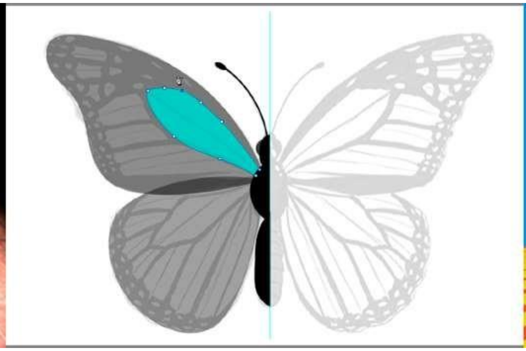
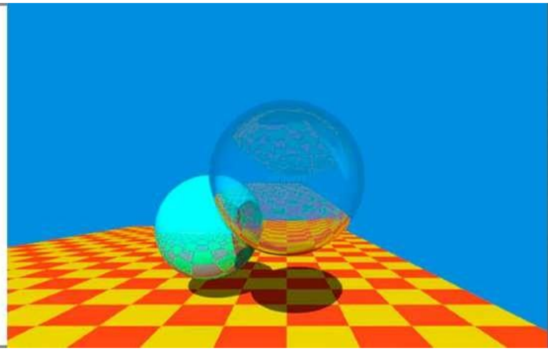
Computer Graphics

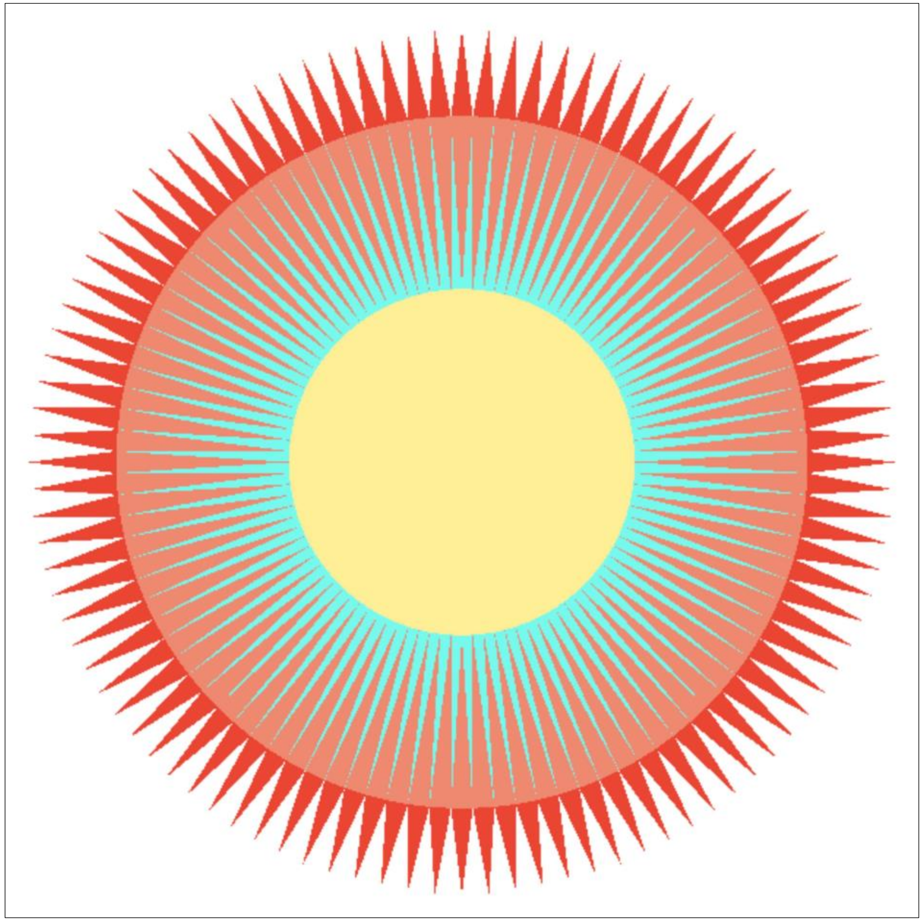
Rasterization 2
(Antialiasing and Z-Buffering)

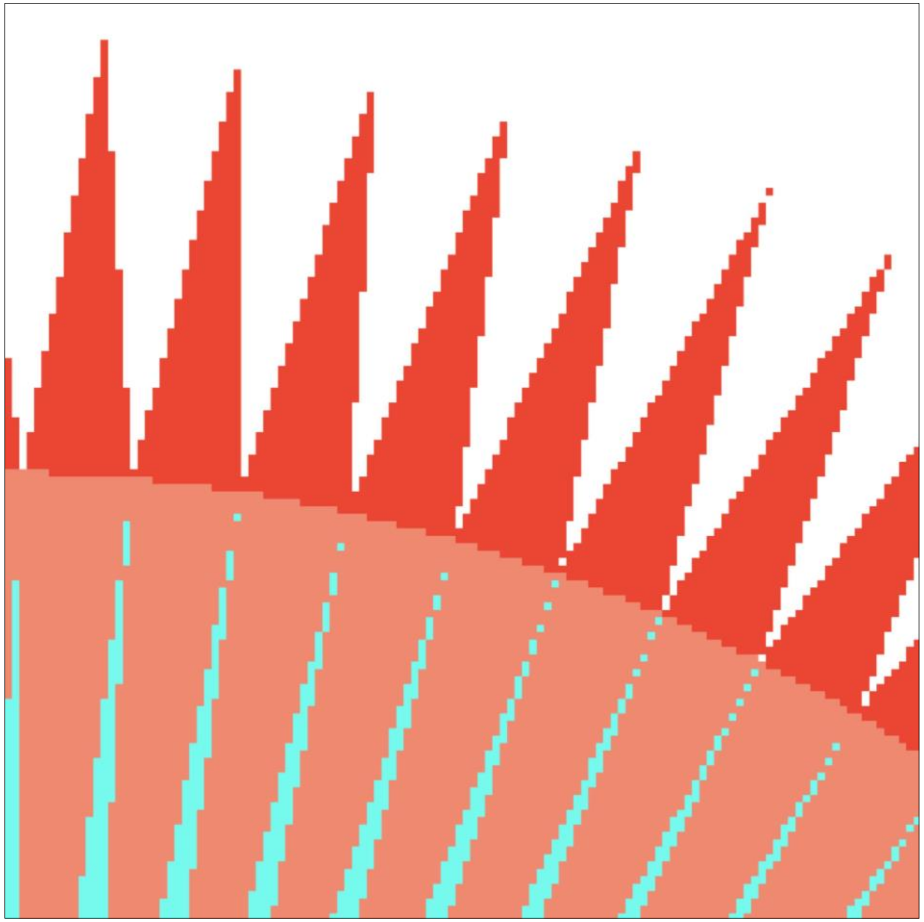
   

Today

- Antialiasing
  - Sampling theory
  - Antialiasing in practice

Aliasing (走样)





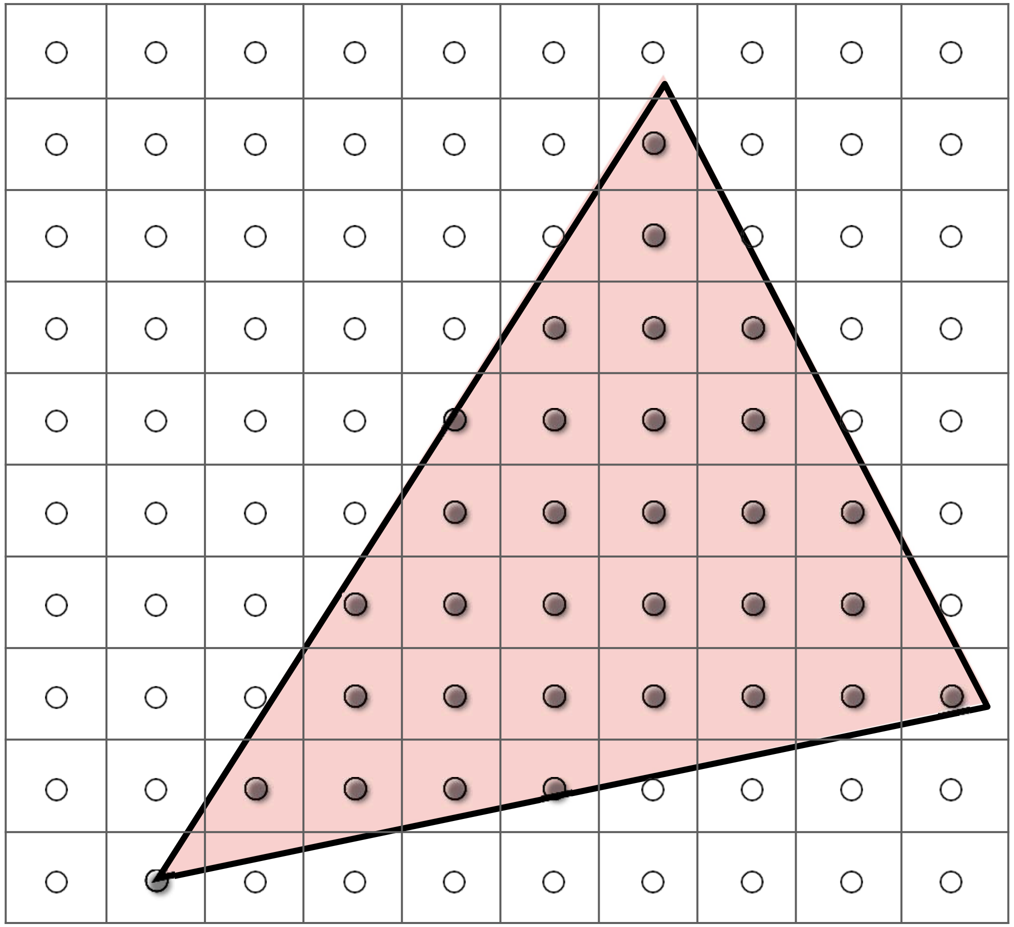
Is this the best we can do?

