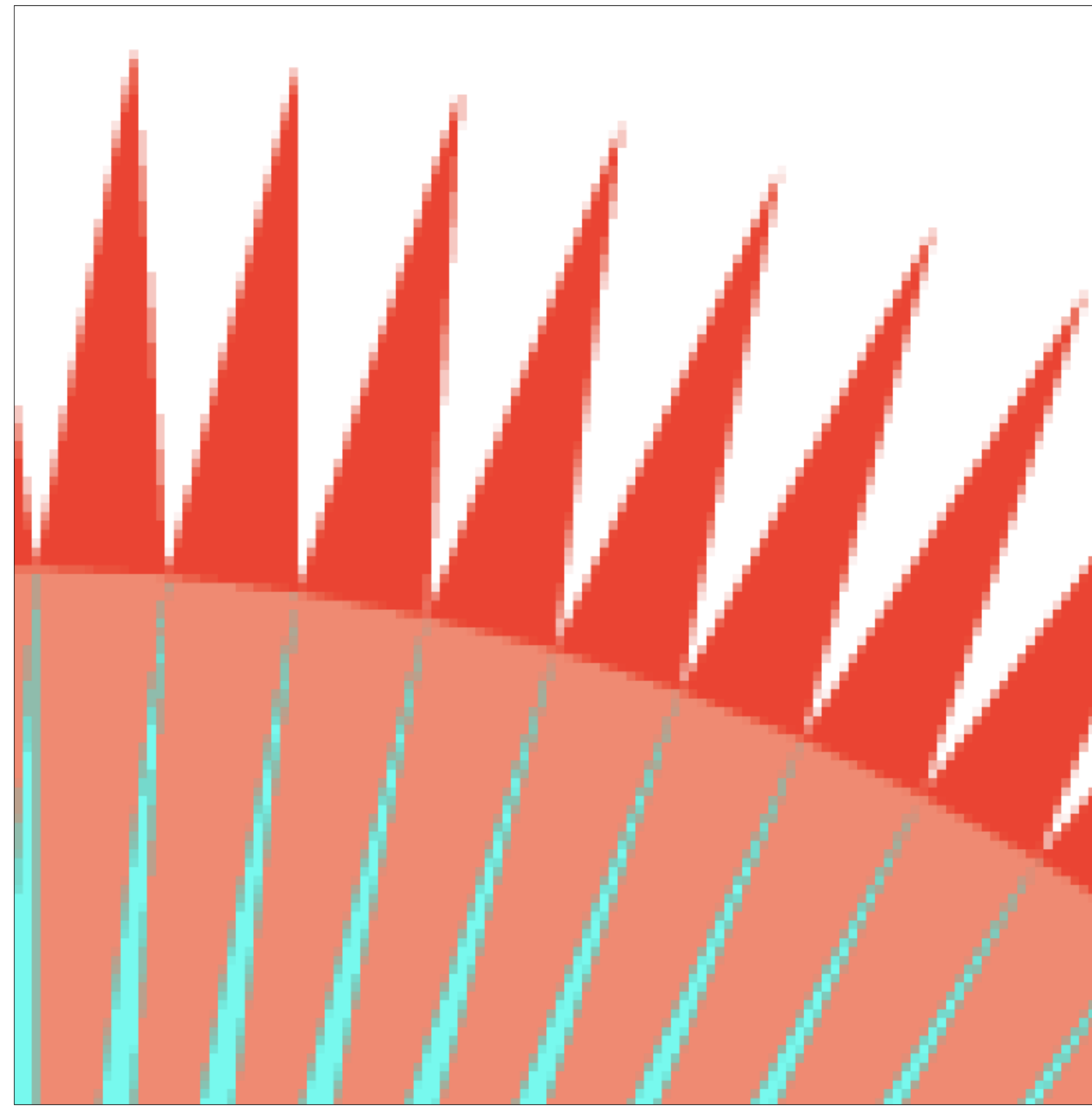
Point Sampling



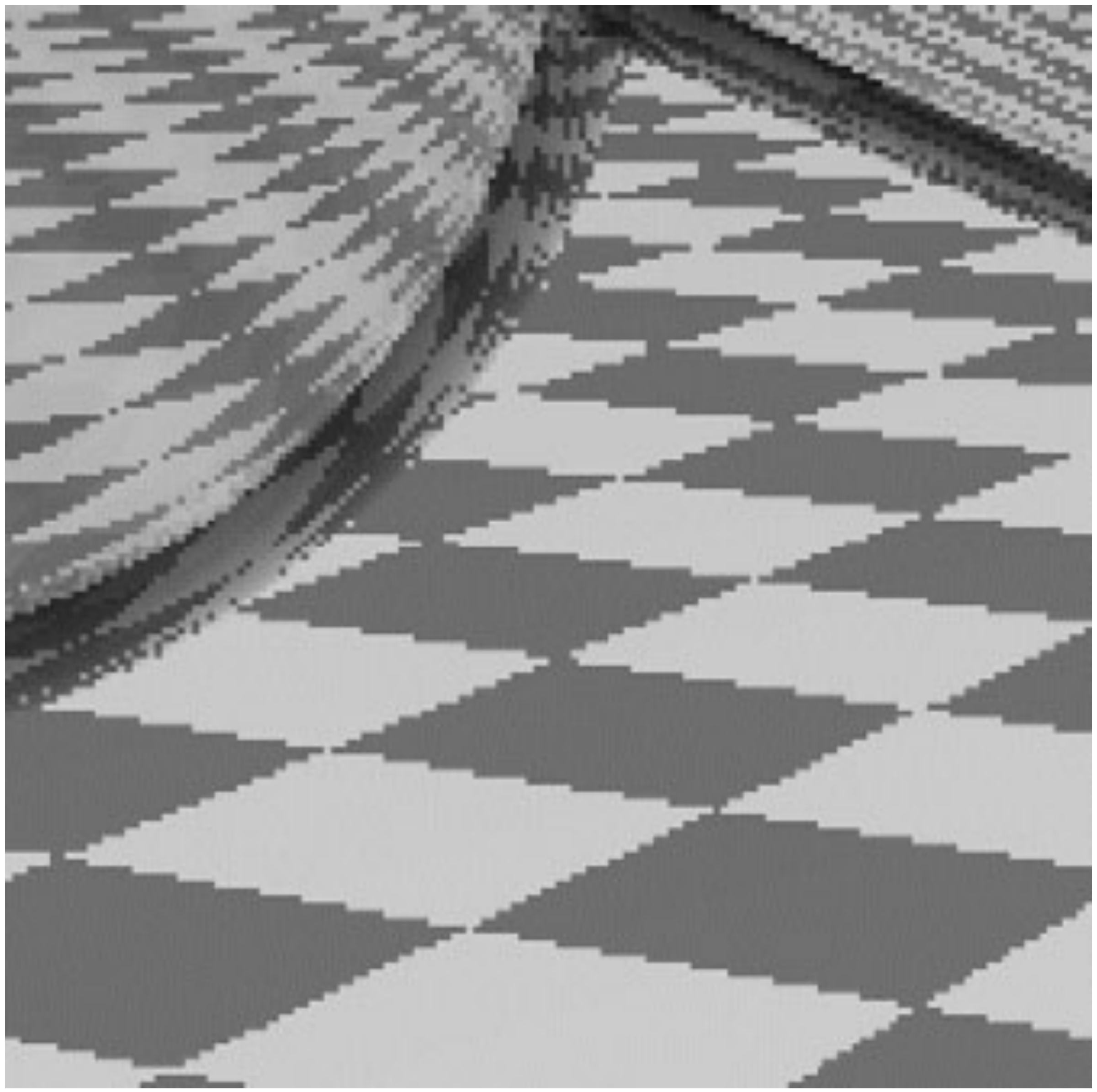


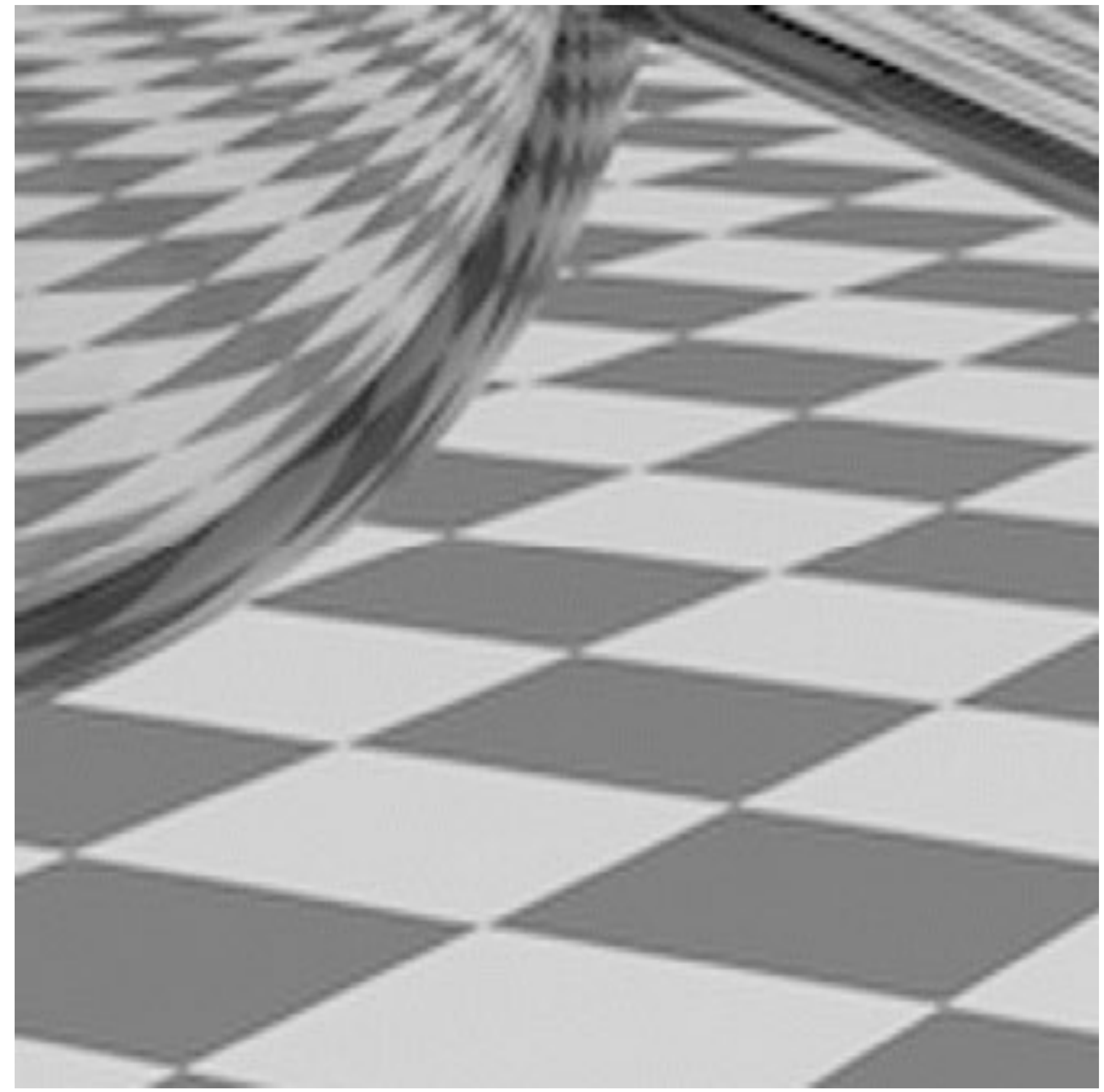
Antialiasing



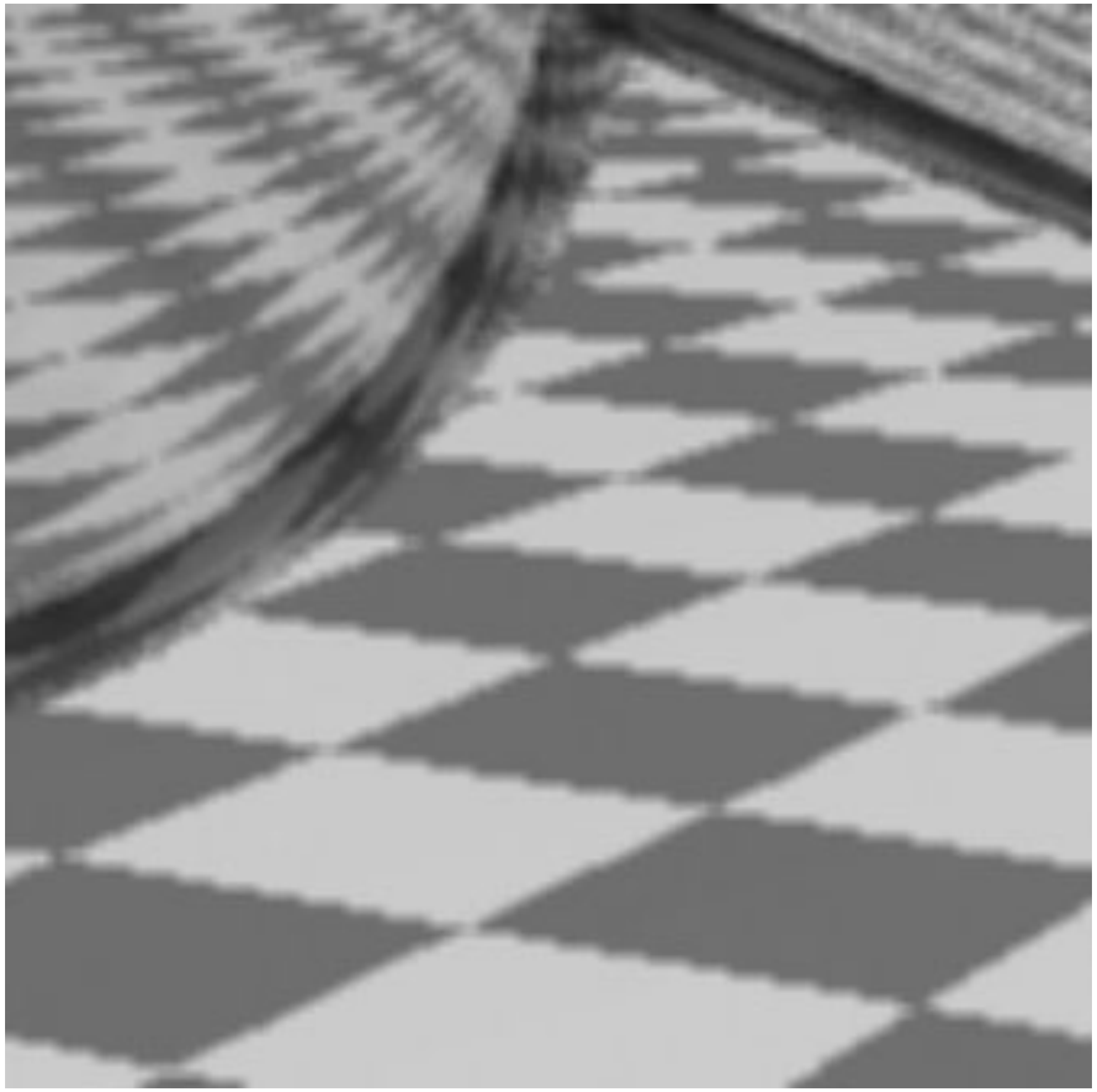


Point Sampling vs Antialiasing

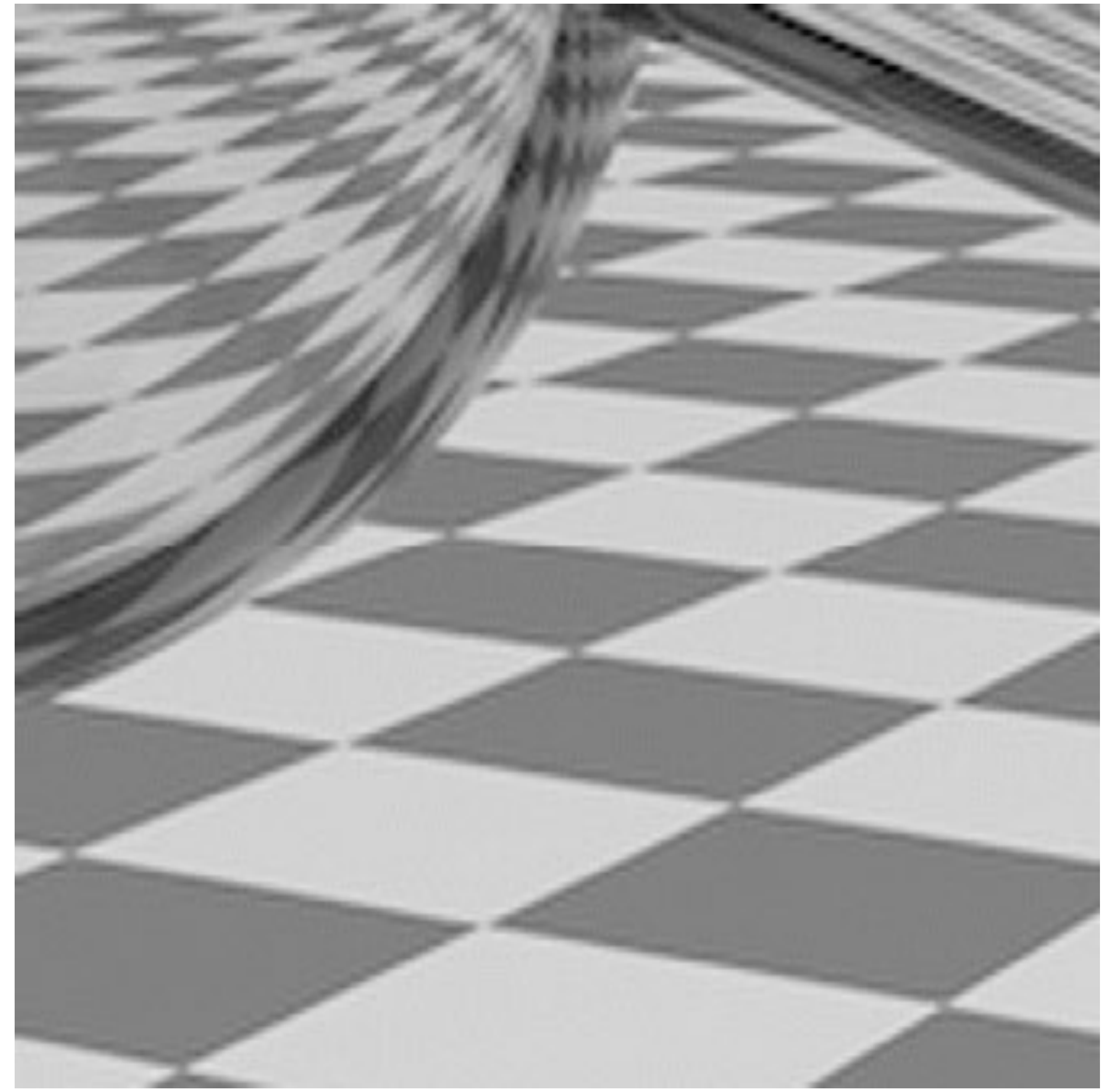




Antialiasing vs Blurred Aliasing



(Sample then filter, WRONG!)



(Filter then sample)

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?

