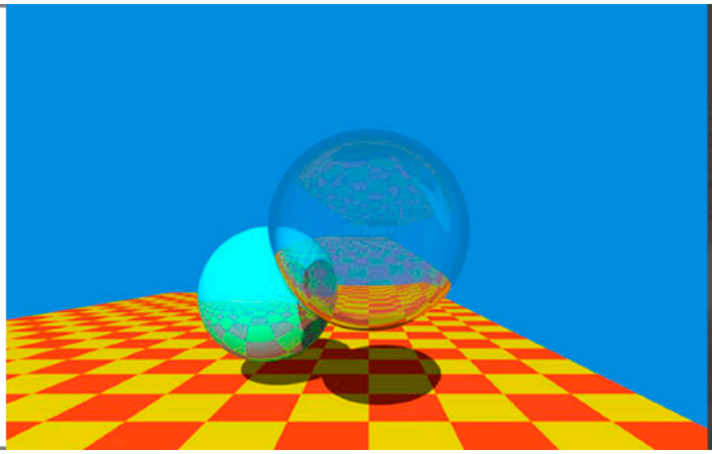
Introduction to Computer Graphics

GAMES101, Lingqi Yan, UC Santa Barbara

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

反走样 深度缓冲






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

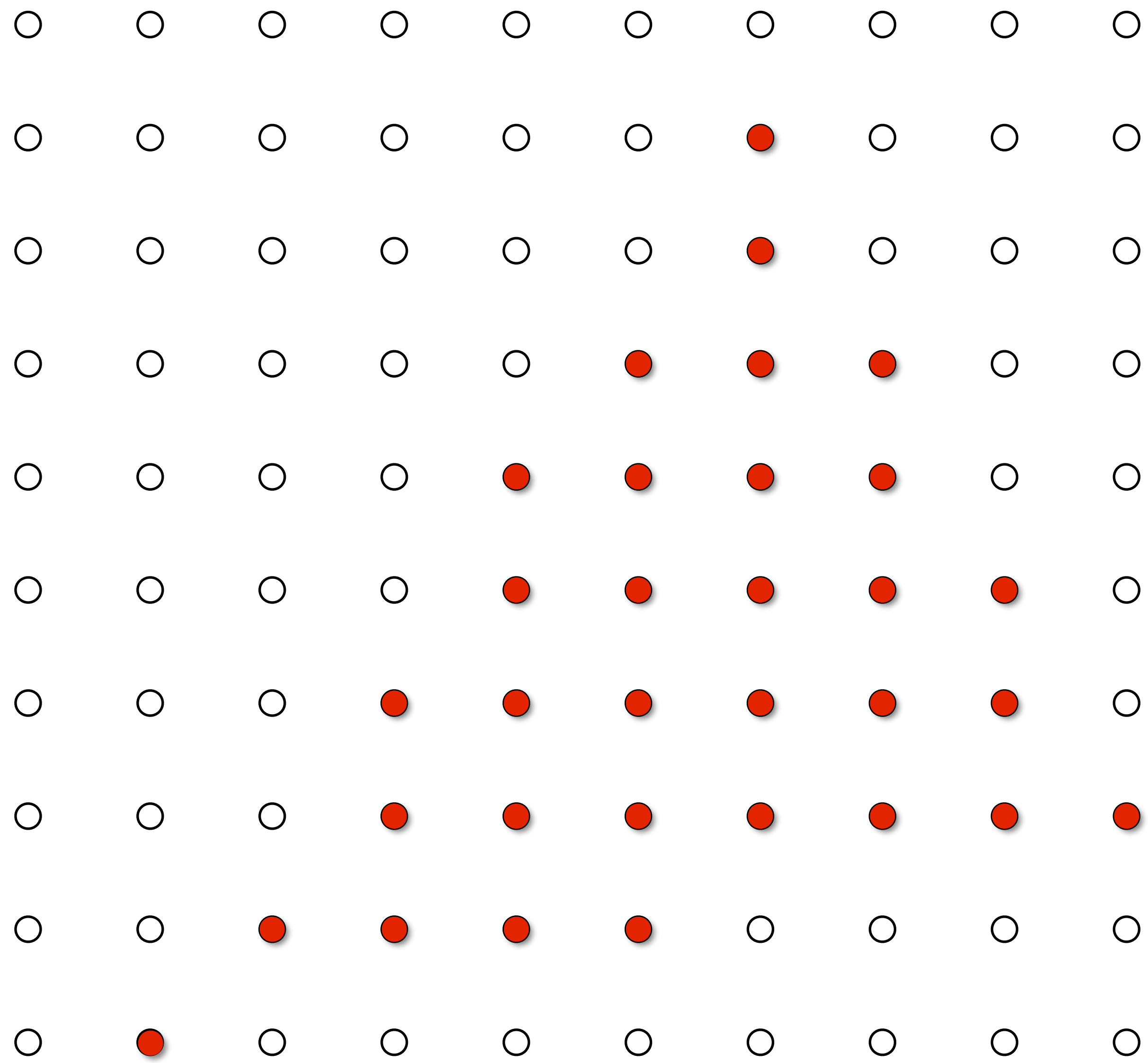
Last Lectures

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

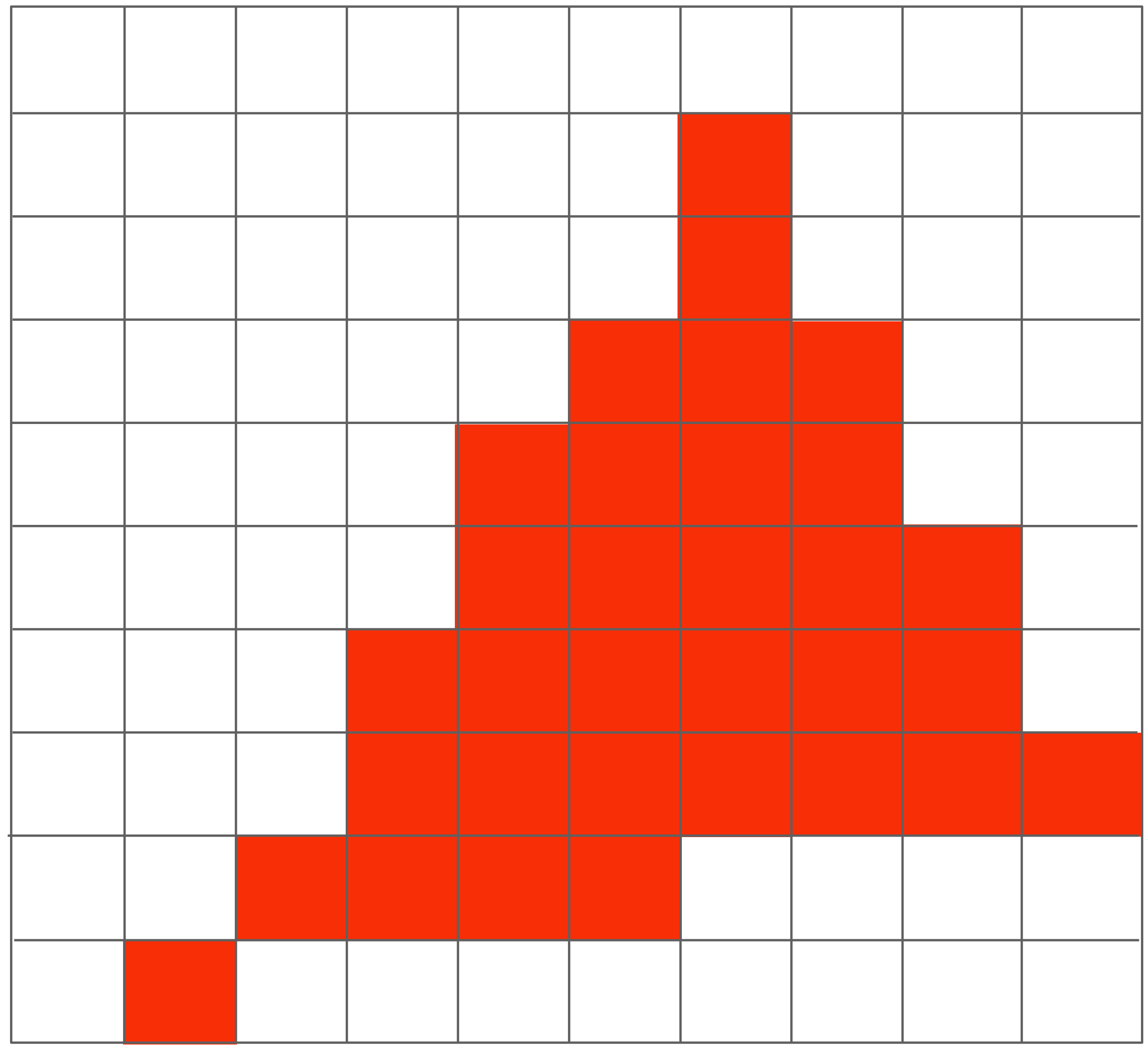
Today

- Antialiasing
  - Sampling theory
  - Antialiasing in practice
- Visibility / occlusion 遮挡与可见性
  - Z-buffering

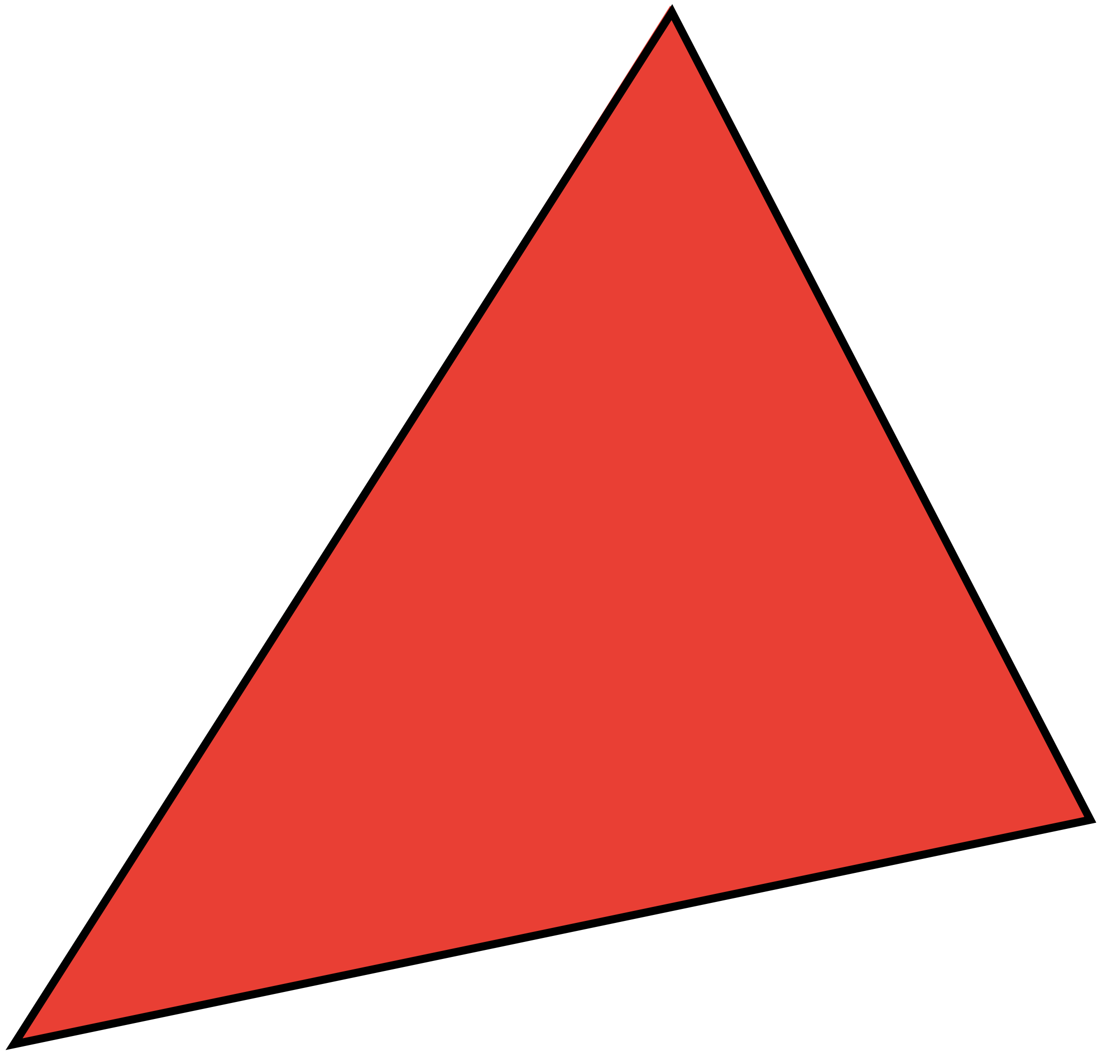
Recap: Testing in/out △ at pixels' centers



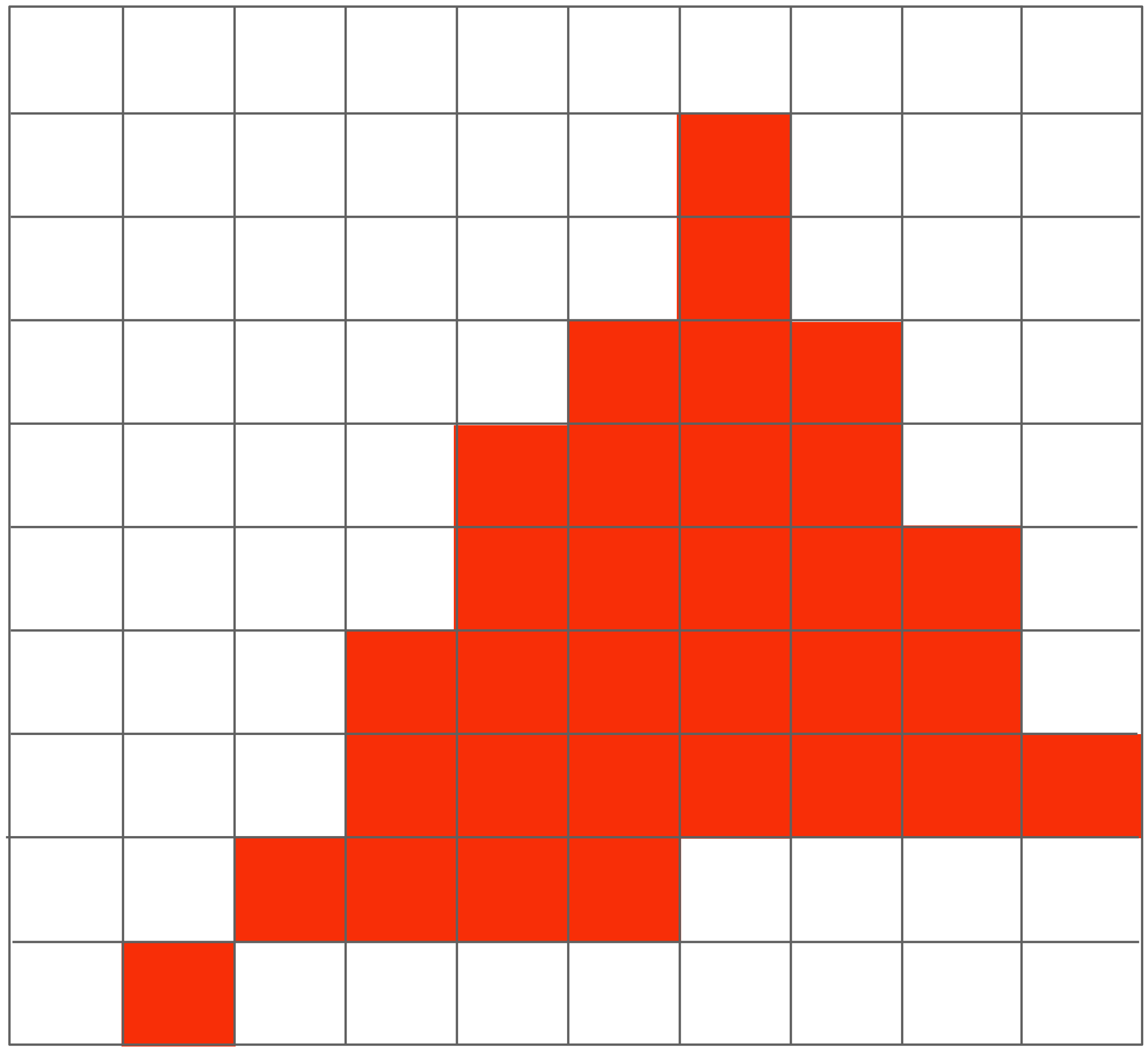
Pixels are uniformly-colored squares



Compare: The Continuous Triangle Function

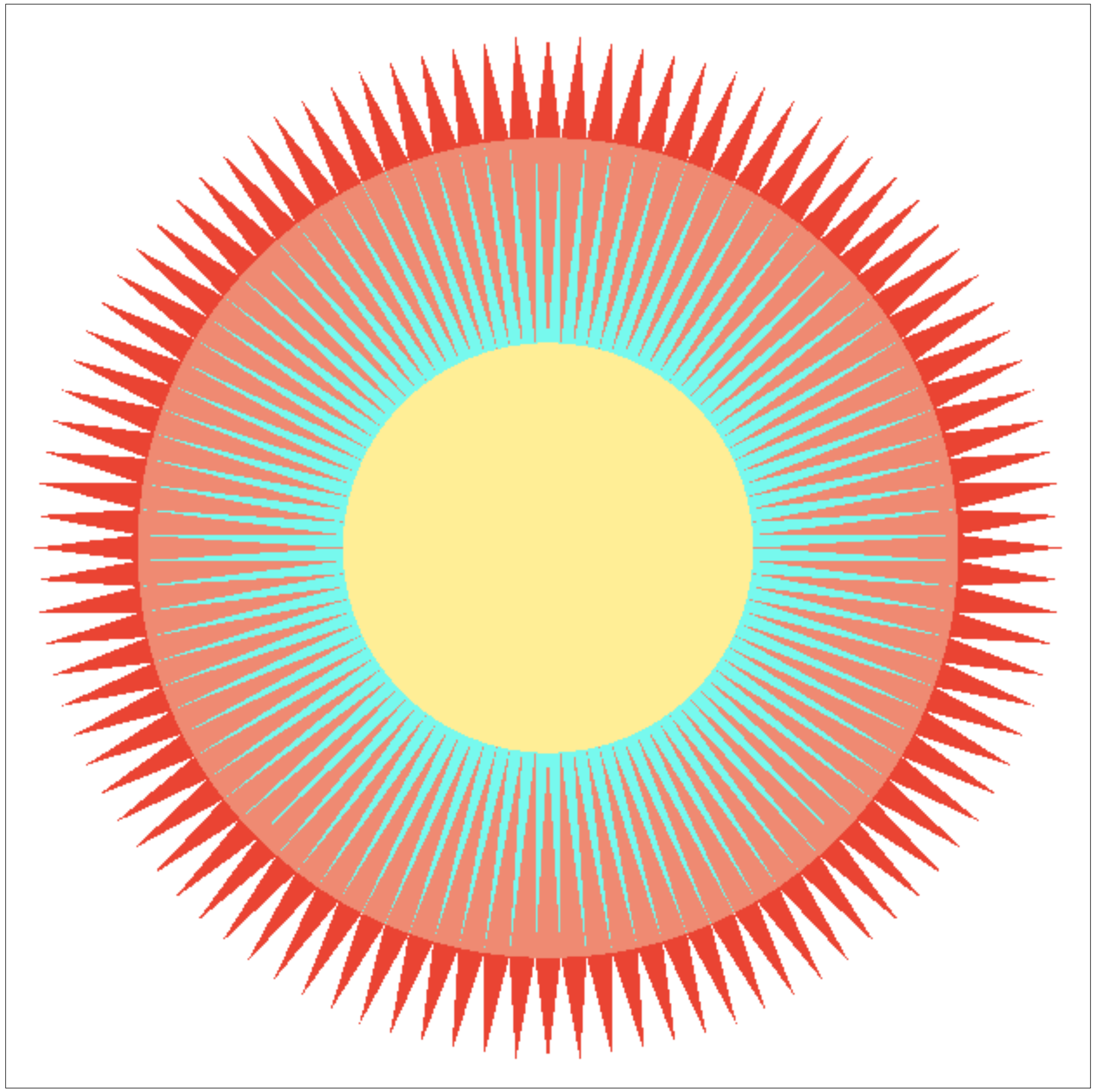


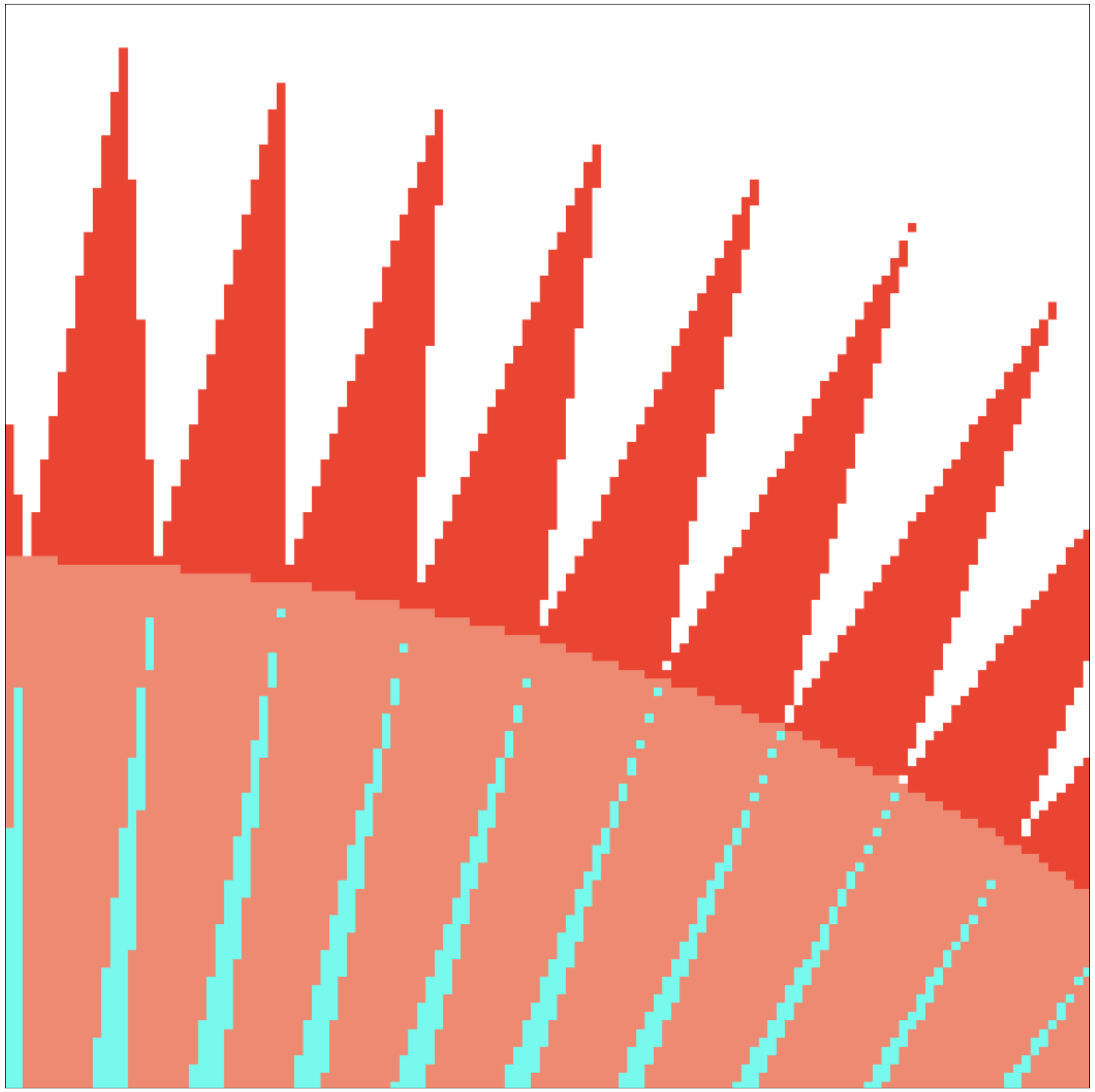
What's Wrong With This Picture?



锯齿：学名“走样” Jaggies!

Aliasing



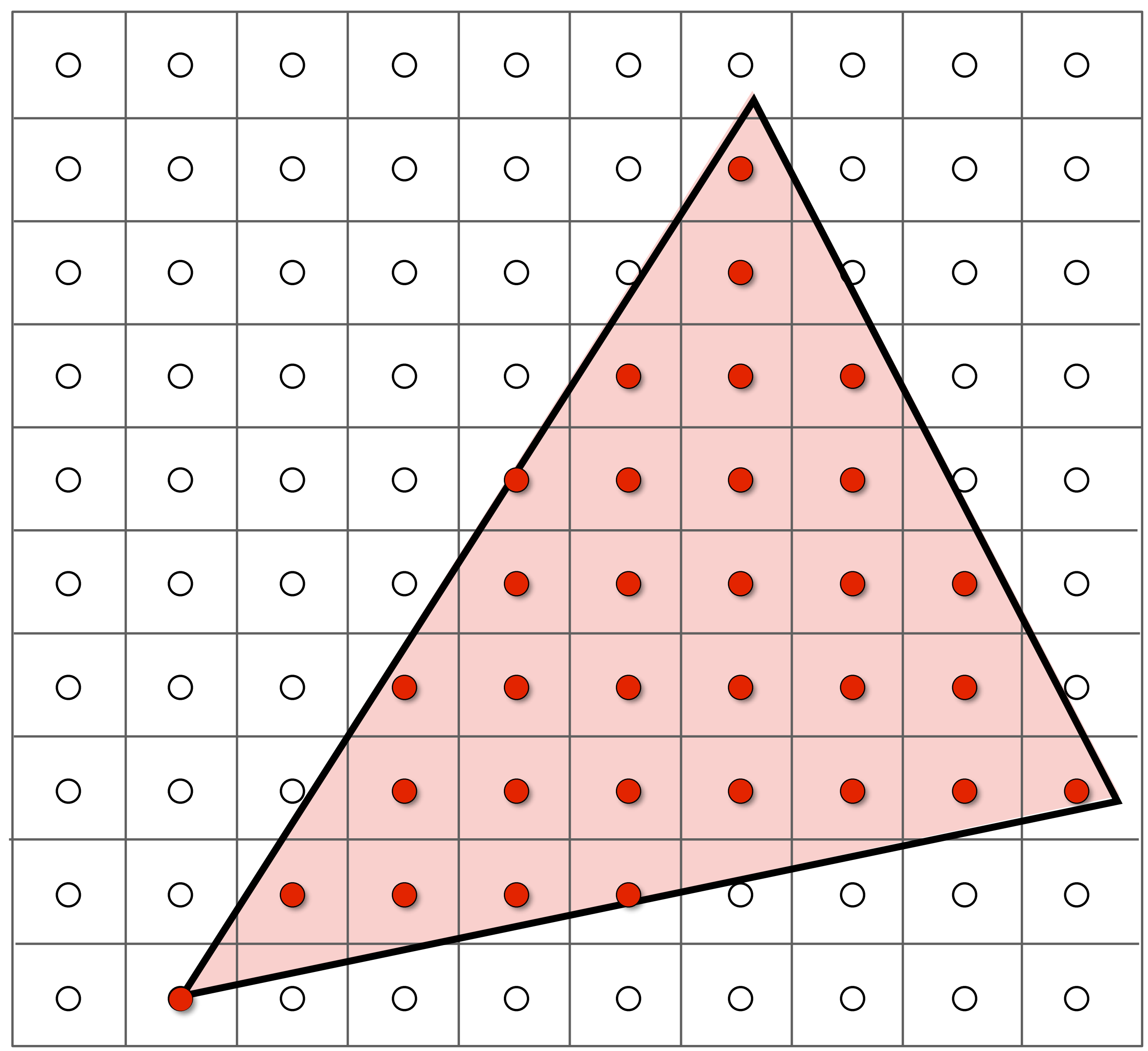


Is this the best we can do?