Slide courtesy of Prof. Ren Ng, UC Berkeley

Sampling is Ubiquitous in Computer Graphics

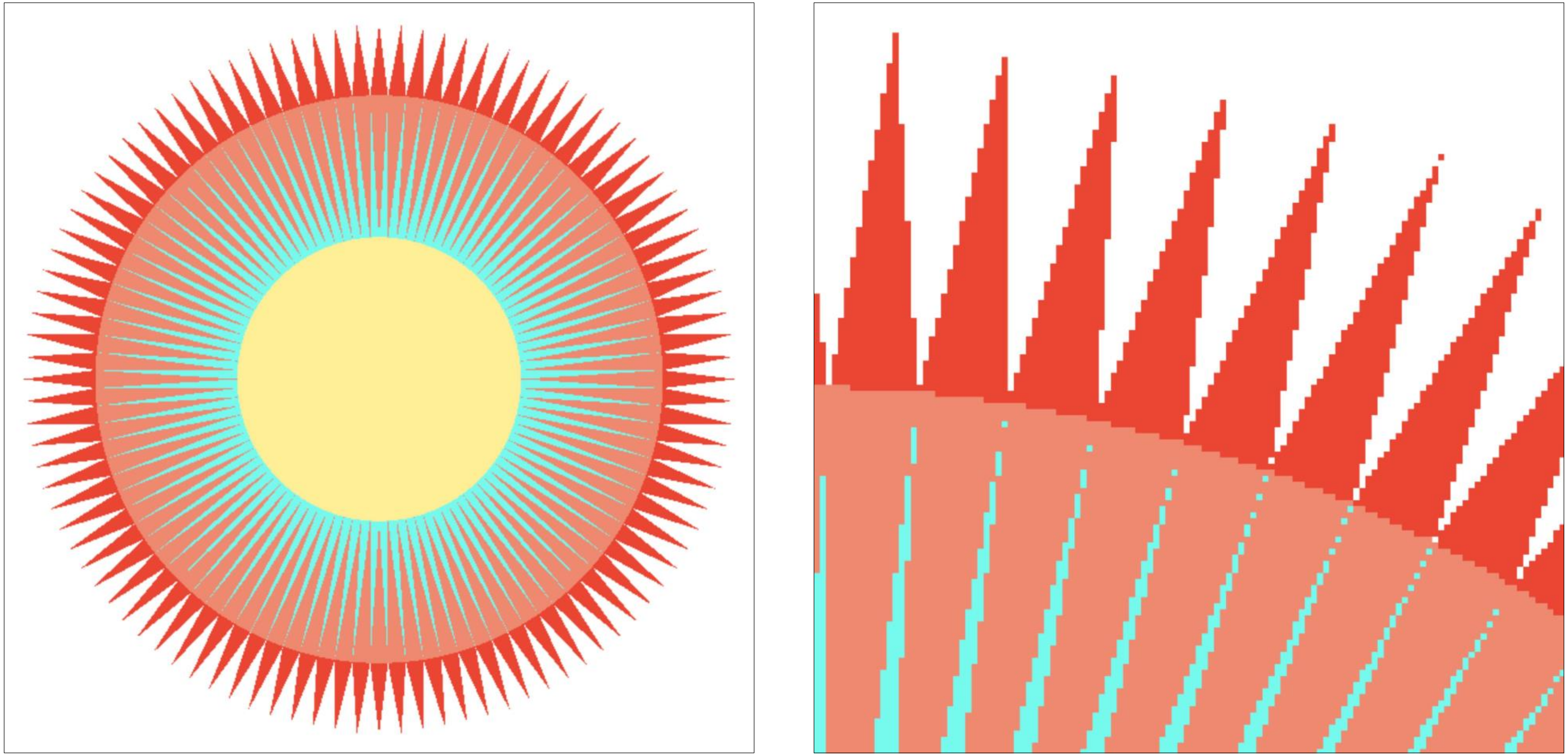
在CG中的采样普遍存在

Rasterization = Sample 2D Positions



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

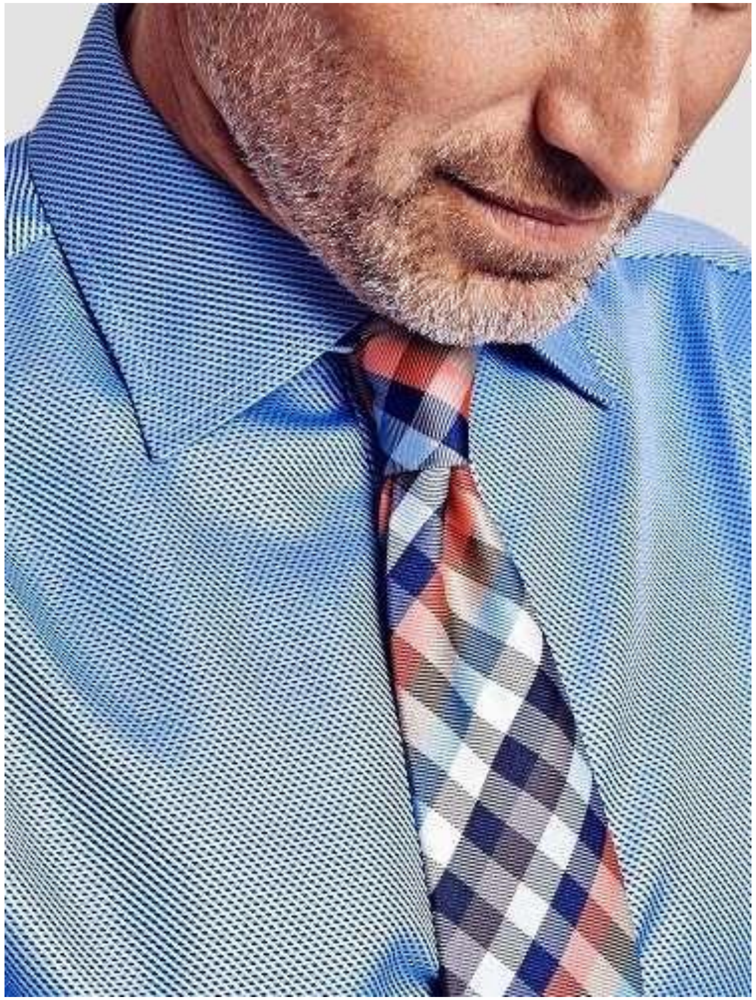
Jaggies (Staircase Pattern)



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

Moiré Patterns in Imaging

[mwɑː]



lystit.com

Skip odd rows and columns

Sampling Artifacts in Computer Graphics

Artifacts due to sampling - “Aliasing”

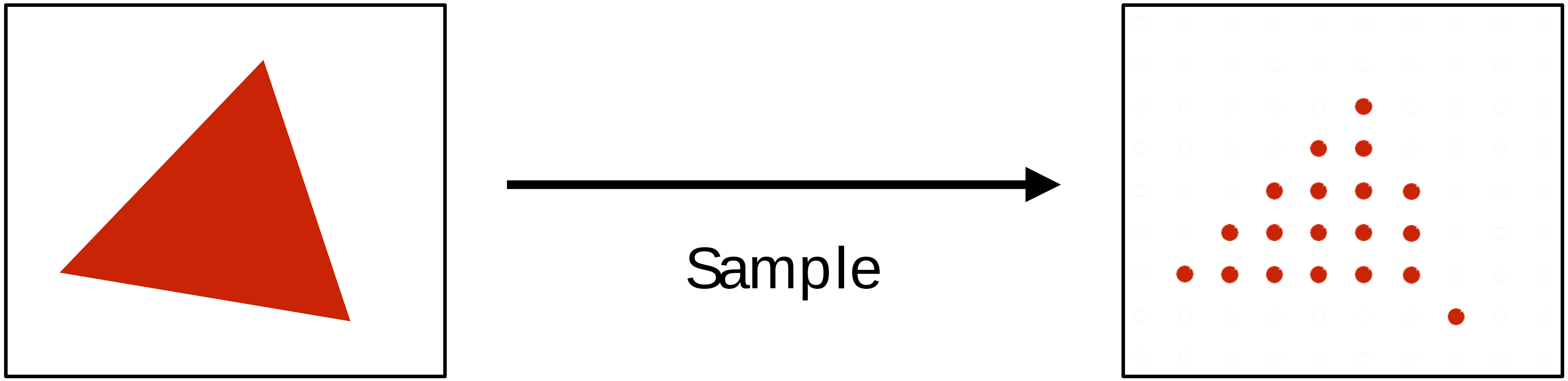
- Jaggies – sampling in space
- Moire – undersampling images
- Wagon wheel effect – sampling in time
- [Many more] …

Behind the Aliasing Artifacts

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

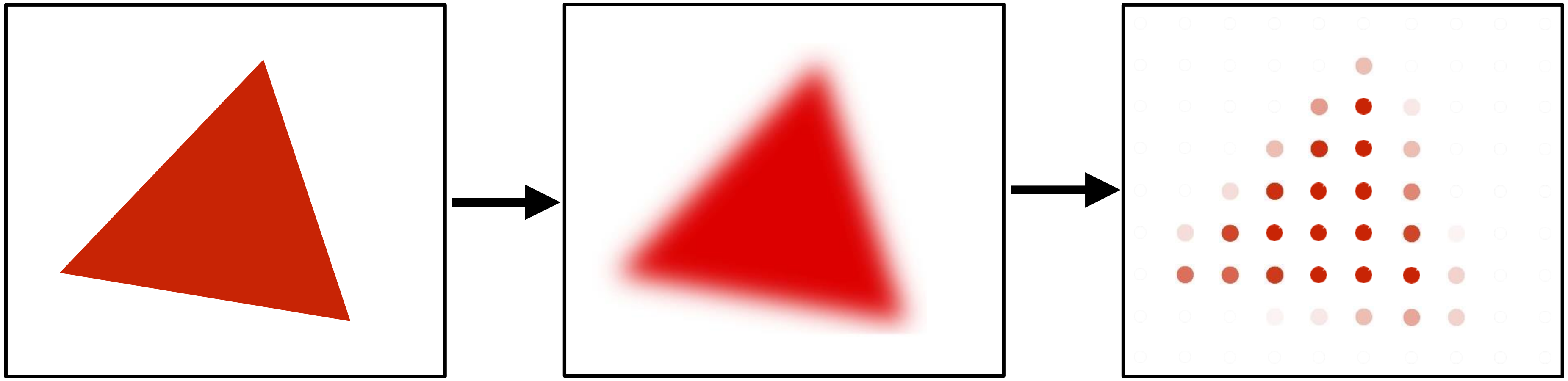
Antialiasing Idea:
Blurring (Pre-Filtering) Before Sampling