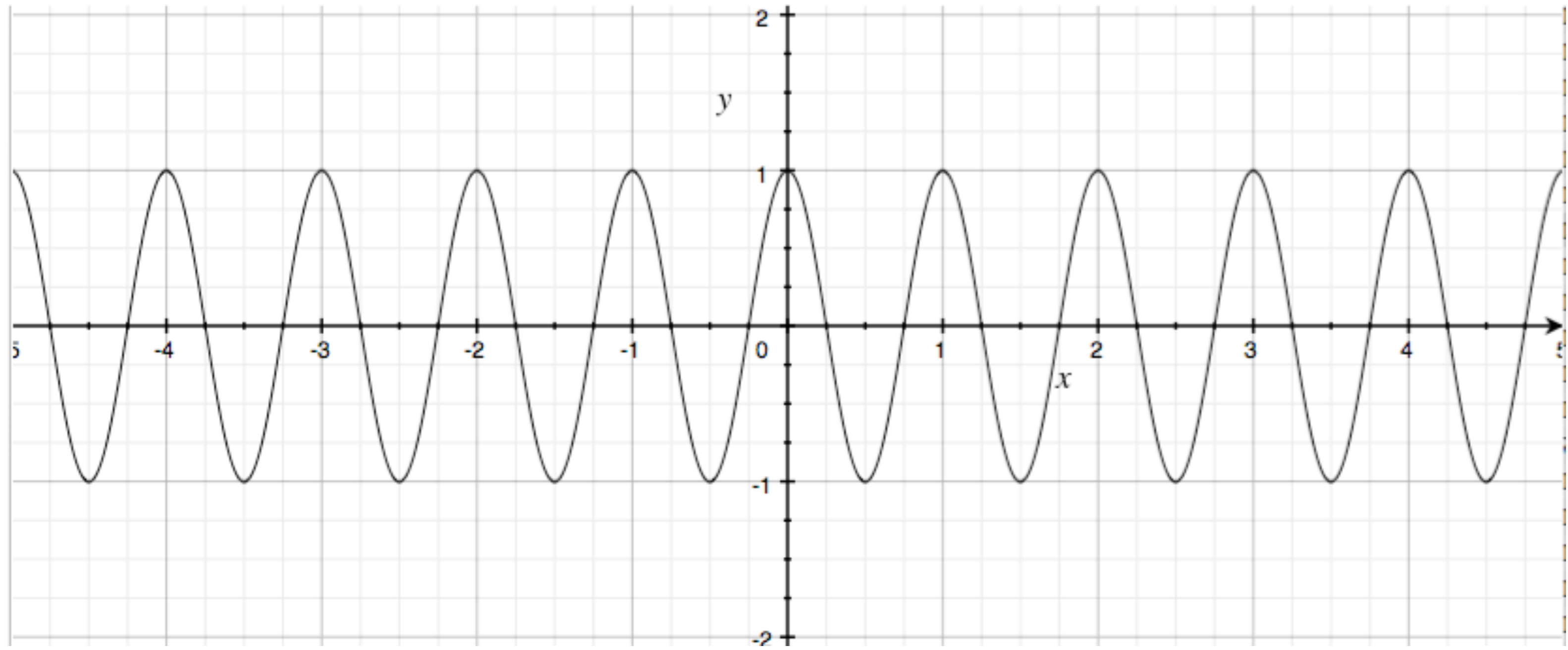
1. Why undersampling introduces aliasing?
2. Why pre-filtering then sampling can do antialiasing?

Let's dig into fundamental reasons

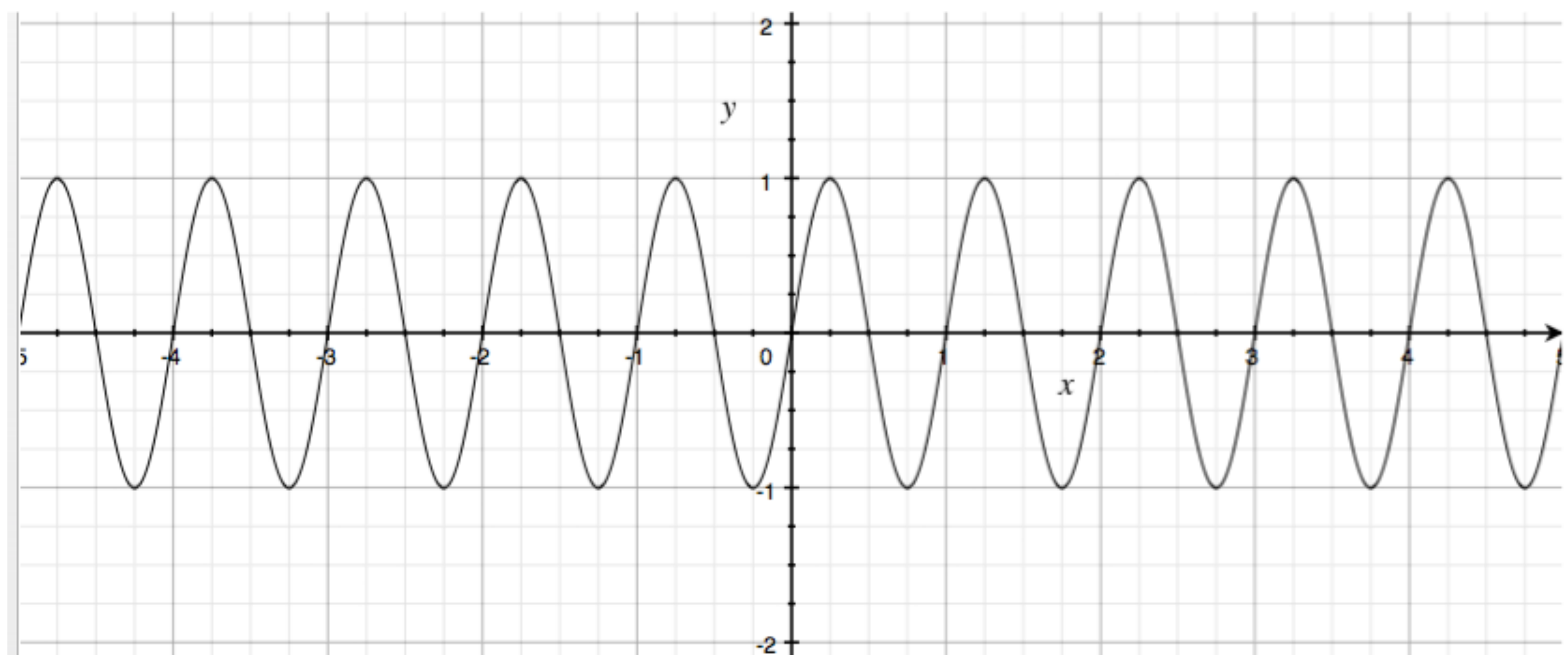
And look at how to implement antialiased rasterization

Frequency Domain

Sines and Cosines



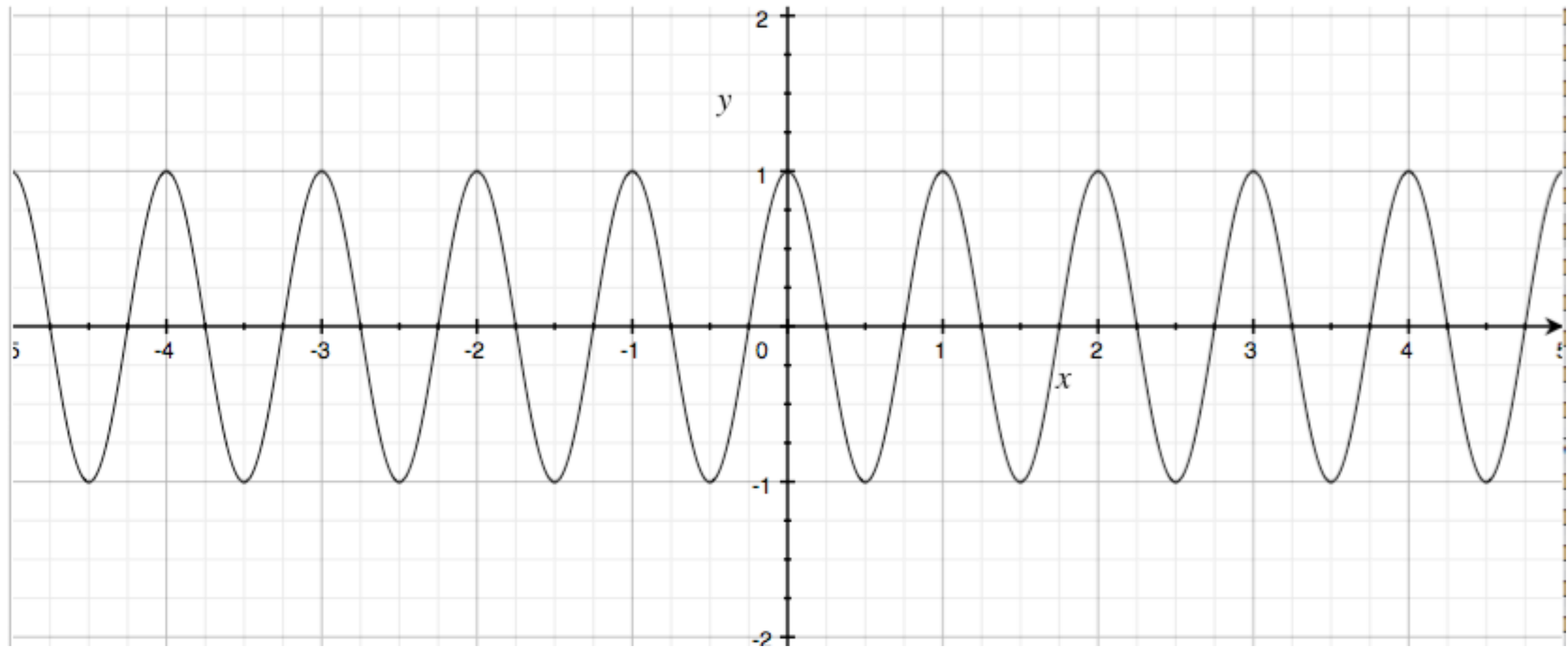
$$\cos 2\pi x$$



$$\sin 2\pi x$$

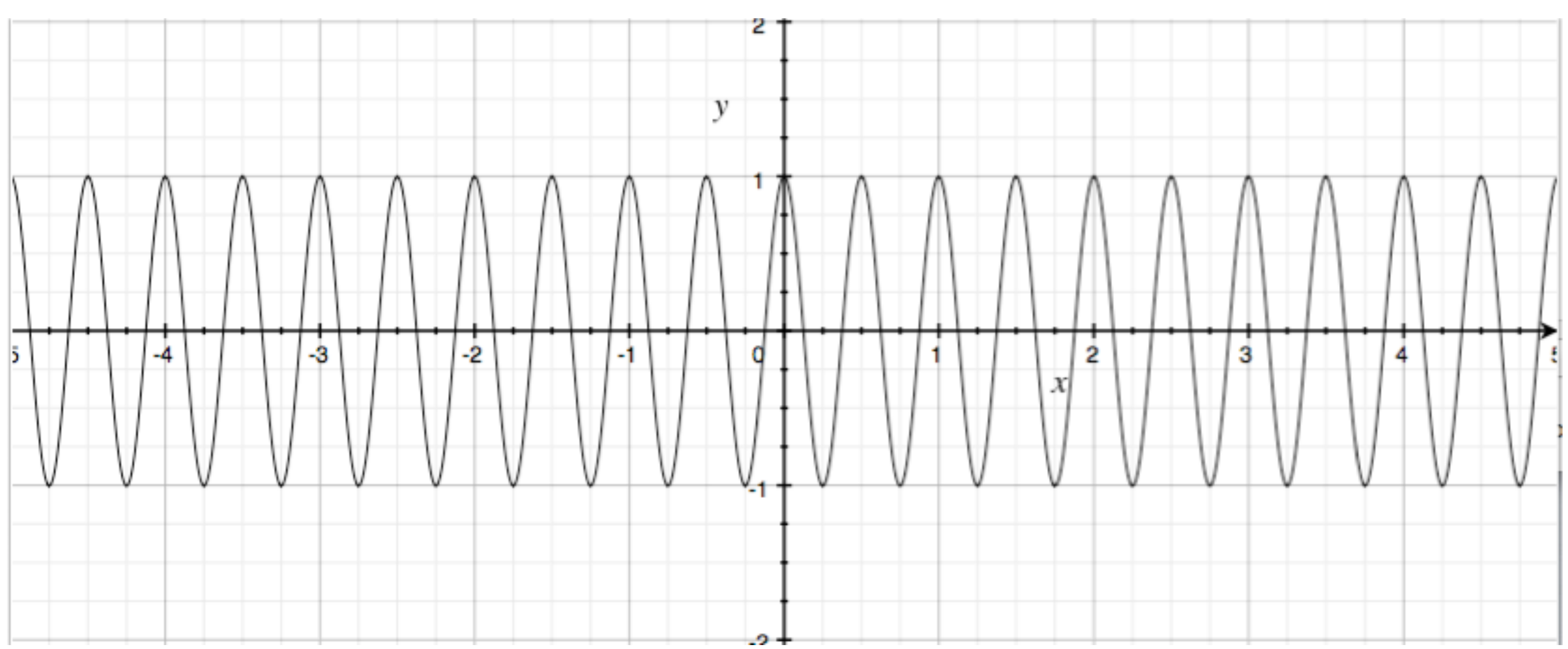
Frequencies $\cos 2\pi f x$

$$f = \frac{1}{T}$$



$$\cos 2\pi x$$

$$f = 1$$



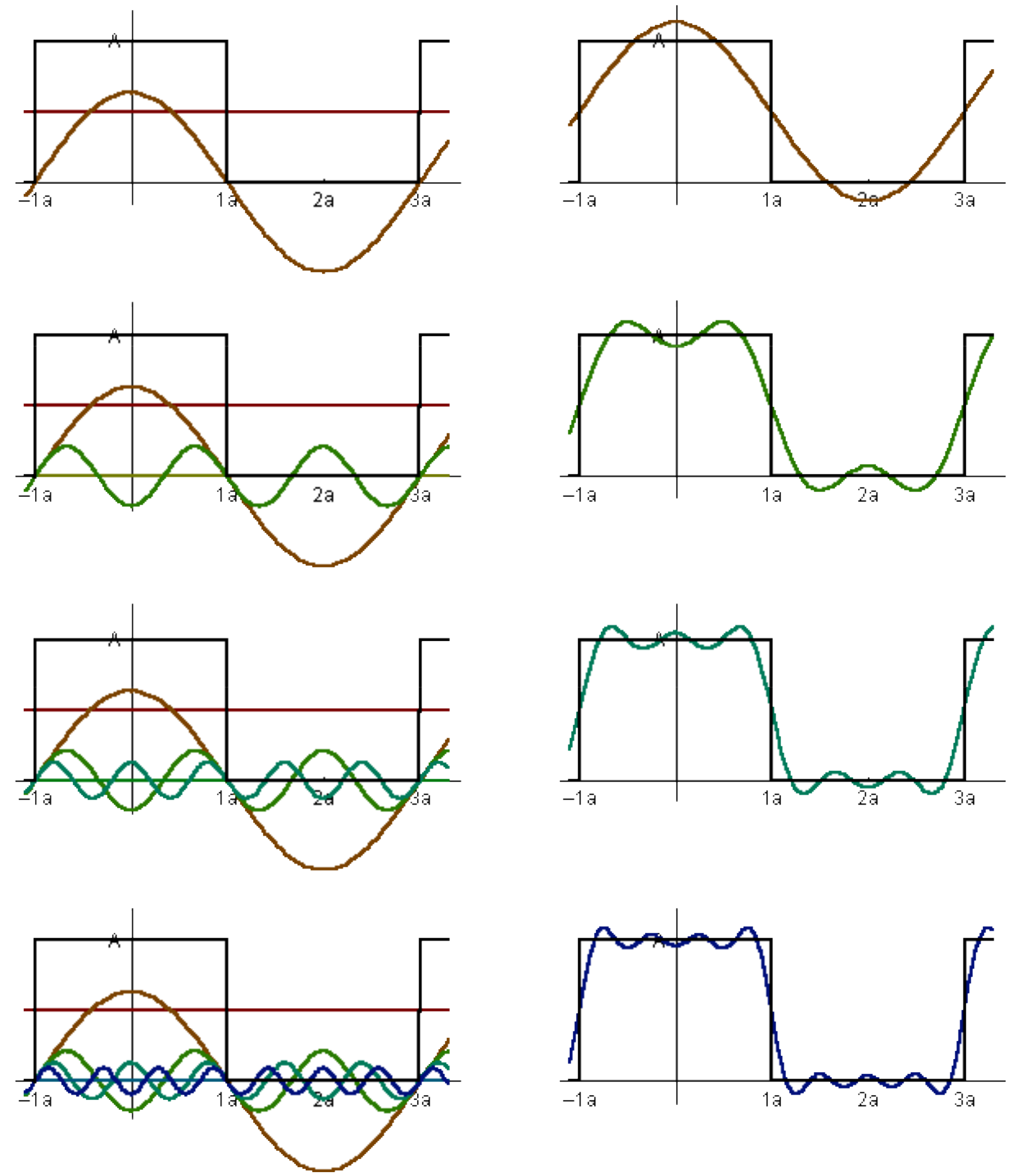
$$\cos 4\pi x$$

$$f = 2$$

Fourier Transform

Represent a function as a weighted sum of sines and cosines

Joseph Fourier 1768 - 1830



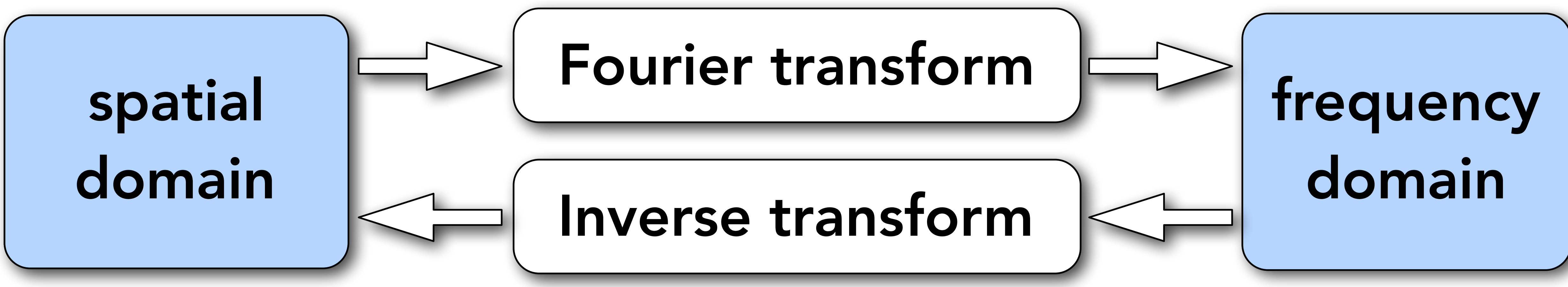
$$f(x) = \frac{A}{2} + \frac{2A\cos(t\omega)}{\pi} - \frac{2A\cos(3t\omega)}{3\pi} + \frac{2A\cos(5t\omega)}{5\pi} - \frac{2A\cos(7t\omega)}{7\pi} + \cdots$$