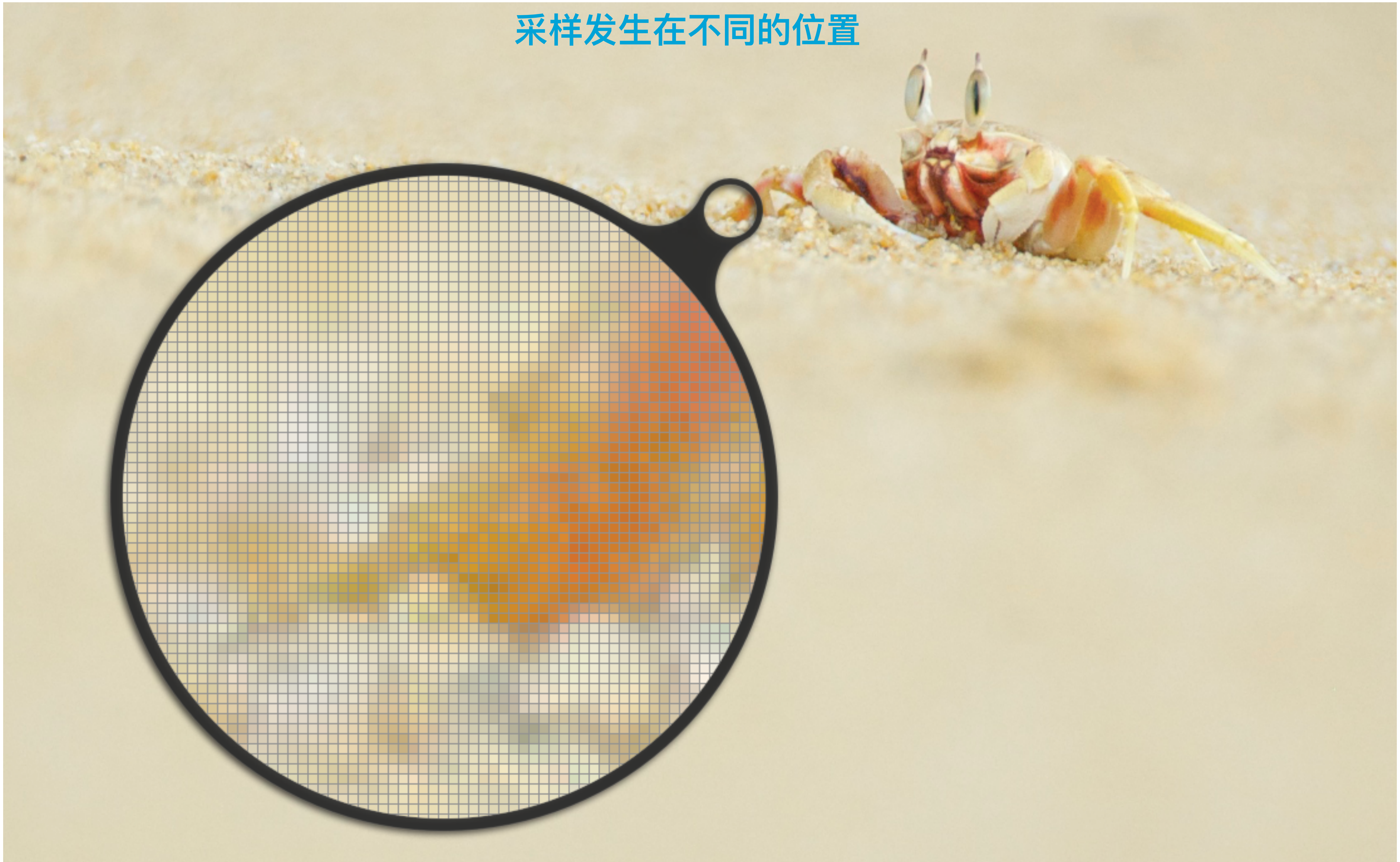
Slide courtesy of Prof. Ren Ng, UC Berkeley

Sampling is Ubiquitous in Computer Graphics

Rasterization = Sample 2D Positions

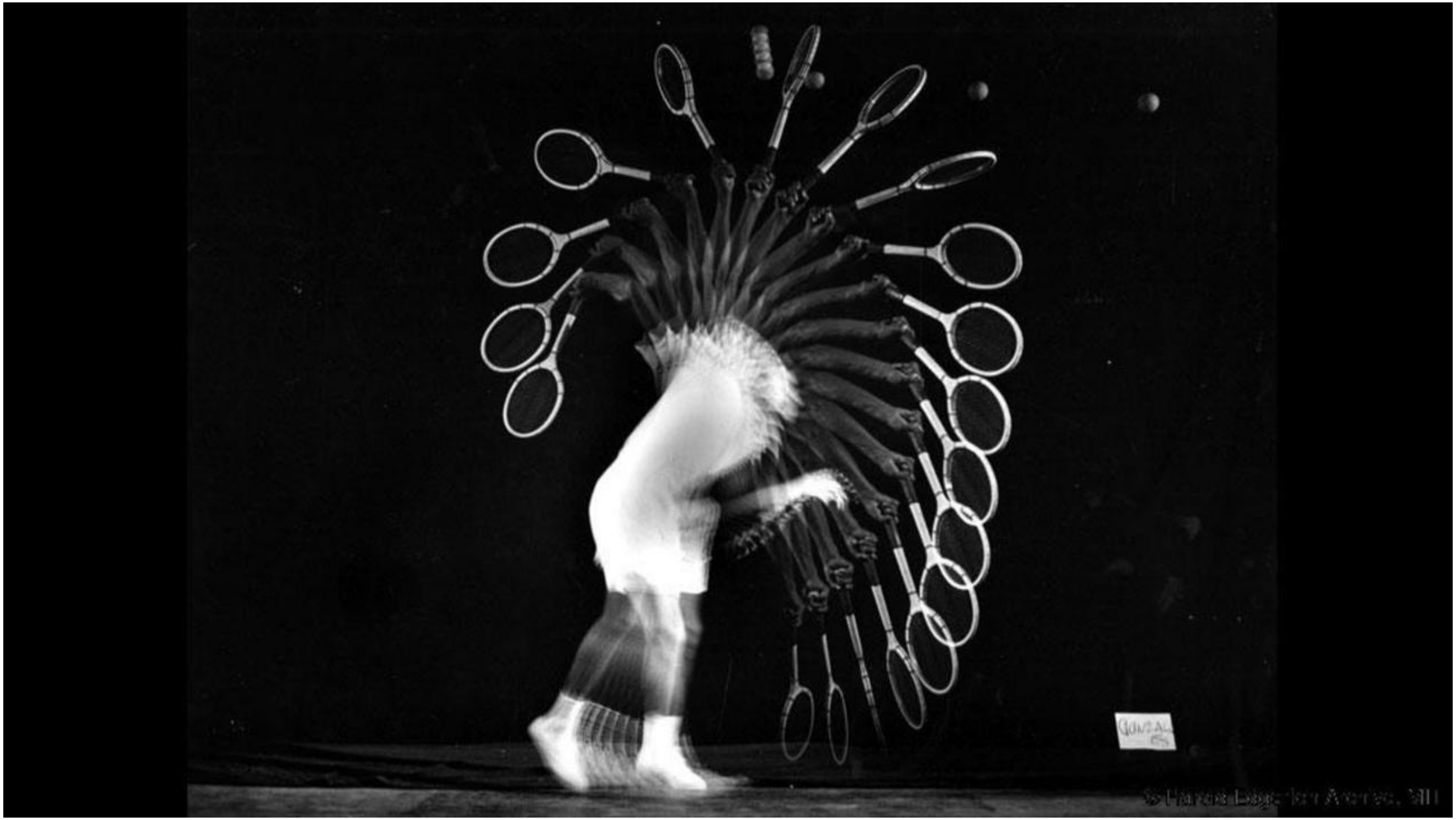


Photograph = Sample Image Sensor Plane



Video = Sample Time

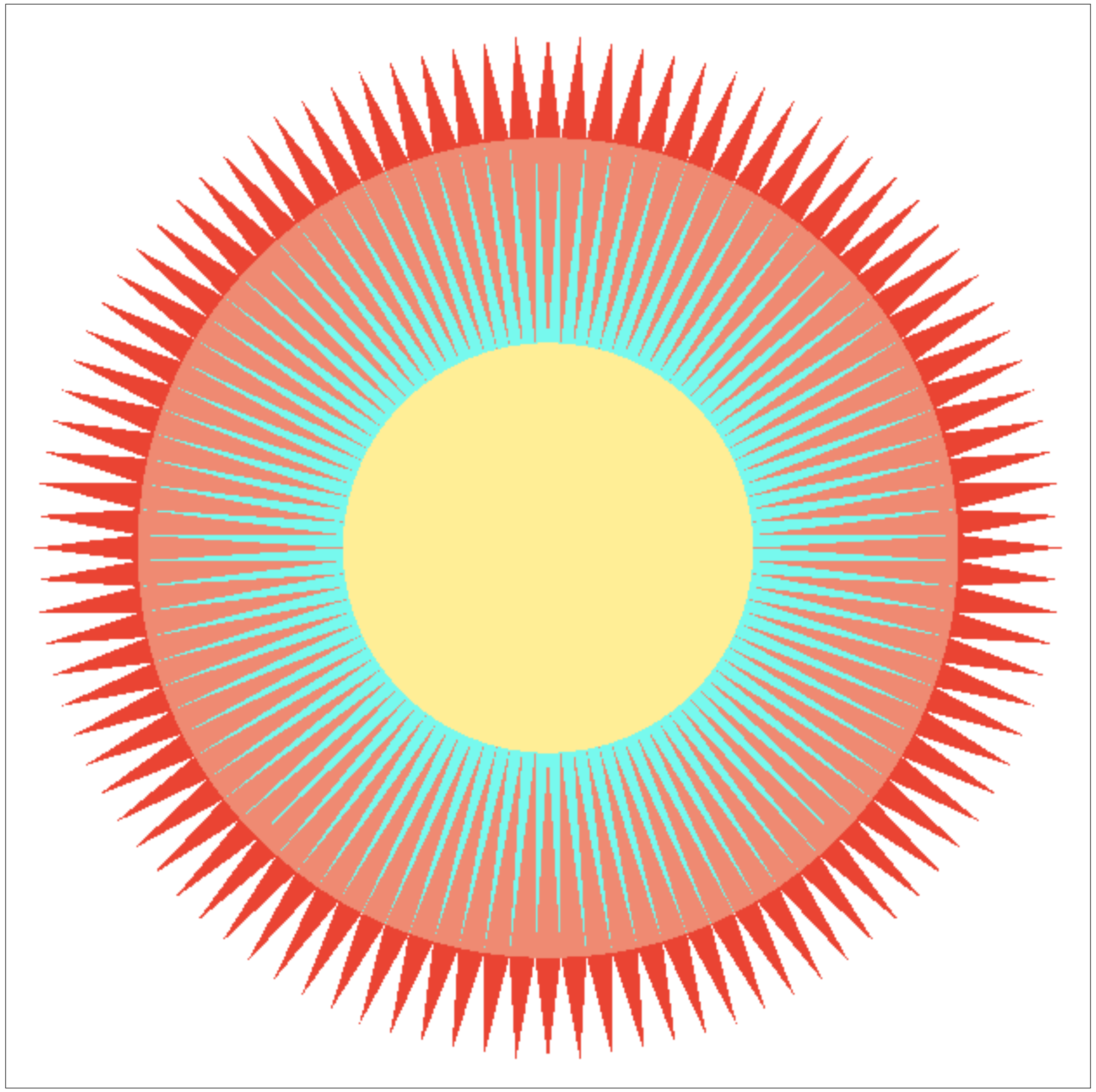
在时间中发生的采样

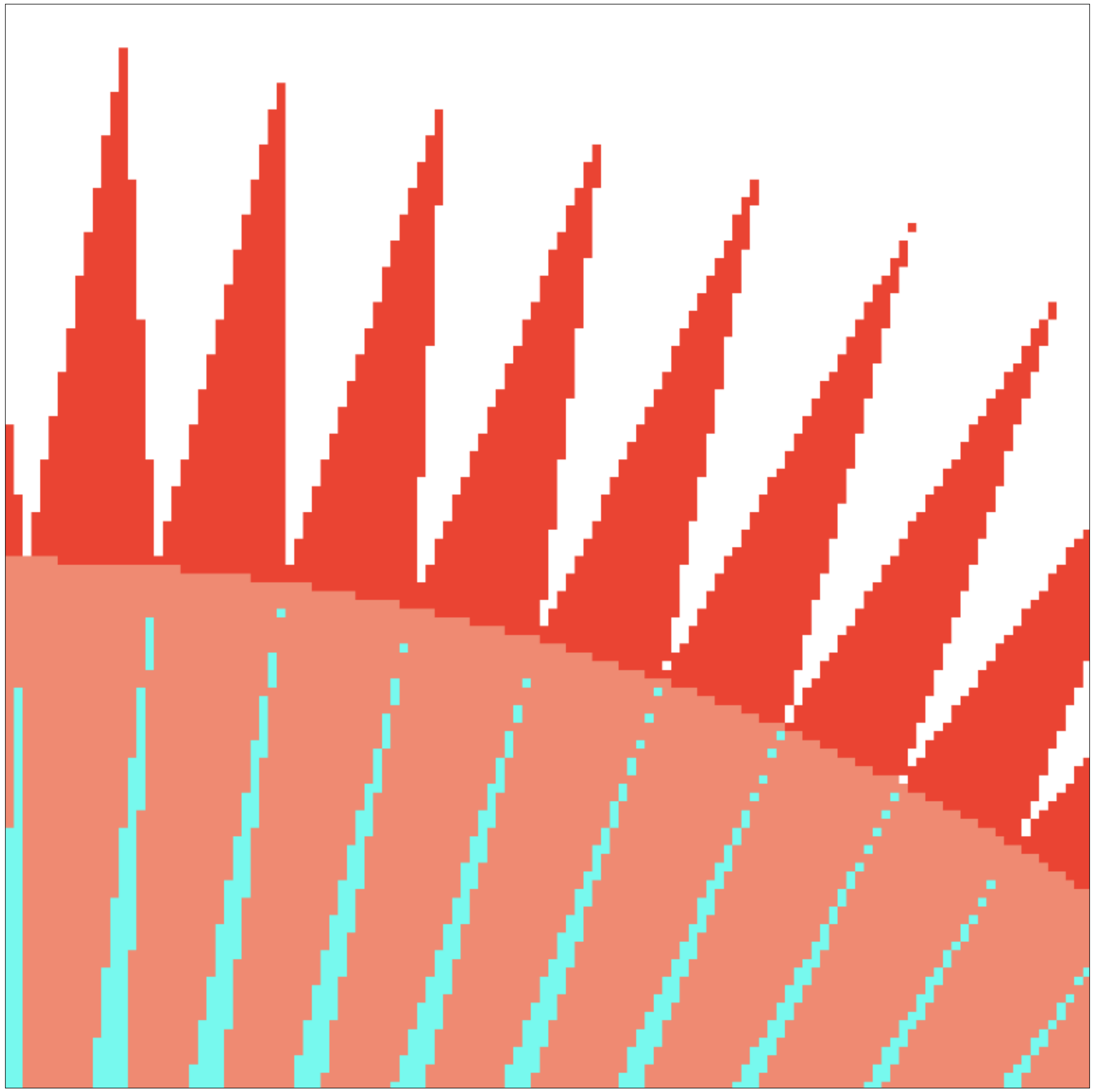


Harold Edgerton Archive, MIT

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

Jaggies (Staircase Pattern)

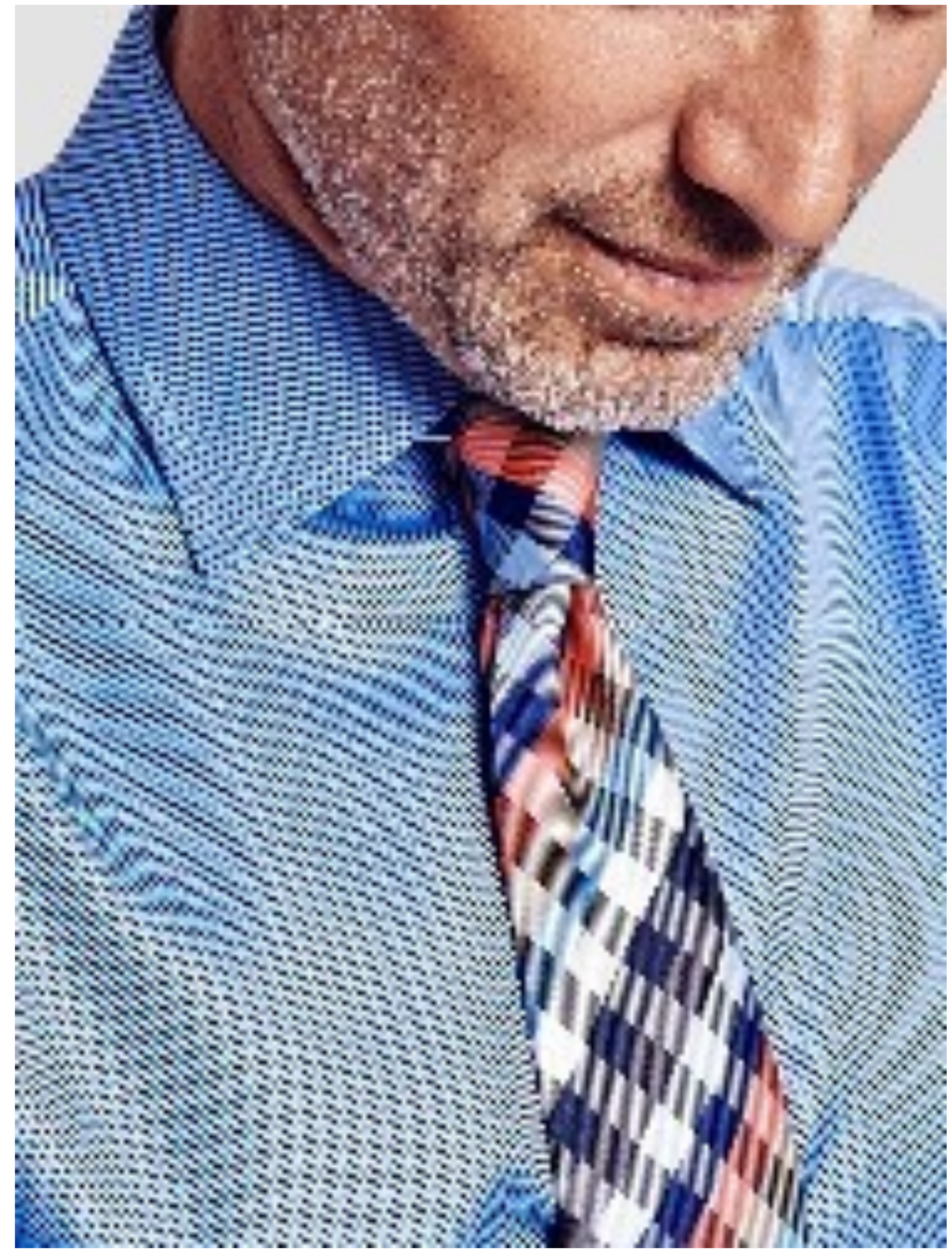




This is also an example of “aliasing” – a sampling error

Moiré Patterns in Imaging

[mwɑː]



lystit.com

Skip odd rows and columns

Wagon Wheel Illusion (False Motion)

Sampling Artifacts in Computer Graphics

Artifacts due to sampling - “Aliasing”

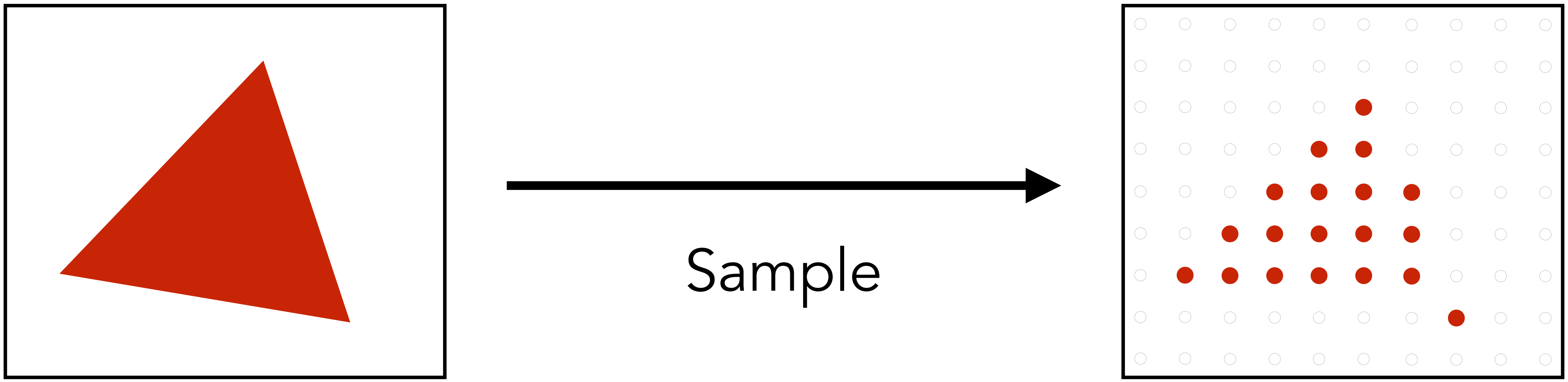
- Jaggies – sampling in space
- Moire – undersampling images
- Wagon wheel effect – sampling in time
- [Many more] …

Behind the Aliasing Artifacts

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

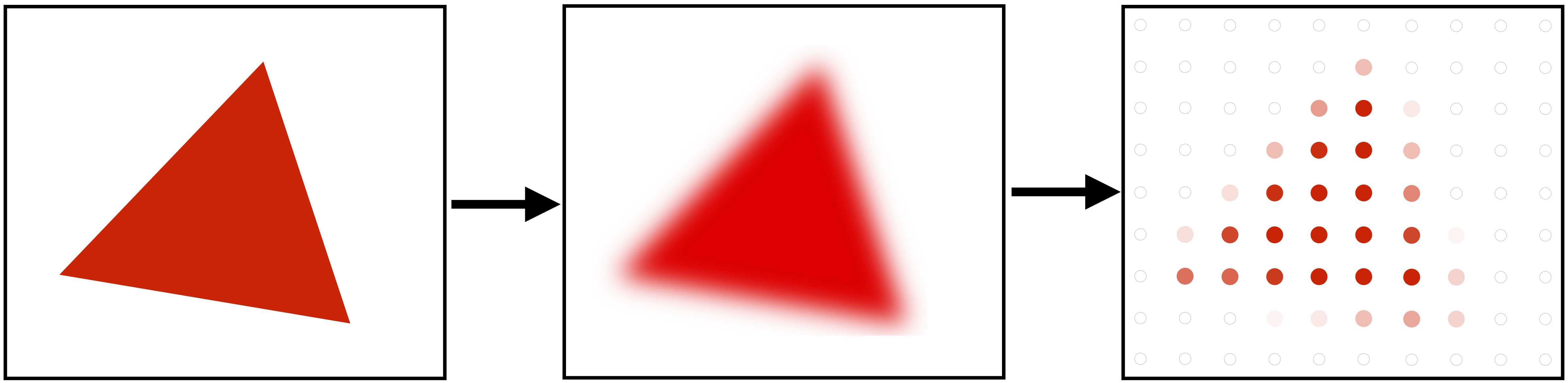
Antialiasing Idea:
Blurring (Pre-Filtering) Before Sampling