Rasterization: Point Sampling in Space



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

Rasterization: Antialiased Sampling



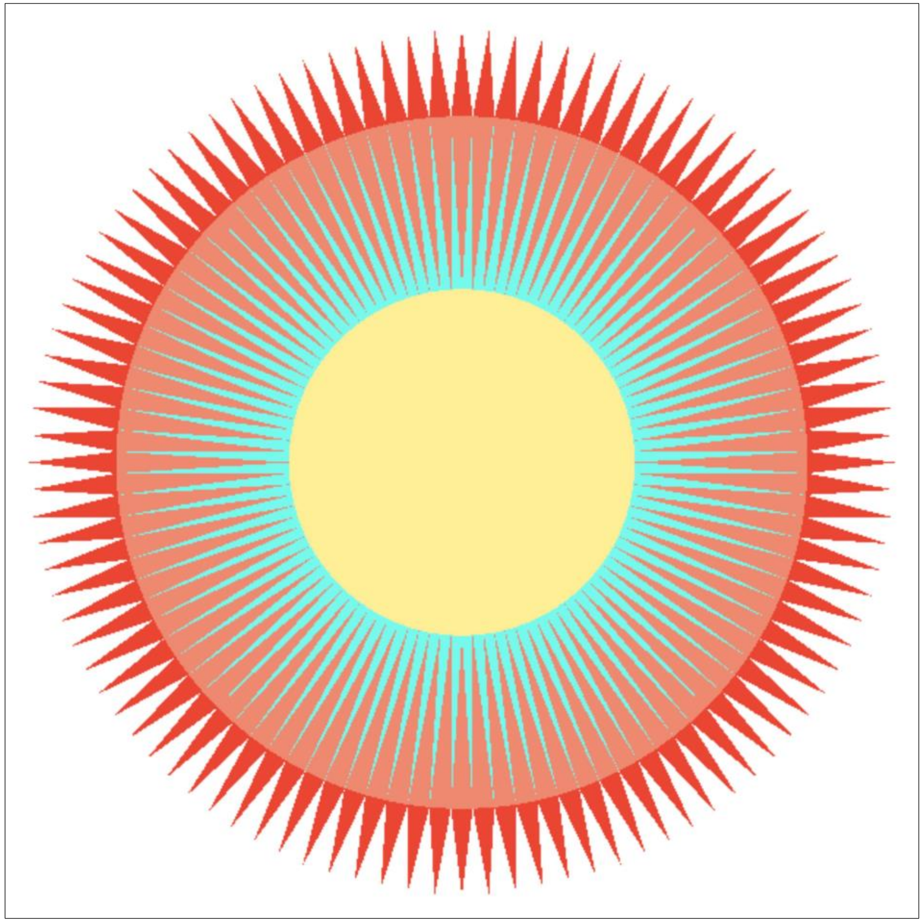
Pre-Filter

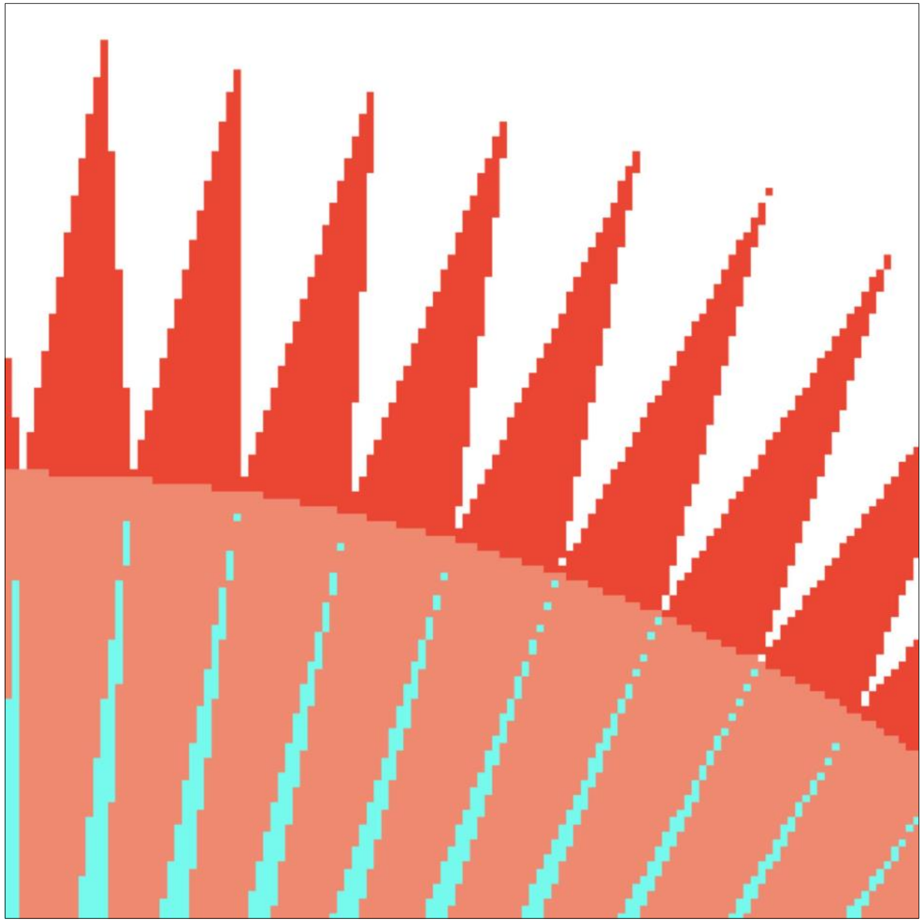
(remove frequencies above Nyquist) (?)

Sample

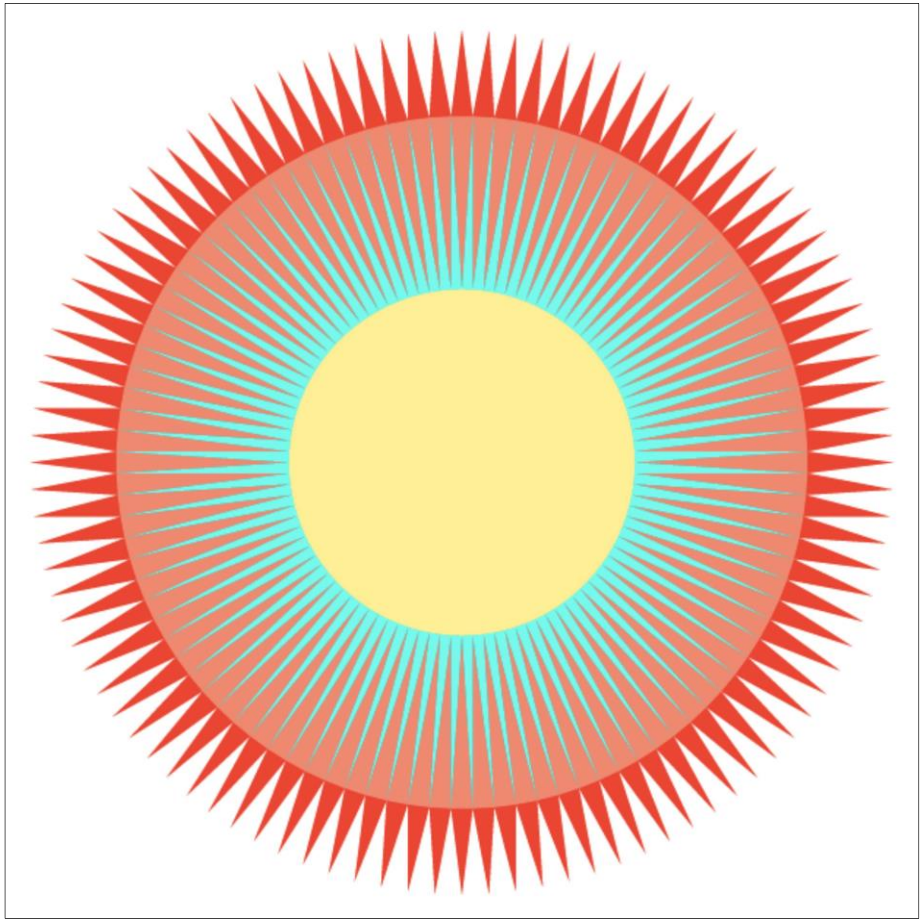
Note antialiased edges in rasterized triangle
where pixel values take intermediate values