傅立叶变换:任何一个周期函数，都能用sin， cos及常数项表示

Fourier Transform Decomposes A Signal Into Frequencies

傅立叶变换性质

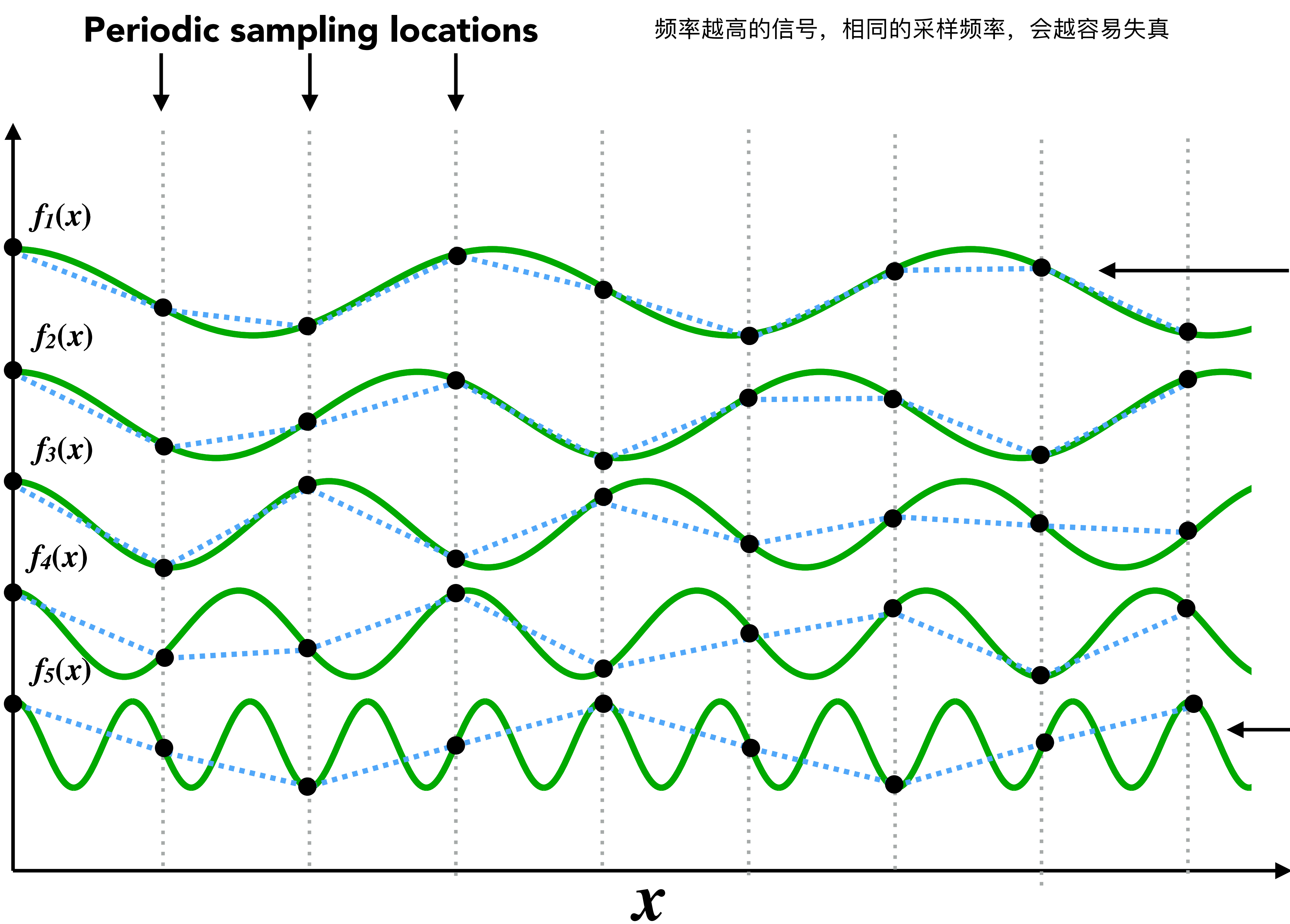
$$f(x) \qquad F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i \omega x} dx \qquad F(\omega)$$



$$f(x) = \int_{-\infty}^{\infty} F(\omega) e^{2\pi i \omega x} d\omega$$

Recall $e^{ix} = \cos x + i \sin x$

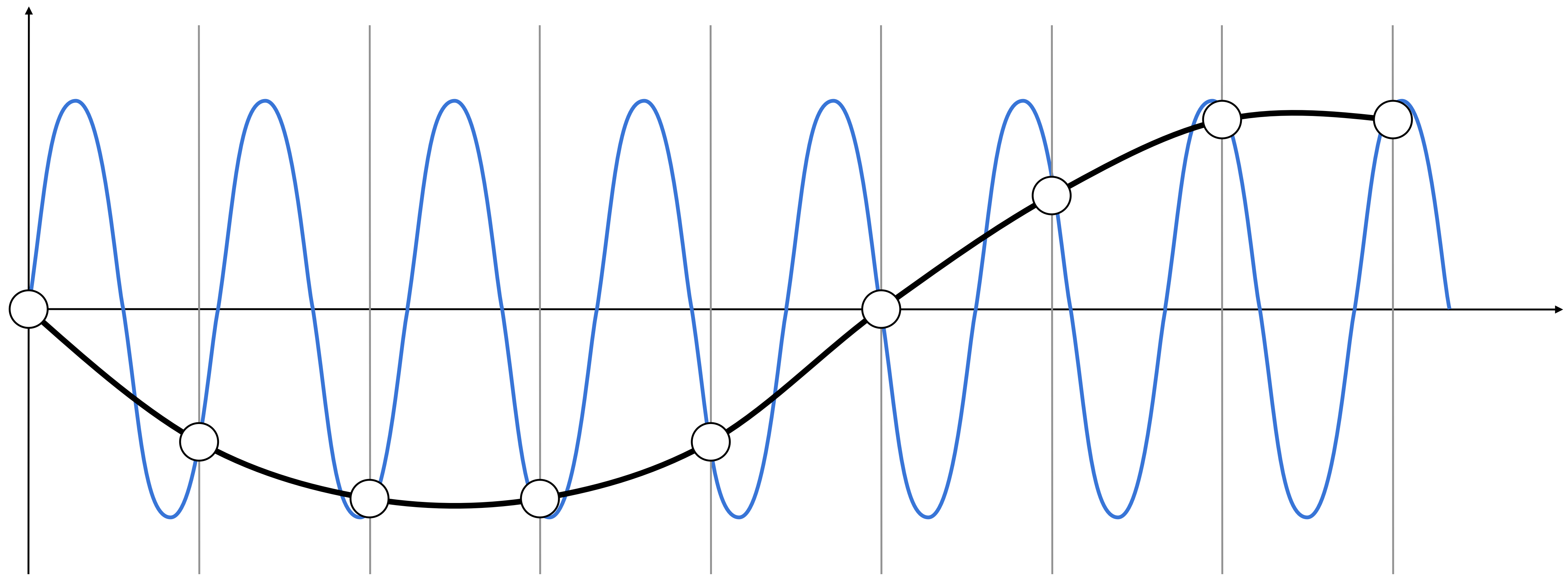
Higher Frequencies Need Faster Sampling



Low-frequency signal: sampled adequately for reasonable reconstruction

High-frequency signal is insufficiently sampled: reconstruction incorrectly appears to be from a low frequency signal

Undersampling Creates Frequency Aliases



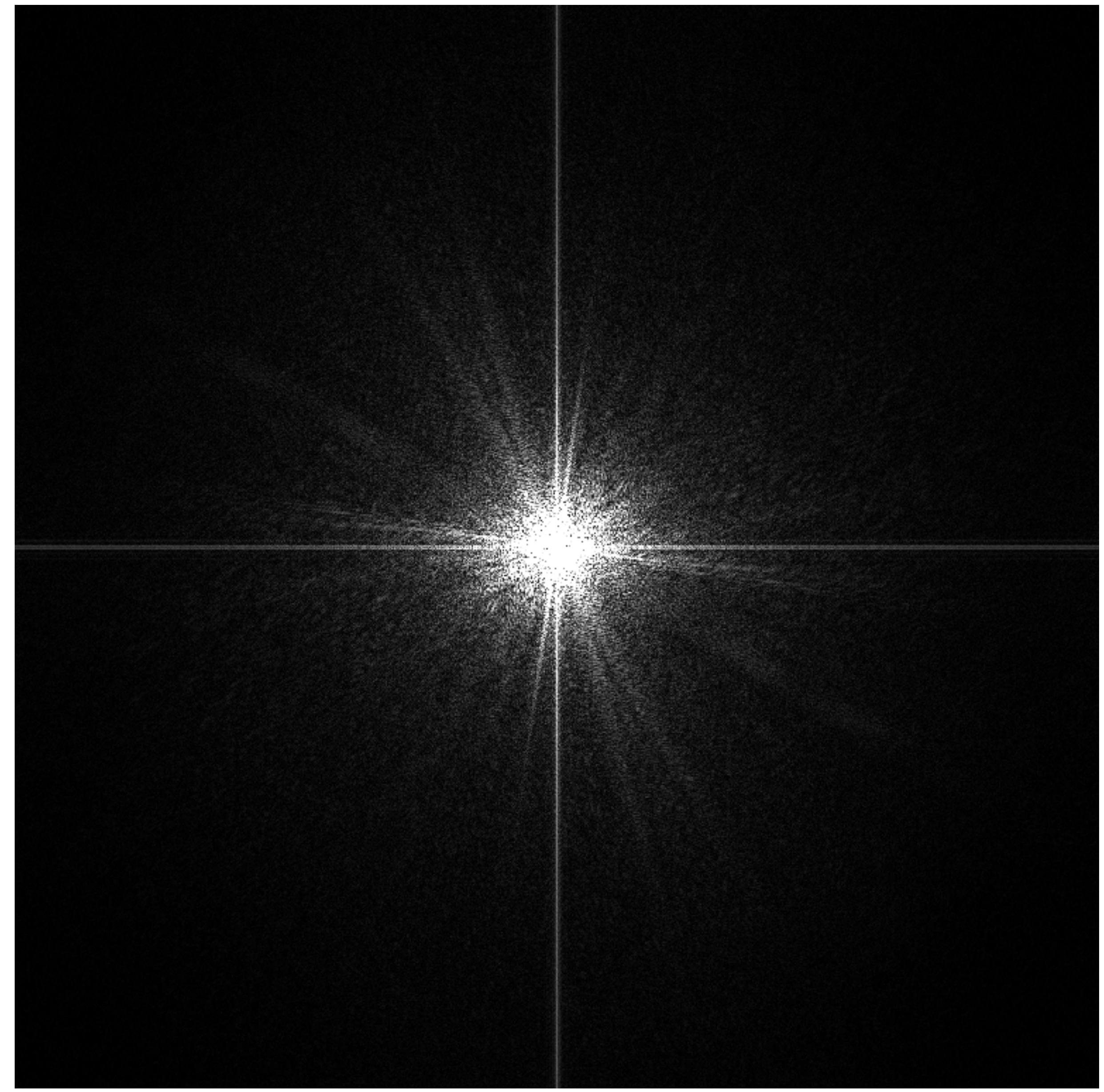
两个不同的函数采样出来的结果是一样的，这就是“走样”

High-frequency signal is insufficiently sampled: samples erroneously appear to be from a low-frequency signal

Two frequencies that are indistinguishable at a given sampling rate are called “aliases”

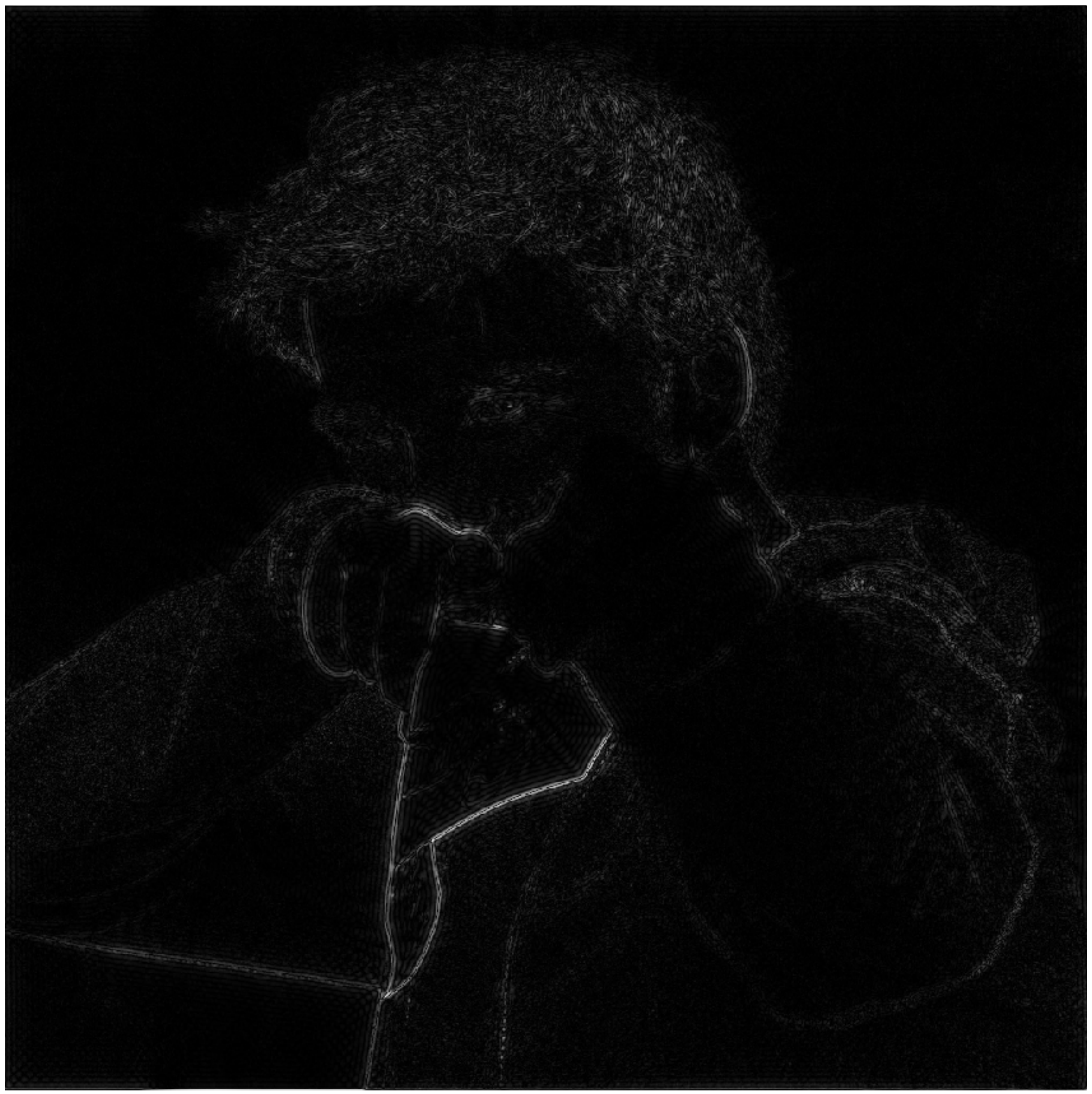
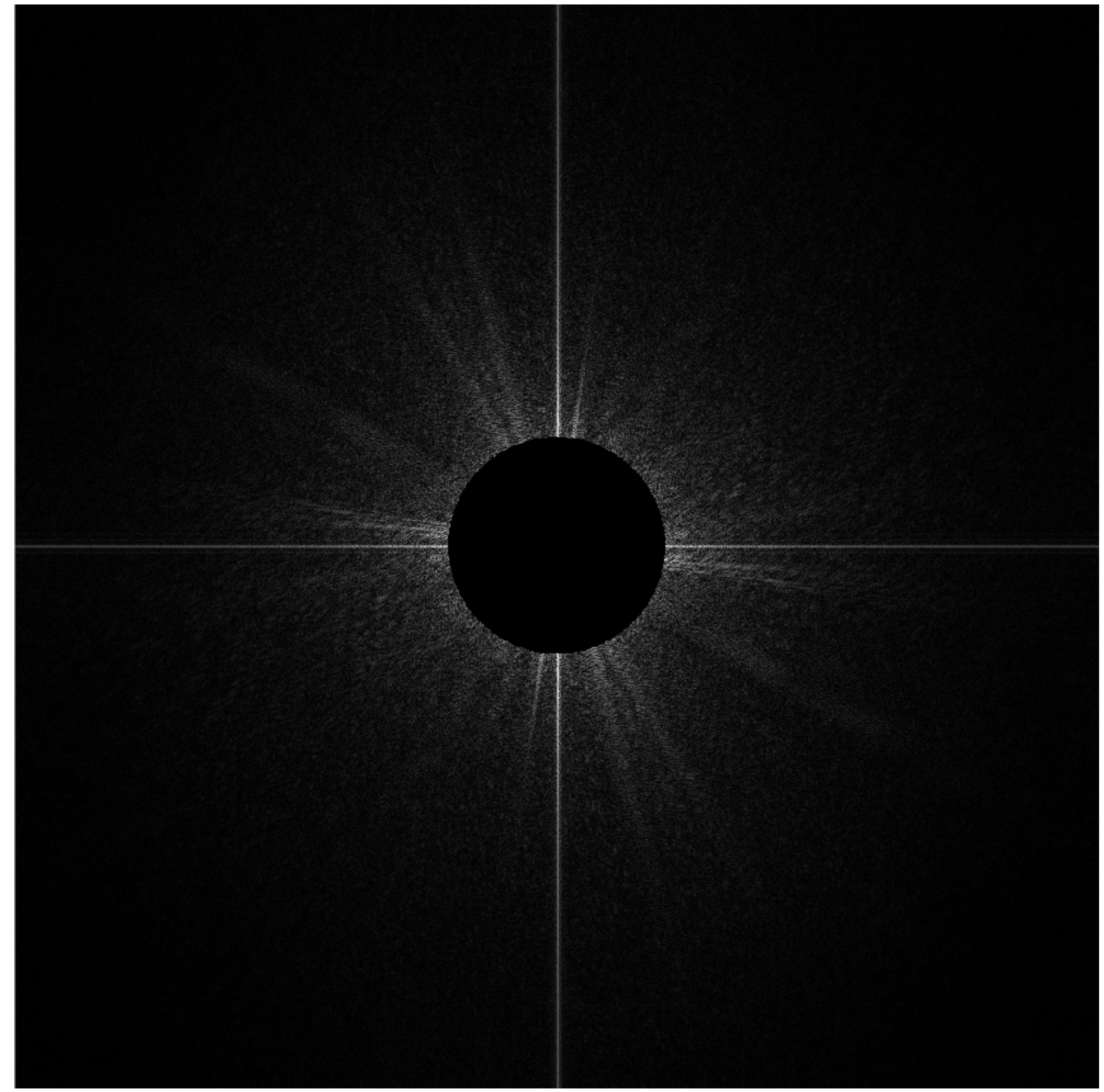
Filtering = Getting rid of certain frequency contents