Rasterization: Point Sampling in Space



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

Rasterization: Antialiased Sampling



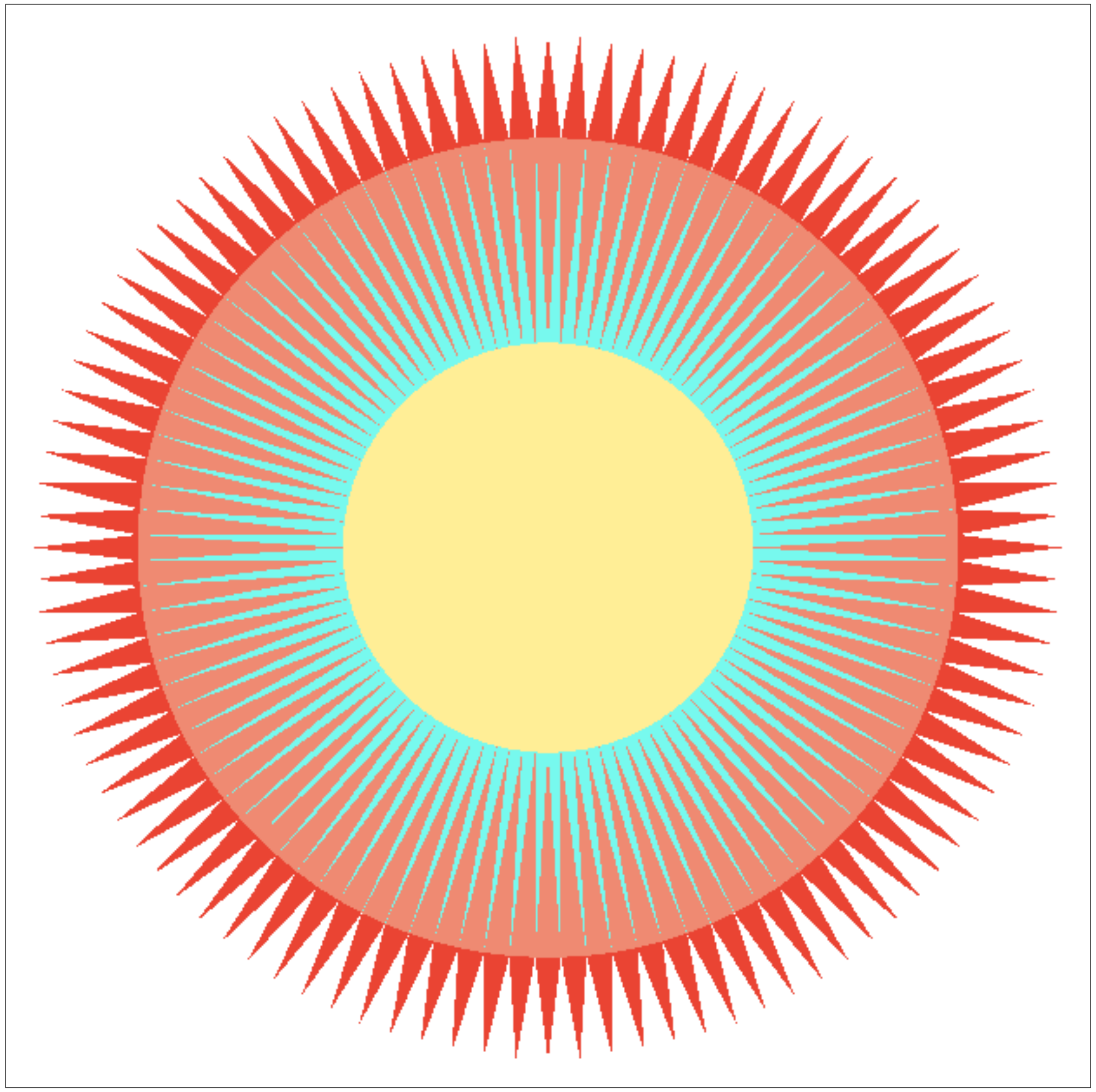
Pre-Filter

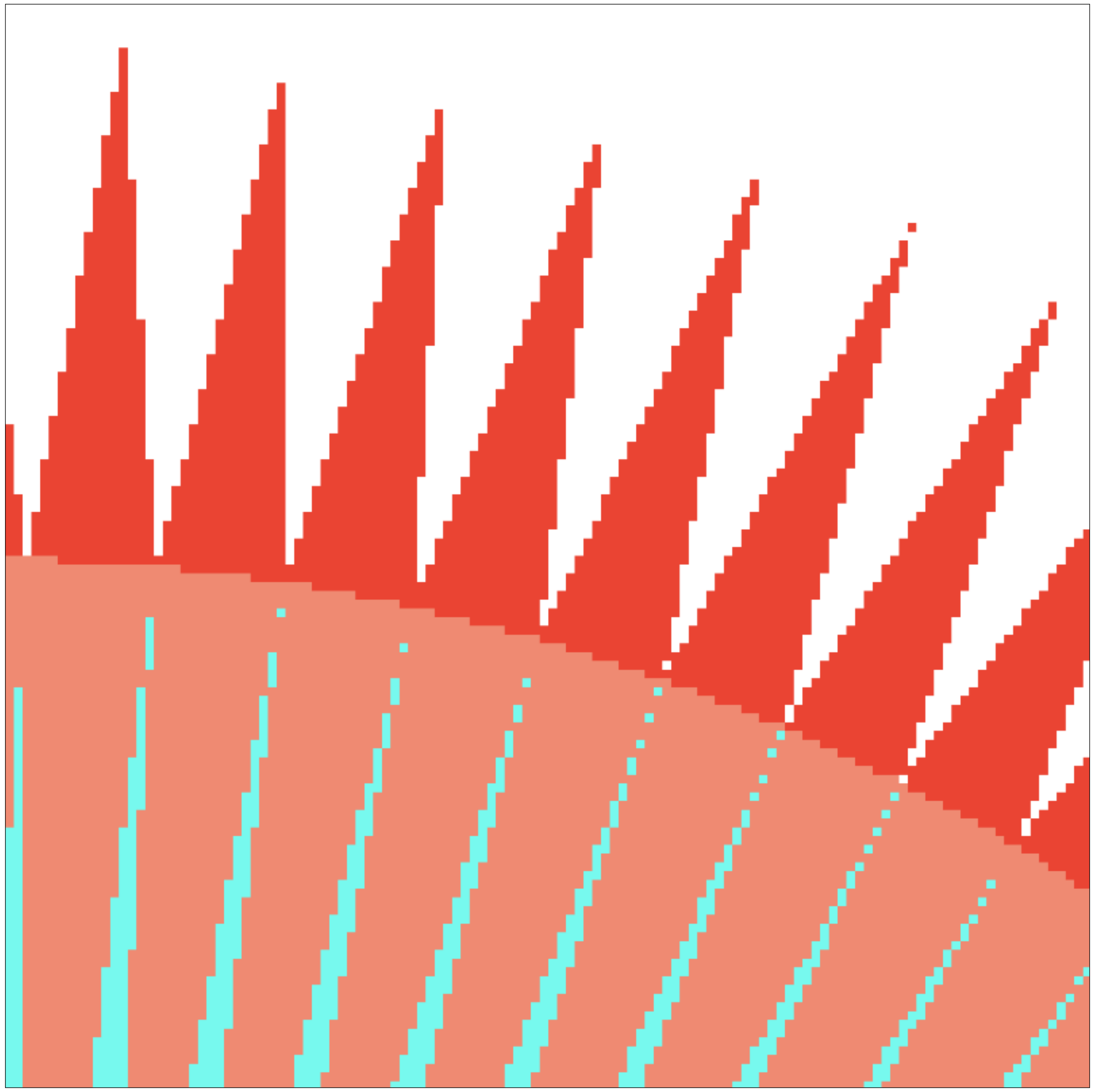
(remove frequencies above Nyquist) (?)

Sample

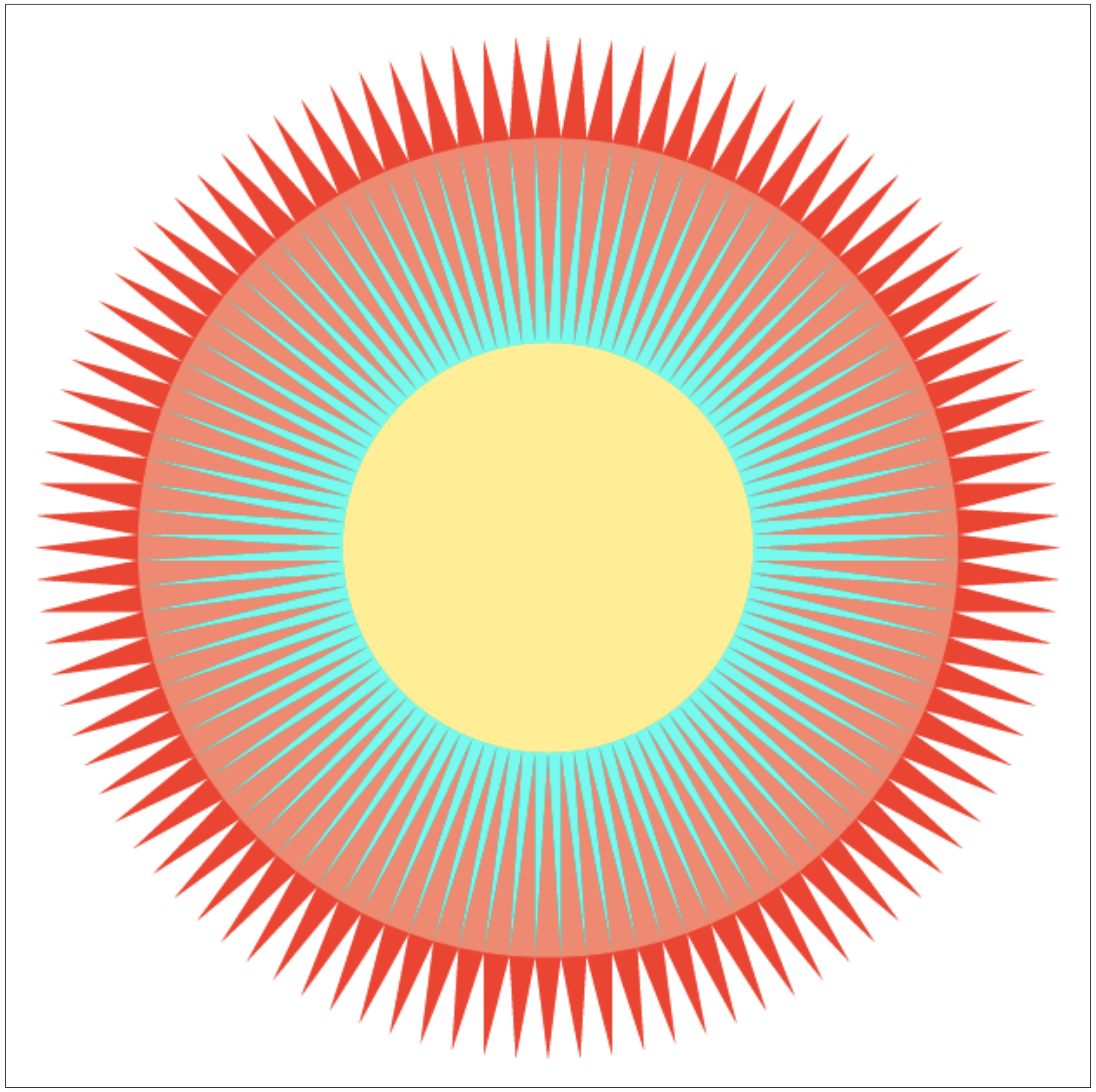
Note antialiased edges in rasterized triangle
where pixel values take intermediate values

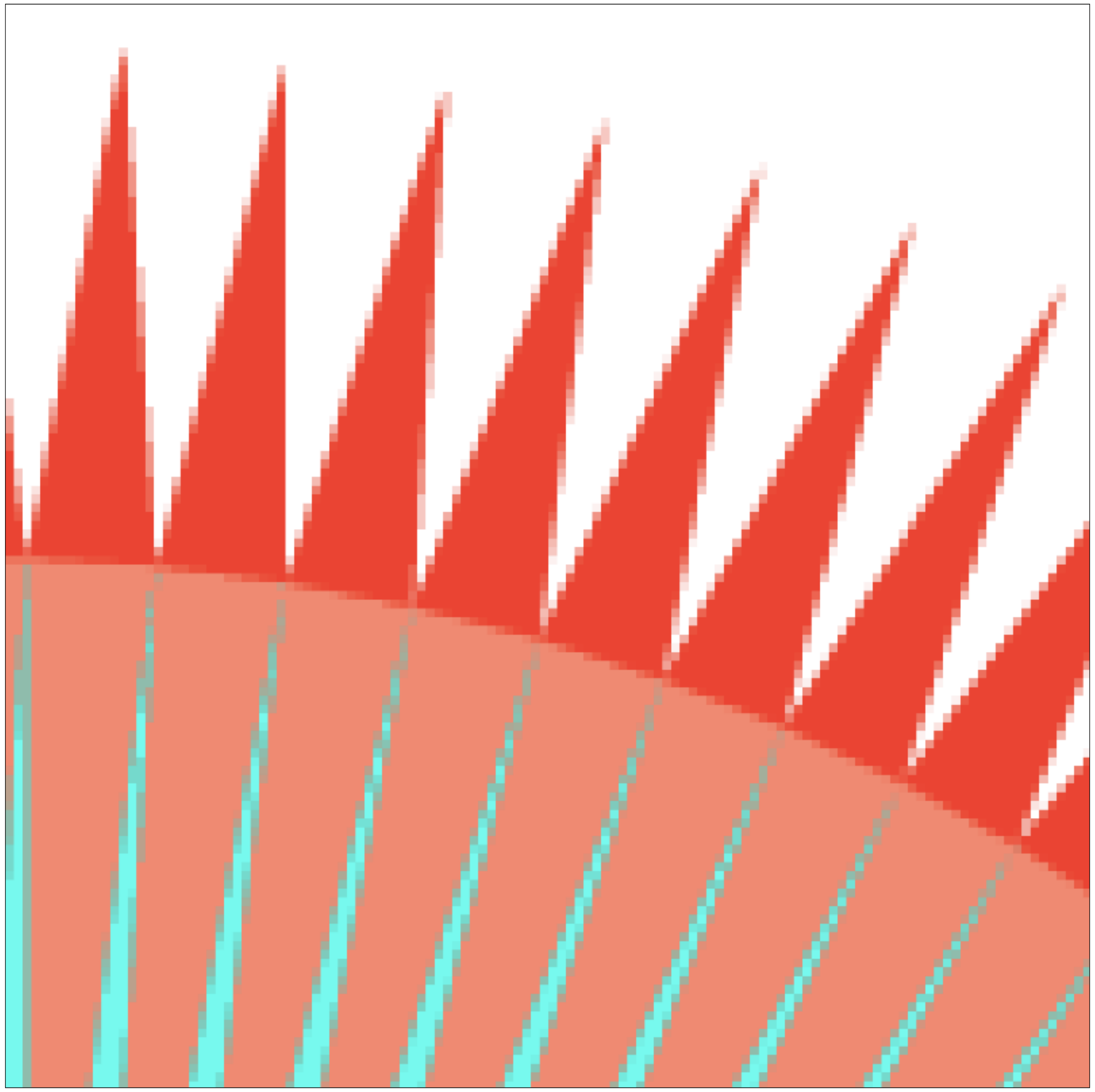
Point Sampling



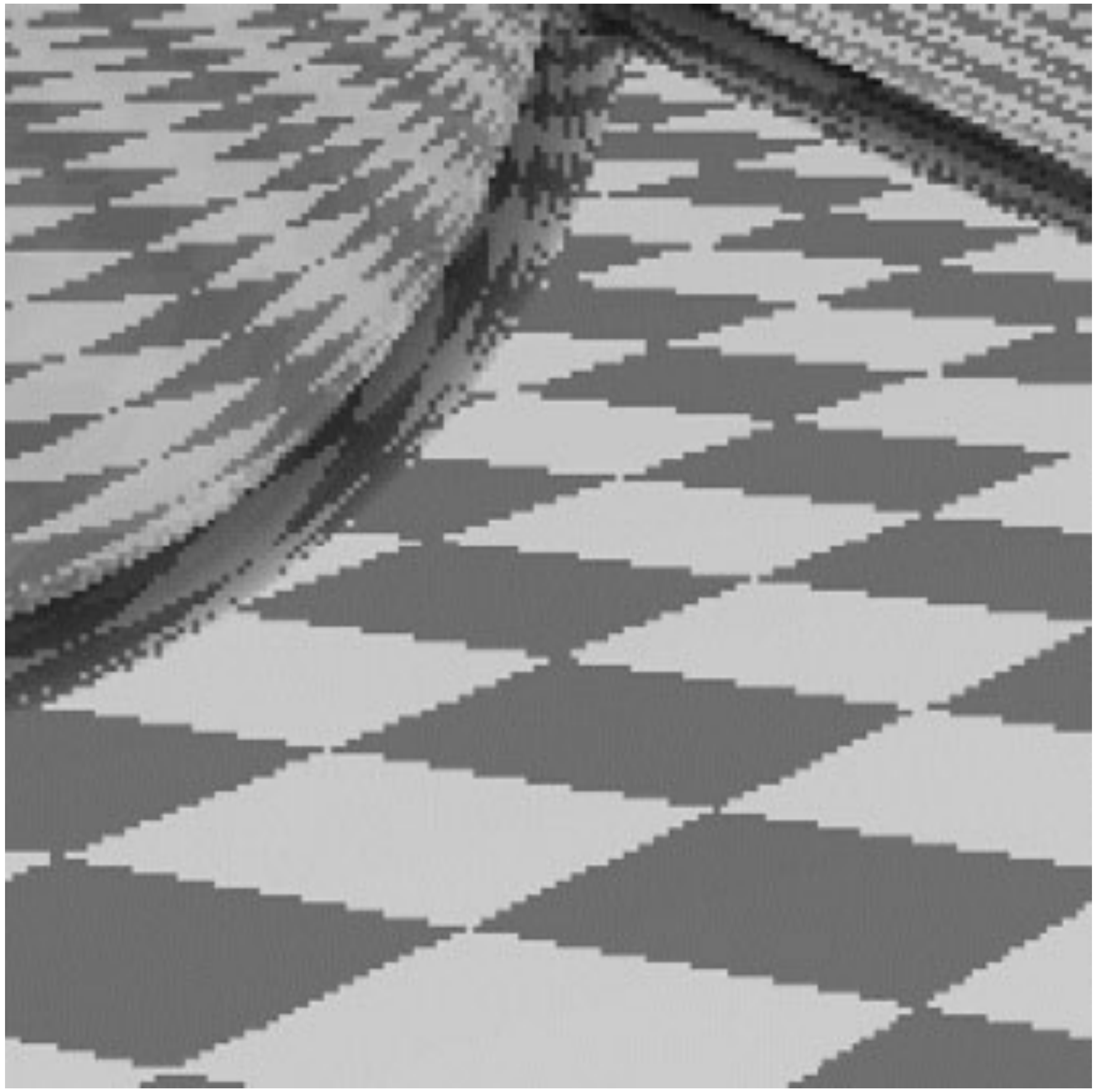


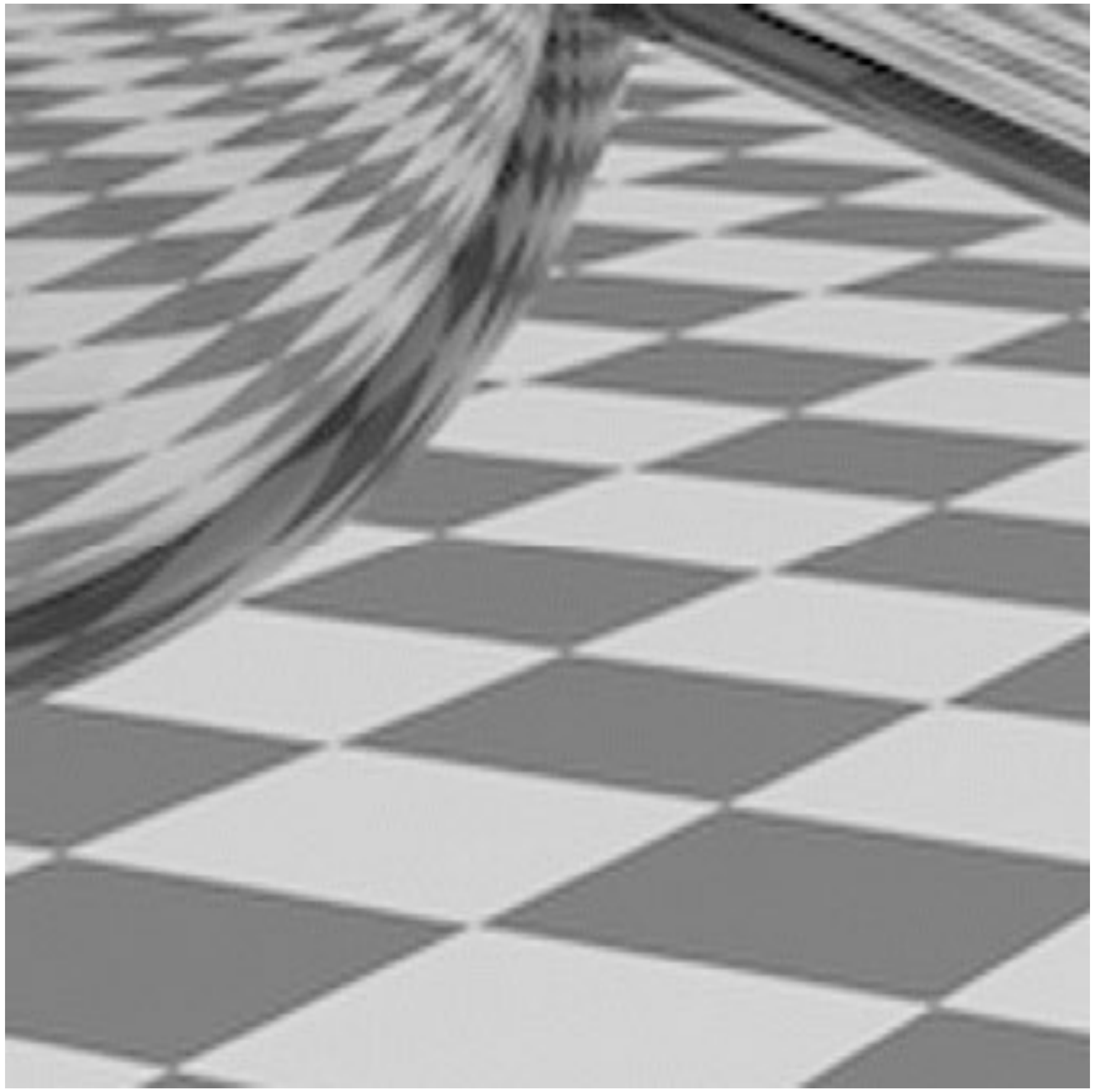
Antialiasing





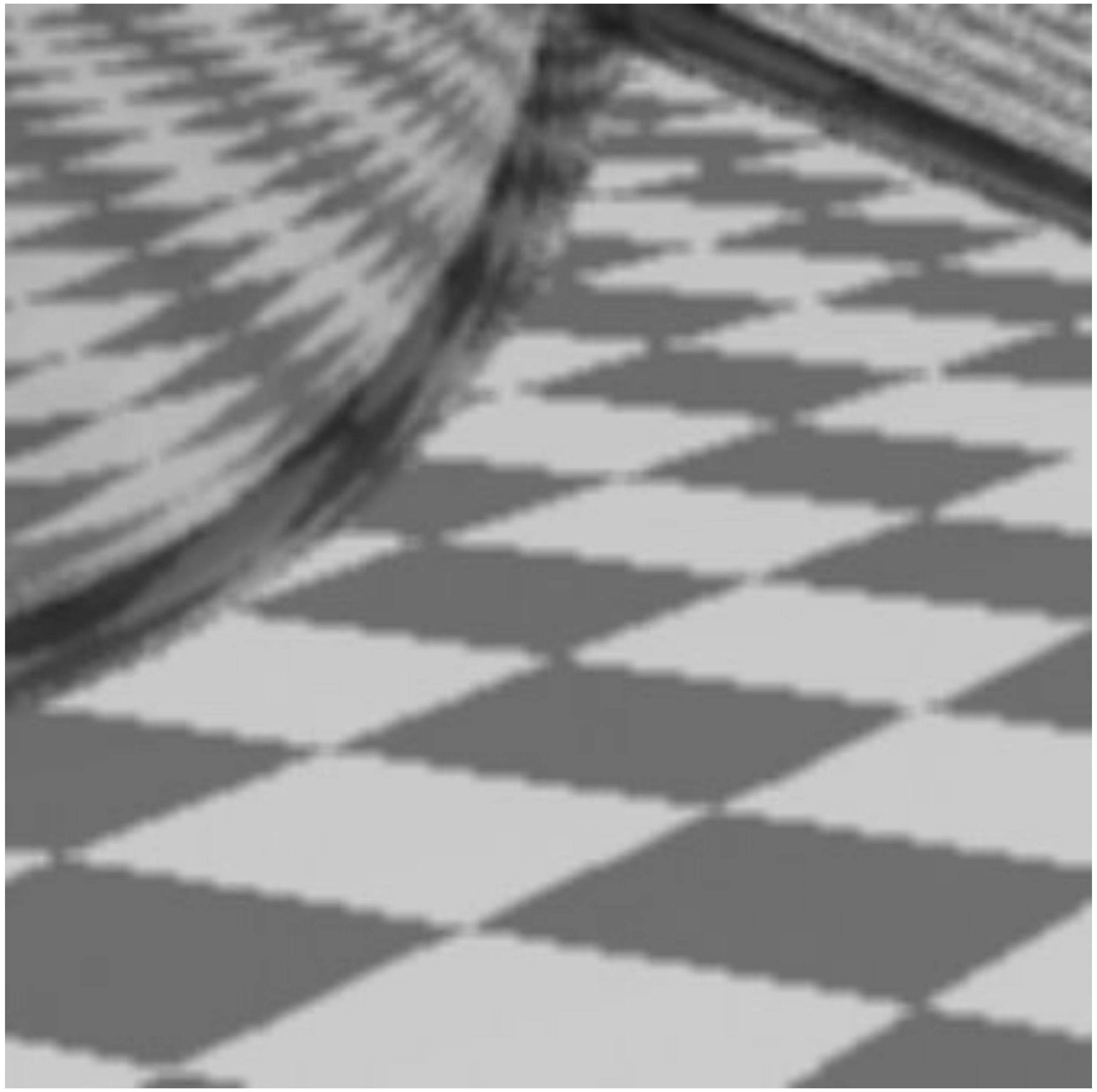
Point Sampling vs Antialiasing



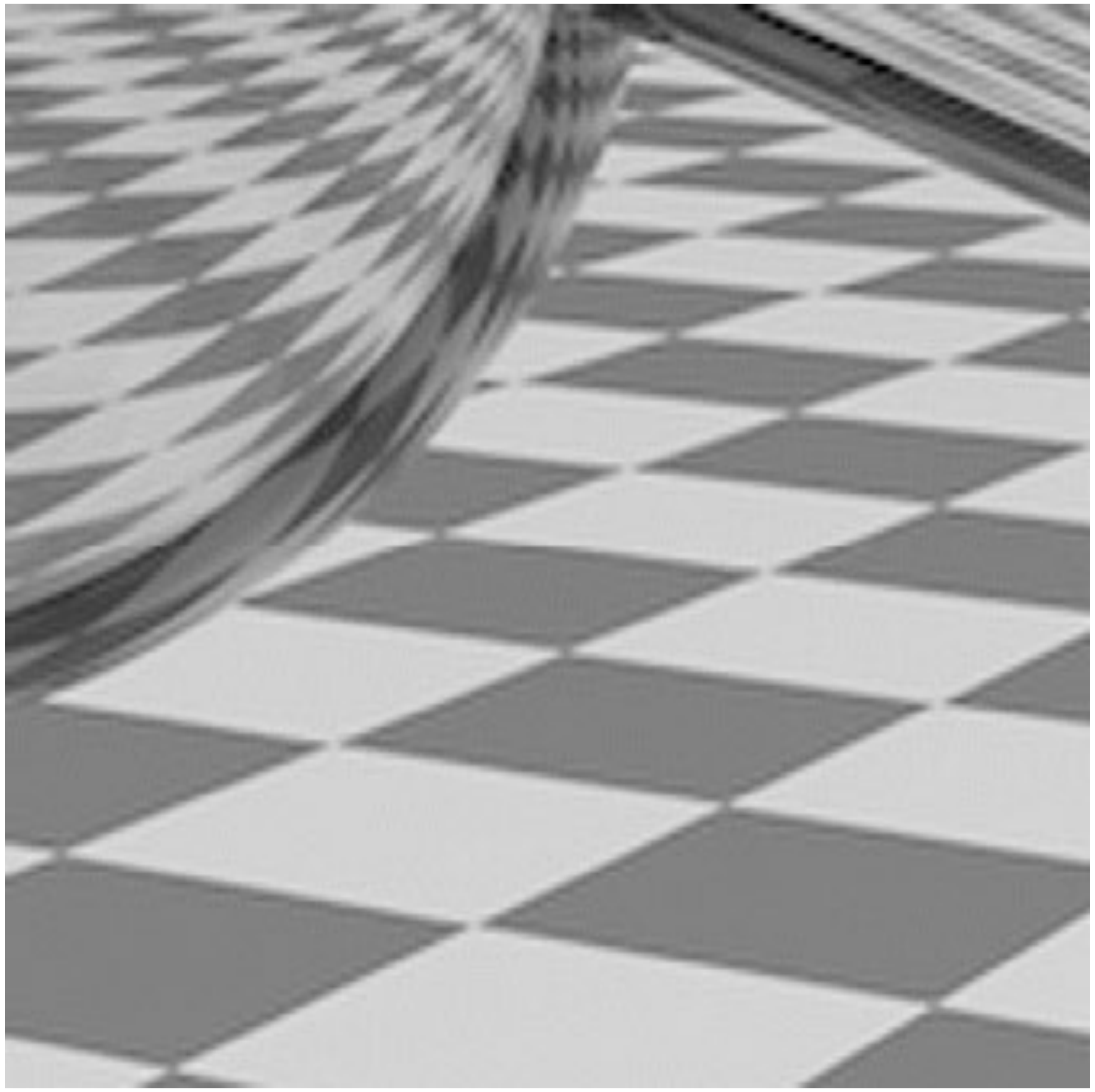


Antialiasing vs Blurred Aliasing

模糊的走样



(Sample then filter, WRONG!)