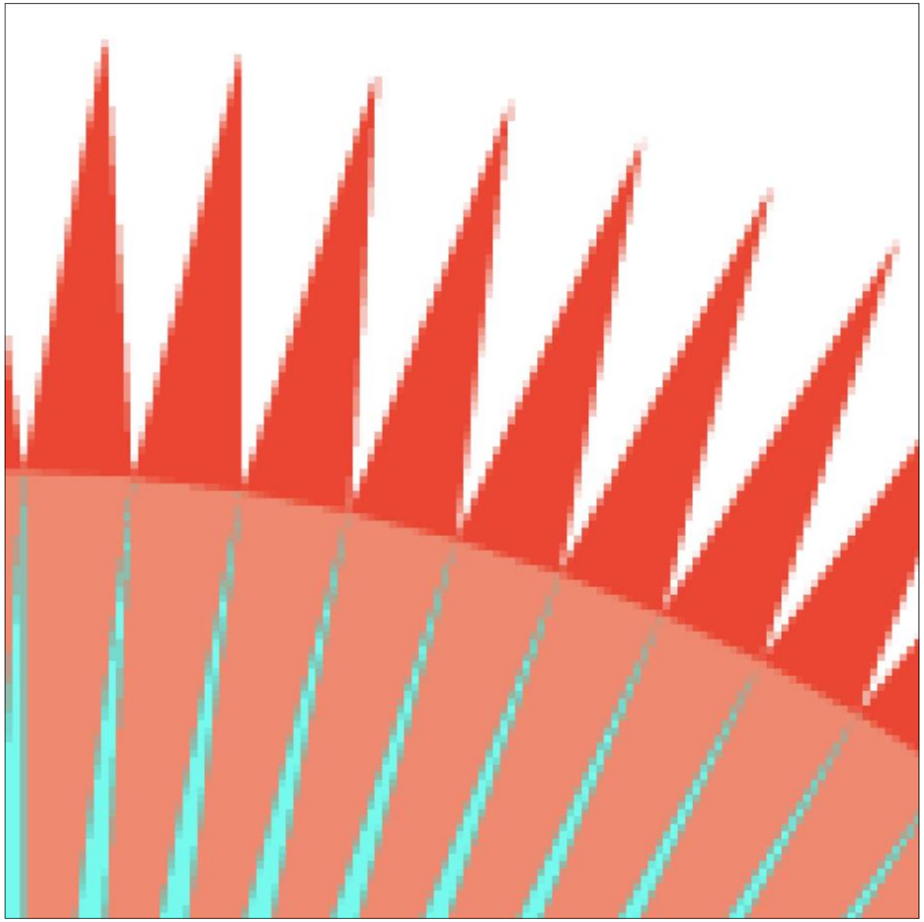
Point Sampling



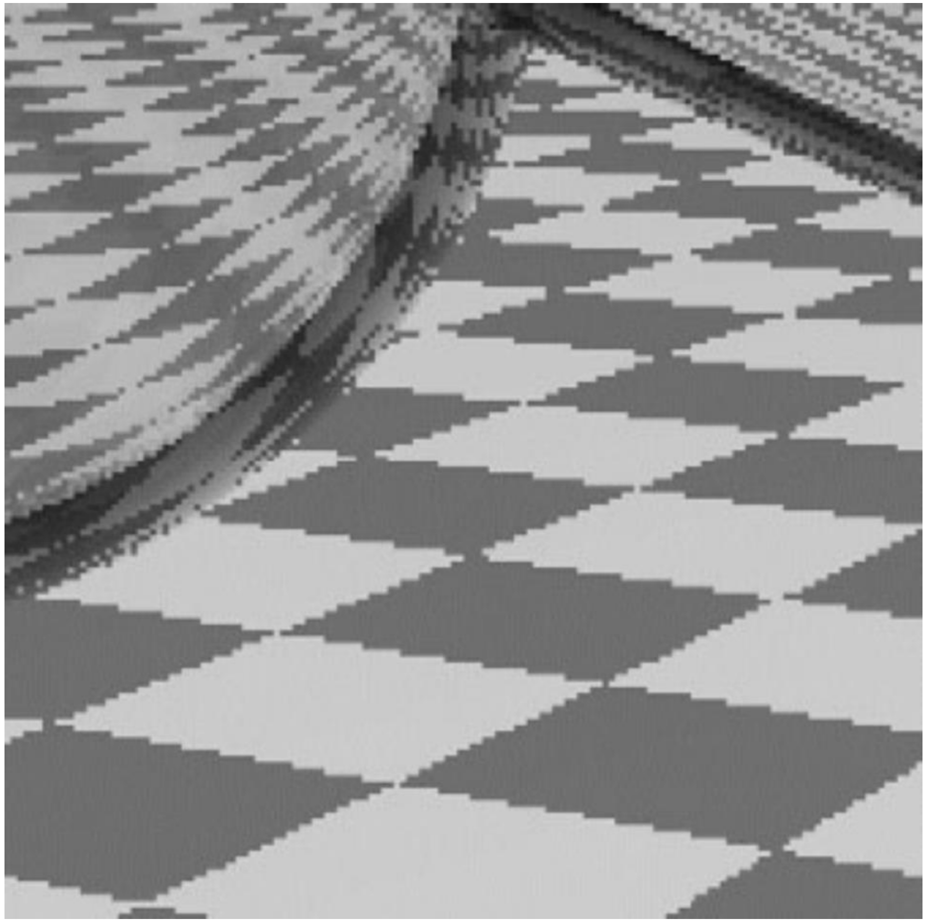


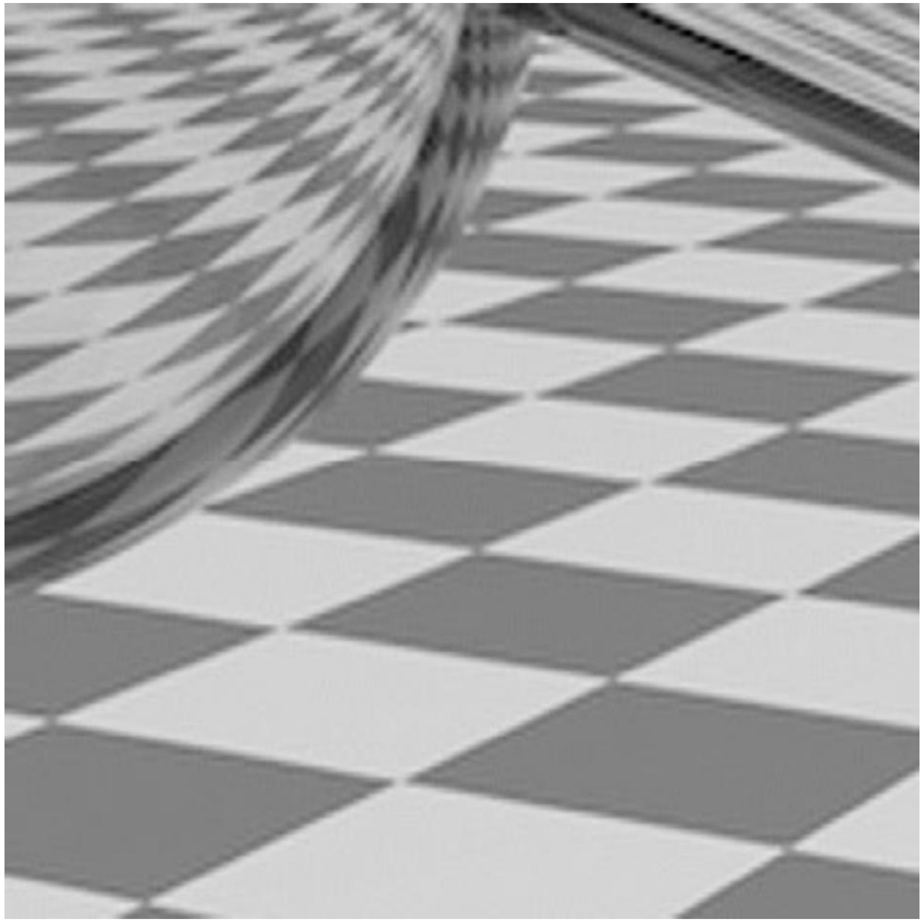
Antialiasing



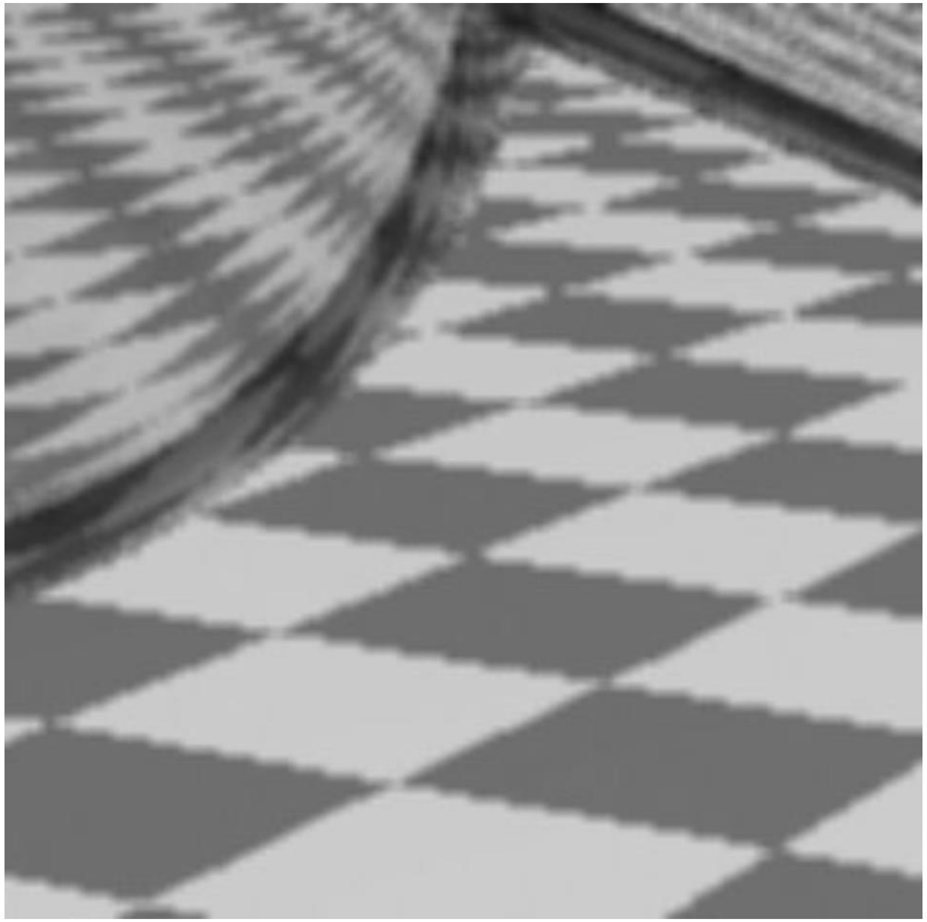


Point Sampling vs Antialiasing

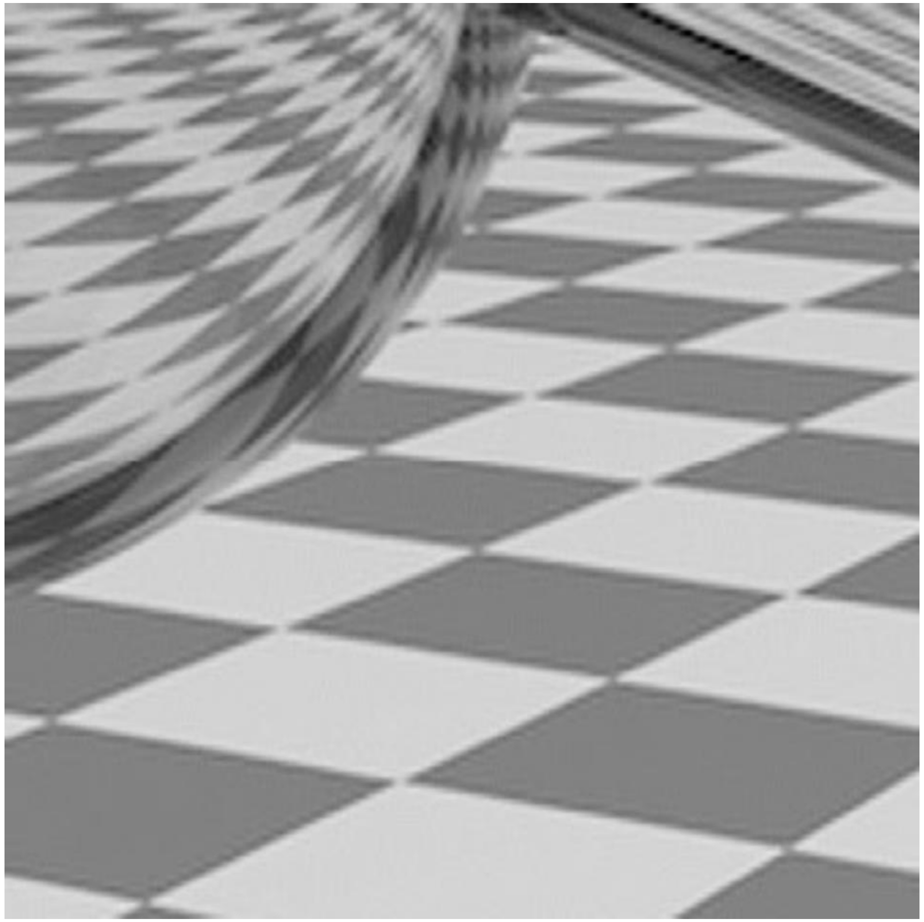




Antialiasing vs Blurred Aliasing



(Sample then filter, WRONG!)



(Filter then sample)

But why?