Visualizing Image Frequency Content



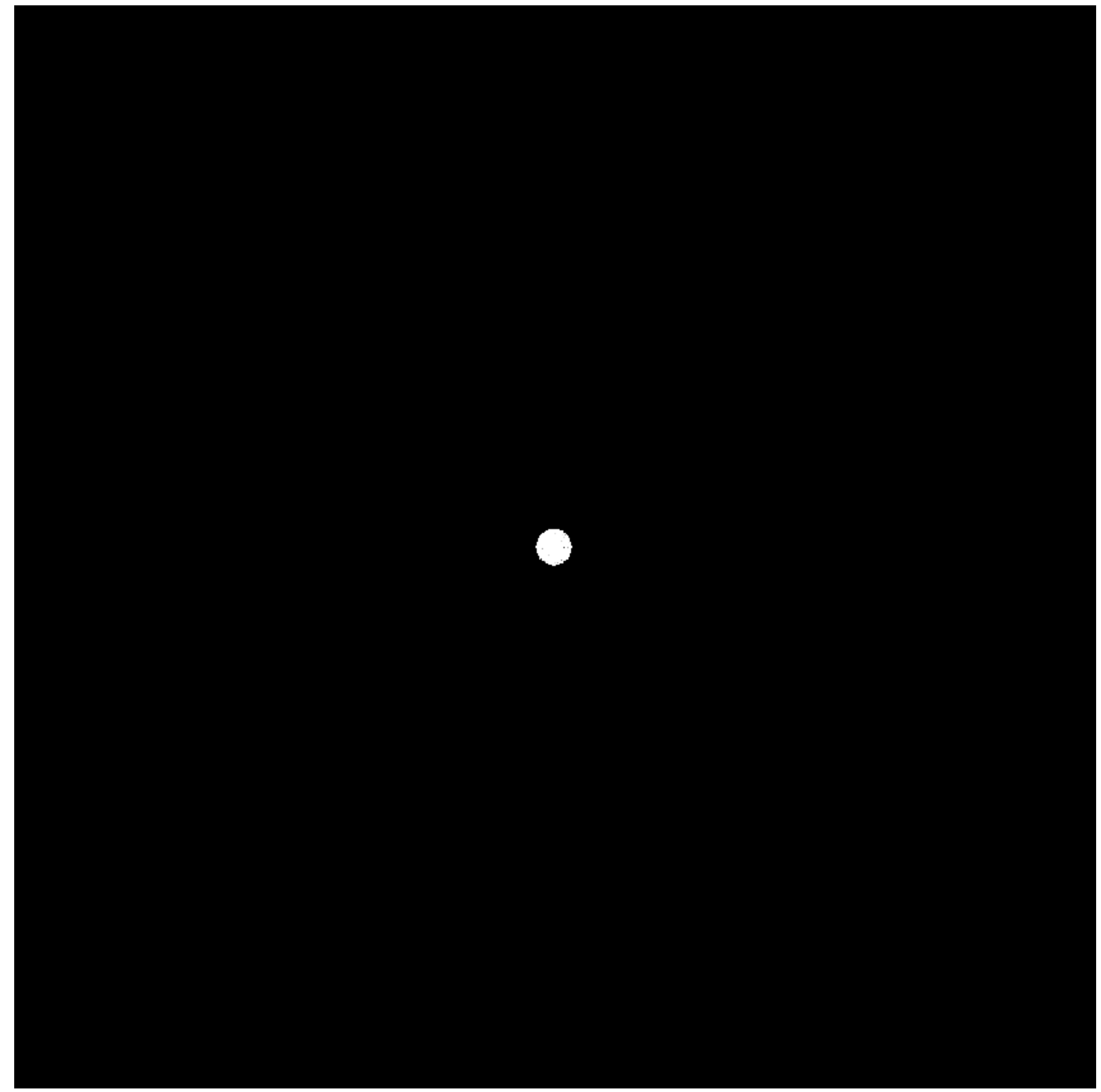
傅立叶图像中间是低频，四周是高频

Filter Out Low Frequencies Only (Edges)

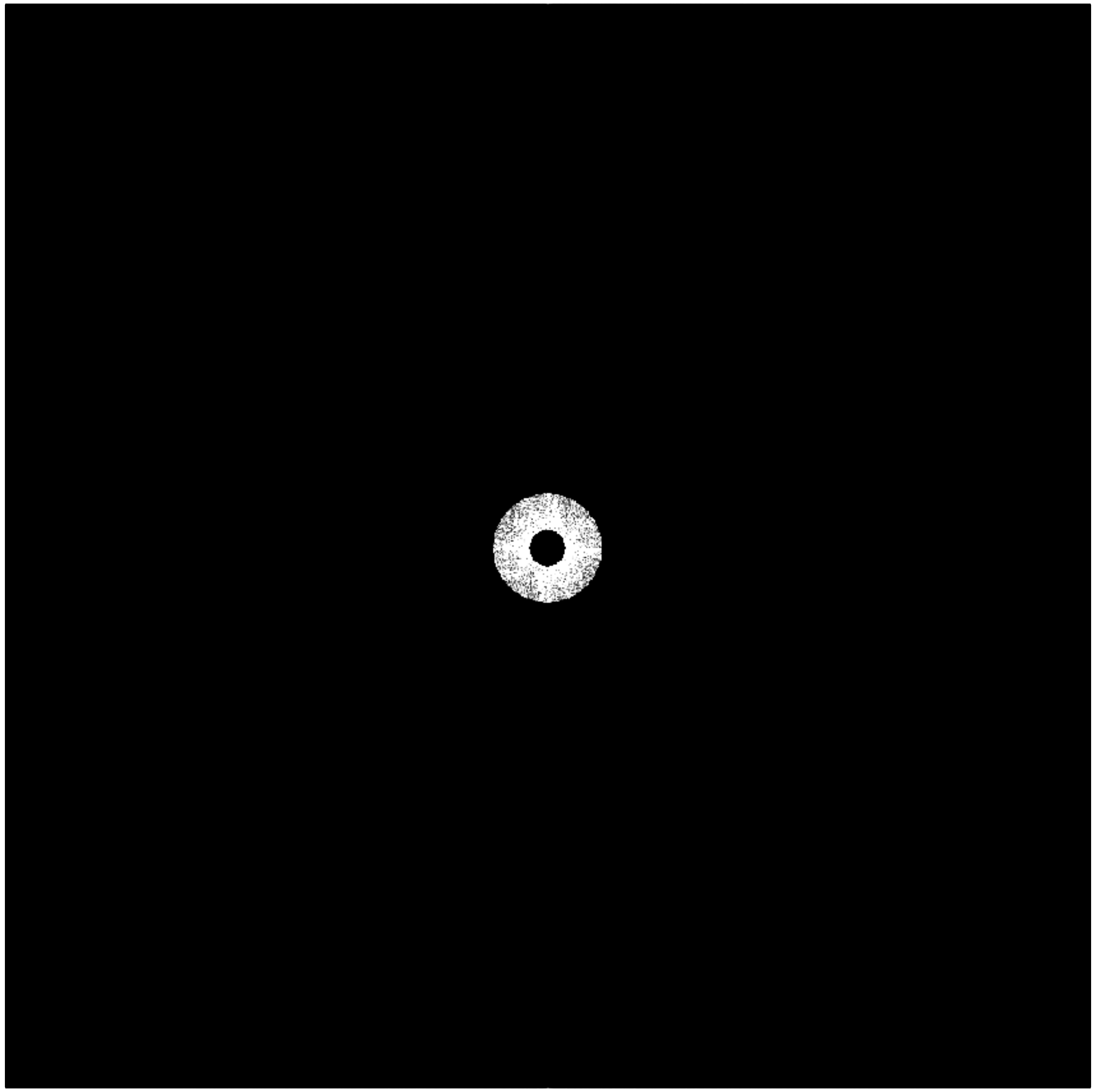
High-pass filter

Filter Out High Frequencies (Blur)



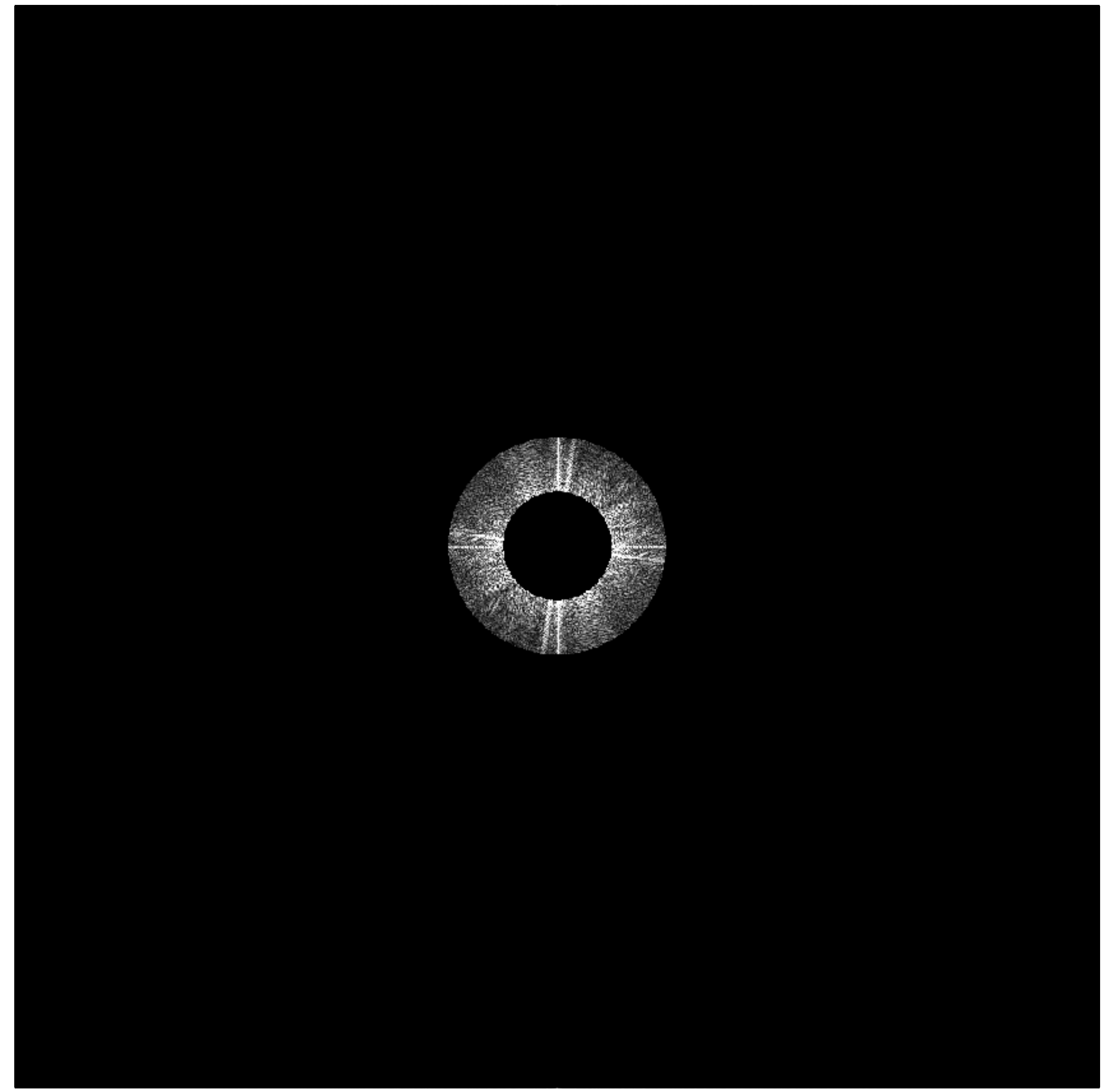
Low-pass filter

Filter Out Low and High Frequencies



带通滤波器

Filter Out Low and High Frequencies



Filtering = Convolution

(= Averaging)

Convolution

Signal	1	3	5	3	7	1	3	8	6	4

Filter	1/4	1/2	1/4

Point-wise local averaging in a “sliding window”

Convolution

Signal

1	3	5	3	7	1	3	8	6	4

Filter

1/4	1/2	1/4

1 x (1/4) + 3 x (1/2) + 5 x (1/4) = 3

Result

	3								

Convolution

Signal

1	3	5	3	7	1	3	8	6	4

Filter

1/4	1/2	1/4

$$3 \times (1/4) + 5 \times (1/2) + 3 \times (1/4) = 4$$

Result

	3	4							

Convolution Theorem

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

空间域的卷积相当于频域的乘积，反之亦然

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)

Convolution Theorem

空域操作与频域操作对应

Spatial Domain

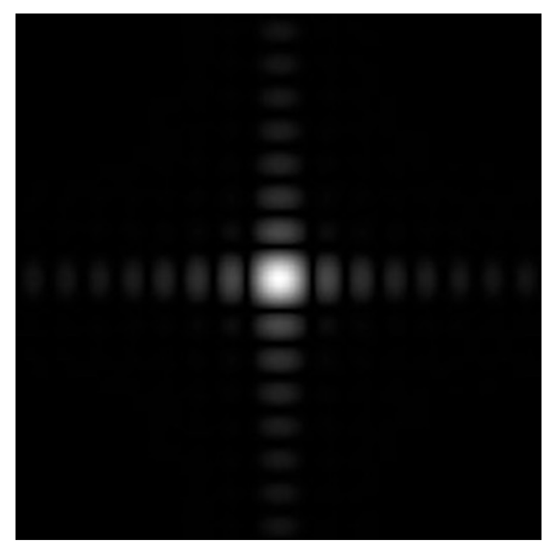
$$* \ \frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix} =$$