(Filter then sample)

But why?

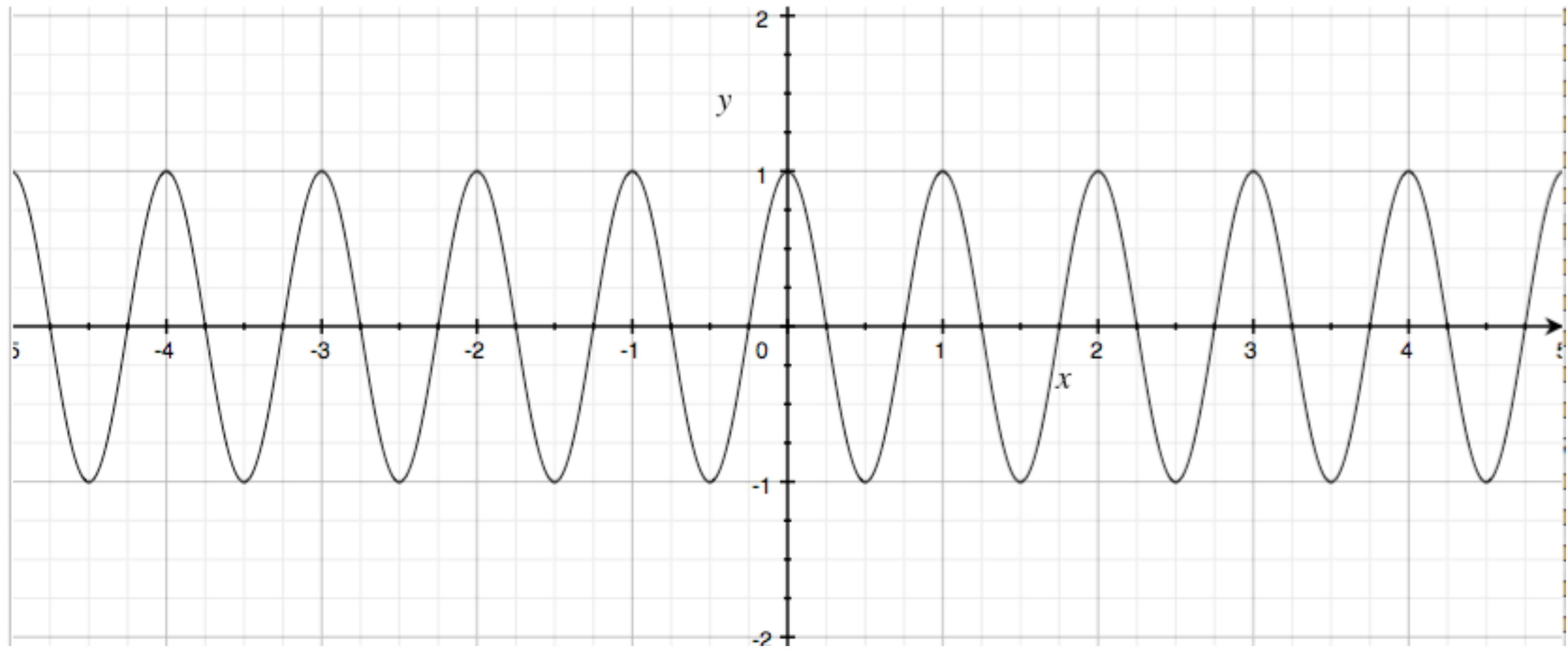
1. Why undersampling introduces aliasing?
采样过疏
2. Why pre-filtering then sampling can do antialiasing?

Let's dig into fundamental reasons

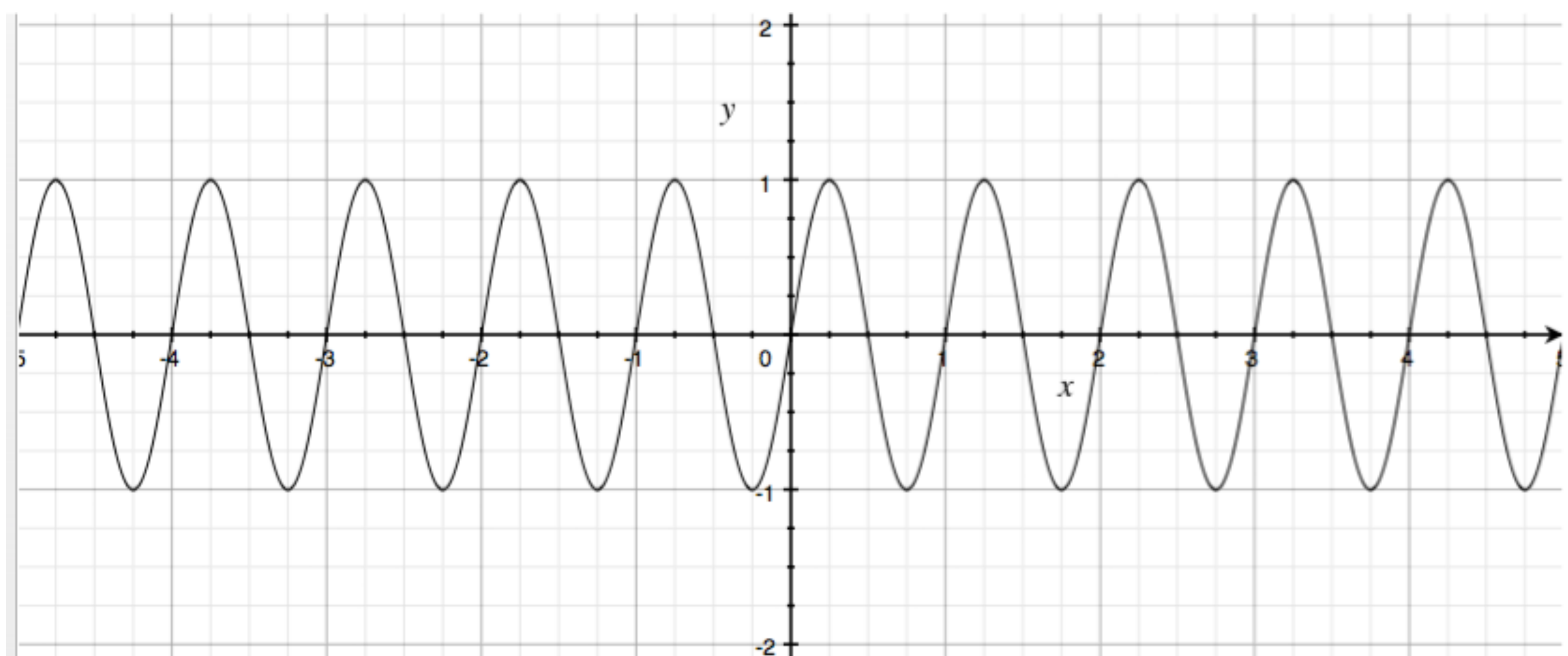
And look at how to implement antialiased rasterization

Frequency Domain

Sines and Cosines



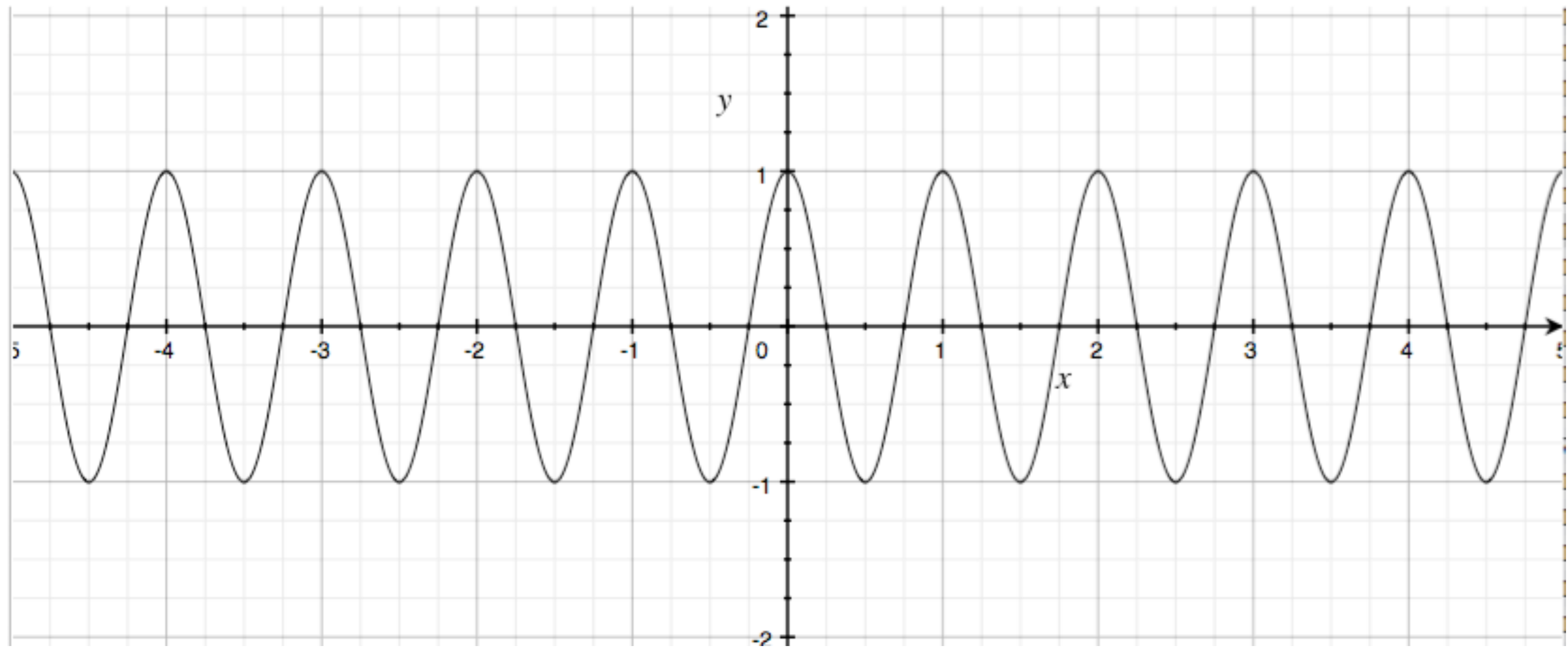
$$\cos 2\pi x$$



$$\sin 2\pi x$$

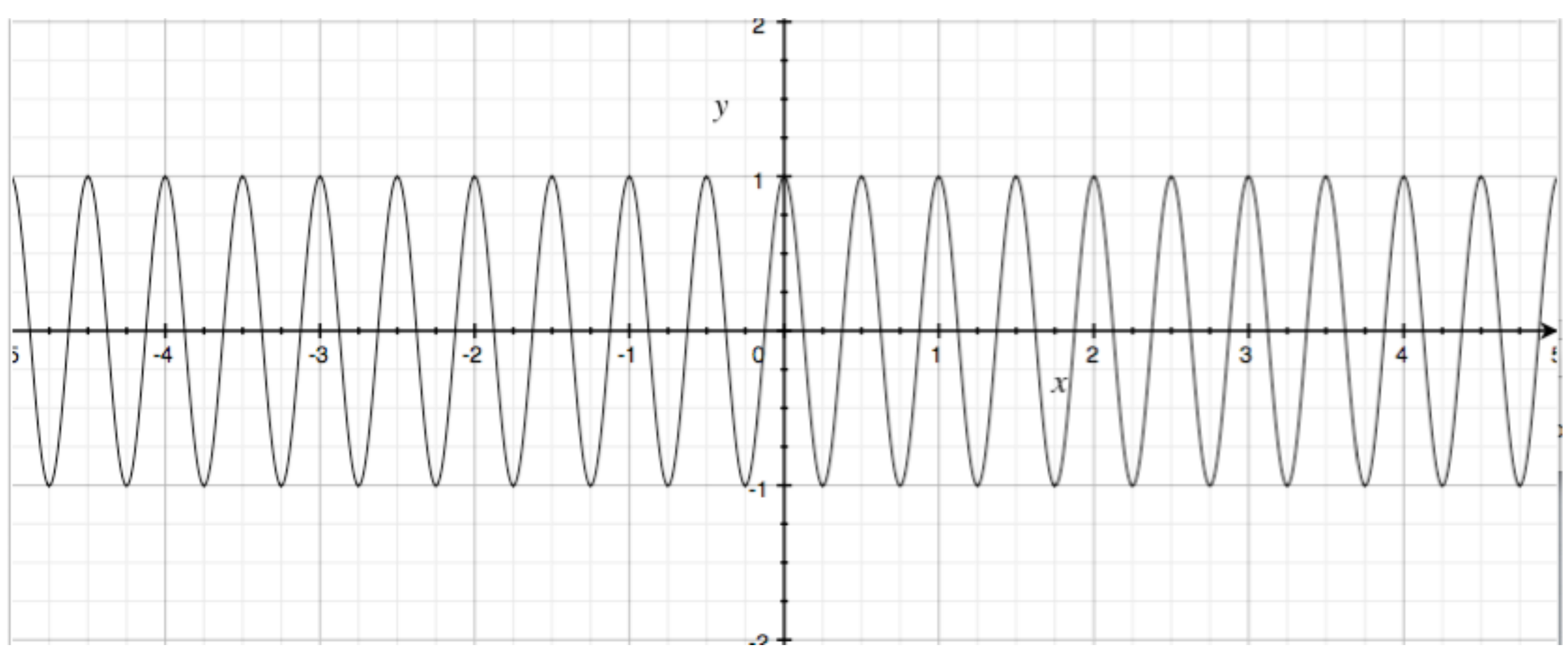
Frequencies $\cos 2\pi f x$

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



$$f = 1$$

$$\cos 2\pi x$$



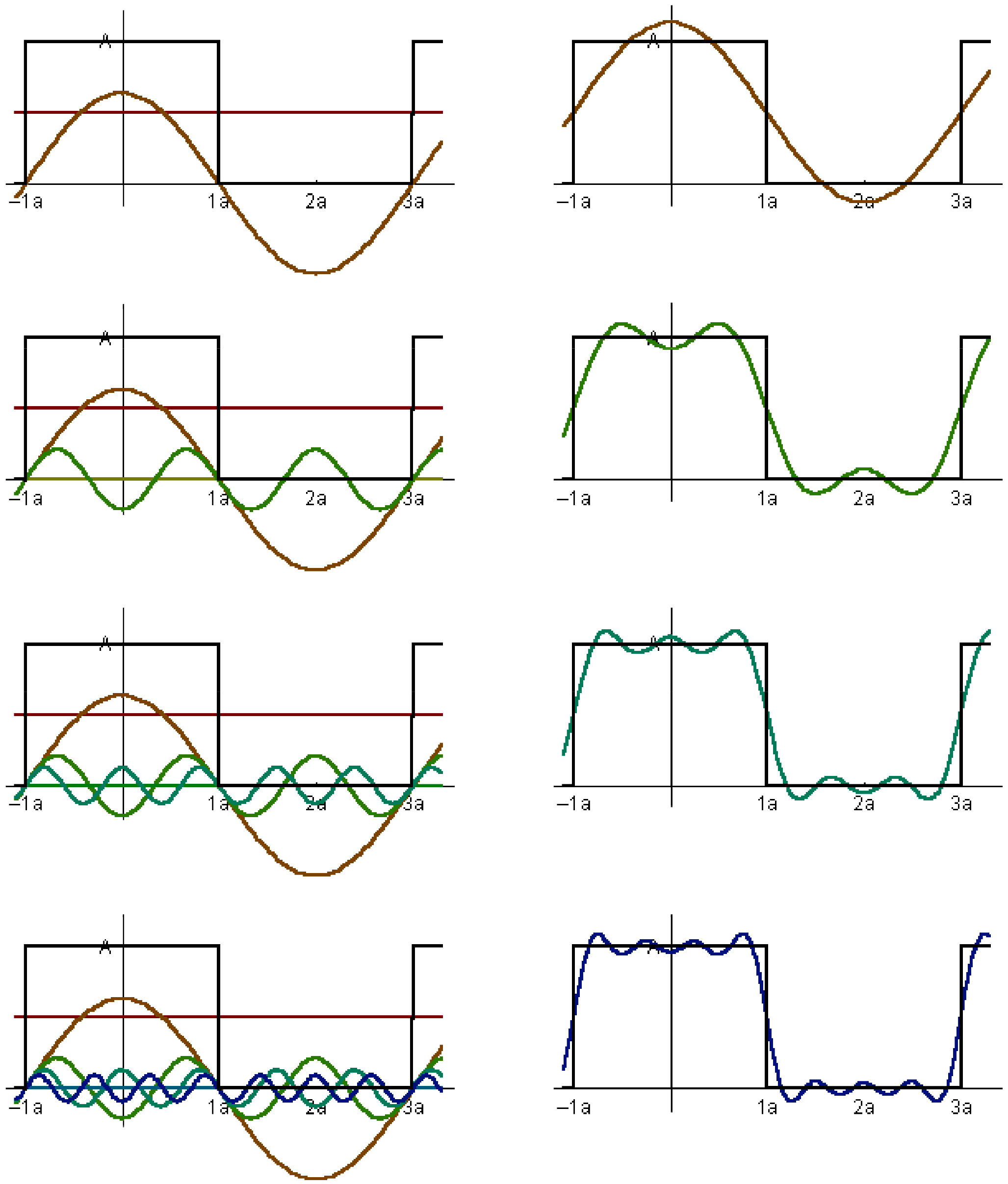
$$f = 2$$

$$\cos 4\pi x$$

Fourier Transform

Represent a function as a weighted sum of sines and cosines

Joseph Fourier 1768 - 1830



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

Fourier Transform Decomposes A Signal Into Frequencies

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

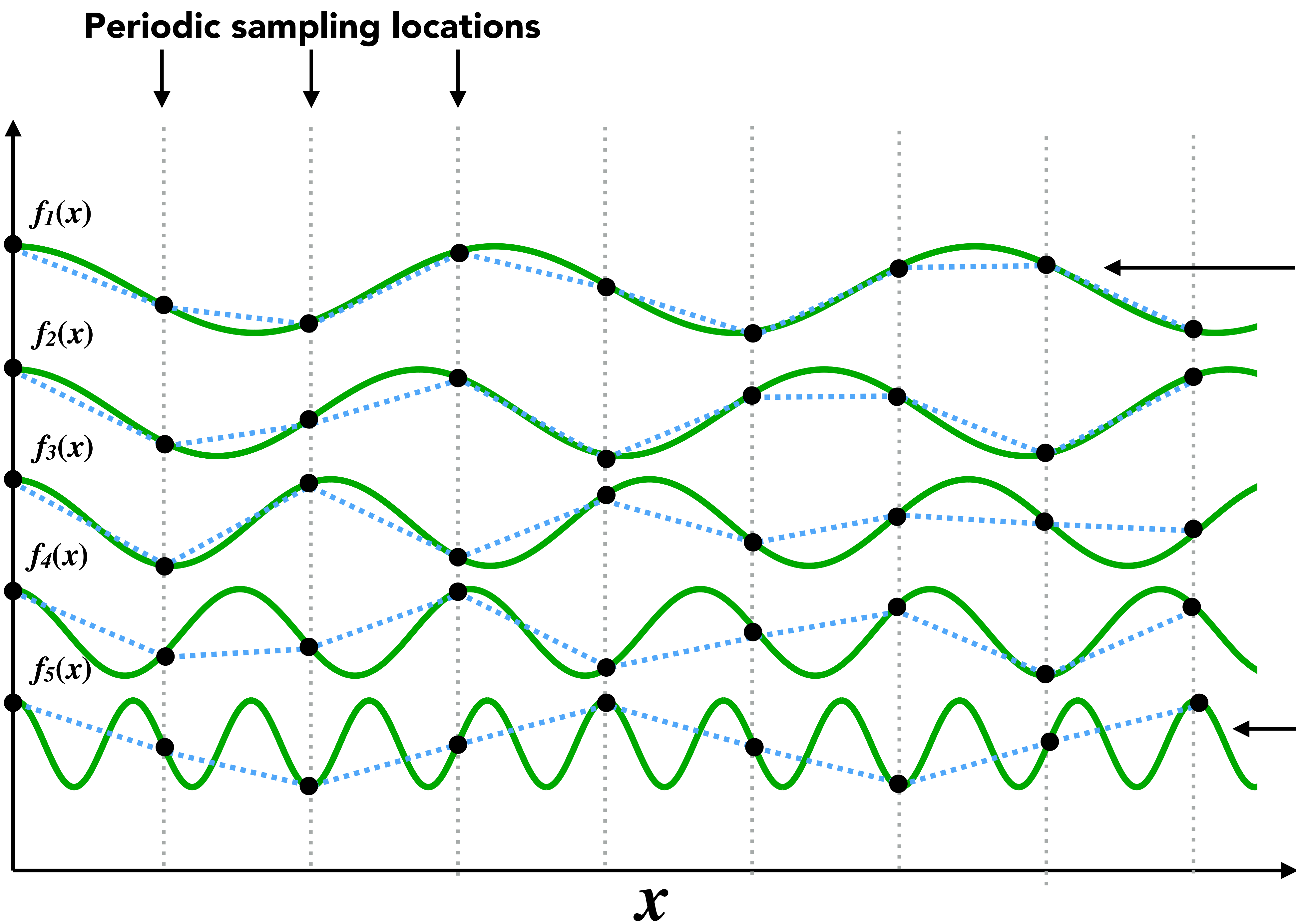
spatial domain → **Fourier transform** → **frequency domain**