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

If you want to dig into fundamental reasons

Please look at how to implement antialiased rasterization

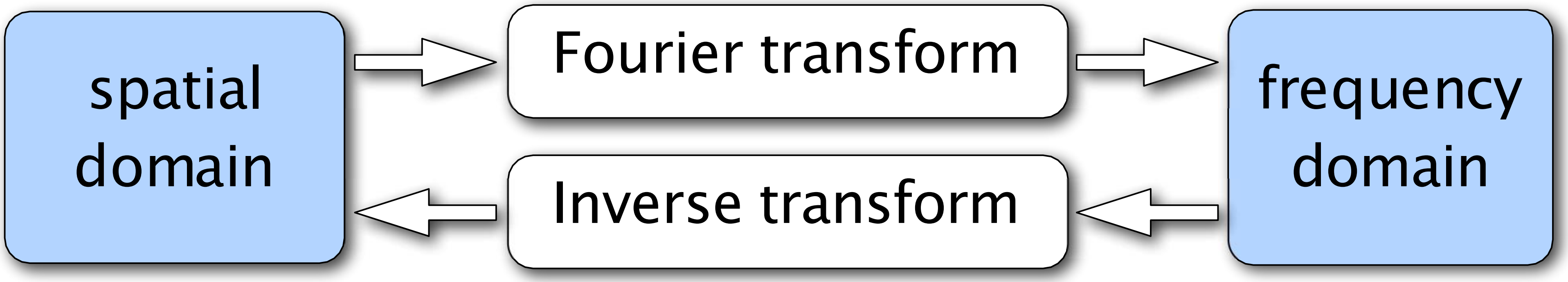
Frequency Domain

Fourier Transform Decomposes A Signal Into Frequencies

$f(x)$

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

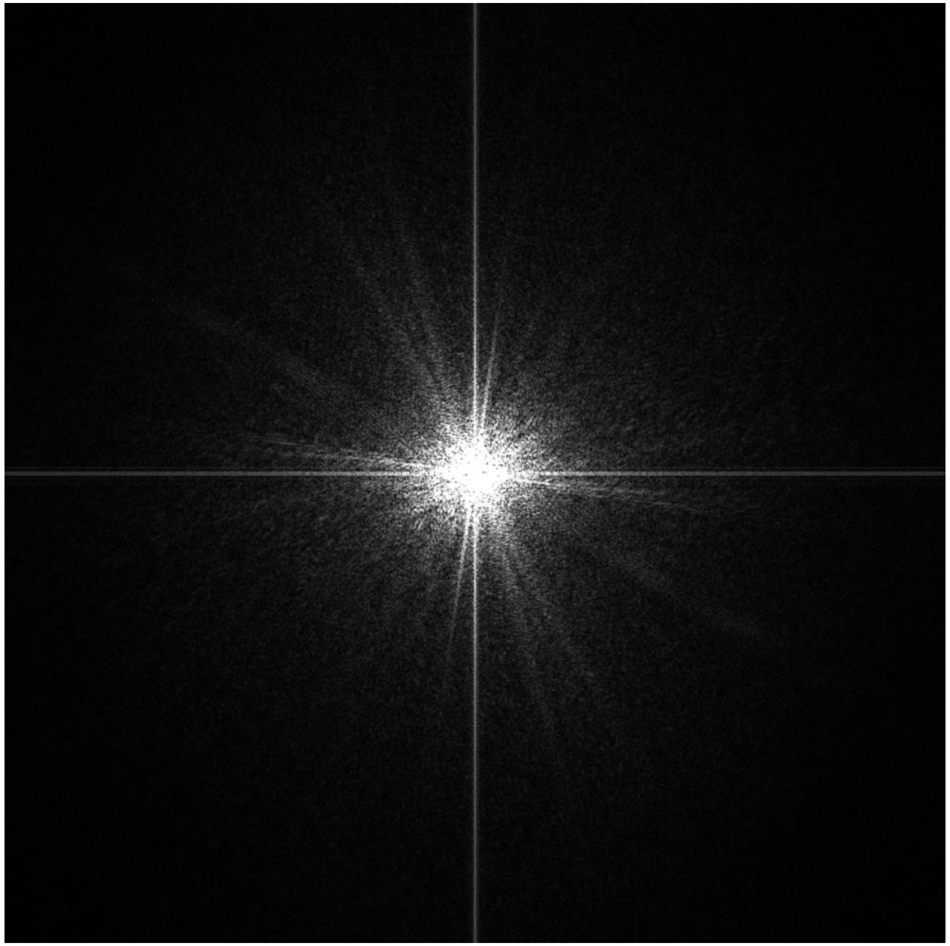
$F(\omega)$



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

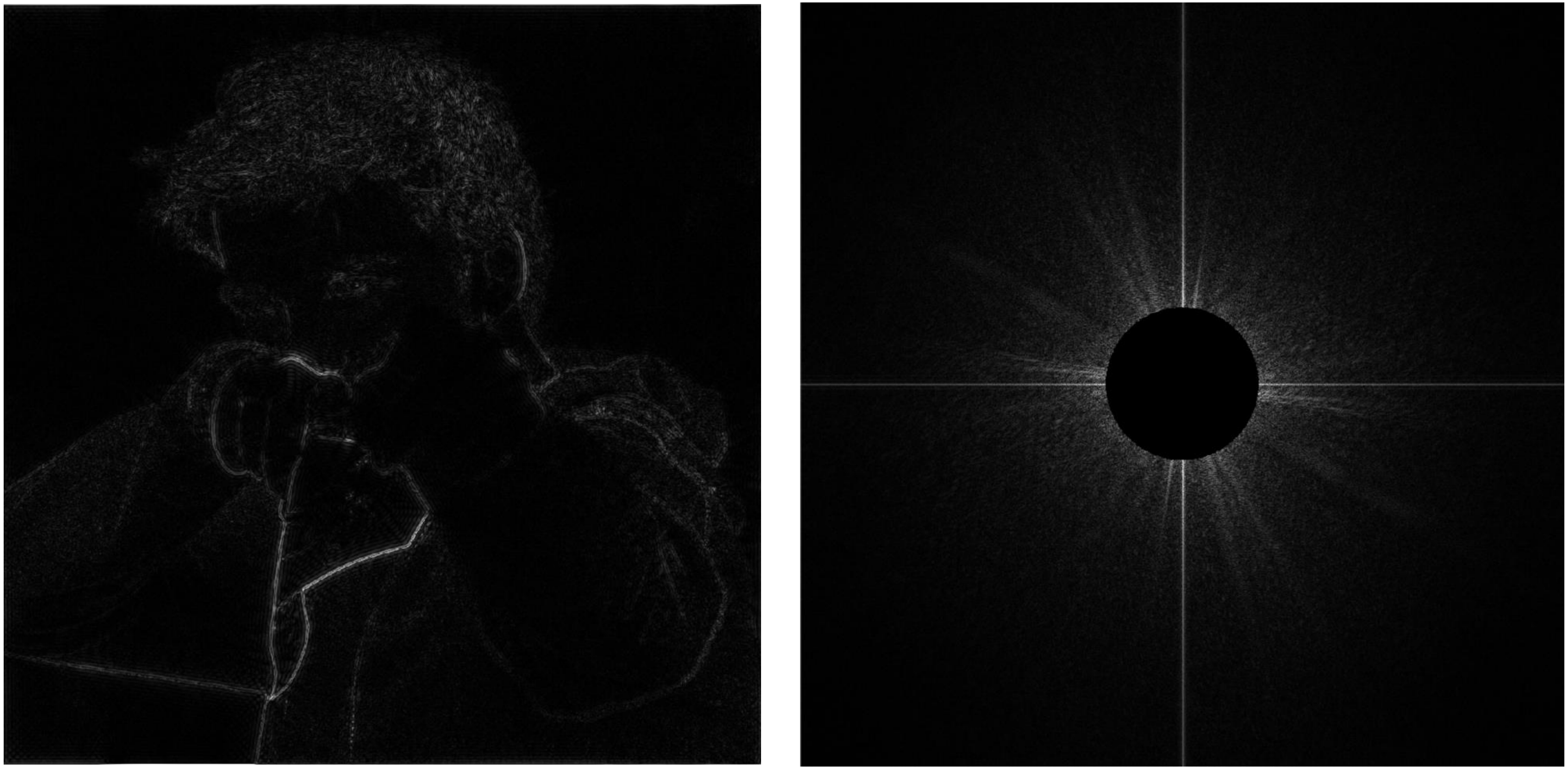
Recall $e^{ix} = \cos x + i \sin x$

Visualizing Image Frequency Content



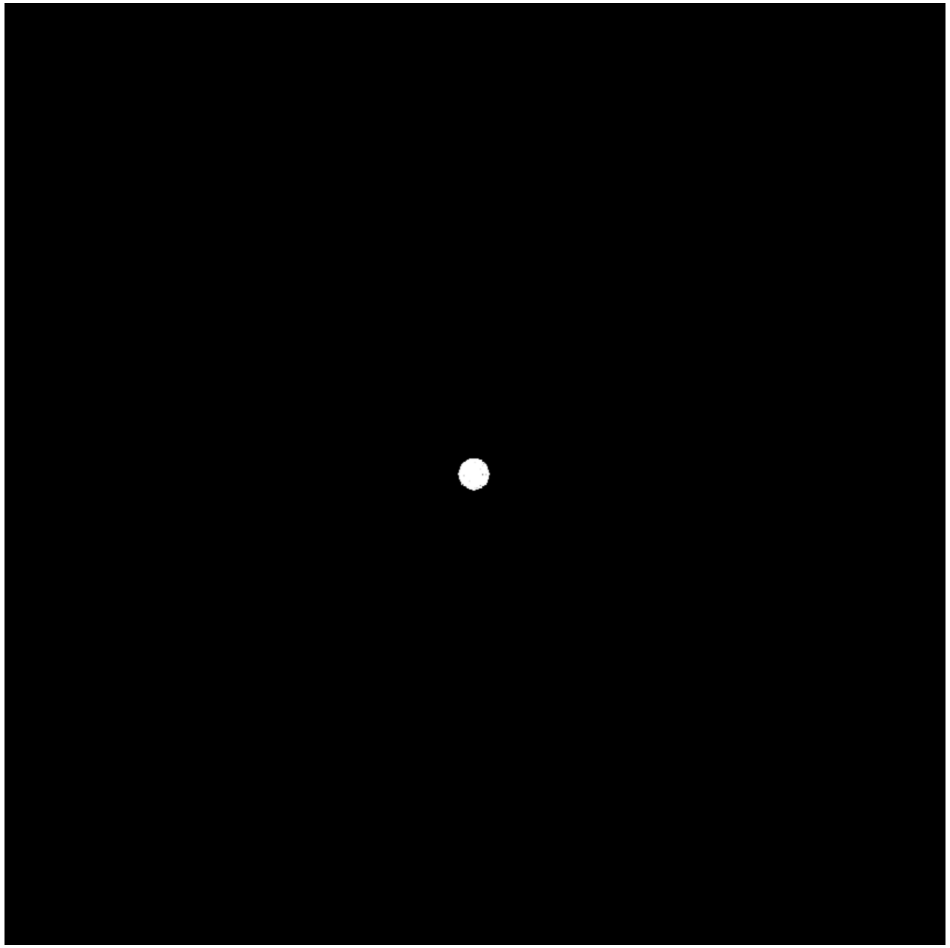
Filtering = Getting rid of certain frequency contents

Filter Out Low Frequencies Only (Edges)



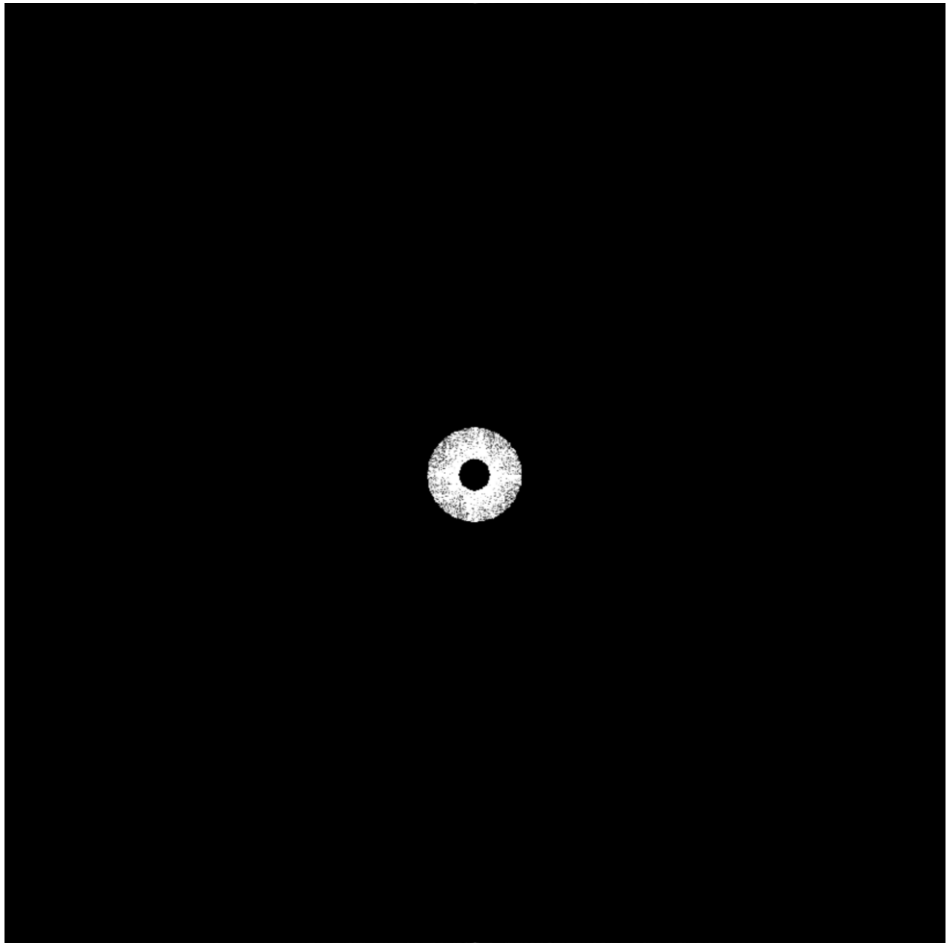
High-pass filter

Filter Out High Frequencies (Blur)