Fourier Transform ↓ ↓

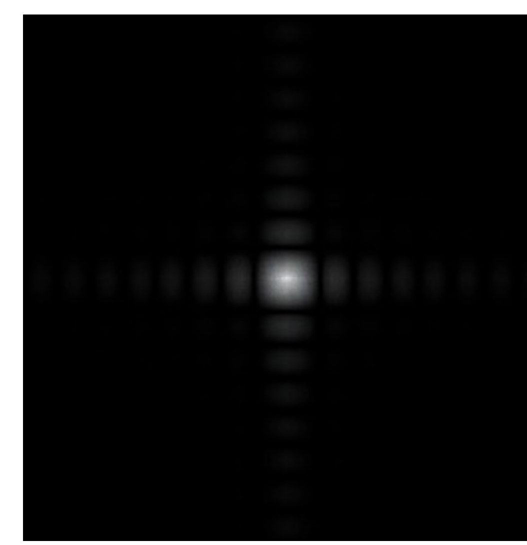
Inv. Fourier Transform ↑

Frequency Domain

x



=

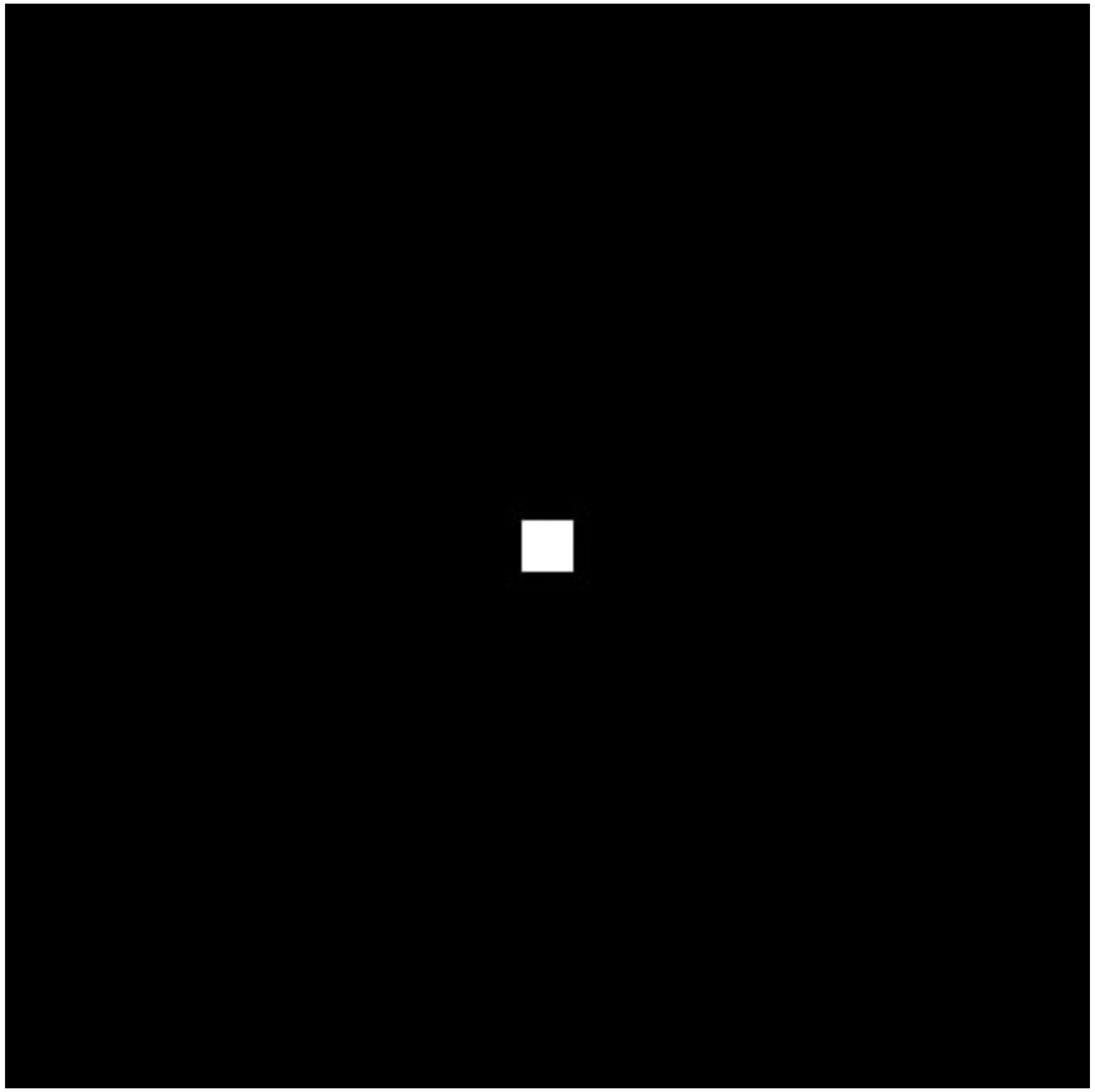


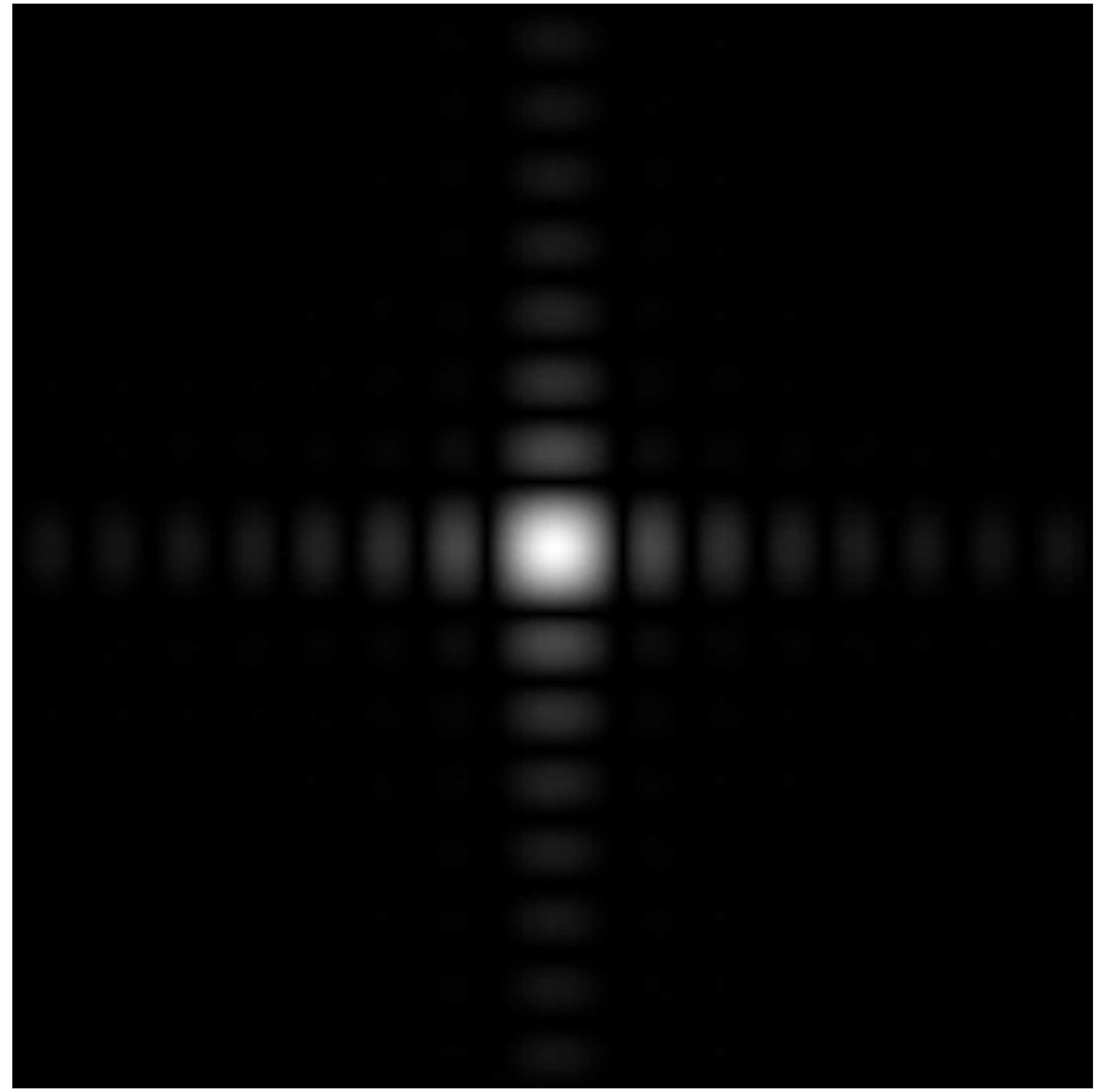
Box Filter

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

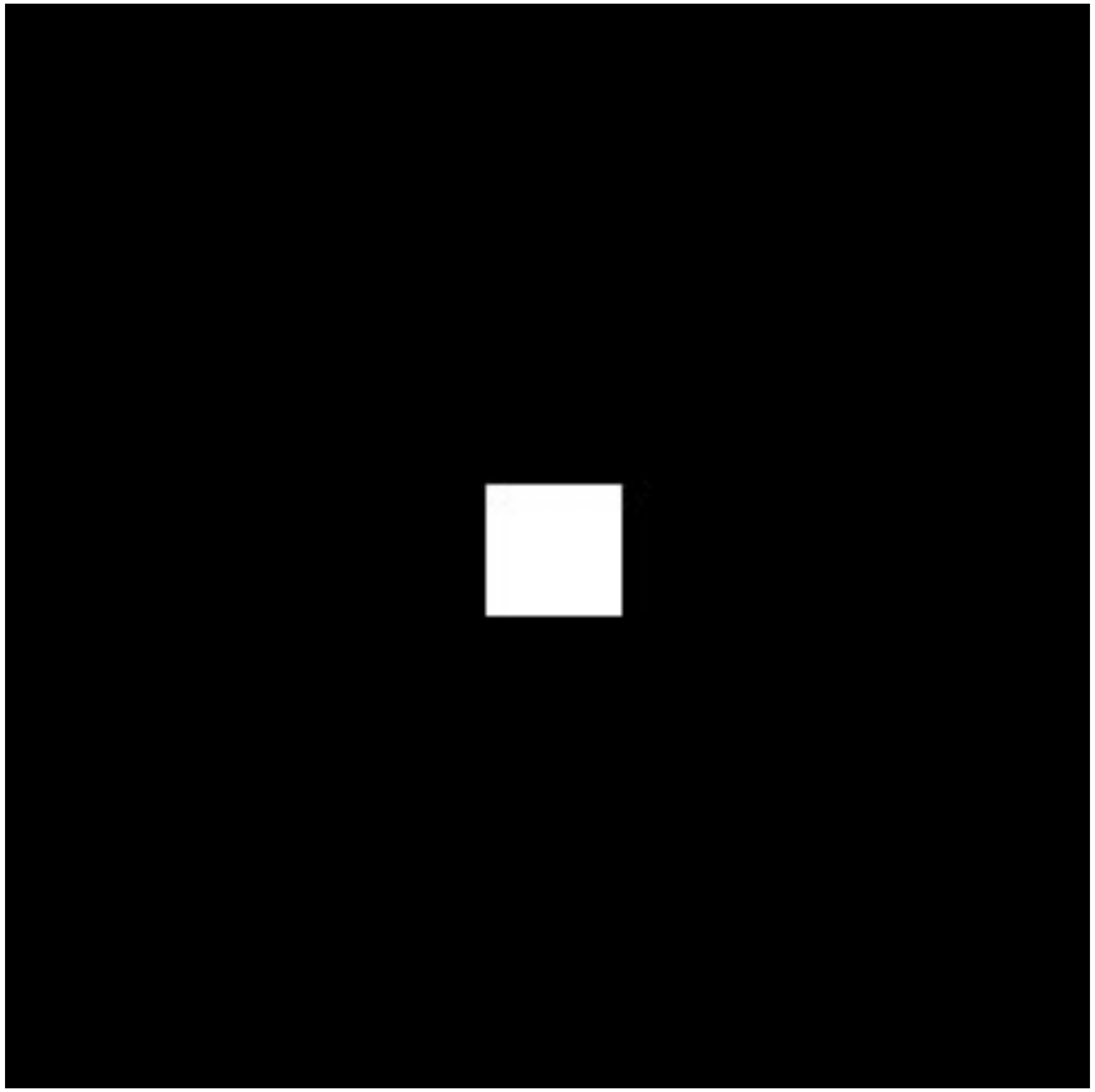
Example: 3x3 box filter

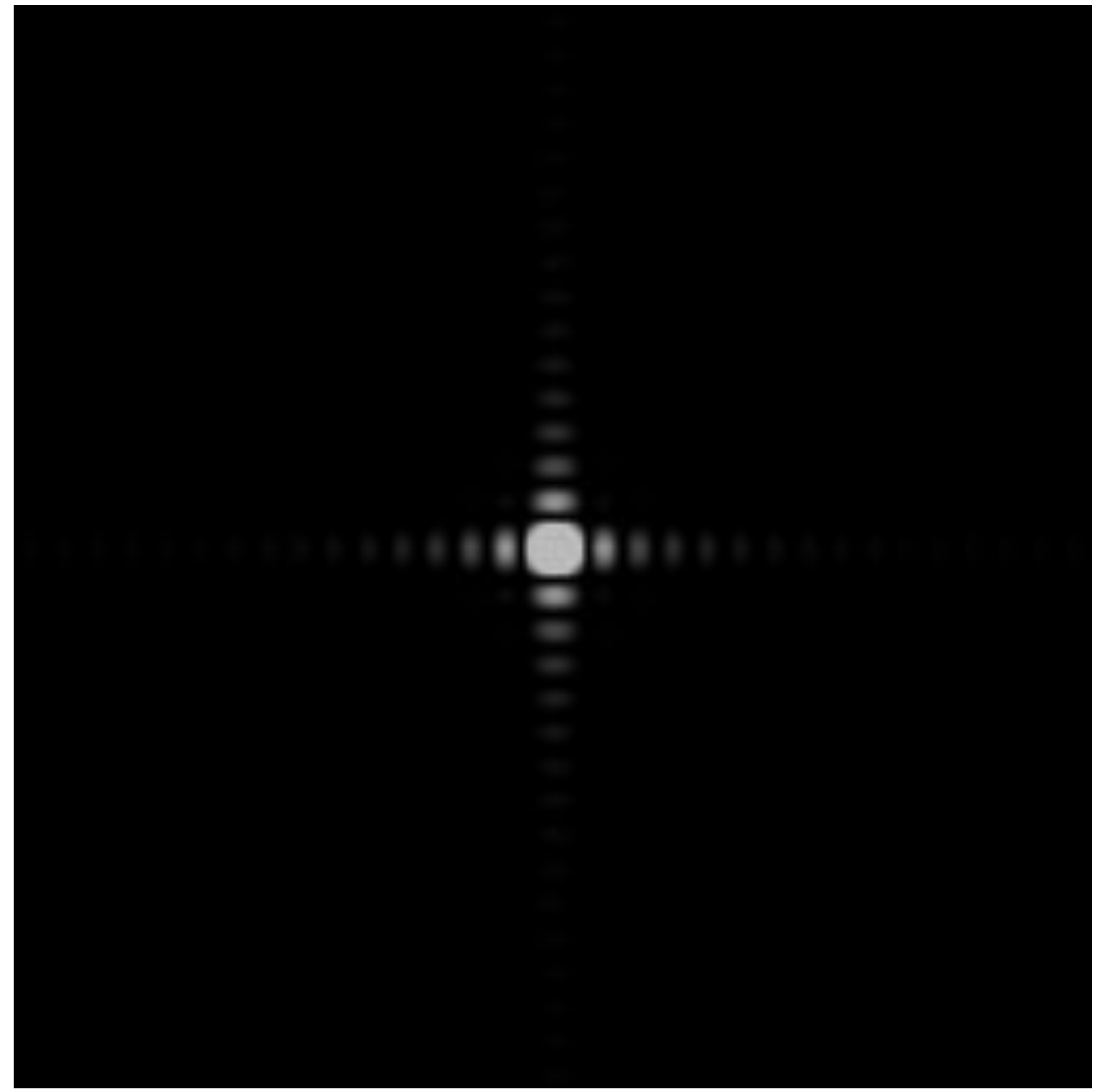
Box Function = "Low Pass" Filter





Wider Filter Kernel = Lower Frequencies



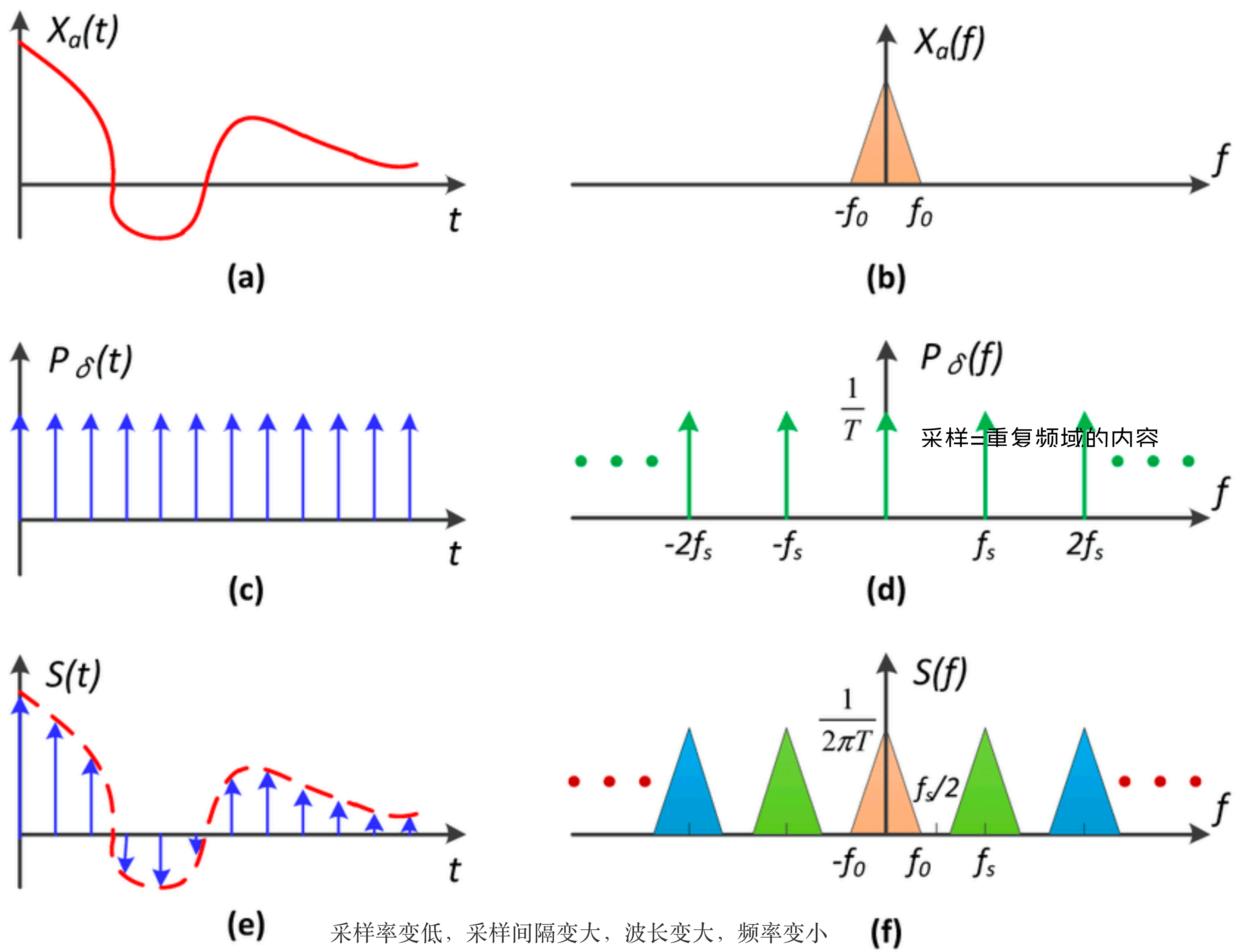


卷积核越大，保留的高频信息越少，低频信息越多，对应到频域图上，高频区域的亮度就降低，低频区域信息越多

Sampling = Repeating Frequency Contents

Sampling = Repeating Frequency Contents

空域采样=重复频域的内容



采样率变低，采样间隔变大，波长变大，频率变小

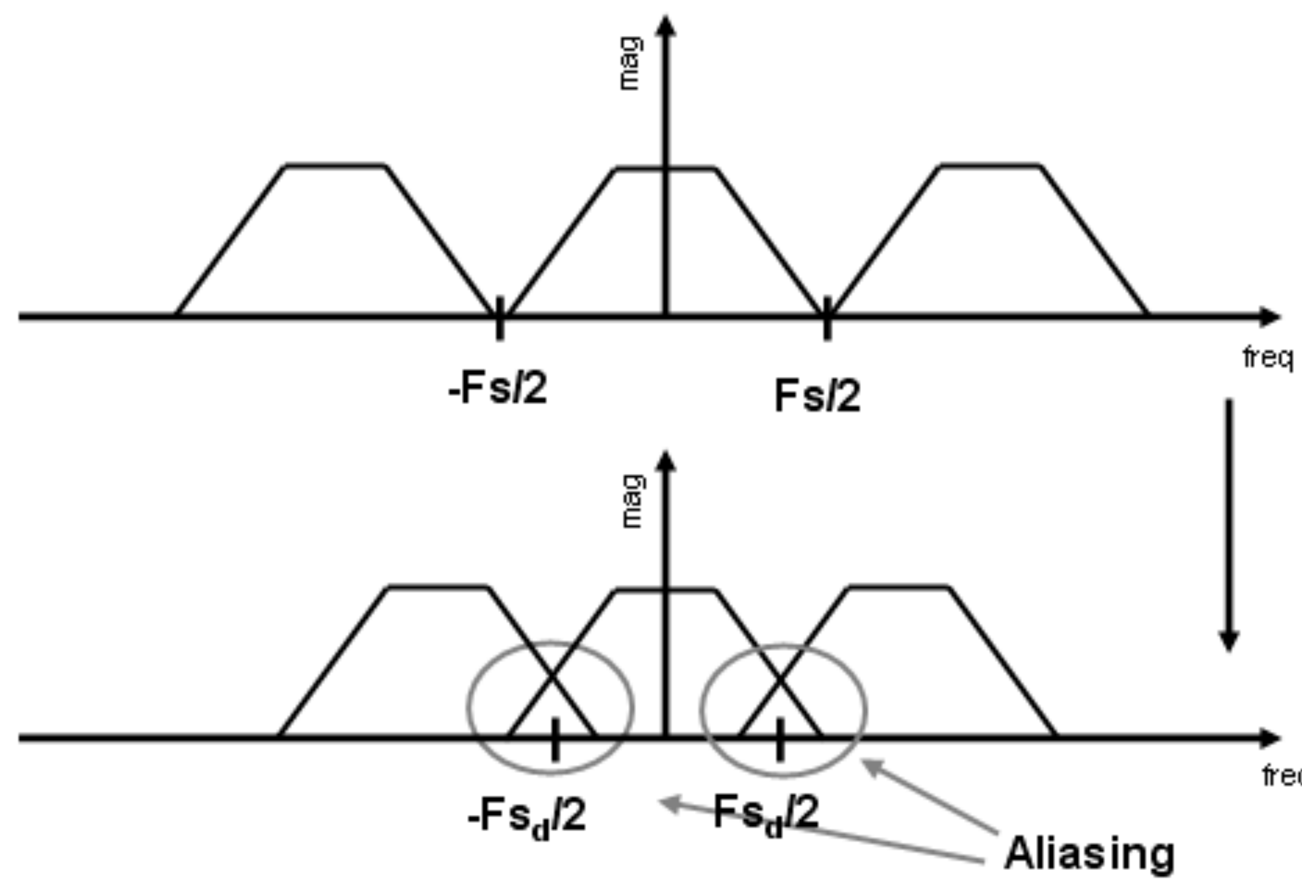
(f)

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

Aliasing = Mixed Frequency Contents

Dense sampling:

Sparse sampling:



Dense sampling稀疏采样，意味着频率Fs变小，间隔变小，就会产生混叠
Sparse sampling ，图中信号已经首尾相接，意味着当前的采样频率Fs是不发生走样的最低限值

Antialiasing

反走样技术思路

1、增加屏幕分辨率，增加采样频率(成本高)
2、在采样之前，进行模糊(/滤波)处理，(注意，先模糊处理在采样，反过来是不可行的)，模糊以后，将图像的 边界弱化了，采样的时候，该区域对应的像素值可以起到过度缓冲的效果(低通滤波降低信号最高频率，使得可以用 更低的采样频率完成采样)