spatial domain ← **Inverse transform** ← **frequency domain**

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

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

Higher Frequencies Need Faster Sampling

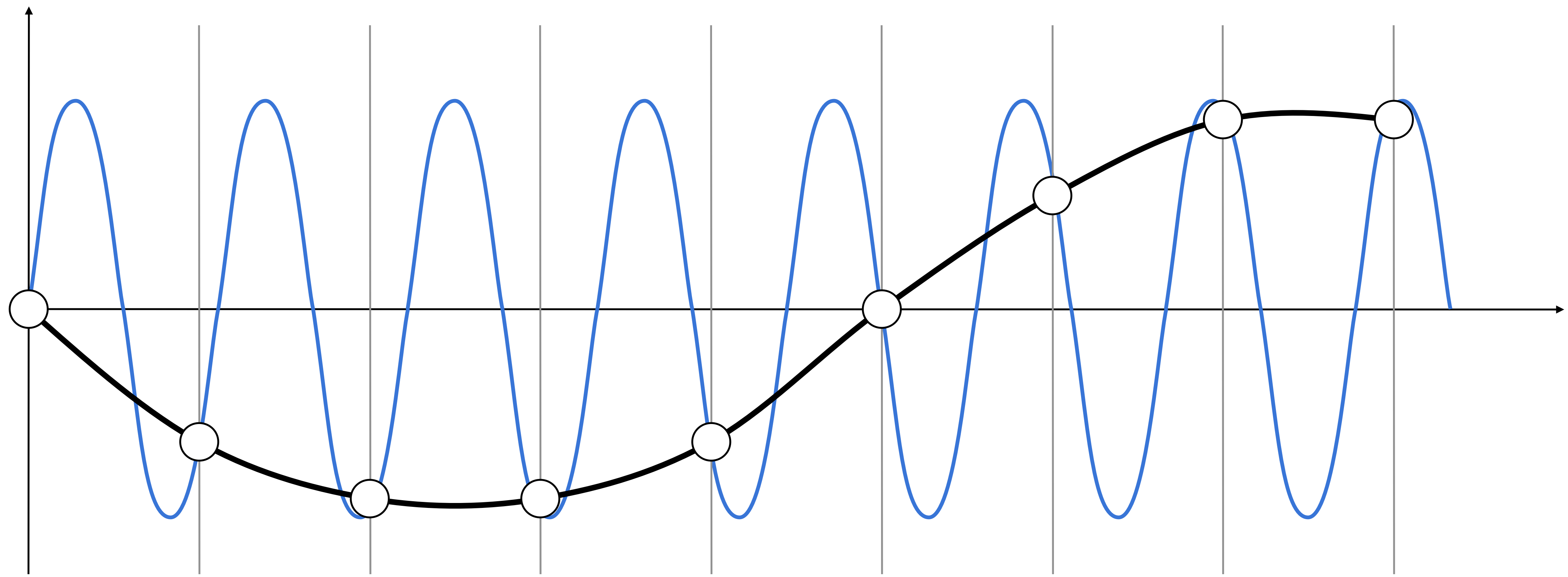


Low-frequency signal: sampled adequately for reasonable reconstruction

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

Undersampling Creates Frequency Aliases

采样过疏



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

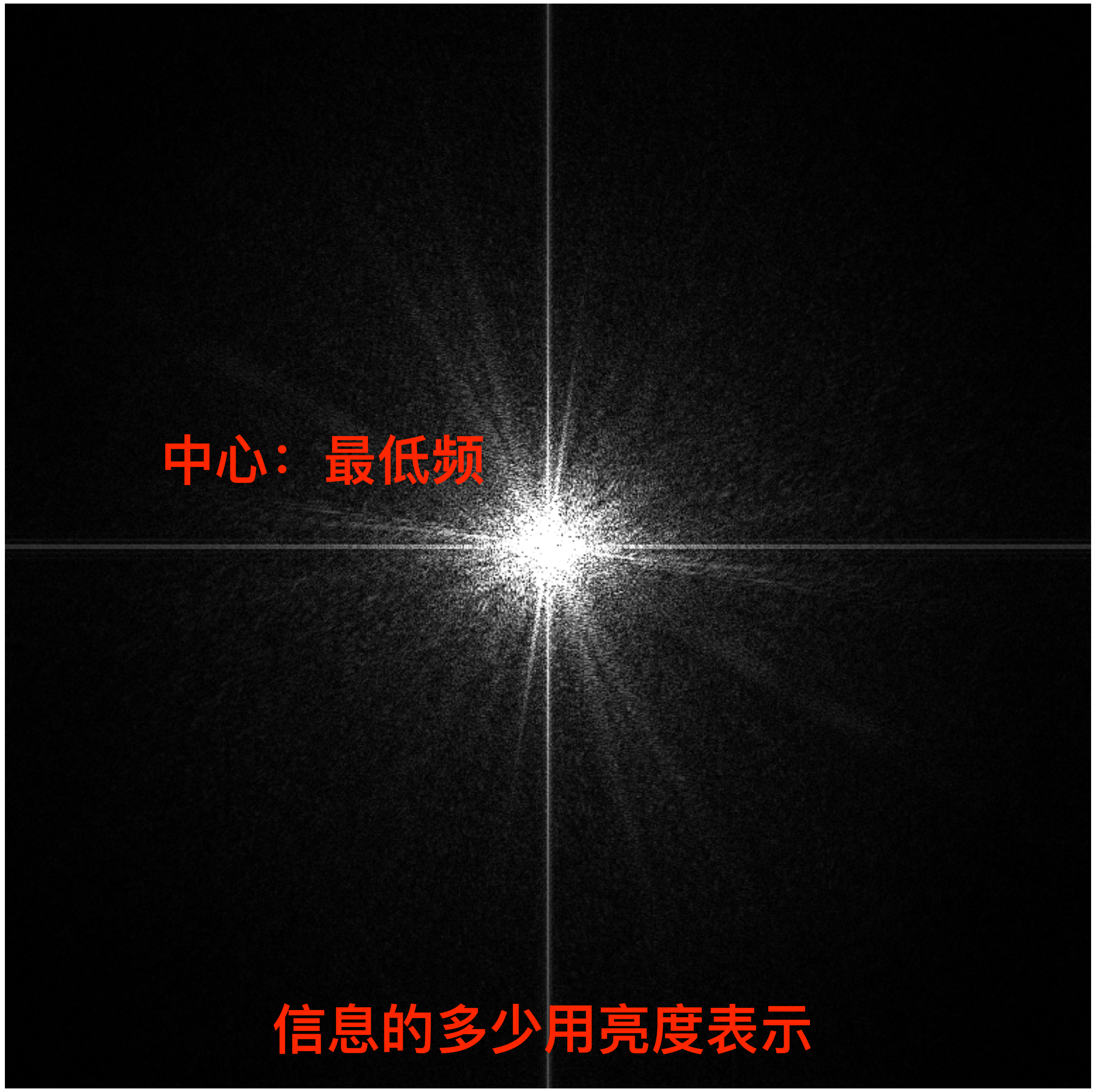
Two frequencies that are indistinguishable at a given sampling rate are called "aliases"

走样

Filtering = Getting rid of certain frequency contents

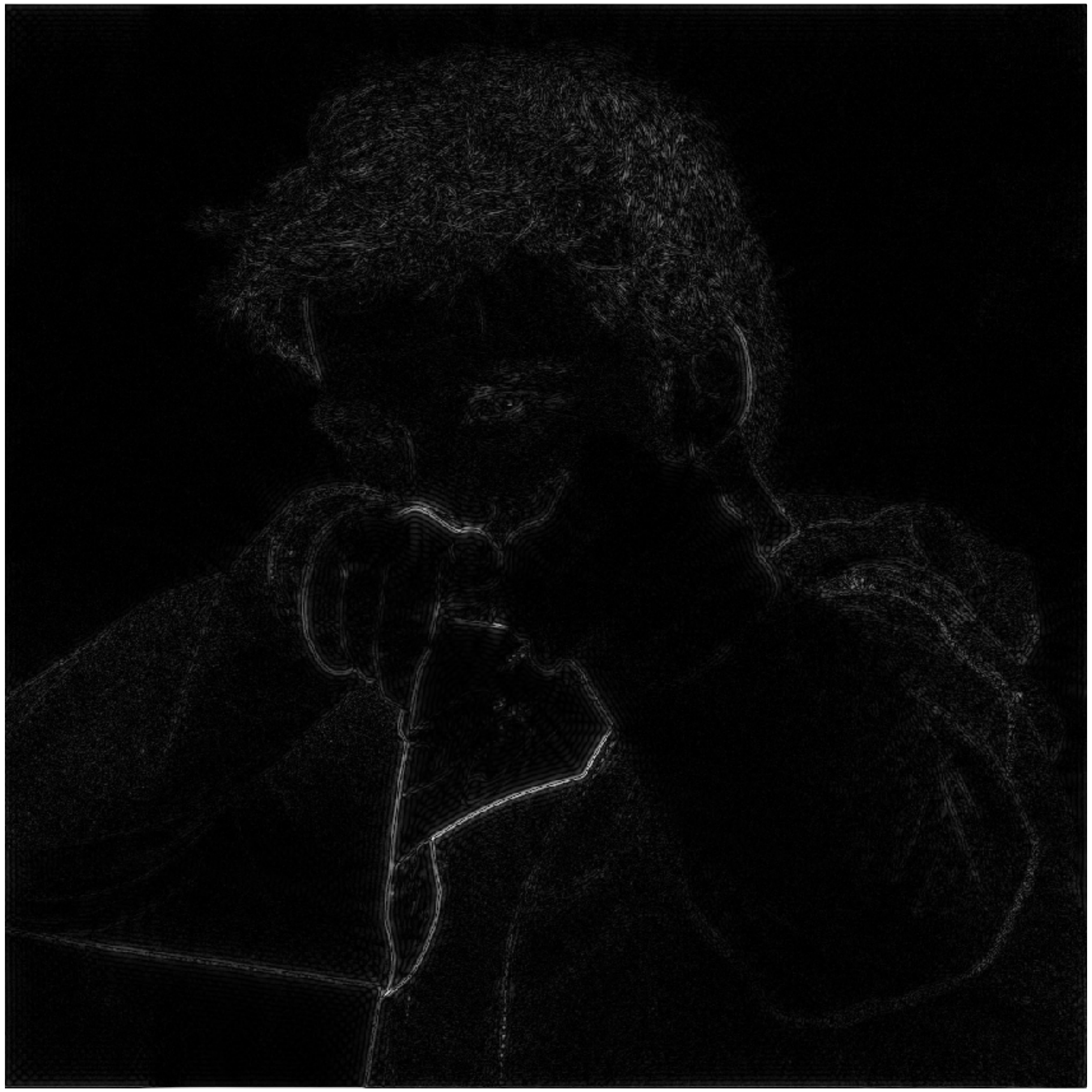
滤波：去掉某些频率

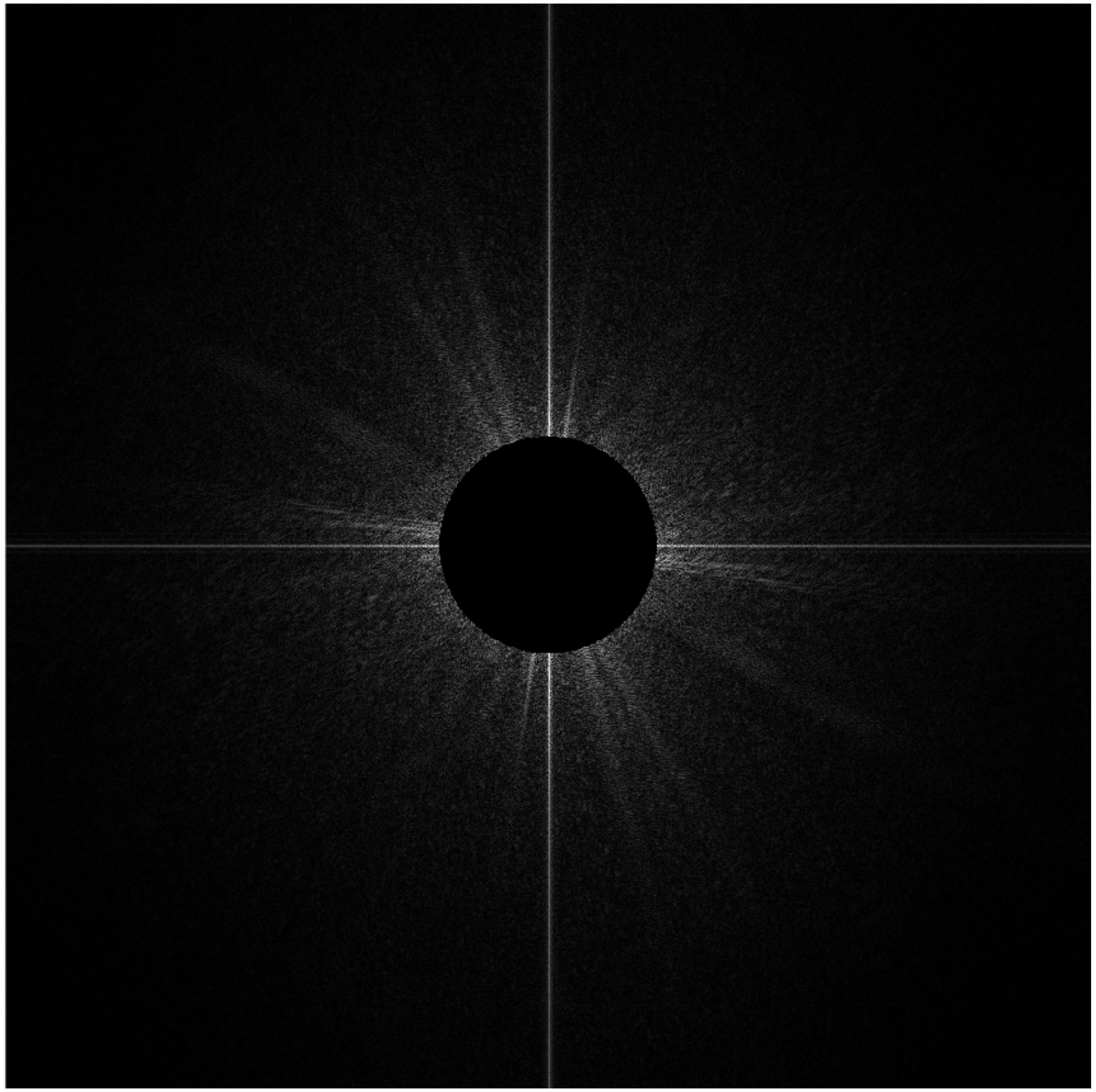
Visualizing Image Frequency Content



FT针对周期信号进行变换，但原始图片不是周期性的，所以认为原始图片上下左右均存在周期性复制。但在边界处存在信号突变，所以导致频域图上出现四个极高频的直线。

Filter Out Low Frequencies Only (Edges)





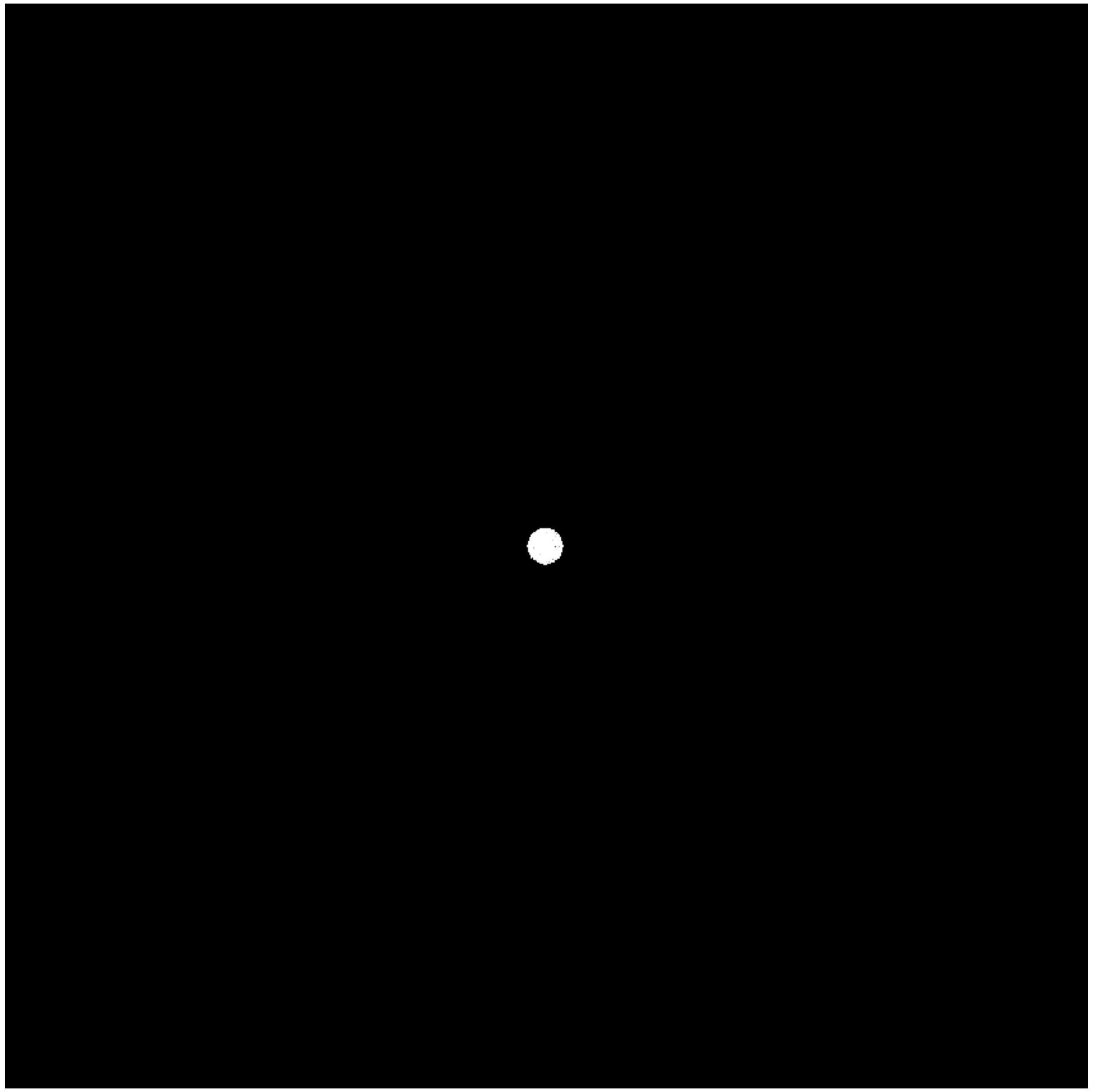
High-pass filter

表示图像内容中的“边界” 高通滤波

信号发生突变的地方

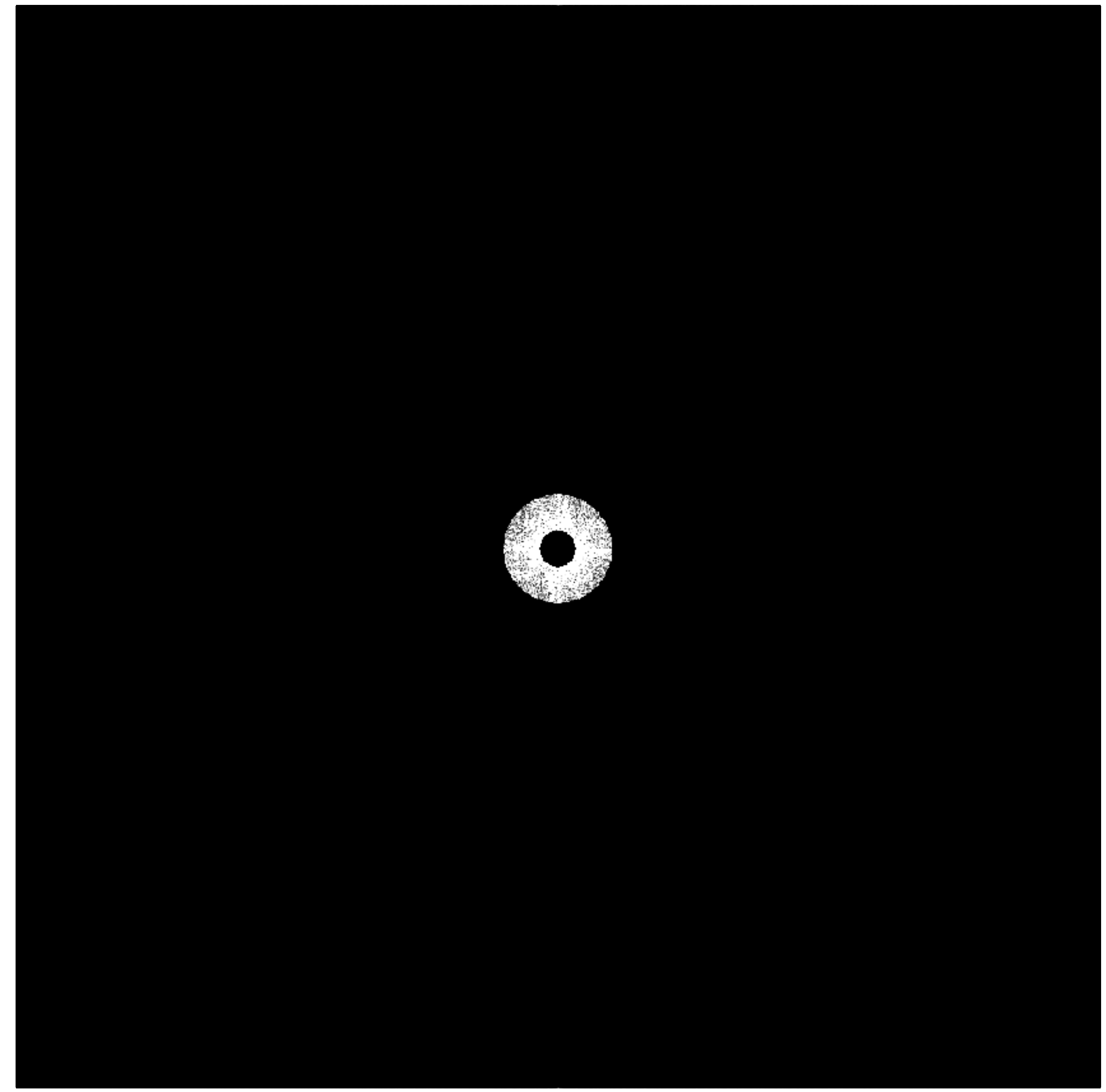
Filter Out High Frequencies (Blur)

低通滤波造成模糊

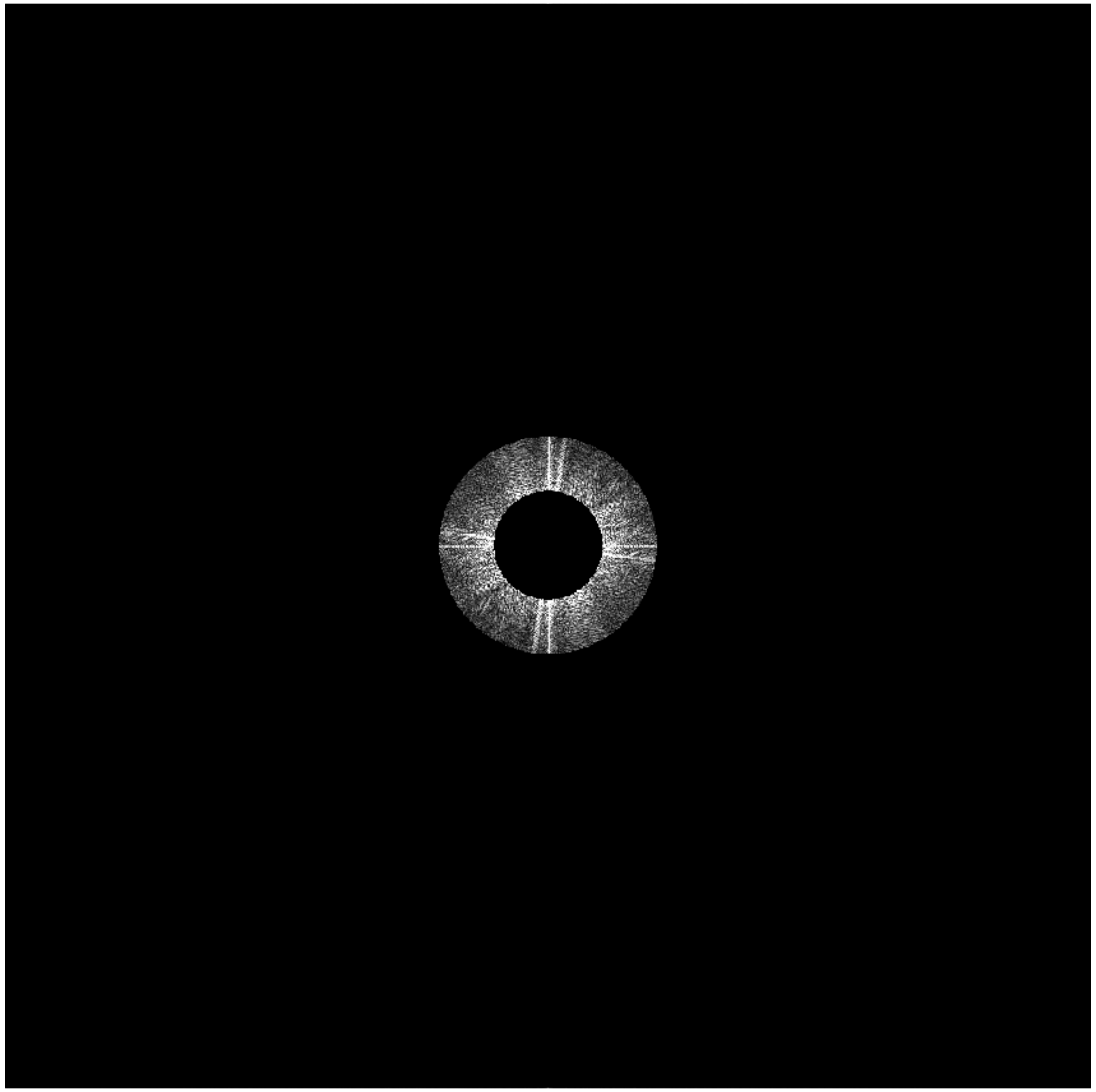


Low-pass filter

Filter Out Low and High Frequencies



Filter Out Low and High Frequencies



相关课程：《数字图像处理》

Filtering = Convolution

卷积

(= Averaging)

模糊～取平均

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

就是加权取周围平均的过程

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

Result

	3	4							

Convolution Theorem

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

Option 1:

- Filter by convolution in the spatial domain

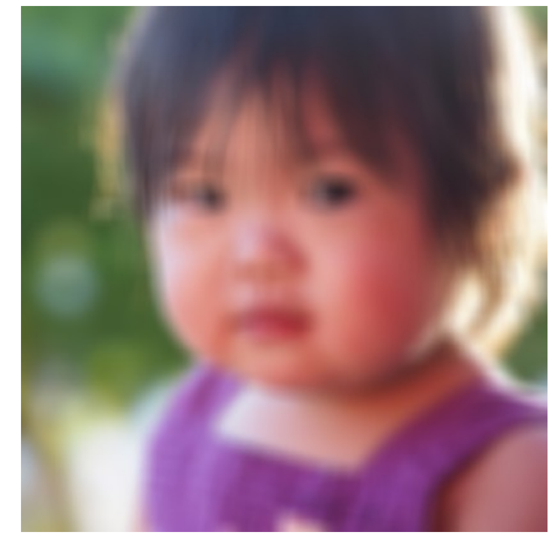
Option 2:

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

Convolution Theorem

Spatial Domain

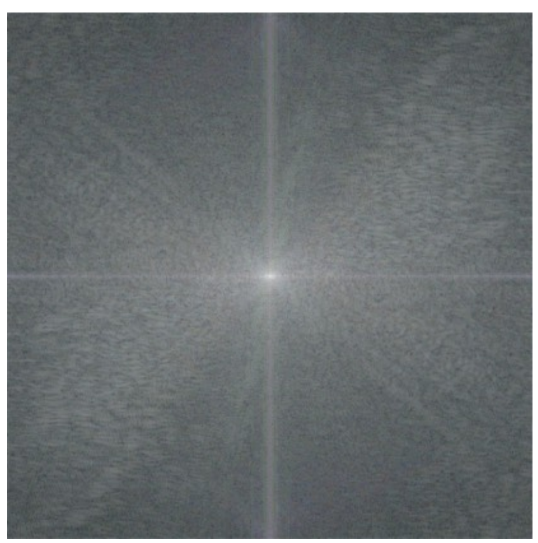
$$* \ \frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix} =$$