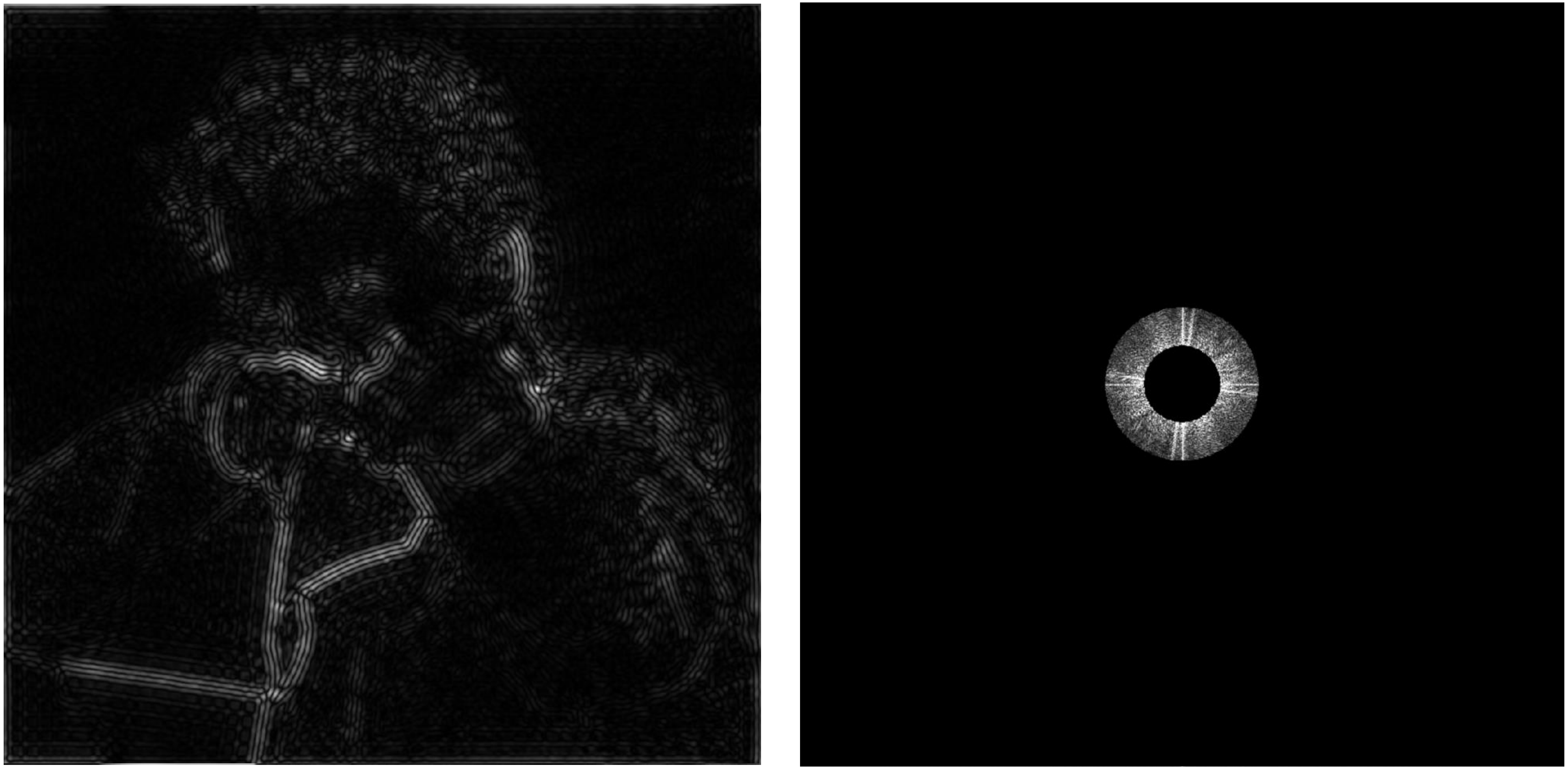


Low-pass filter

Filter Out Low and High Frequencies



Filter Out Low and High Frequencies



Filtering = Convolution

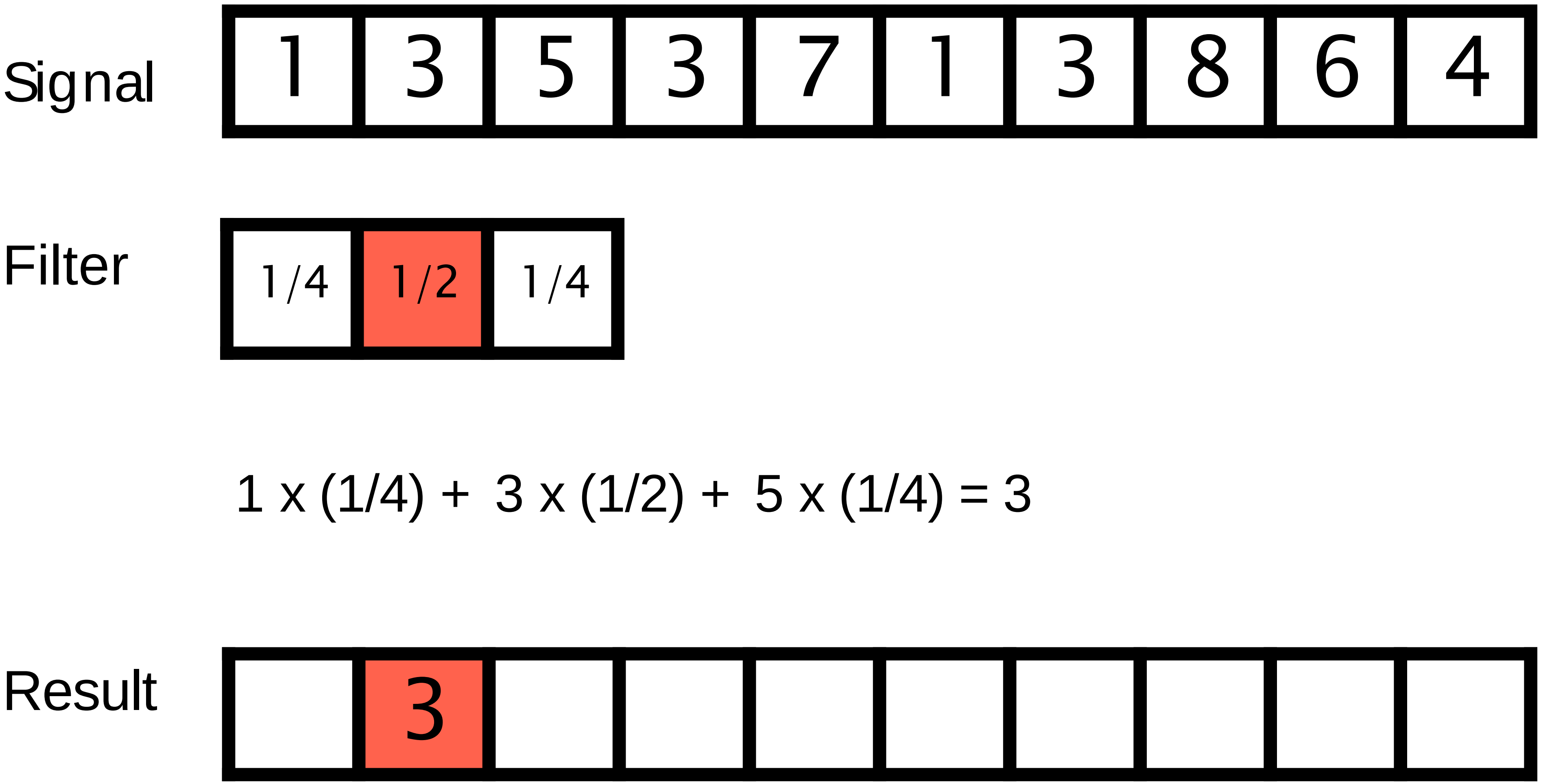
(= Averaging)

Convolution

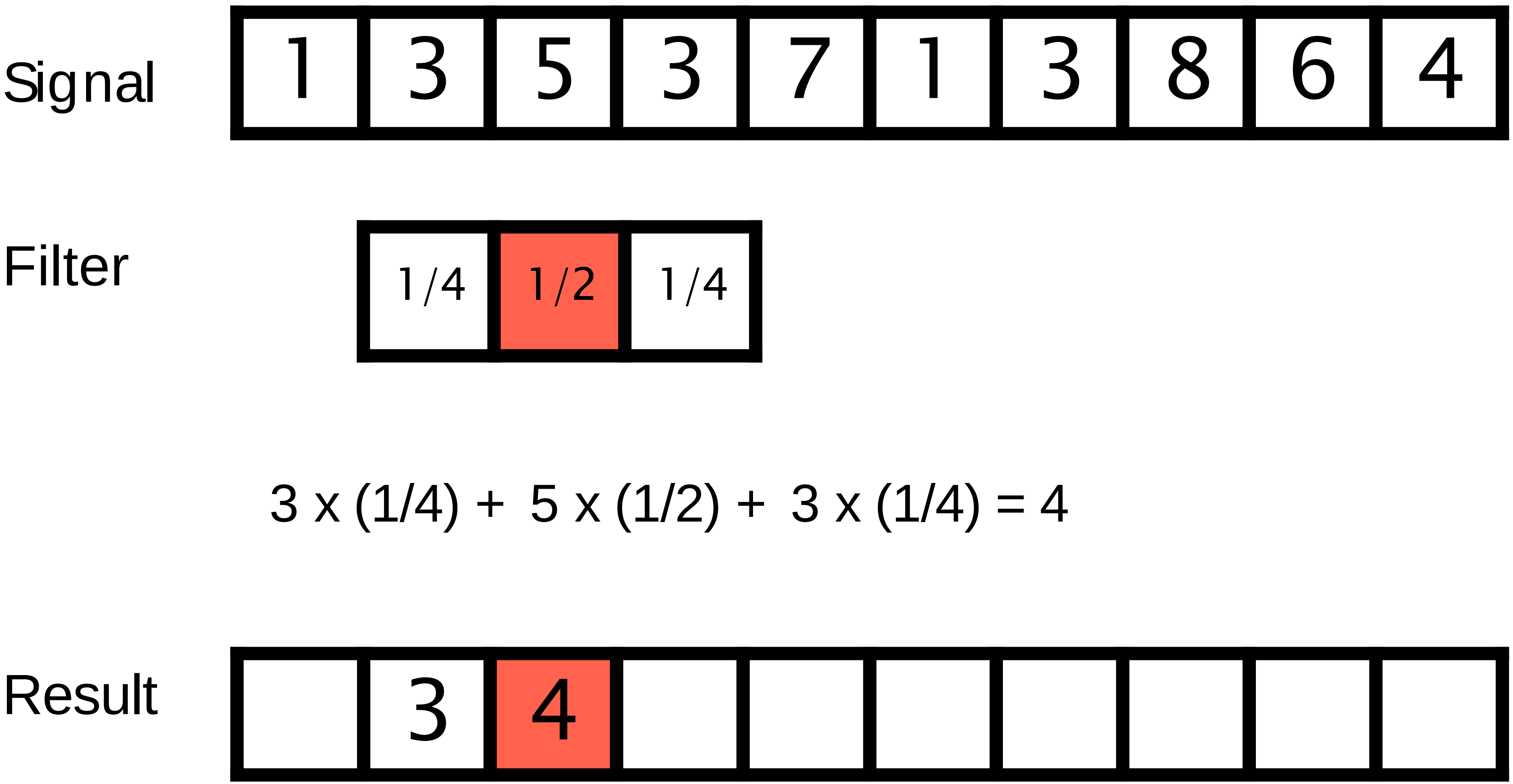
Signal	1	3	5	3	7	1	3	8	6	4
Filter	1/4	1/2	1/4							

Point-wise local averaging in a “sliding window”

Convolution



Convolution



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

Convolution Theorem

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

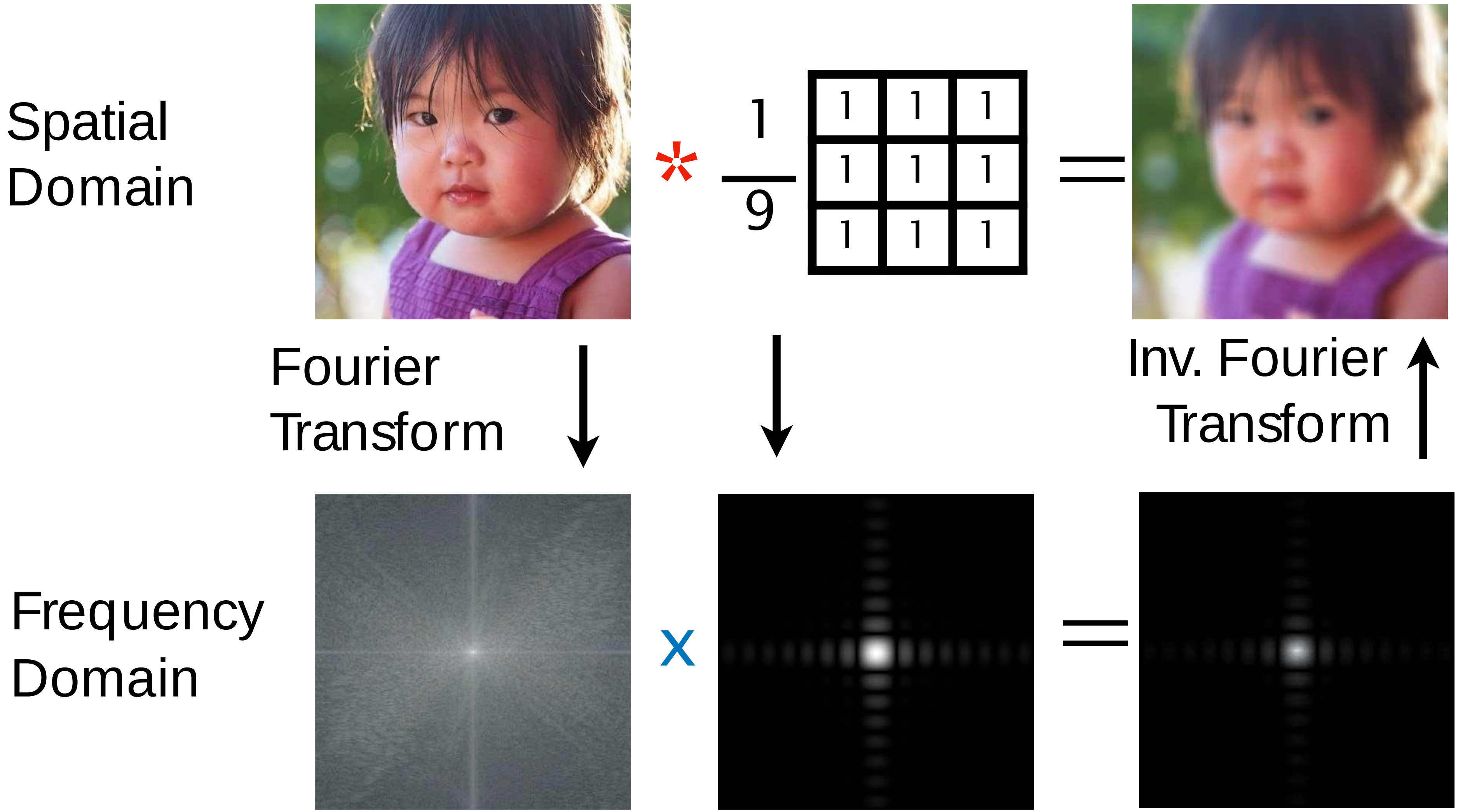
Option 1:

- 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

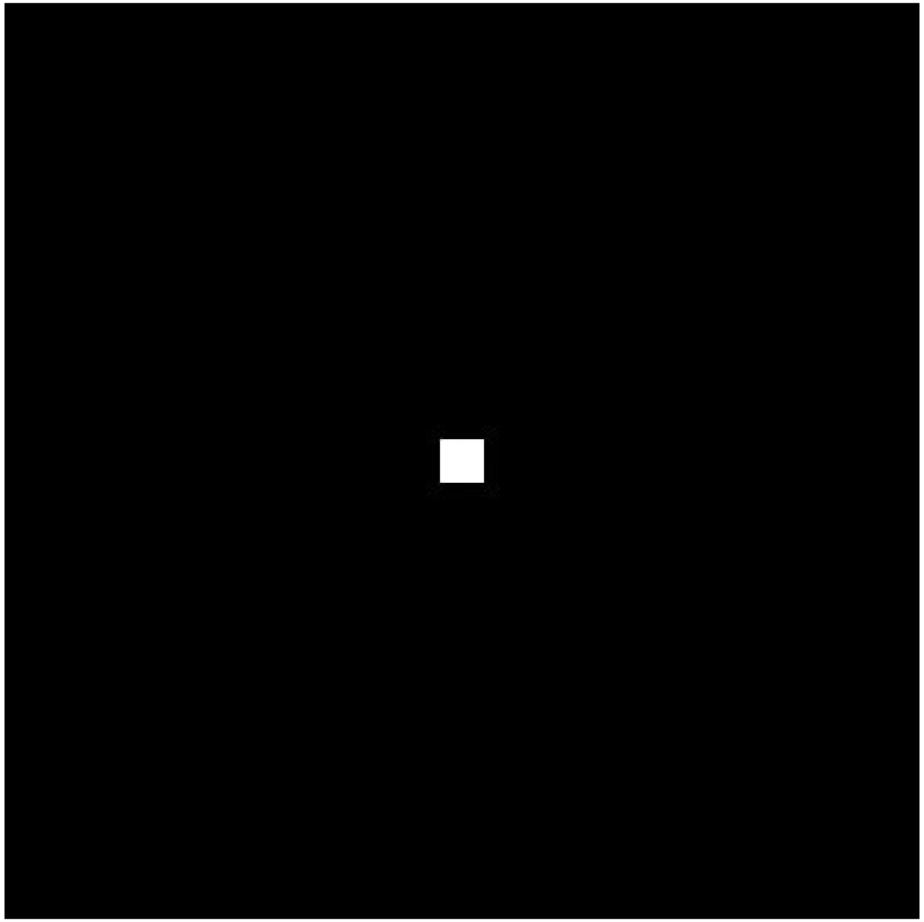


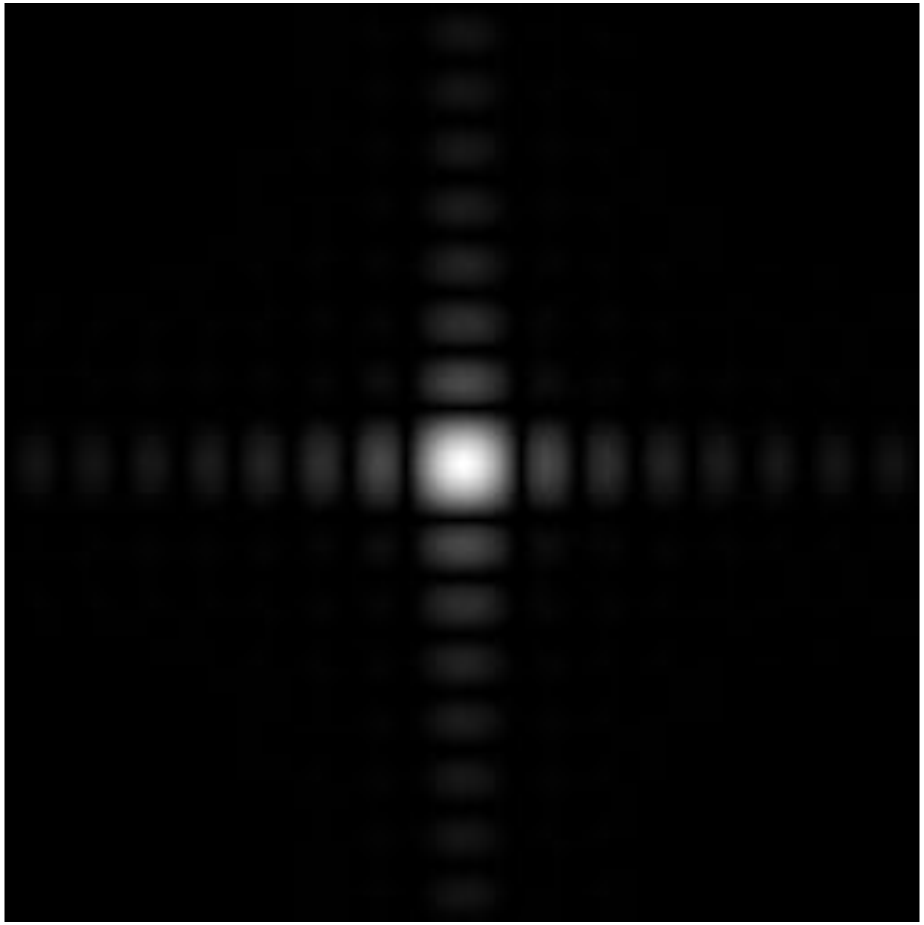
Box Filter

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

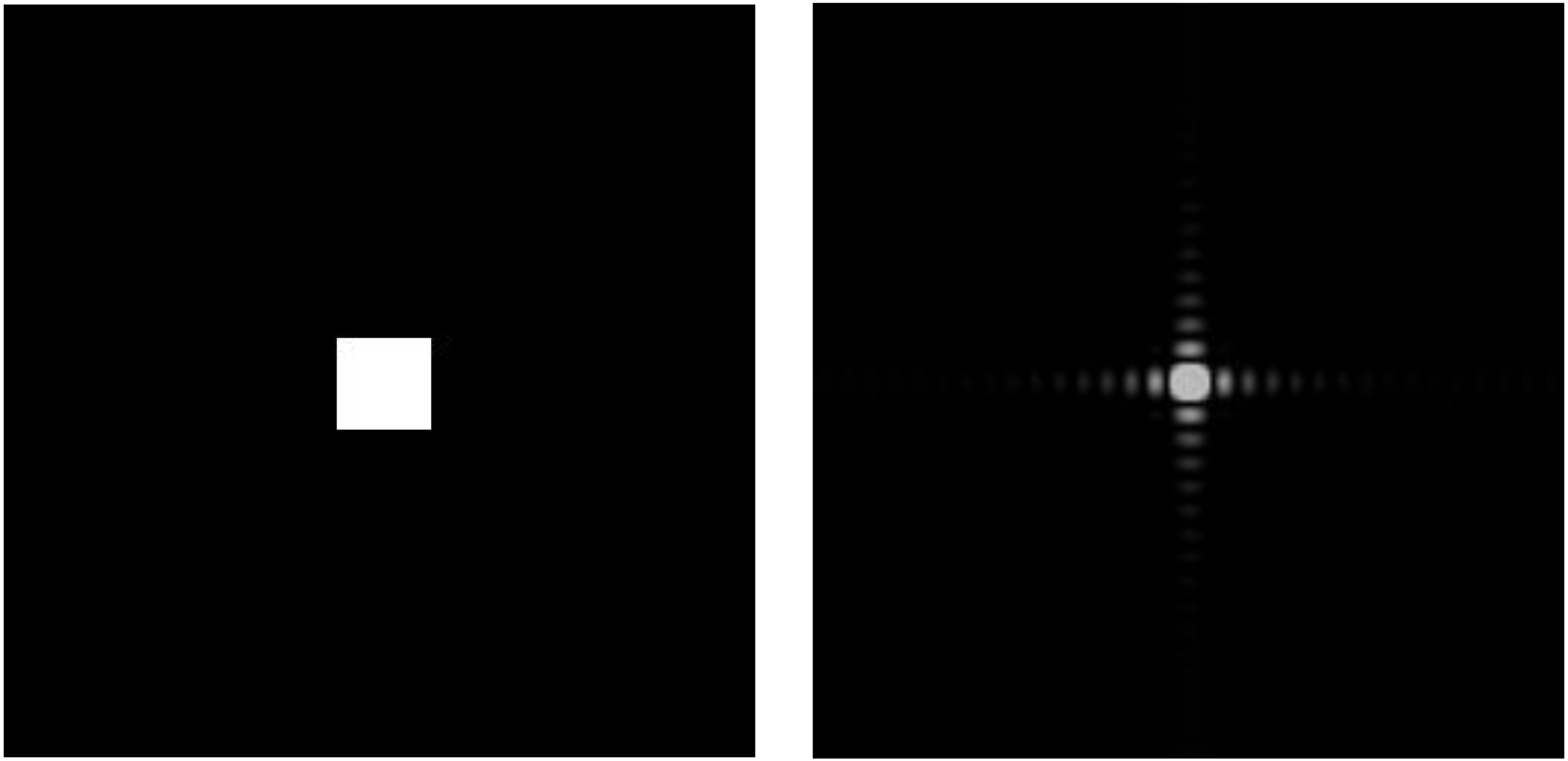
Example: 3x3 box filter

Box Function = “Low Pass” Filter





Wider Filter Kernel = Lower Frequencies



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

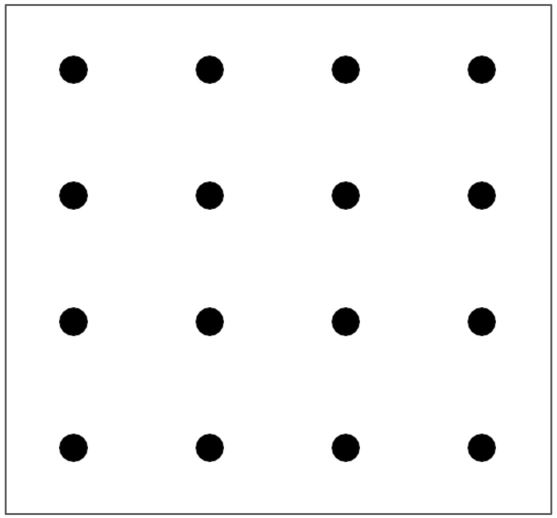
- Making Fourier contents “narrower” before repeating
- i.e. Filtering out high frequencies before sampling