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

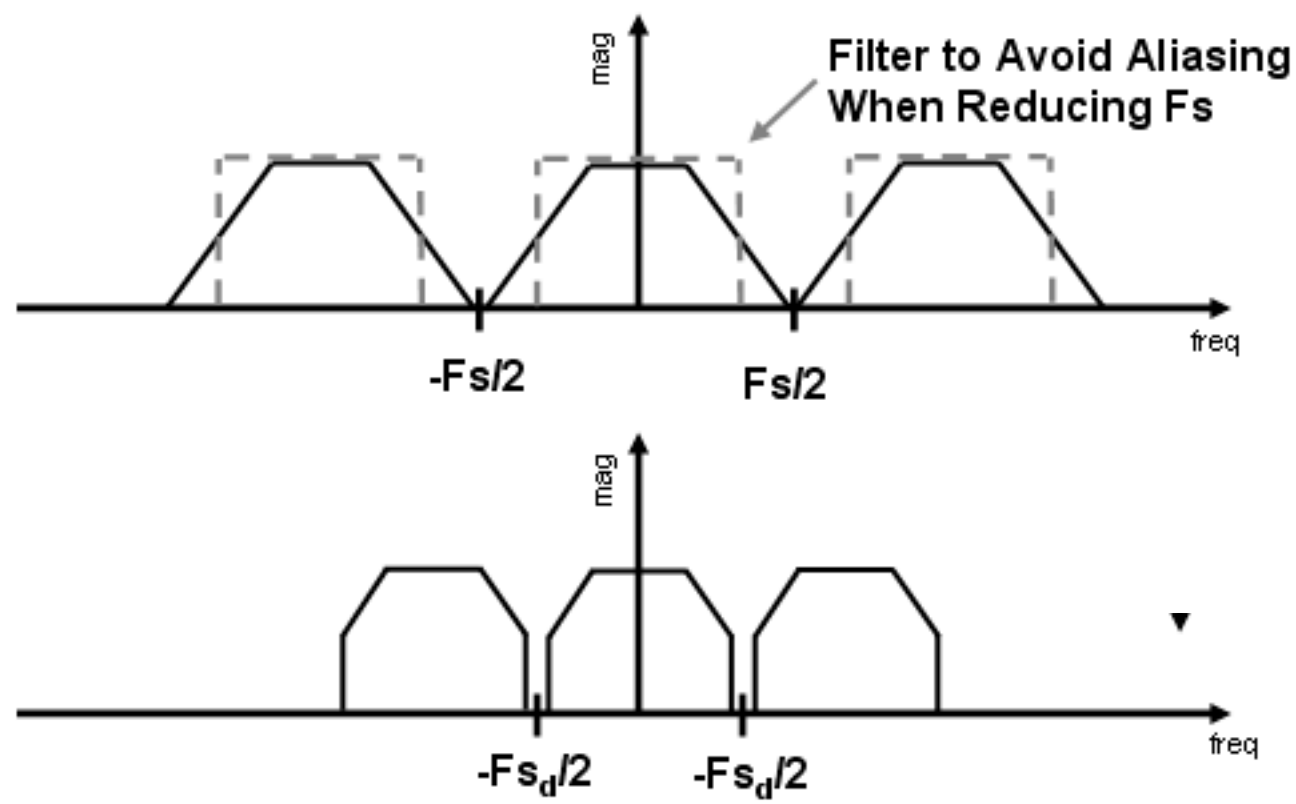
- Making Fourier contents “narrower” before repeating
- i.e. Filtering out high frequencies before sampling

Antialiasing = Limiting, then repeating

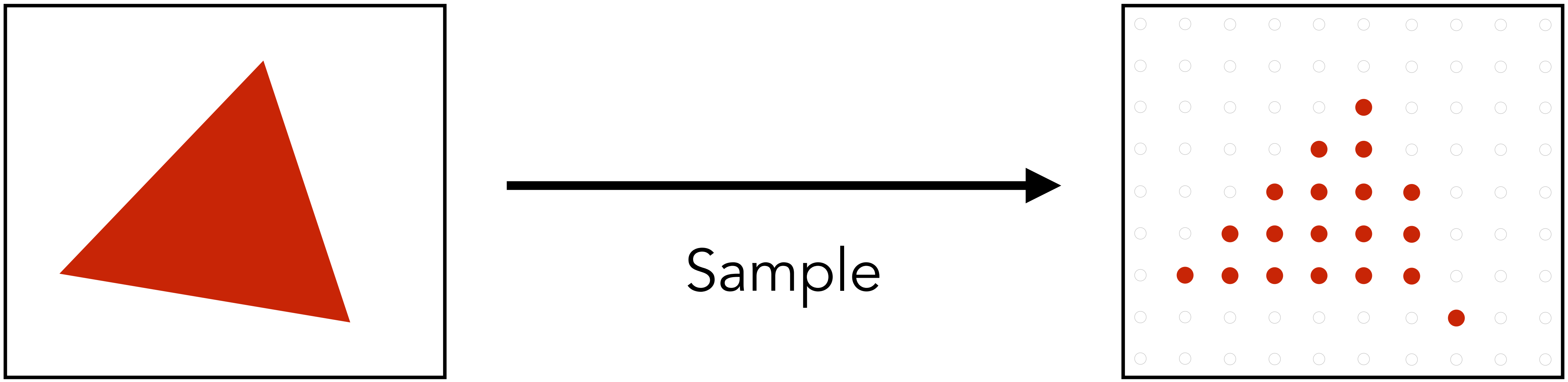
先模糊，相当于先缩短图像的频域跨度，减少采用得到频域交叠可能性
先采样后模糊之所以不可行，就是因为波形重叠的情况下截断依然会有重叠

Filtering

Then sparse sampling

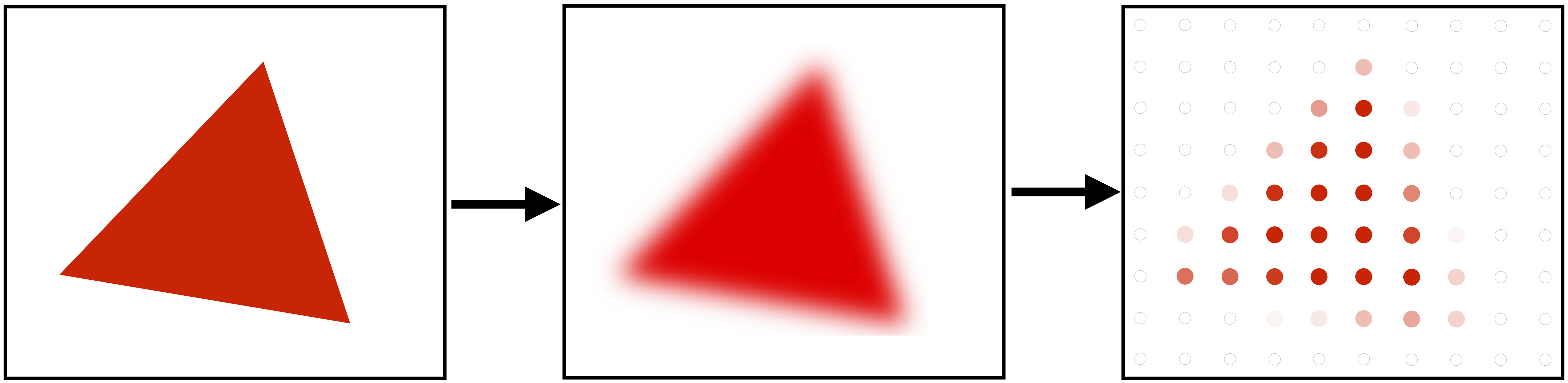


Regular Sampling



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

Antialiased Sampling



Pre-Filter

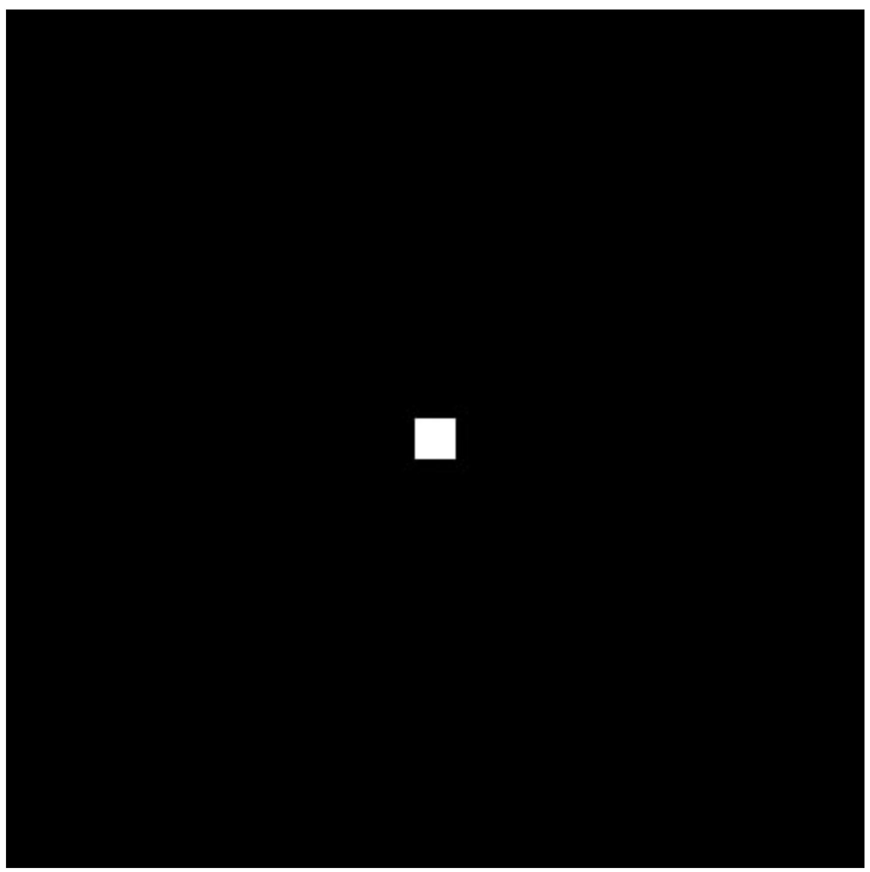
(remove frequencies above Nyquist)

Sample

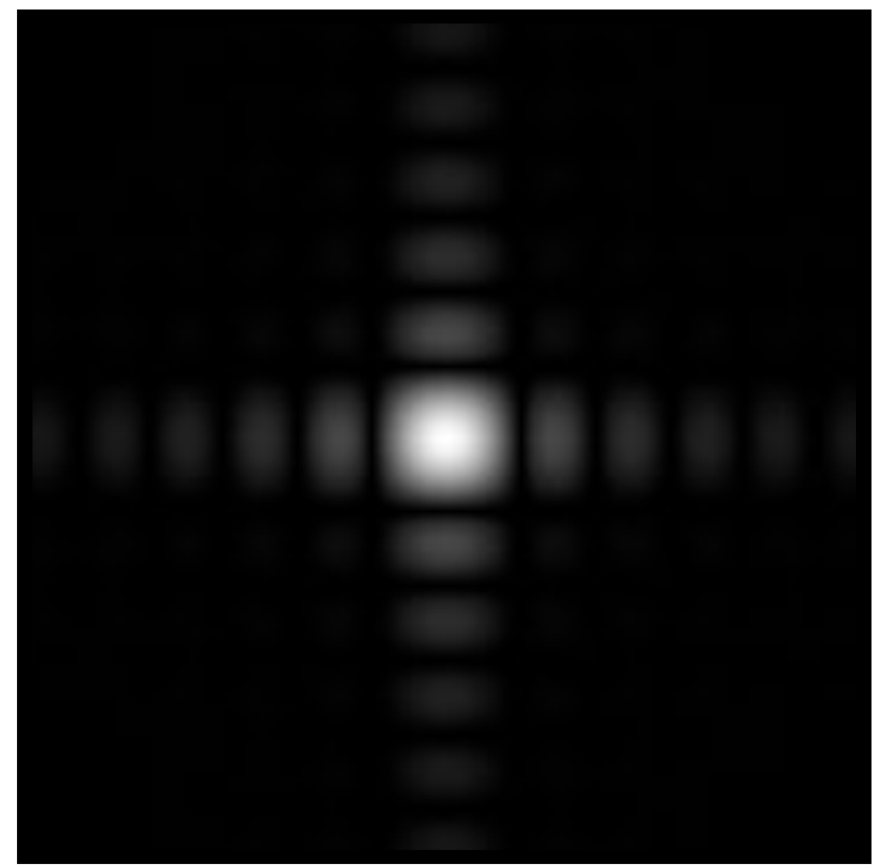
Note antialiased edges in rasterized triangle where pixel values take intermediate values

A Practical Pre-Filter

A 1 pixel-width box filter (low pass, blurring)



Spatial Domain



Frequency Domain

Antialiasing By Averaging Values in Pixel Area

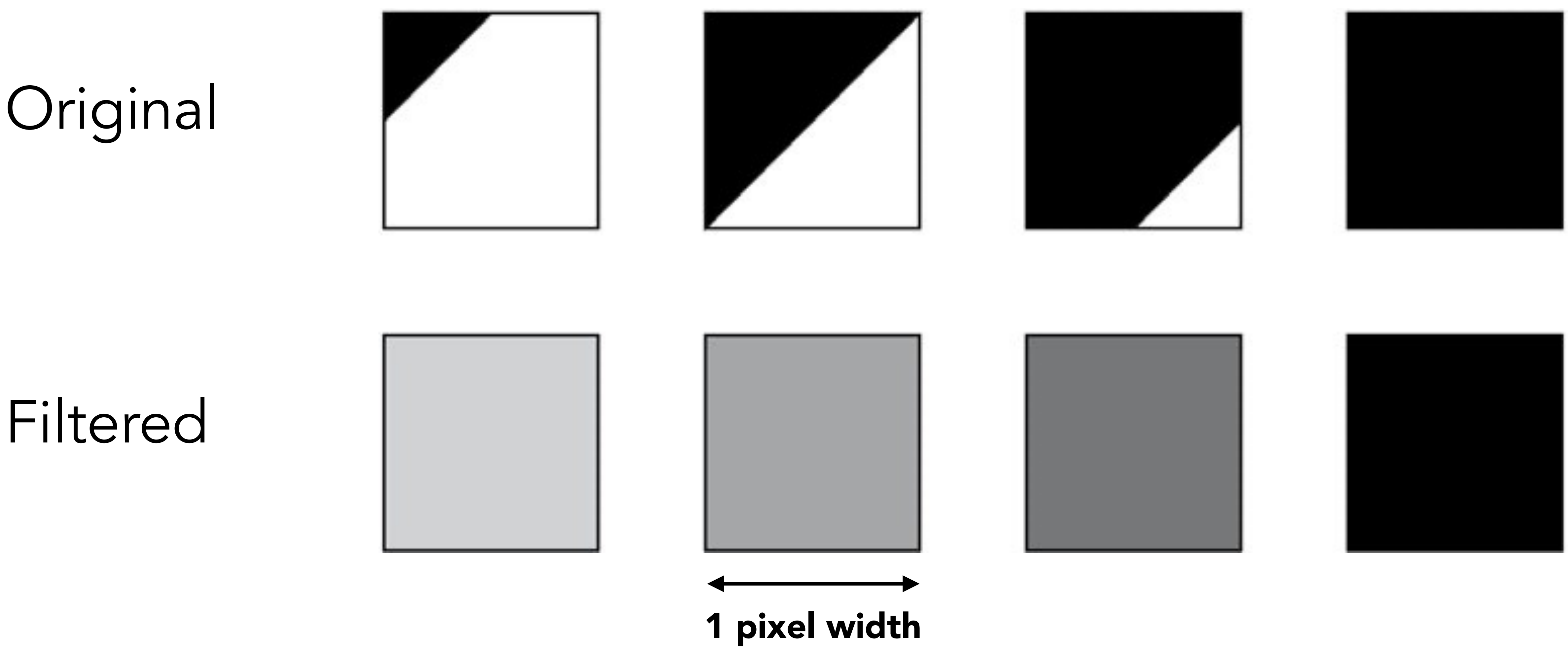
Solution:

- **Convolve** f(x,y) by a 1-pixel box-blur
  - Recall: convolving = filtering = averaging
- **Then sample** at every pixel's center

Antialiasing by Computing Average Pixel Value

In rasterizing one triangle, the average value inside a pixel area of f(x,y) = inside(triangle,x,y) is equal to the area of the pixel covered by the triangle.

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，相比之前非黑即白的锯齿

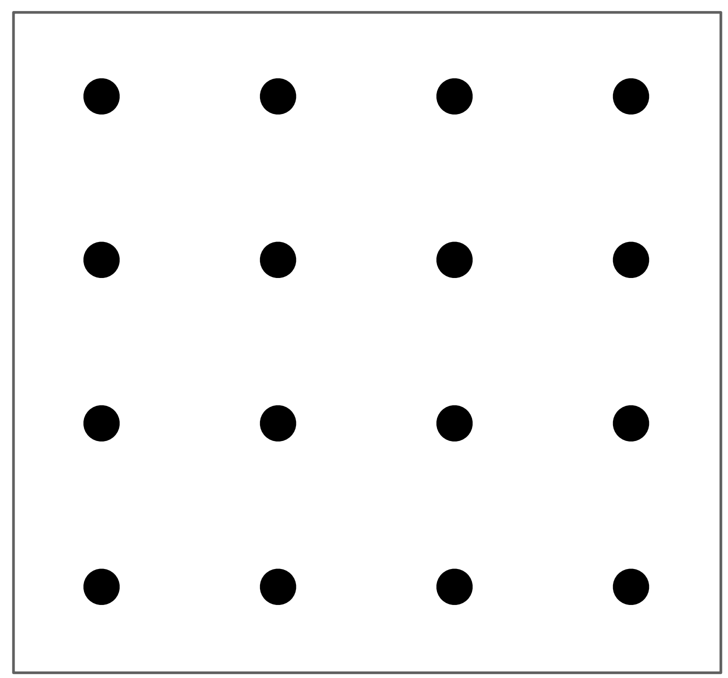


Antialiasing By Supersampling (MSAA)