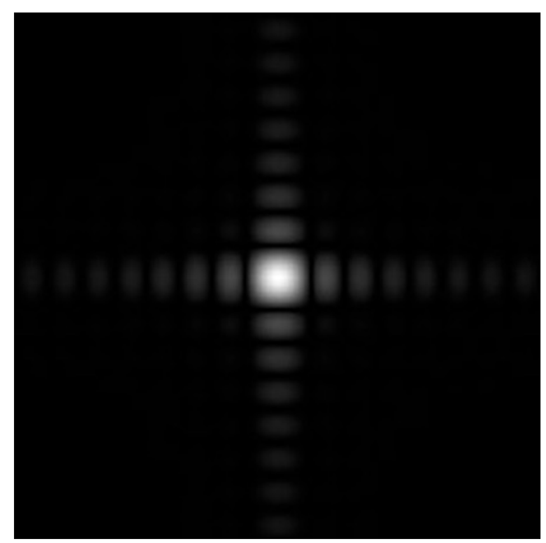
Fourier Transform

Inv. Fourier Transform

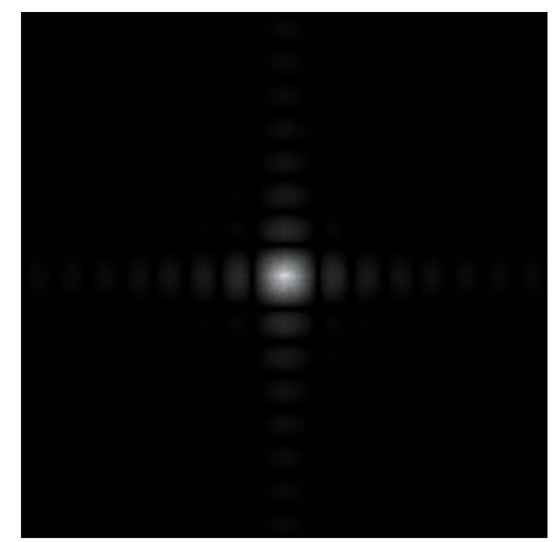
Frequency Domain



$\times$



$=$



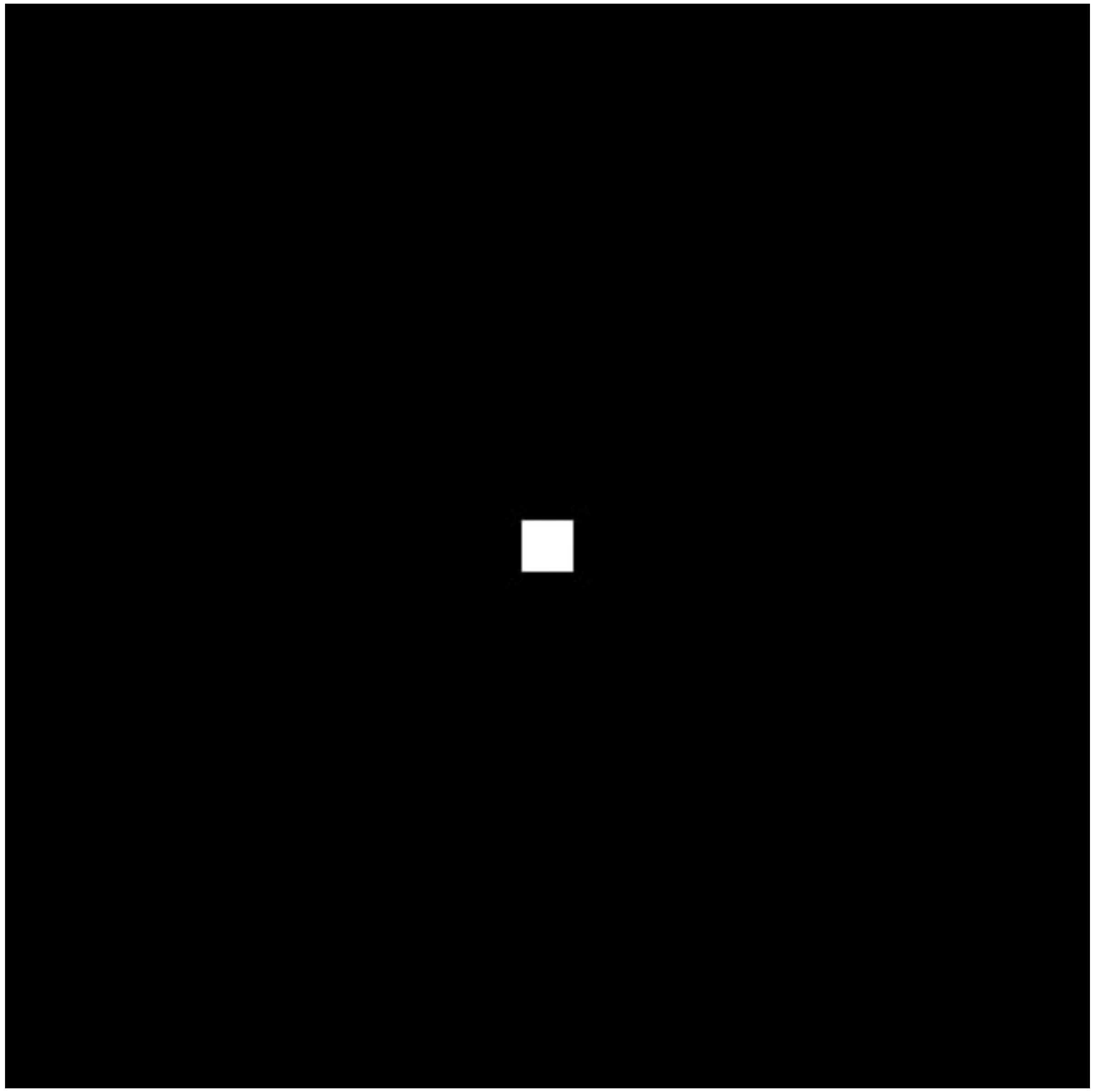
Box Filter 就叫做“低通滤波器”

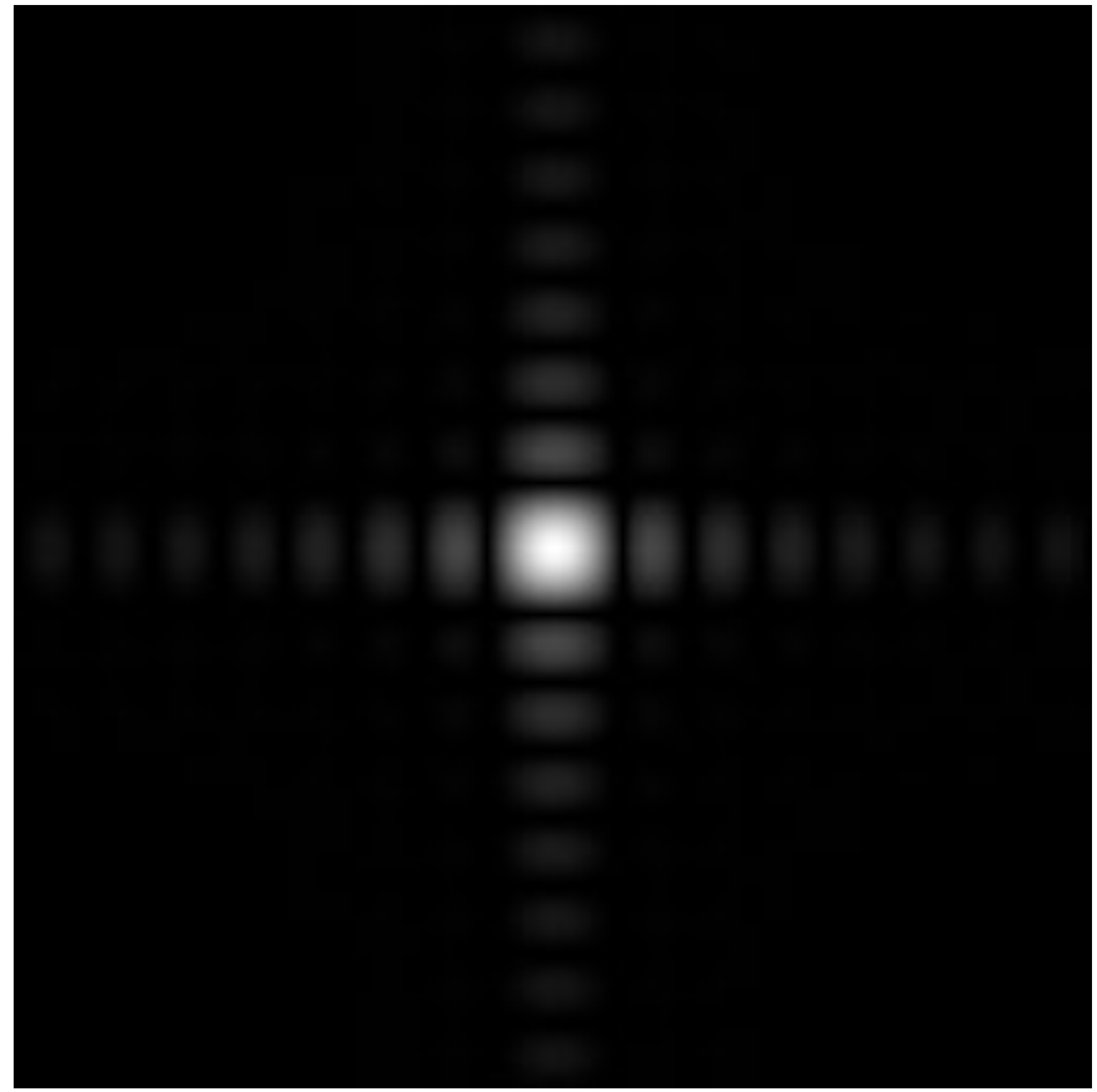
$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

必须做归一化操作，
否则图像亮度变大

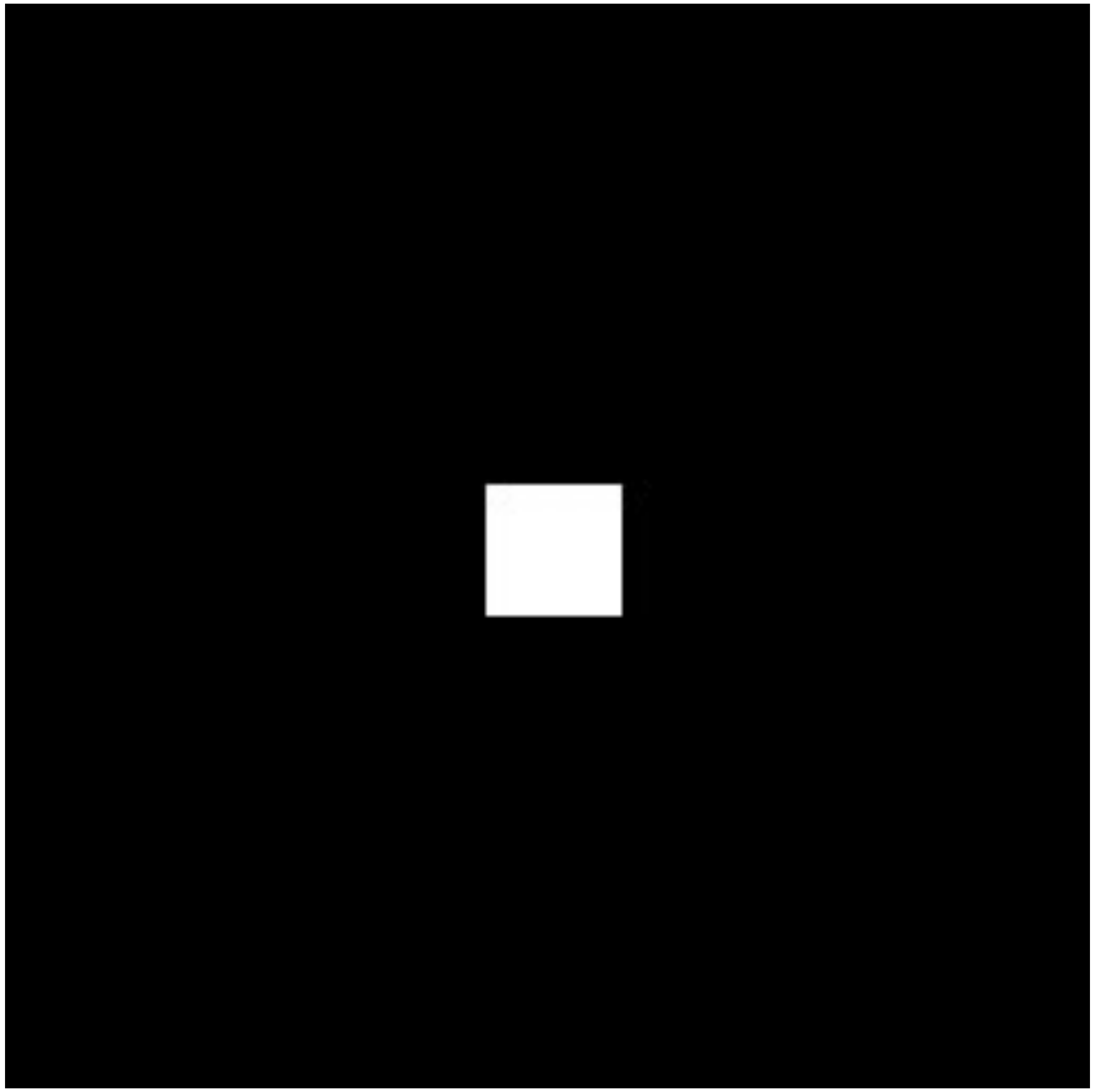
Example: 3x3 box filter

Box Function = “Low Pass” Filter

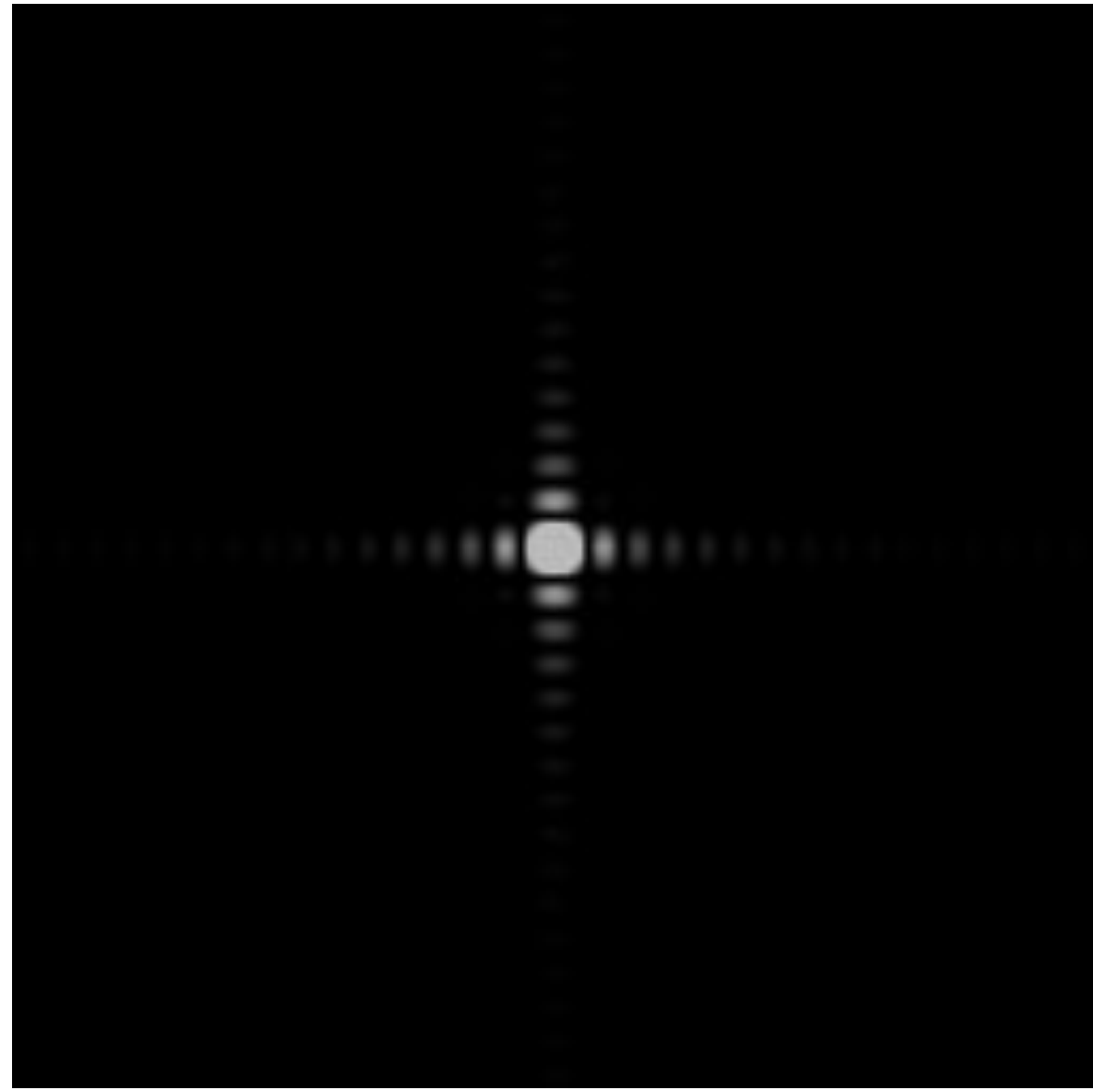




Wider Filter Kernel = Lower Frequencies



越大的box越模糊



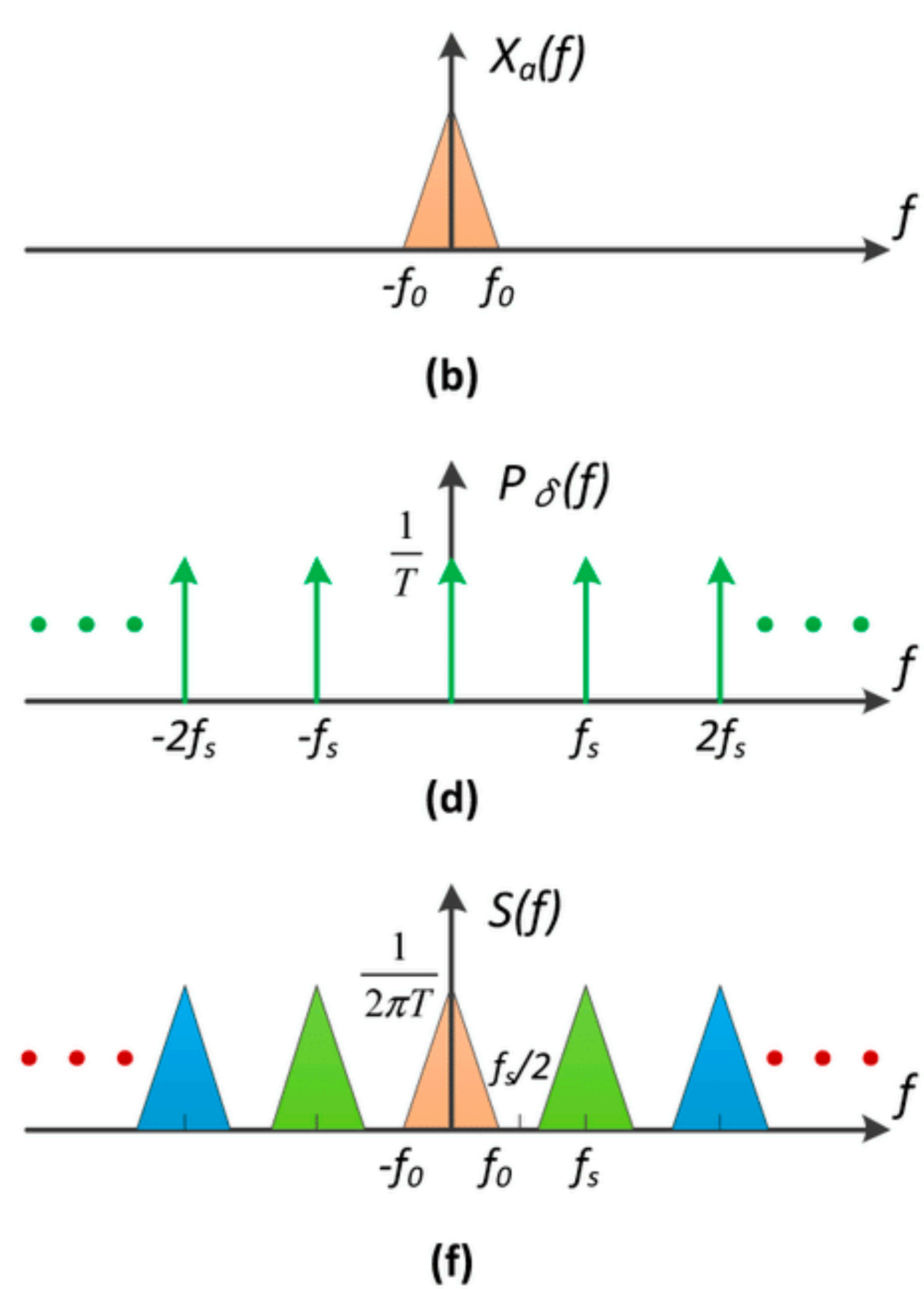
频域通过的低频区域就变小

Sampling = Repeating Frequency Contents

采样：重复频率上的内容

Sampling = Repeating Frequency Contents

冲击函数

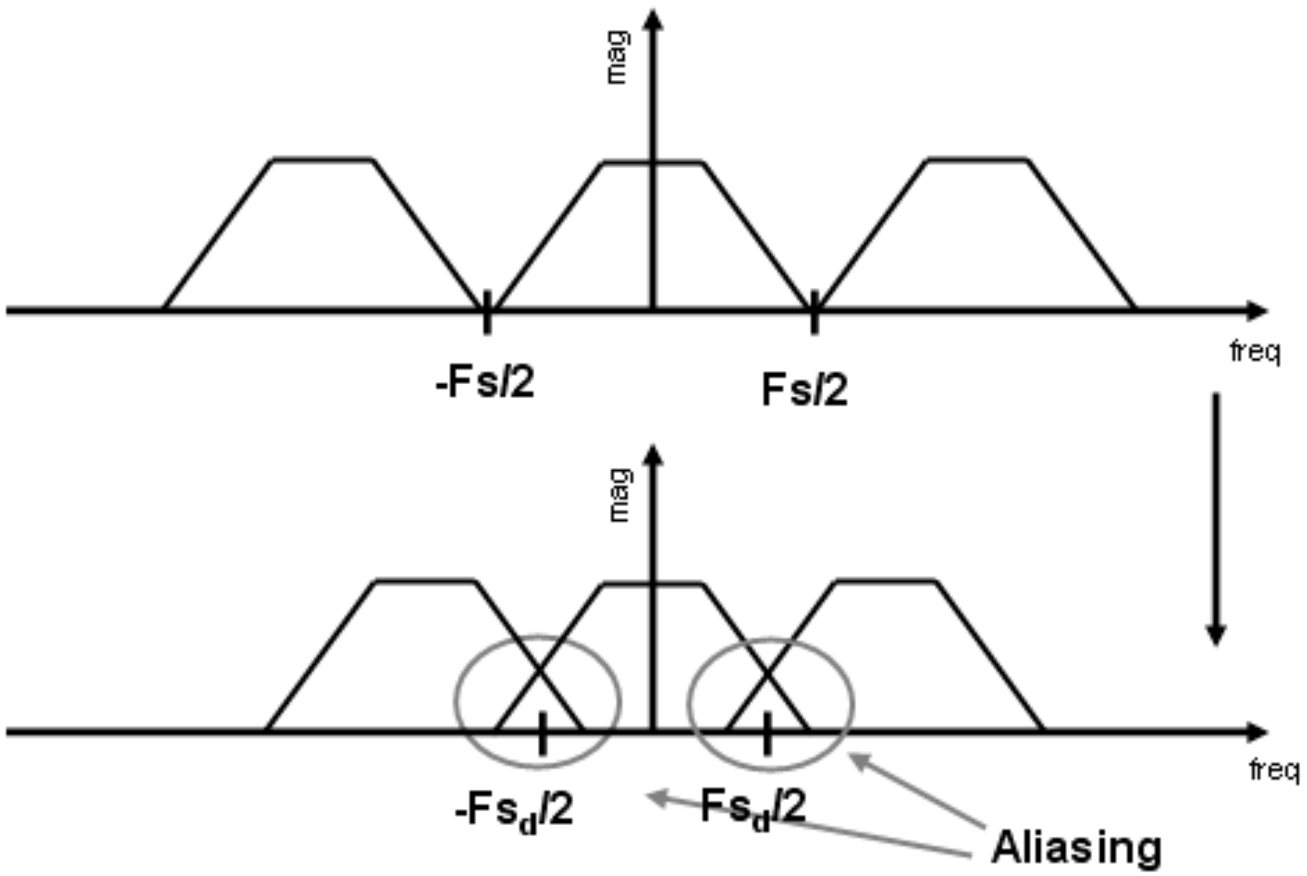


https://www.researchgate.net/figure/The-evolution-of-sampling-theorem-a-The-time-domain-of-the-band-limited-signal-and-b_fig5_301556095

Aliasing = Mixed Frequency Contents

Dense sampling:

Sparse sampling:



Antialiasing

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

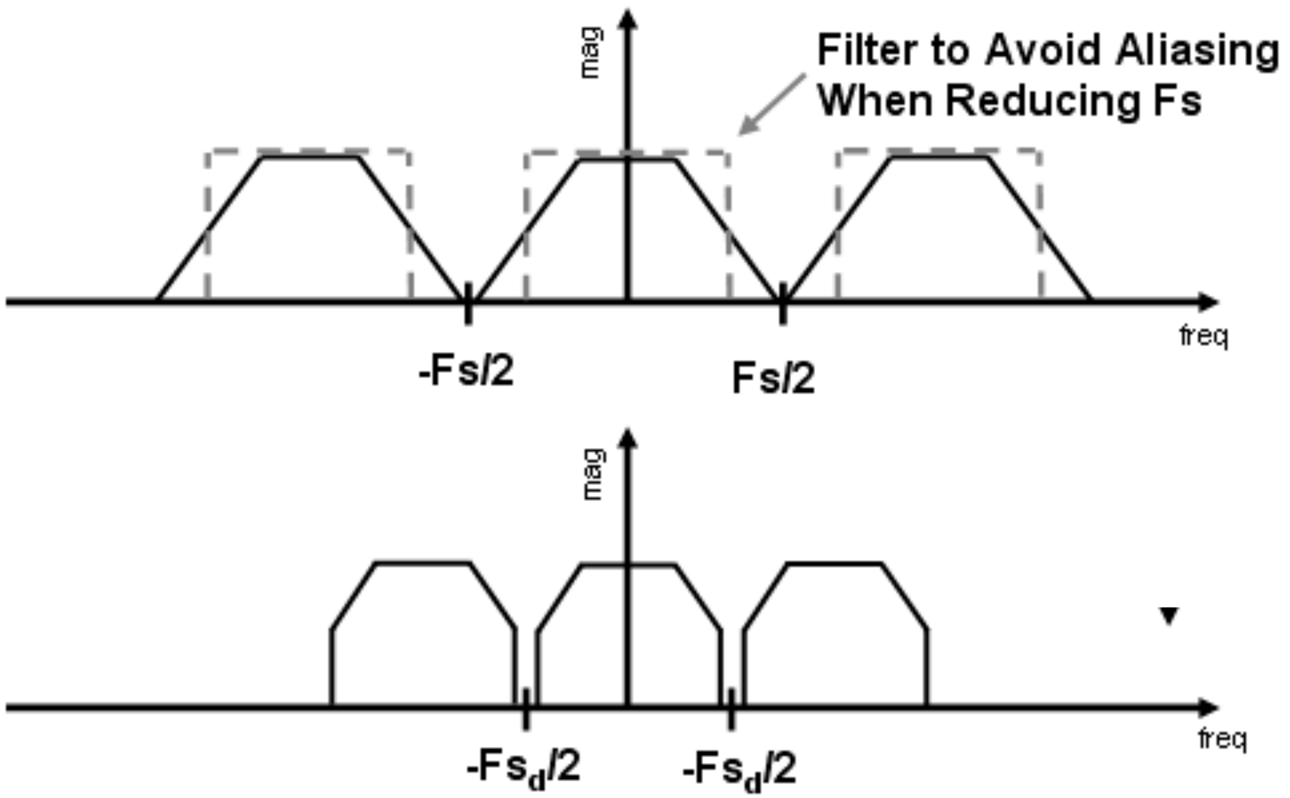
- Making Fourier contents "narrower" before repeating
- i.e. Filtering out high frequencies before sampling

Antialiasing = Limiting, then repeating

Filtering

砍掉高频信号

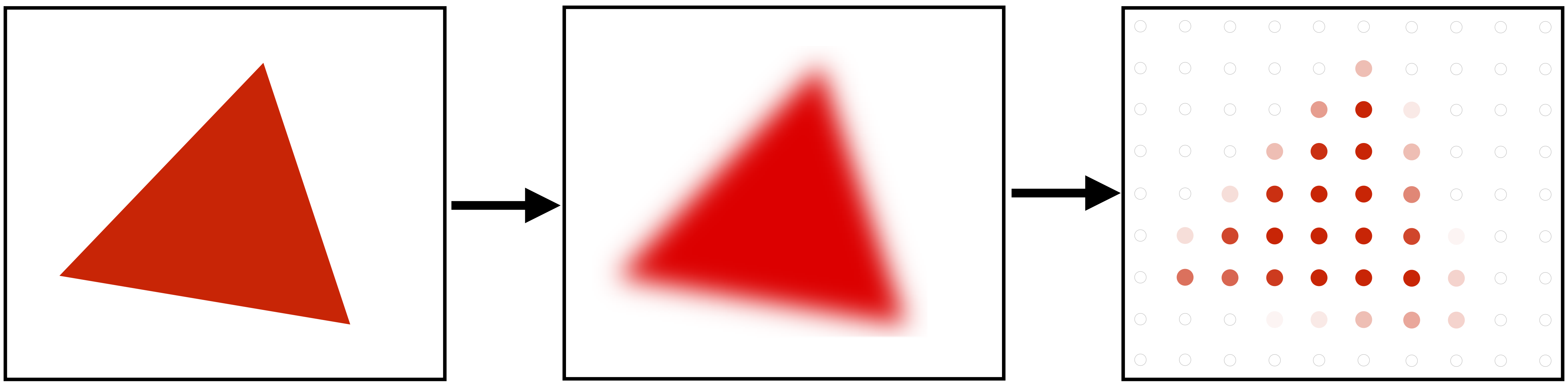
Then sparse sampling



Regular Sampling

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

Antialiased Sampling



Pre-Filter

(remove frequencies above Nyquist)

那么如何做模糊操作呢?

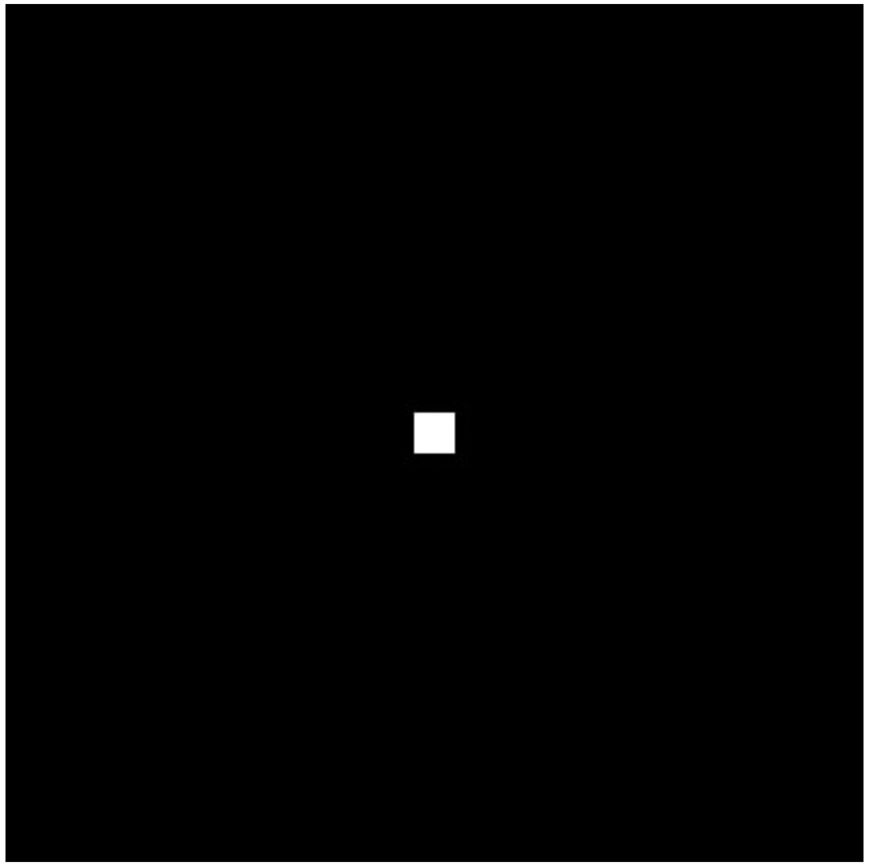
Sample

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

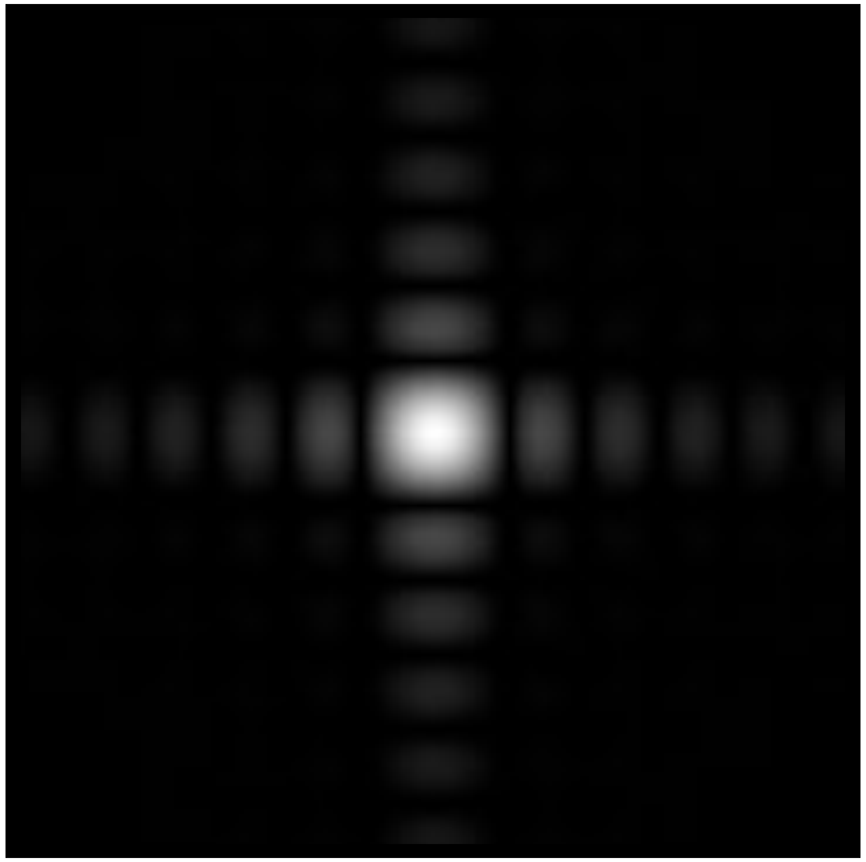
A Practical Pre-Filter

做模糊操作的办法：

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



Spatial Domain



Frequency Domain

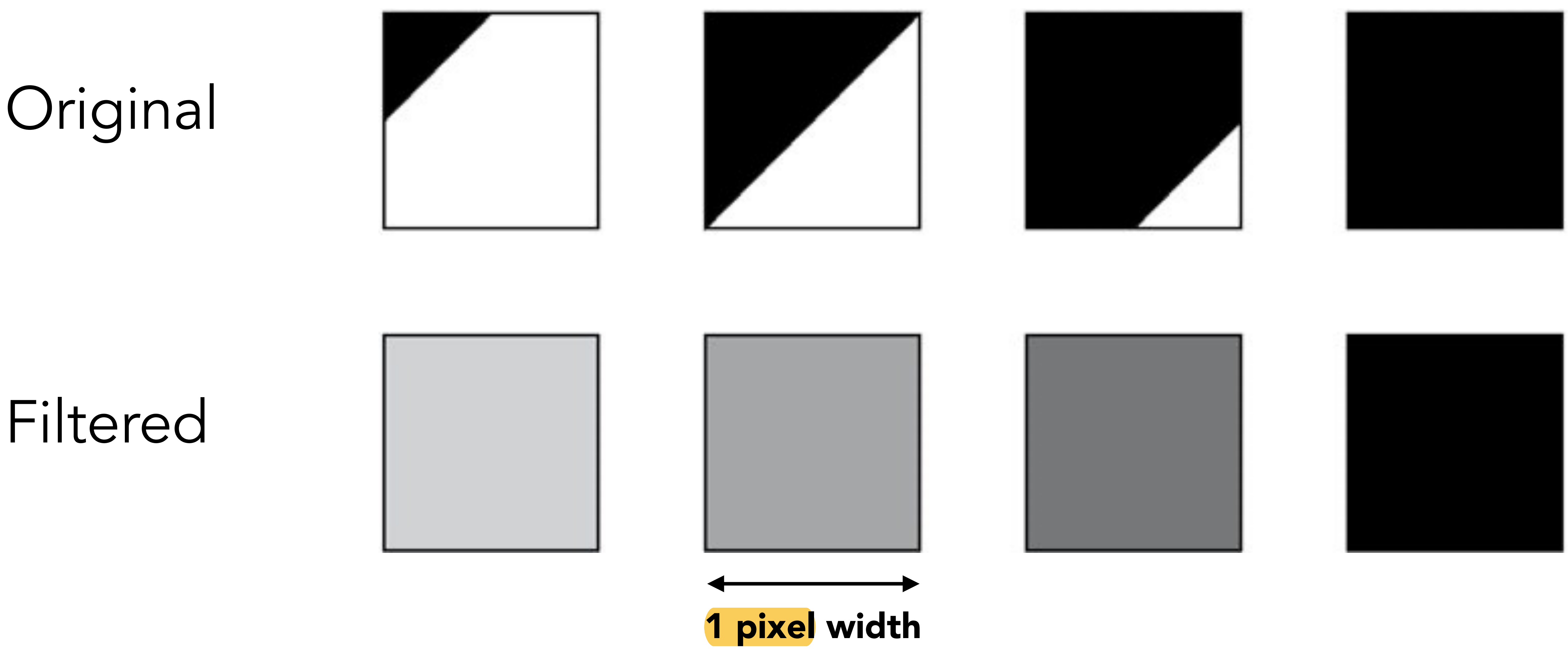
Antialiasing By Averaging Values in Pixel Area

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.

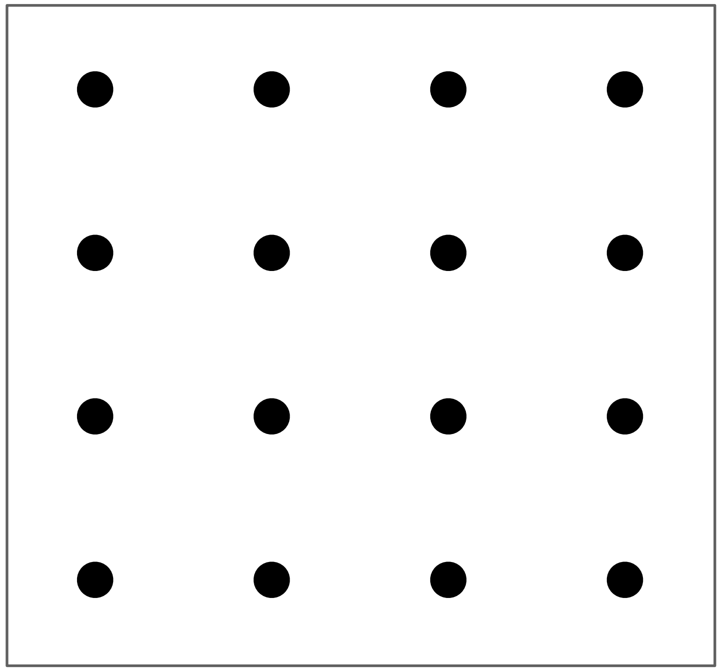


Antialiasing By Supersampling (MSAA)

近似方法：**Multi-Sample Antialiasing，**
并不能完全解决反走样的问题

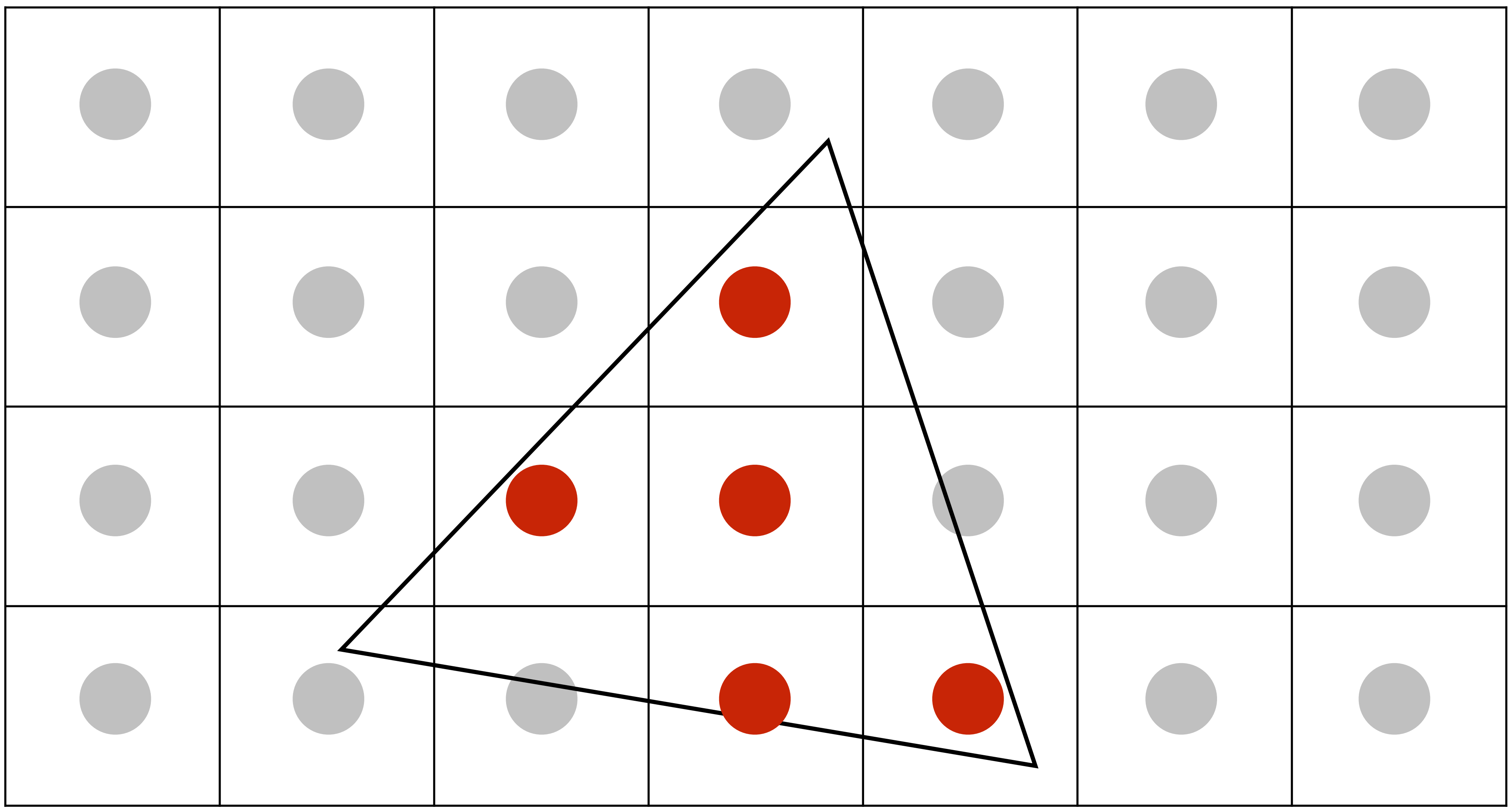
Supersampling

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



4x4 supersampling

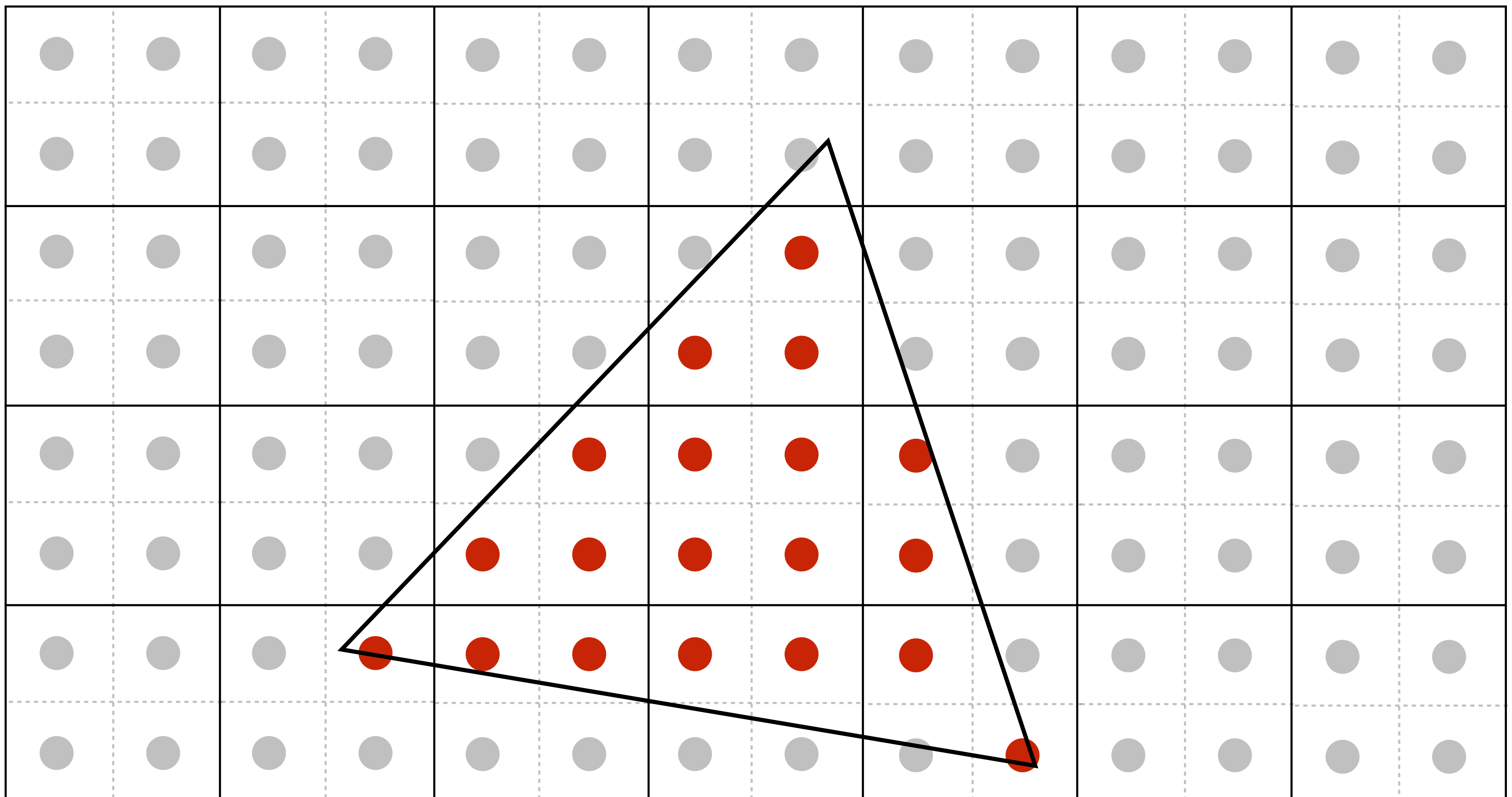
Point Sampling: One Sample Per Pixel



Supersampling: Step 1

Take NxN samples in each pixel.

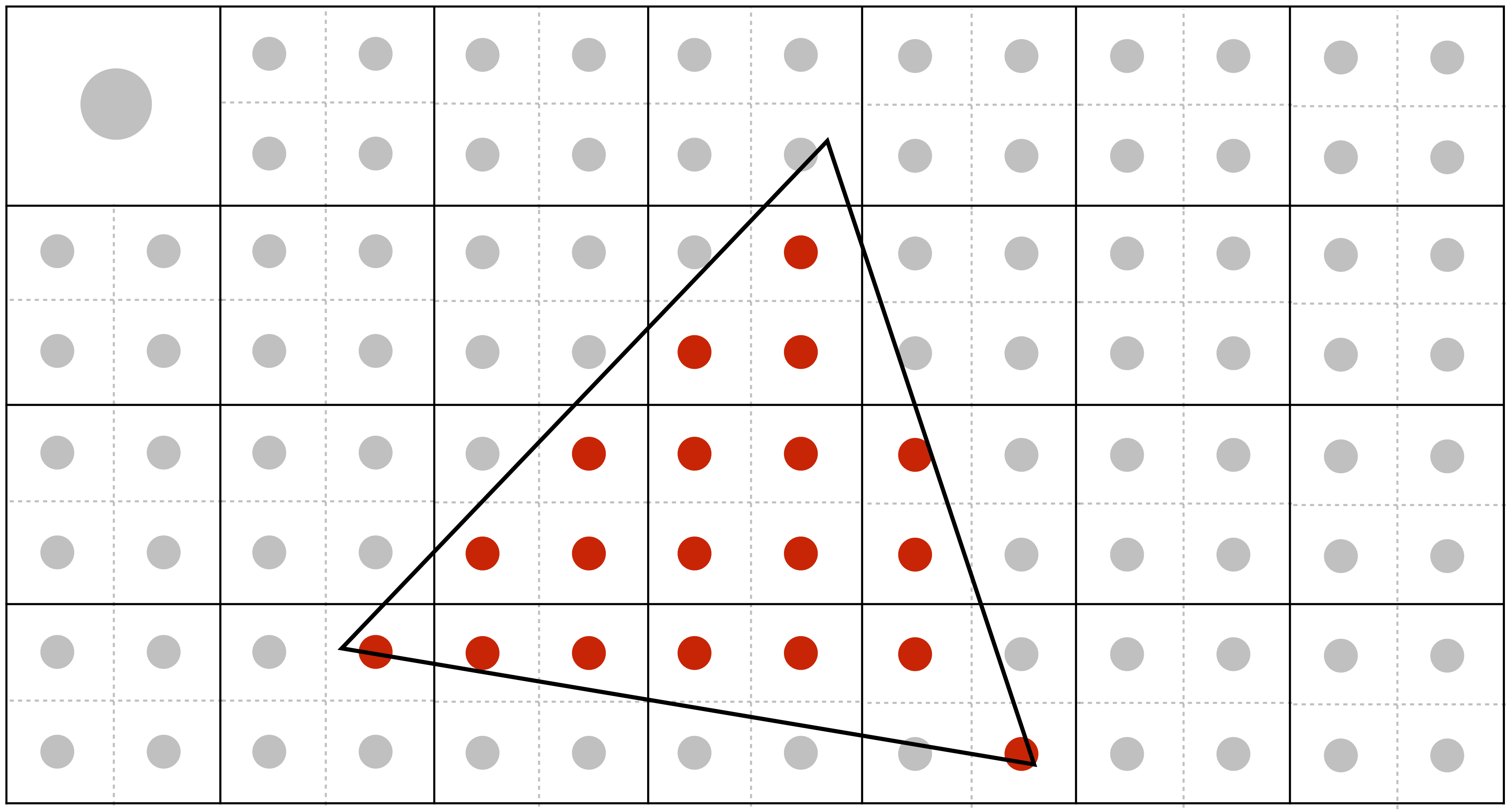
在像素内部多加采样点



2x2 supersampling

Supersampling: Step 2

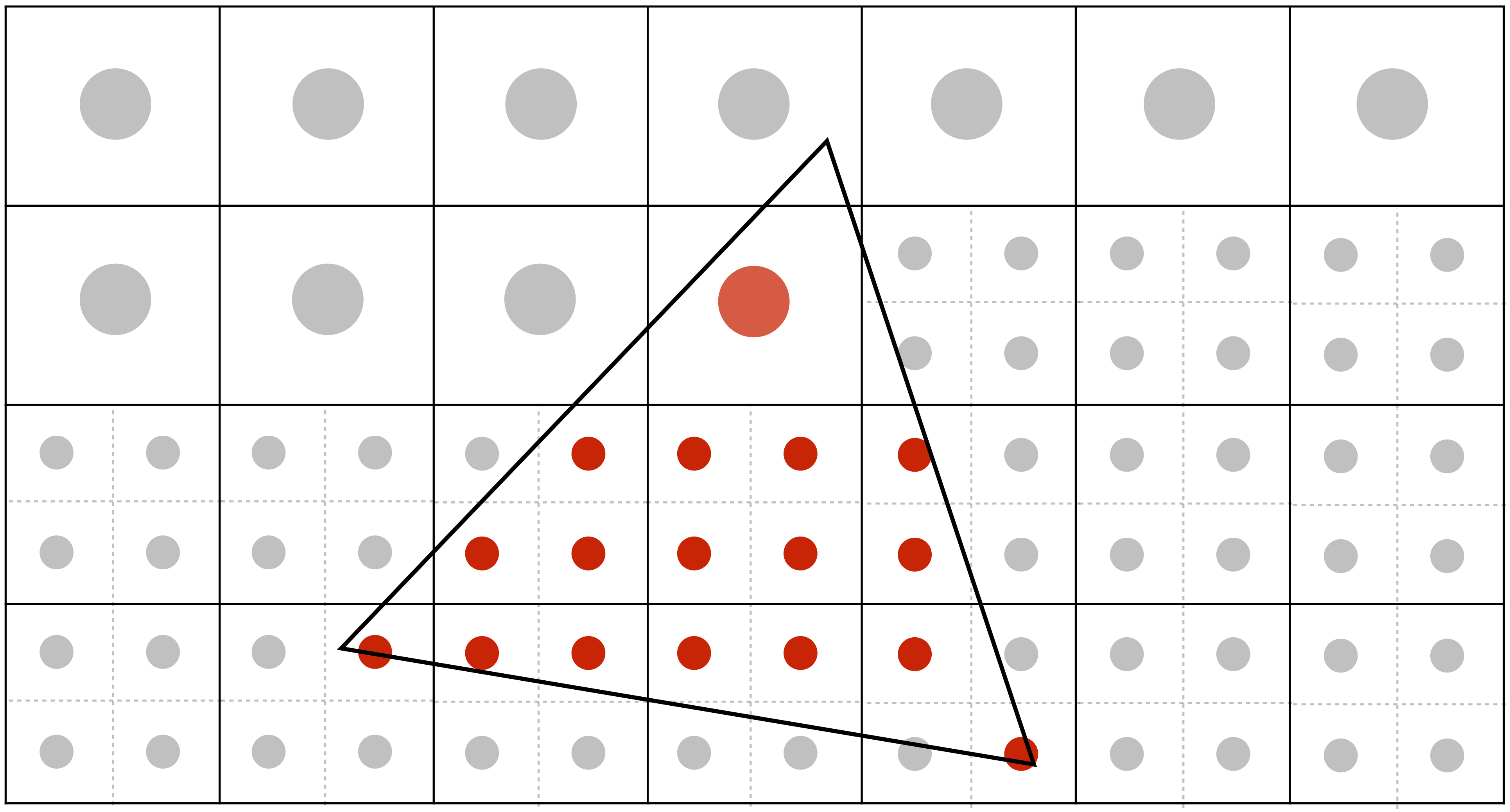
Average the NxN samples “inside” each pixel.



Averaging down

Supersampling: Step 2

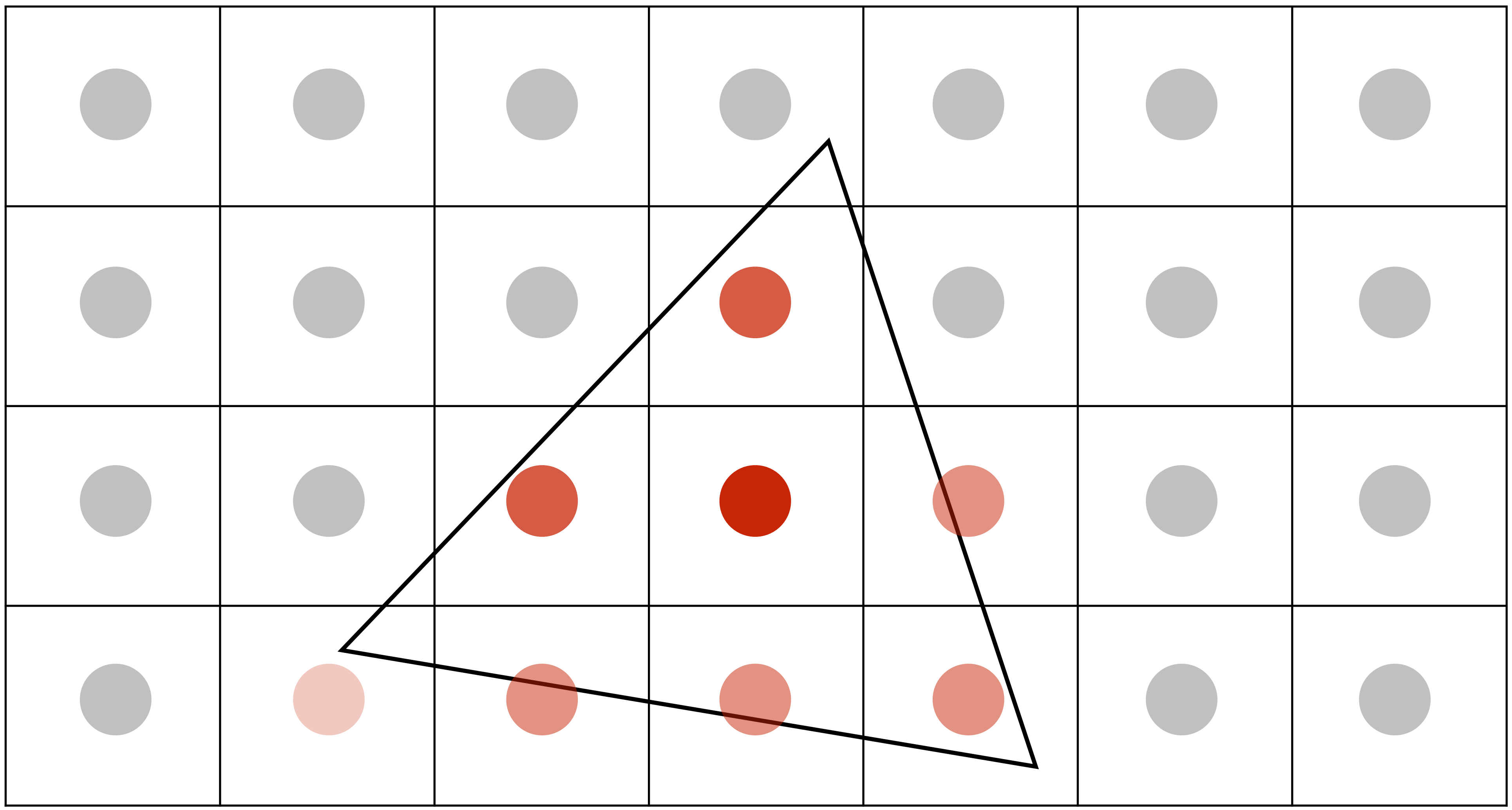
Average the NxN samples "inside" each pixel.



Averaging down

Supersampling: Step 2

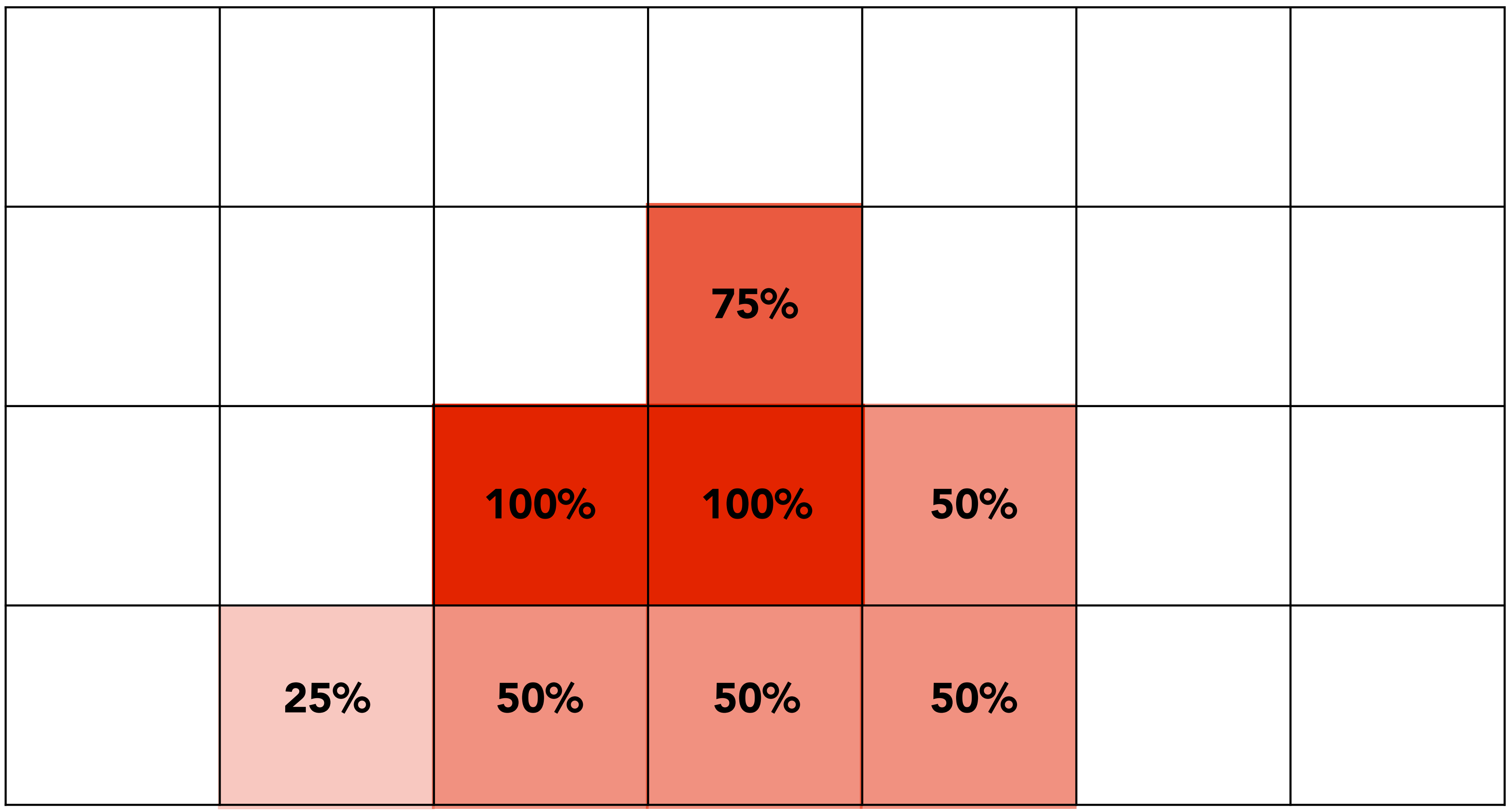
Average the NxN samples "inside" each pixel.



Supersampling: Result

This is the corresponding signal emitted by the display

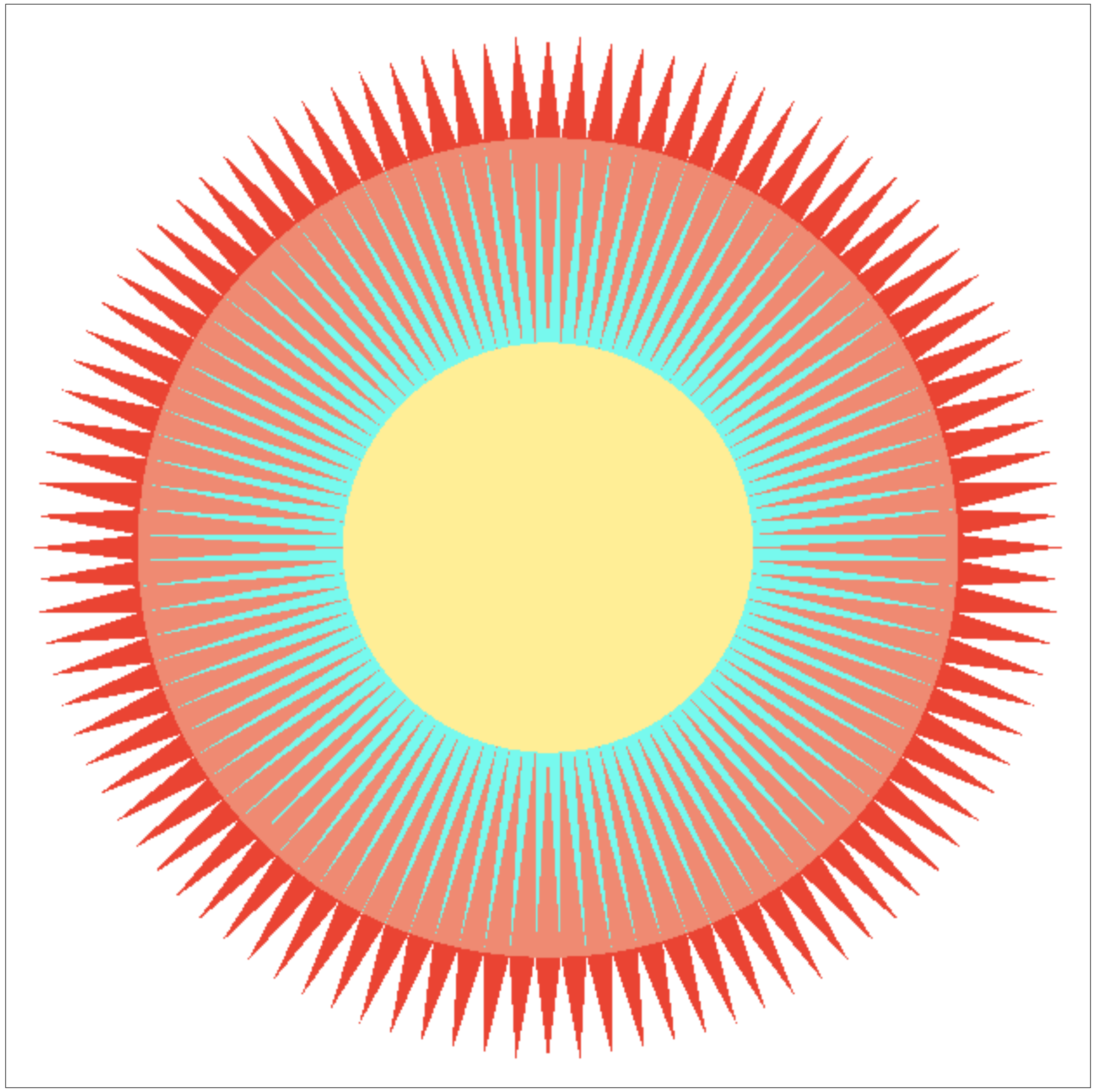
MSAA并没有从根本上解决反走样（只能通过提高分辨率解决），但提供了良好的近似

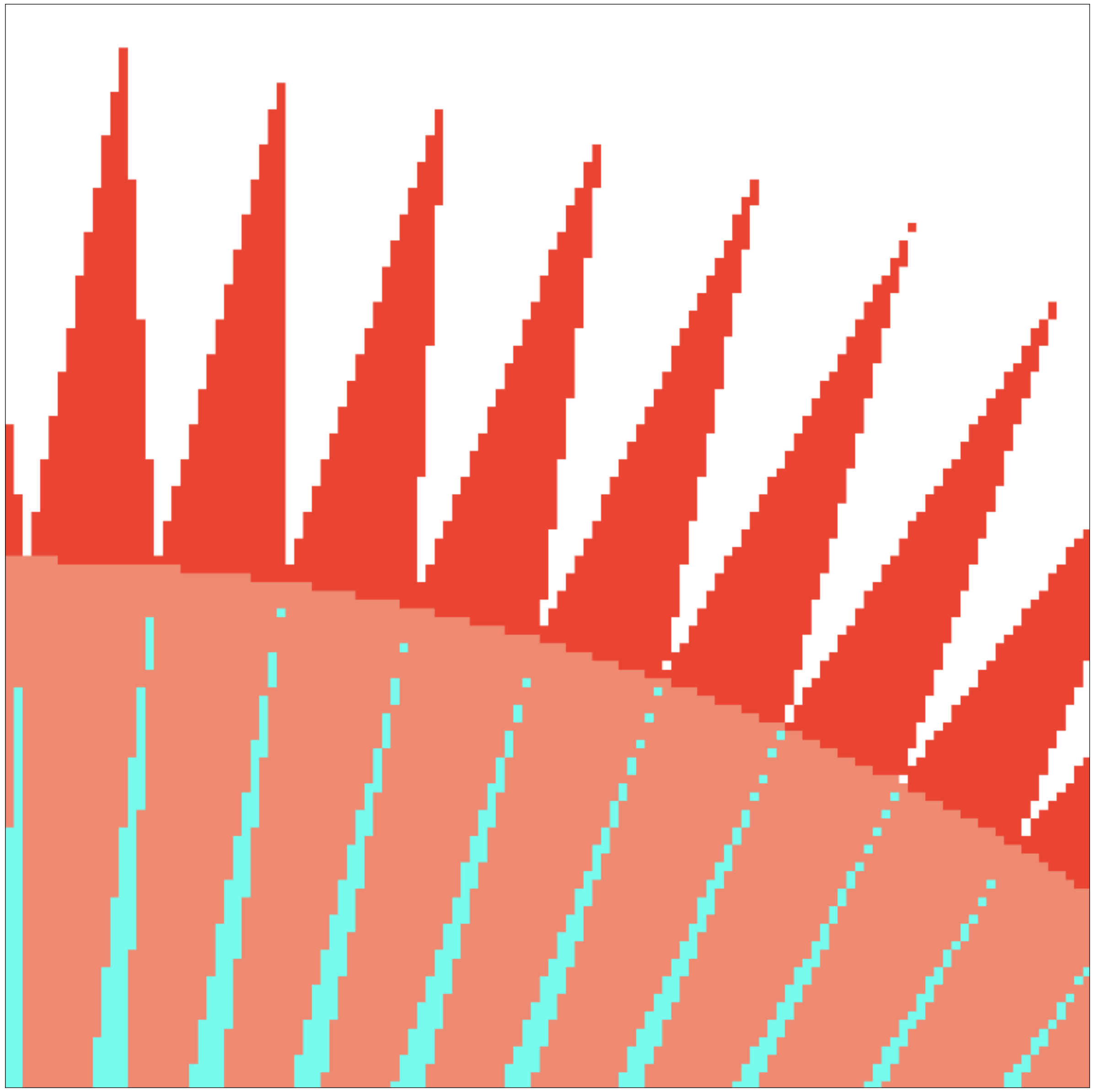


模糊操作做完了

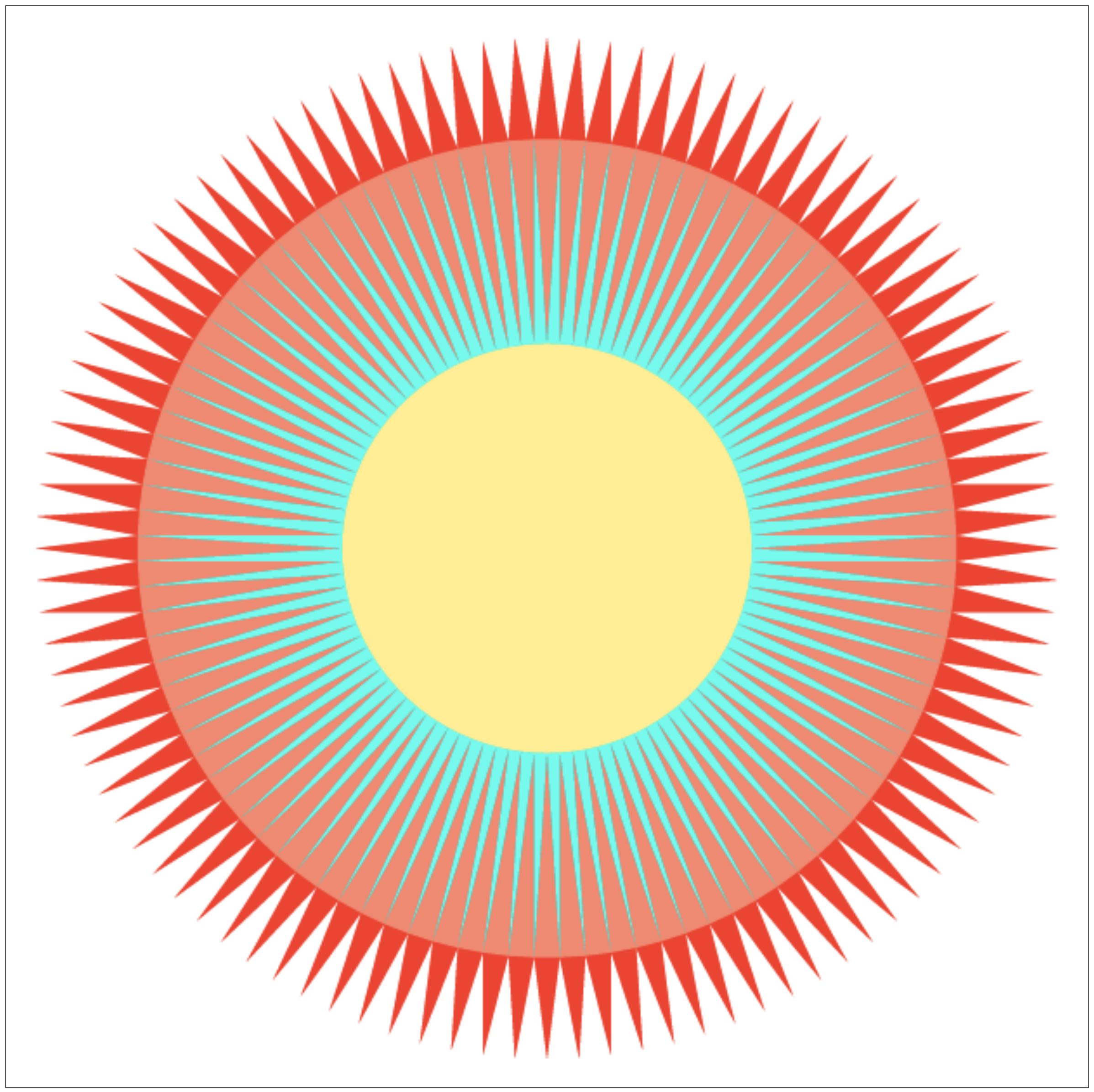
采样就很直接了，就是得到的这种颜色

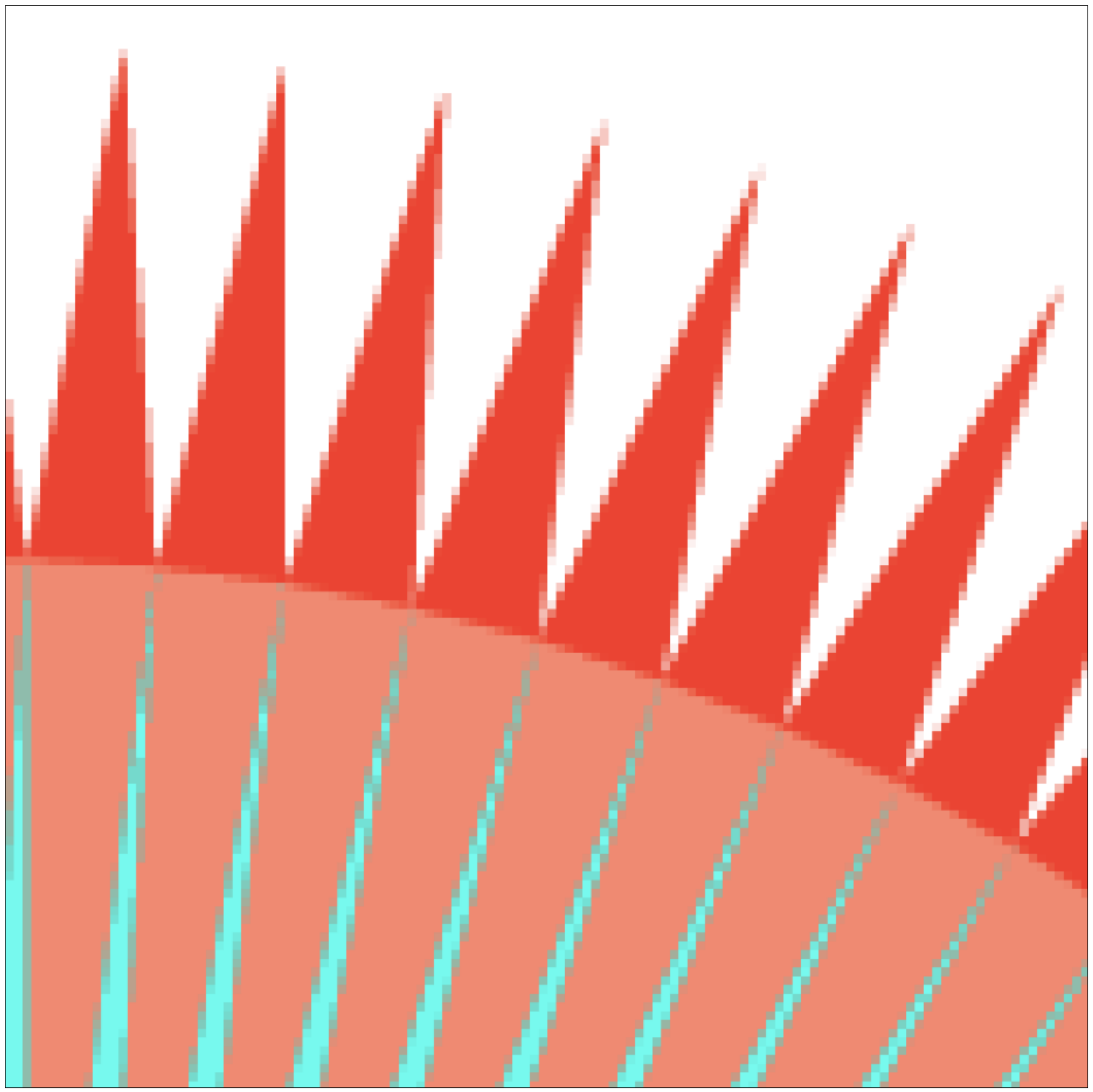
Point Sampling





4x4 Supersampling





Antialiasing Today

No free lunch!

- What's the cost of MSAA? 增大了计算量

Milestones (personal idea)

- FXAA (Fast Approximate AA) 快速近似抗锯齿：图像后期处理（与采样无关）
- TAA (Temporal AA) 复用上一帧的感知效果

Super resolution / super sampling

超分辨率

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

猜测：深度学习，补全细节

Thank you!