每个像素多次采样，求平均，像素的颜色值为负责的区域内取样多次颜色值的平均

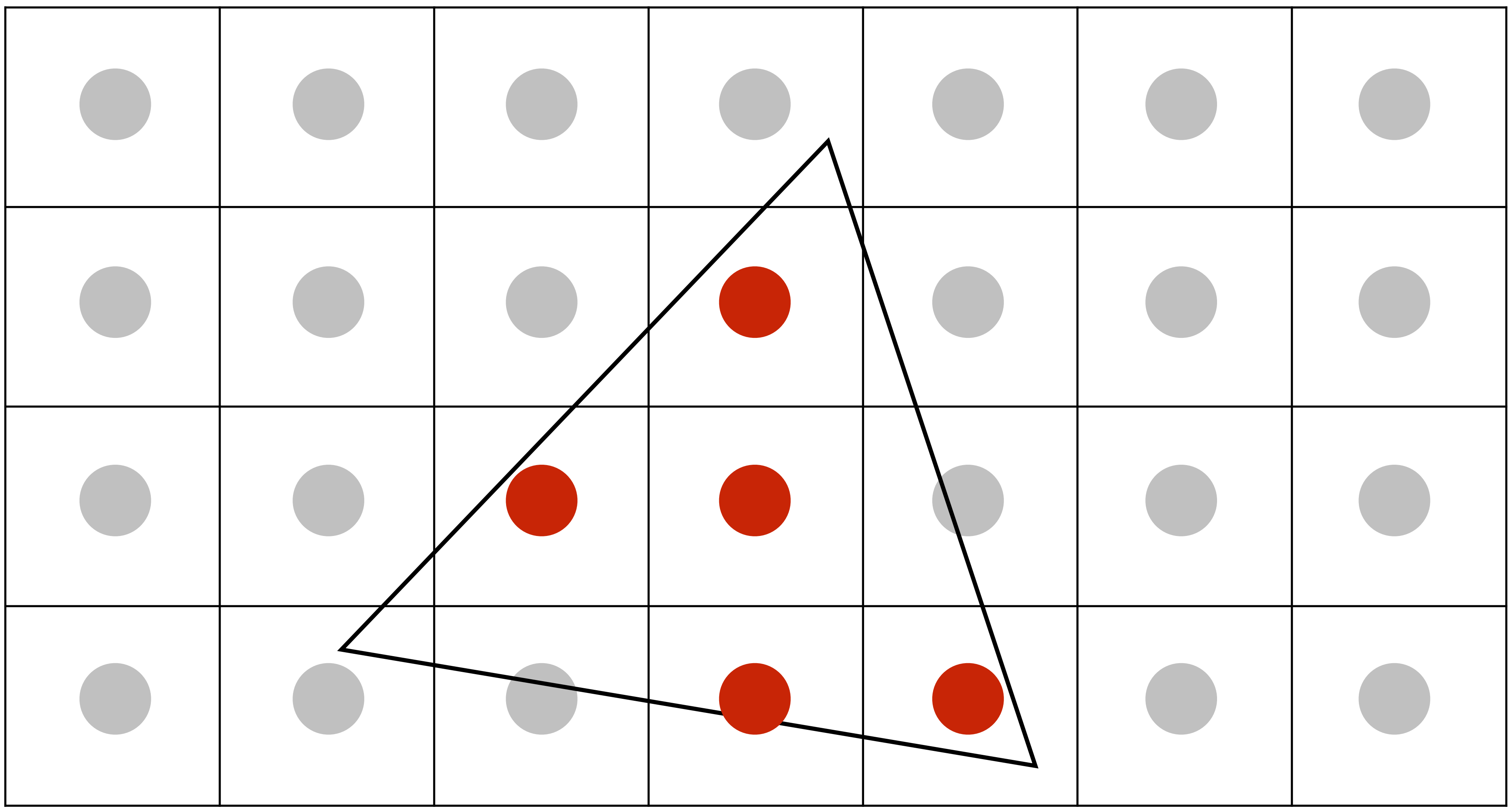
Supersampling

Approximate the effect of the 1-pixel box filter by sampling multiple locations within a pixel and averaging their values:



4x4 supersampling

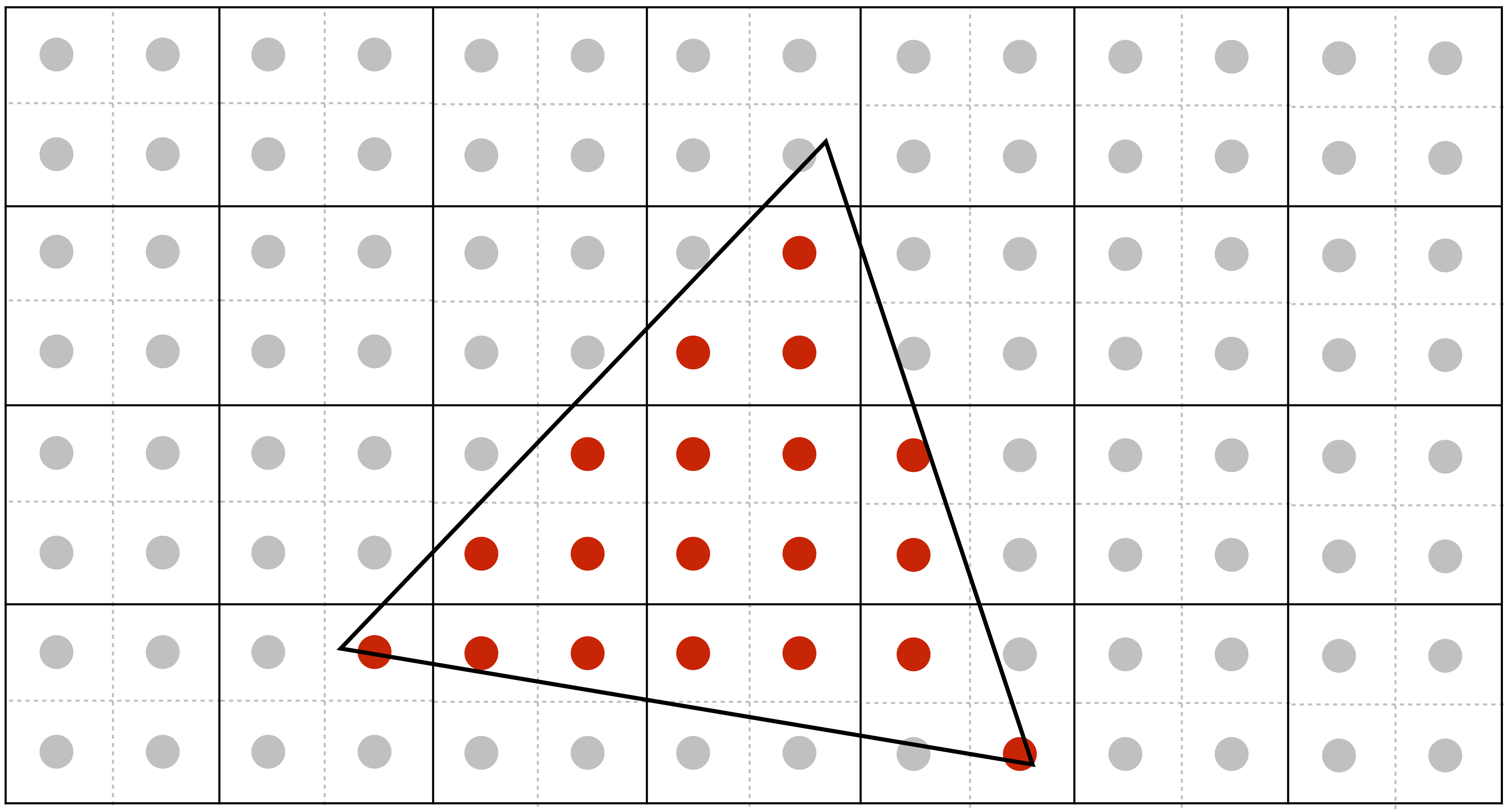
Point Sampling: One Sample Per Pixel



Supersampling: Step 1

Take NxN samples in each pixel.

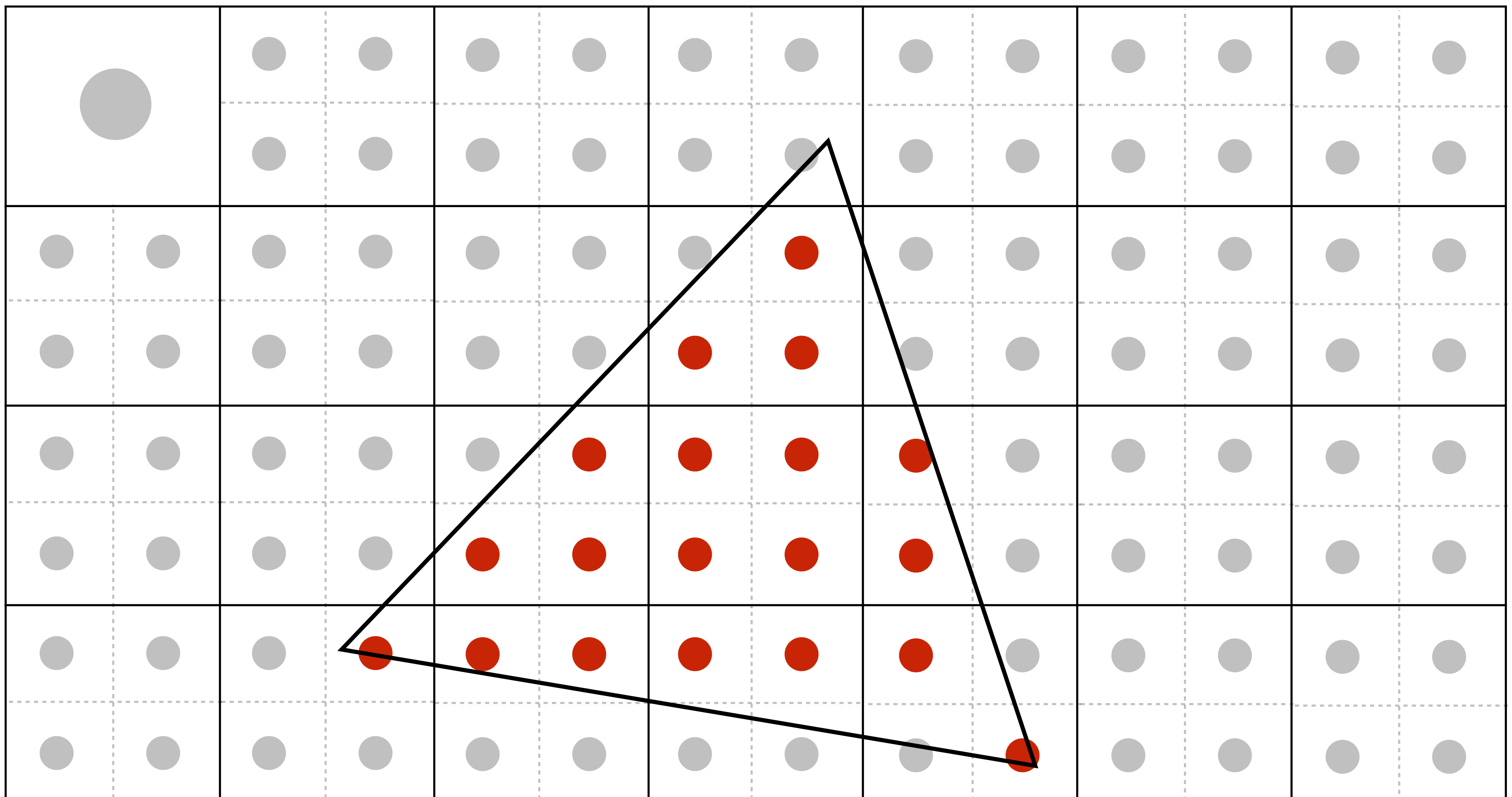
首先，在每个像素点内，取N*N个小的“像素”(采样点)，对每个小采样点判断它们是不是在图 形内



2x2 supersampling

Supersampling: Step 2

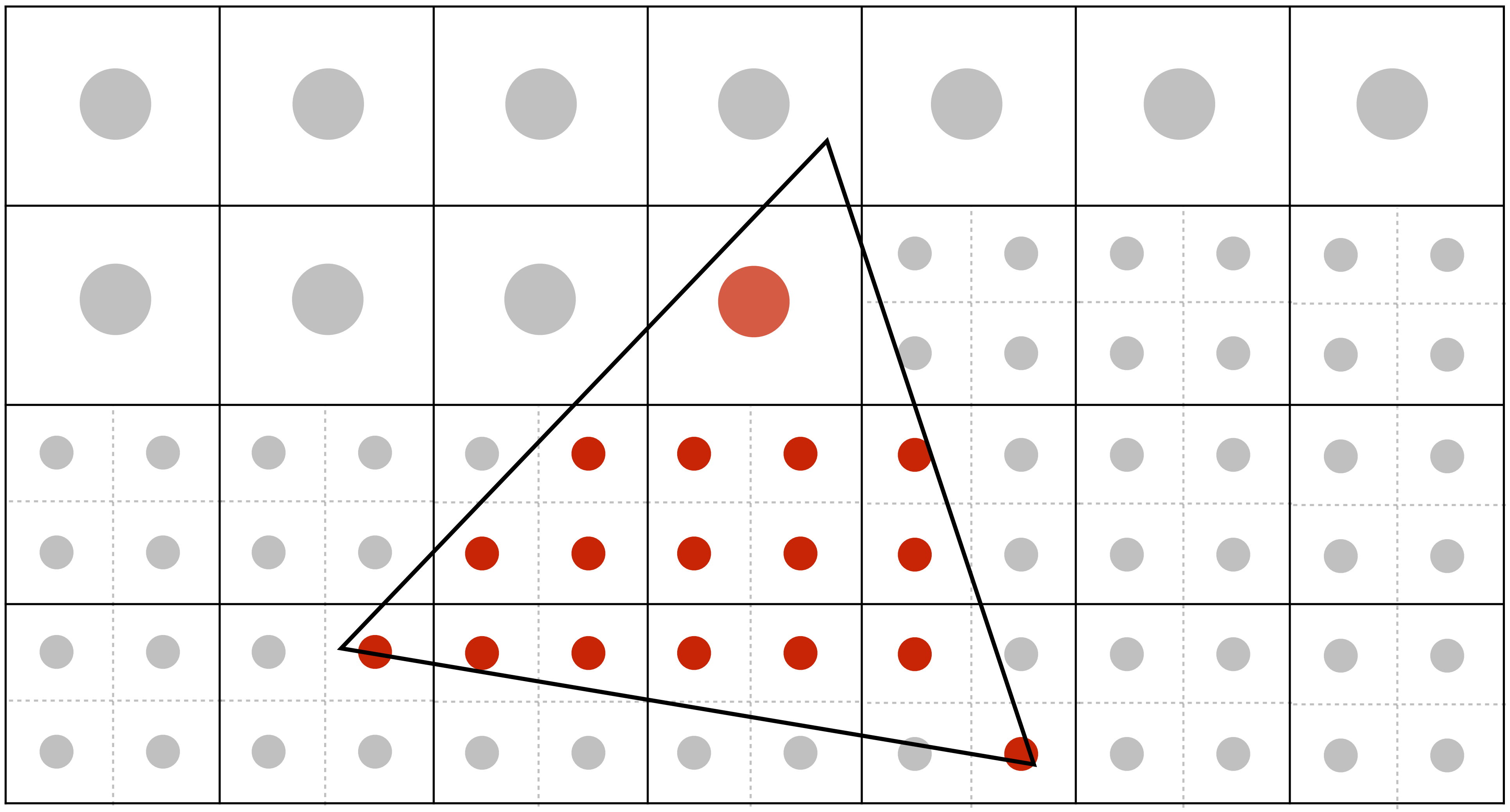
Average the NxN samples "inside" each pixel.



Averaging down

Supersampling: Step 2

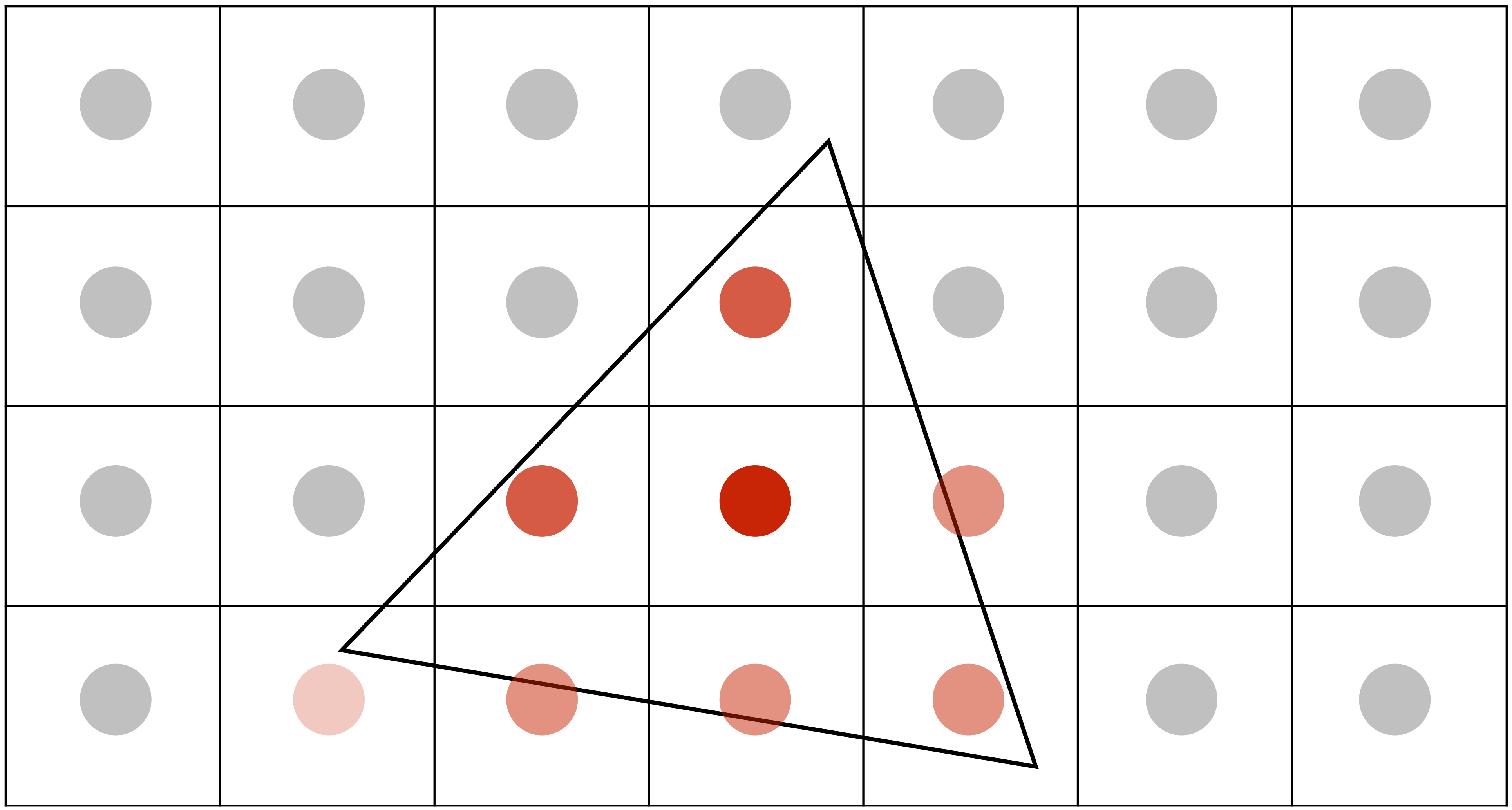
Average the NxN samples “inside” each pixel.



Averaging down

Supersampling: Step 2

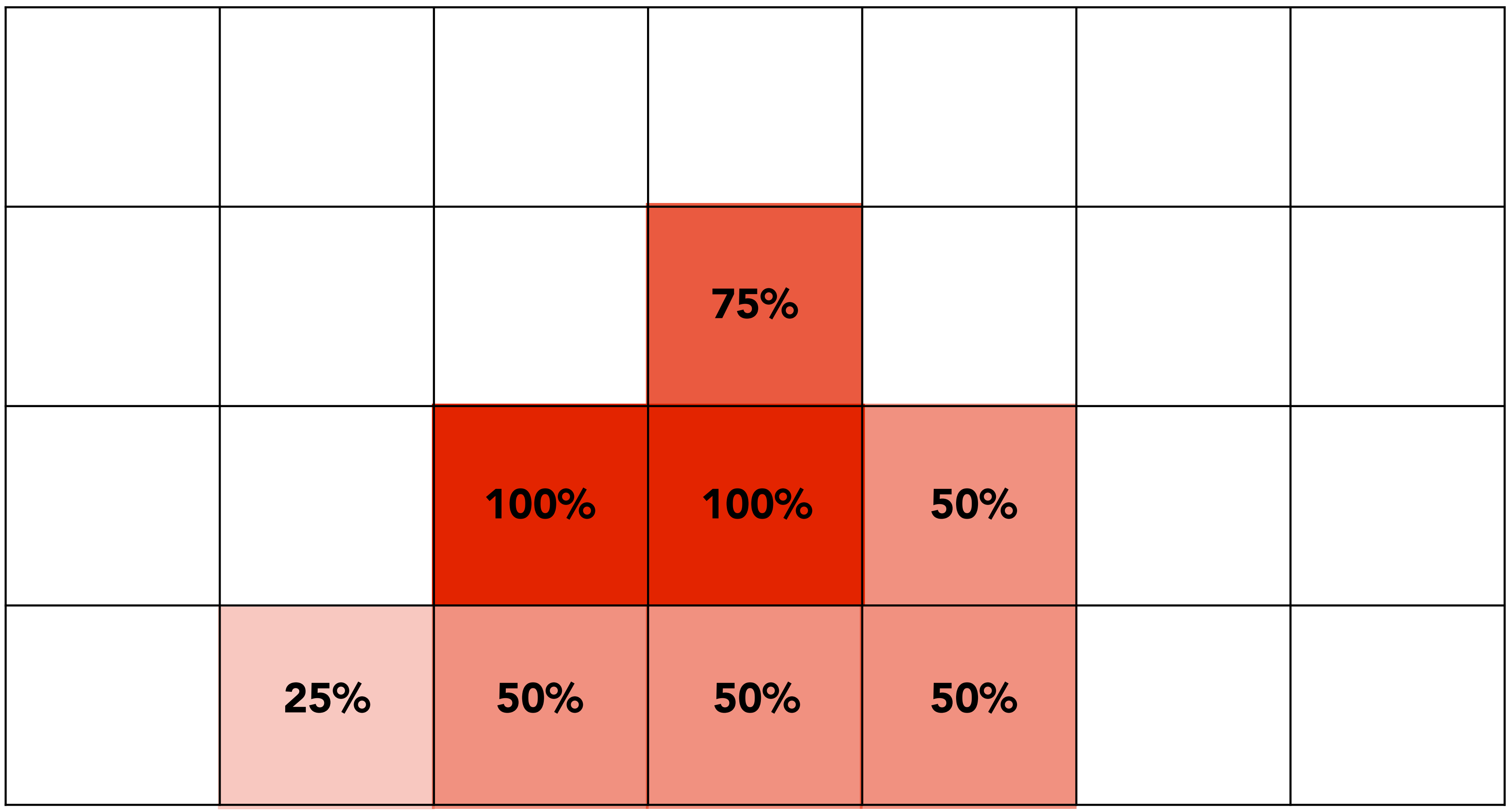
Average the NxN samples “inside” each pixel.



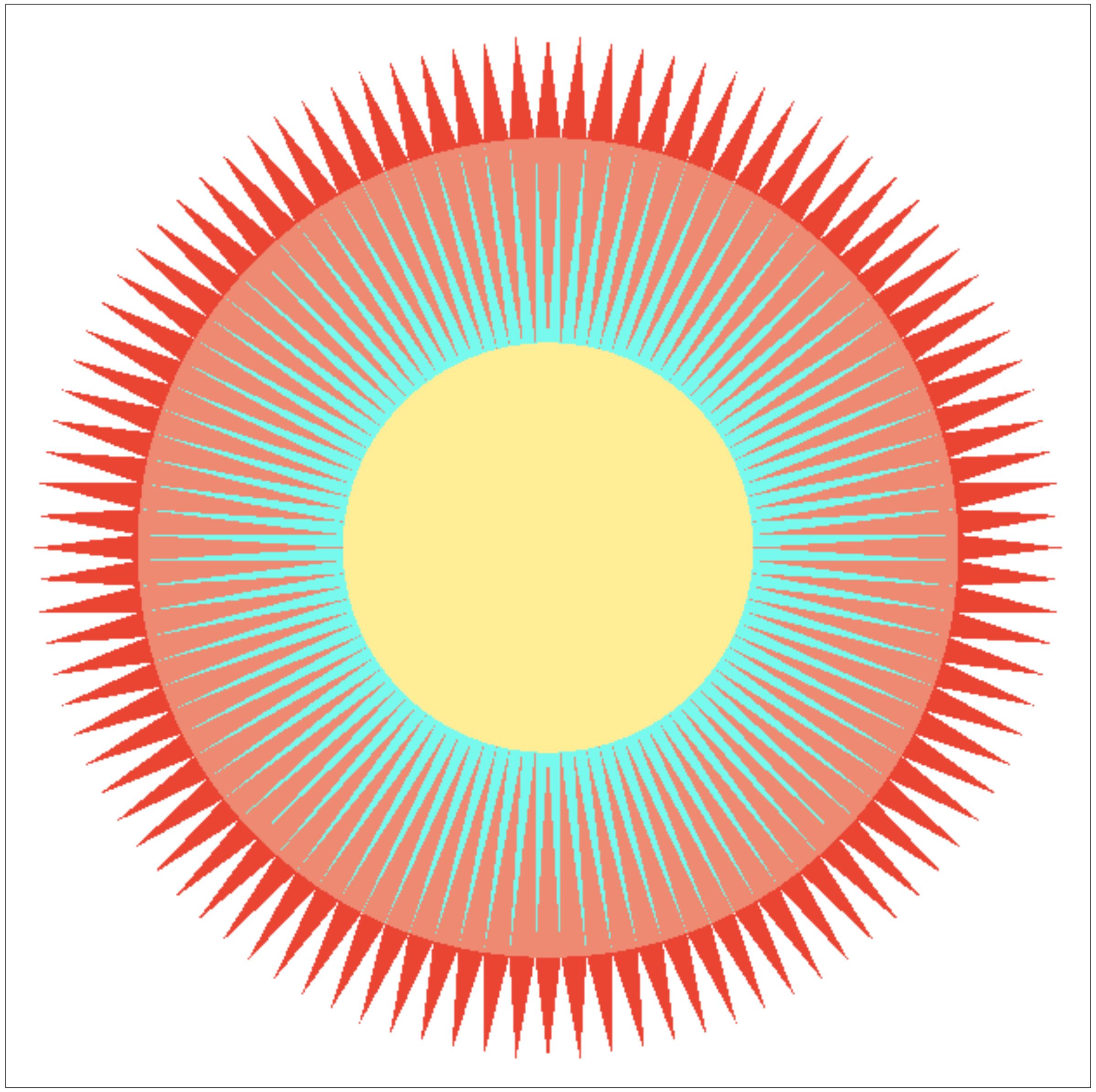
Supersampling: Result

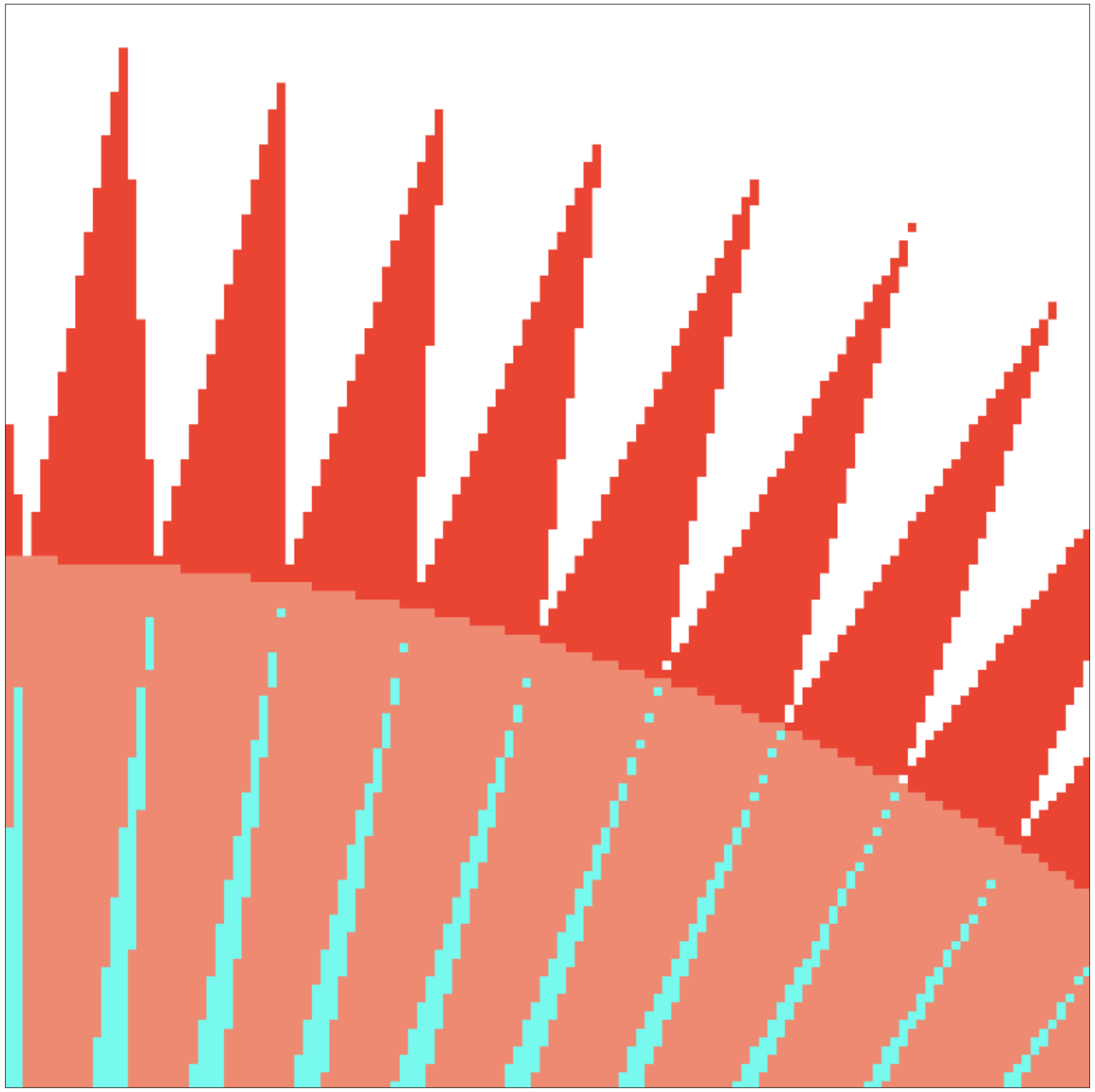
This is the corresponding signal emitted by the display

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 ， 至于比例值是像素的面积占比

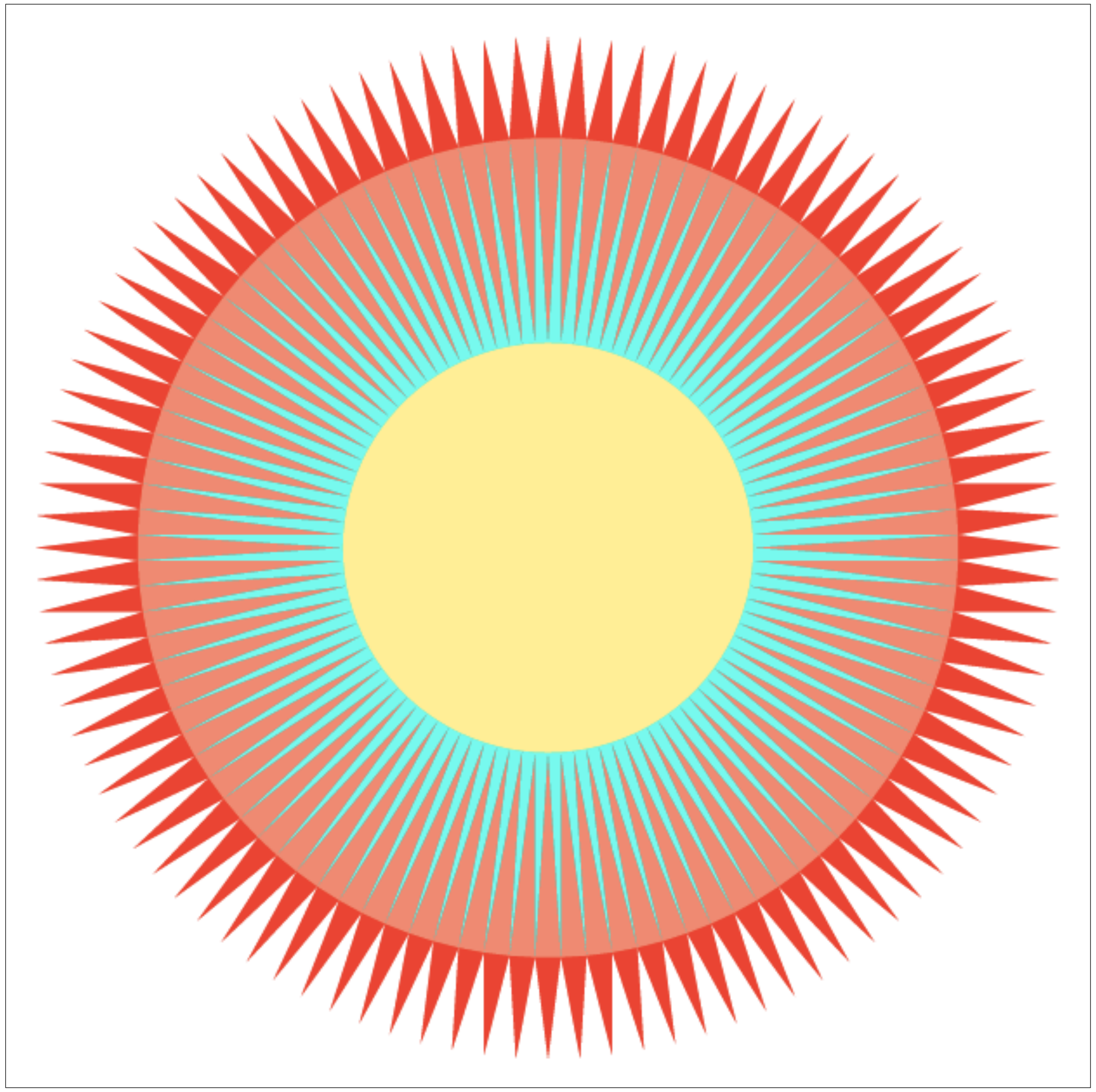


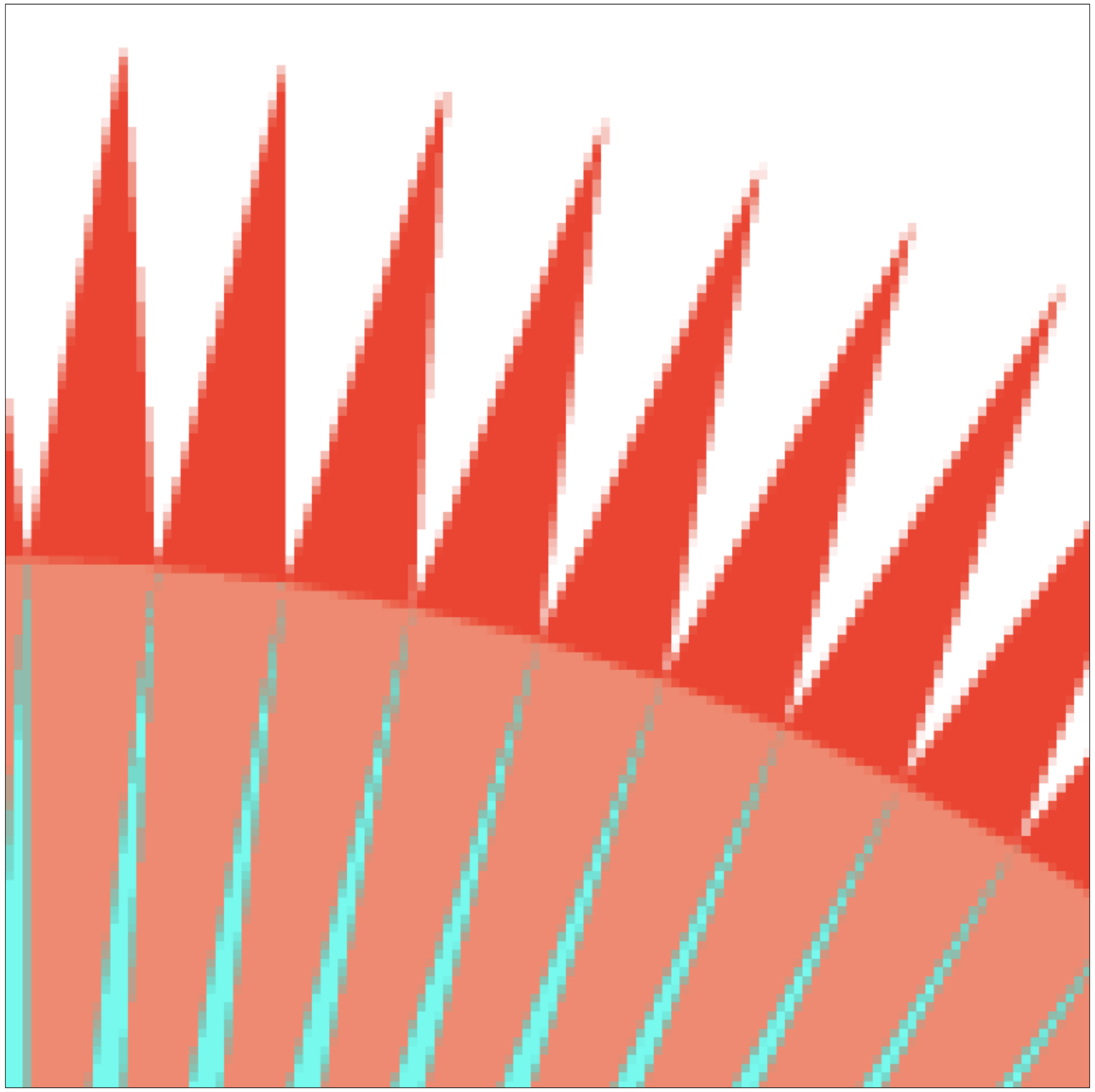
Point Sampling





4x4 Supersampling





Antialiasing Today

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

No free lunch!

- What's the cost of MSAA?

Milestones (personal idea)

- FXAA (Fast Approximate AA)

FXAA (Fast Approximate AA):先获得有锯齿的图，再后处理去除锯齿(很快)
- 找到边界，换成没有锯齿的边界，(图像匹配)非常快
- 方法和采样无关，采样虽然有误，但是这种方法可以弥补

- TAA (Temporal AA)

TAA (Tem'poral AA) :时序信息，借助前面帧的信息
最近刚刚兴起 静态场景，相邻两帧同一像素用不同的位置来sample 把MSAA的Sampling分布在时间上

Super resolution / super sampling

- From low resolution to high resolution
- Essentially still "not enough samples" problem
- DLSS (Deep Learning Super Sampling)

低分辨率显示器还原高分辨率图片，归根结底依旧是"样本不足"，解决方案举例:DLSS (Deep Learning Super Sampling)

Thank you!