Multisample Anti-Aliasing (MSAA)

多重采样抗锯齿

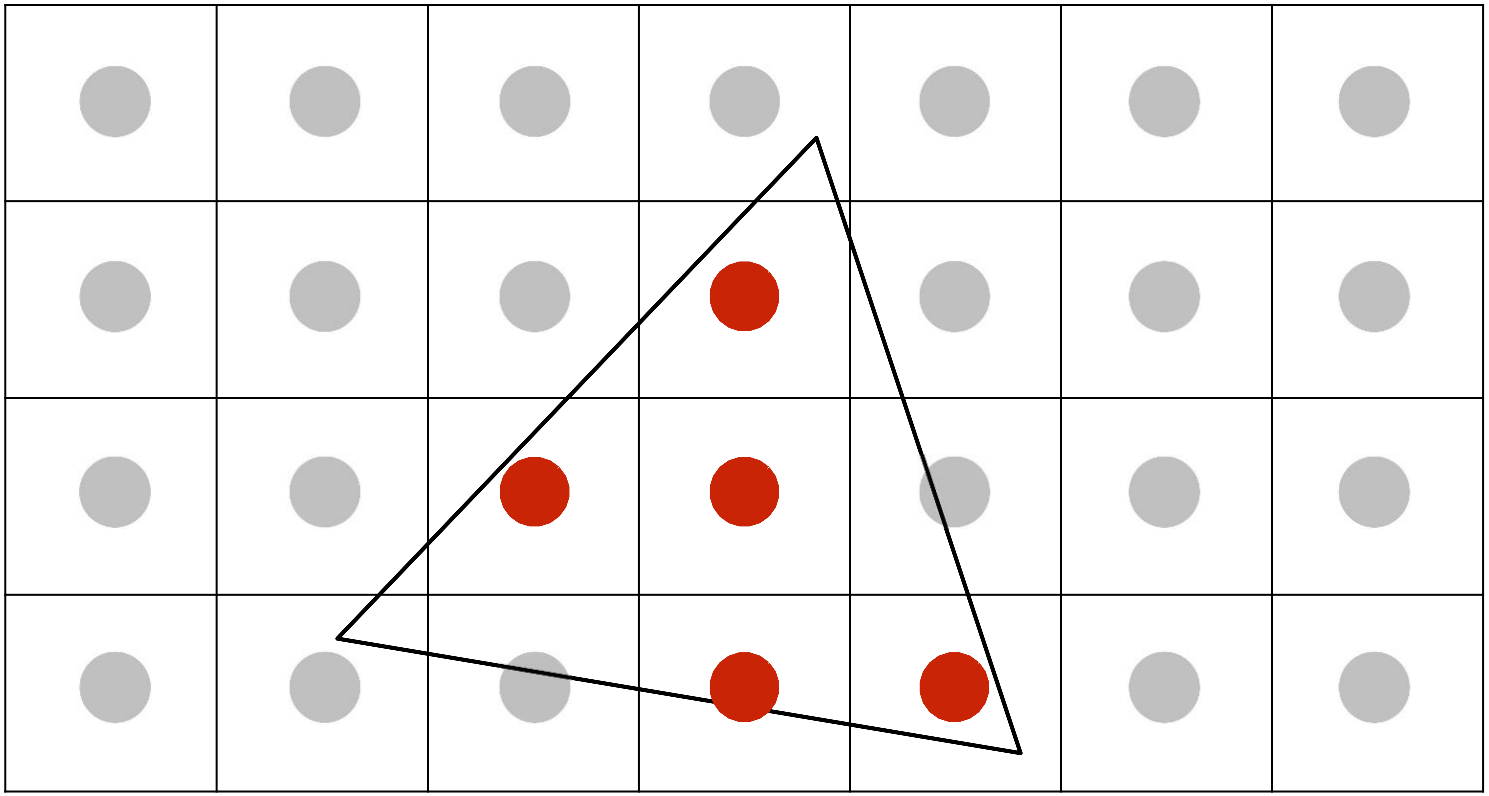
Supersampling

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



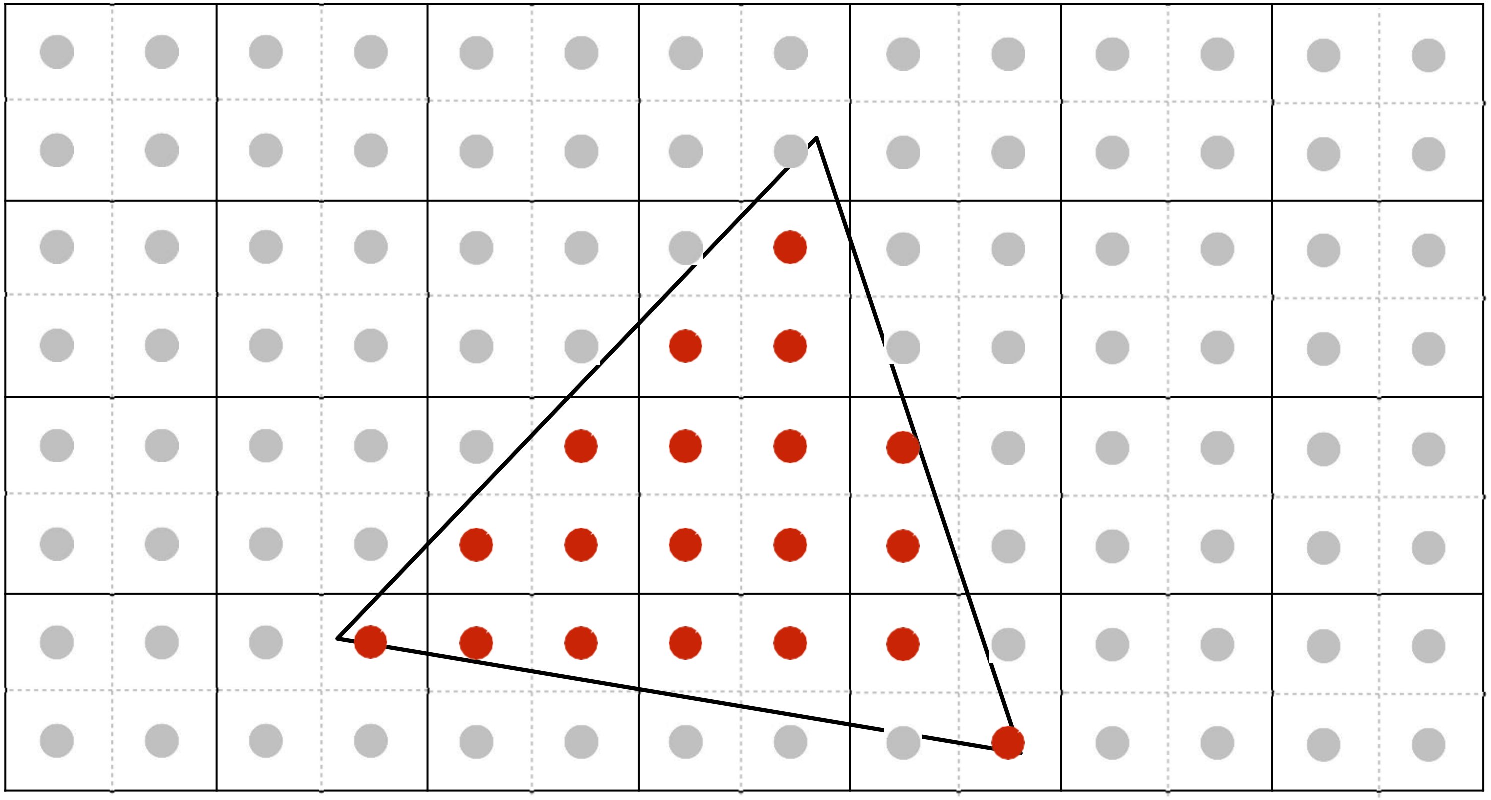
4x4 supersampling

Point Sampling: One Sample Per Pixel



Supersampling: Step 1

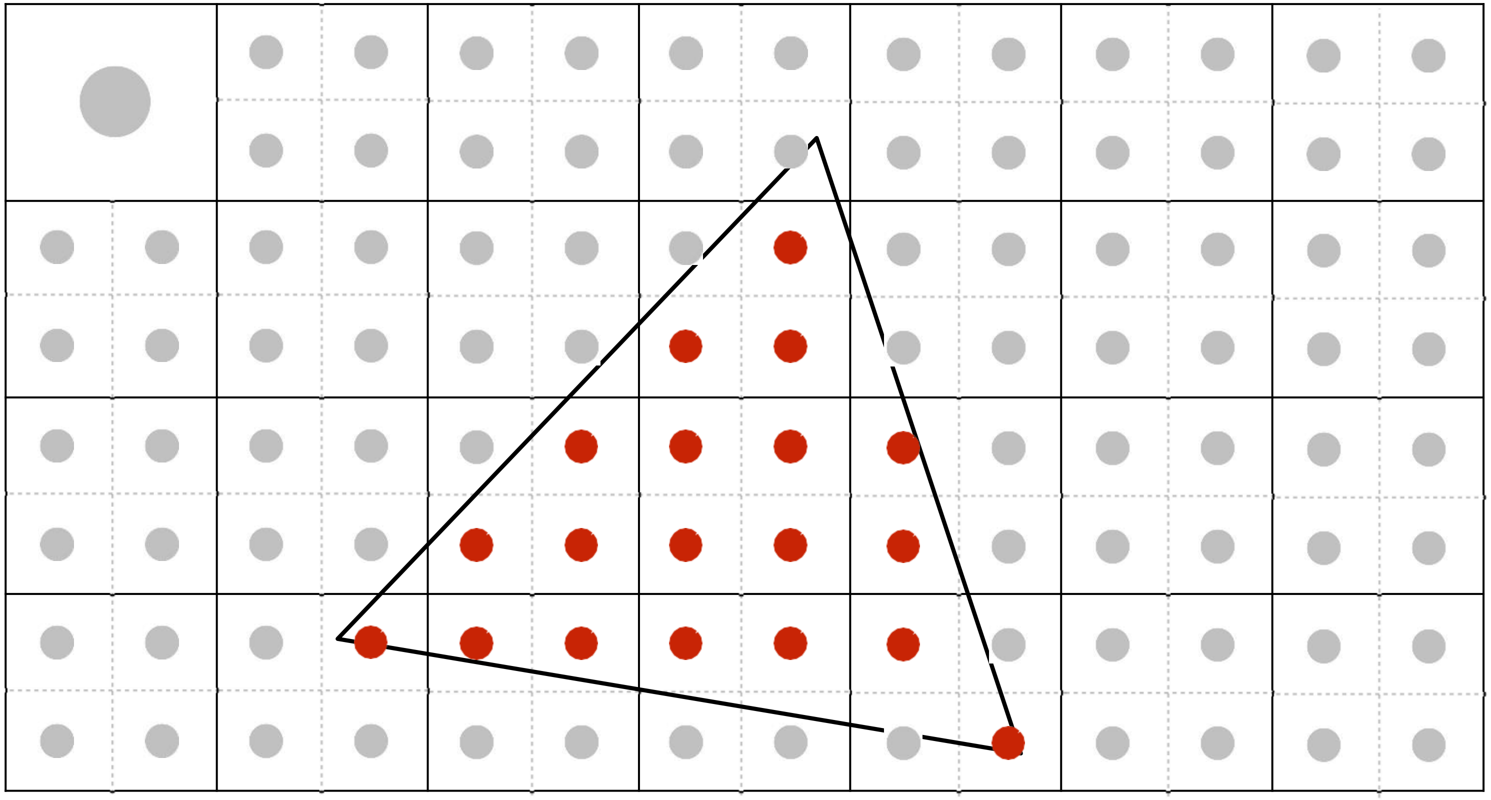
Take NxN samples in each pixel.



2x2 supersampling

Supersampling: Step 2

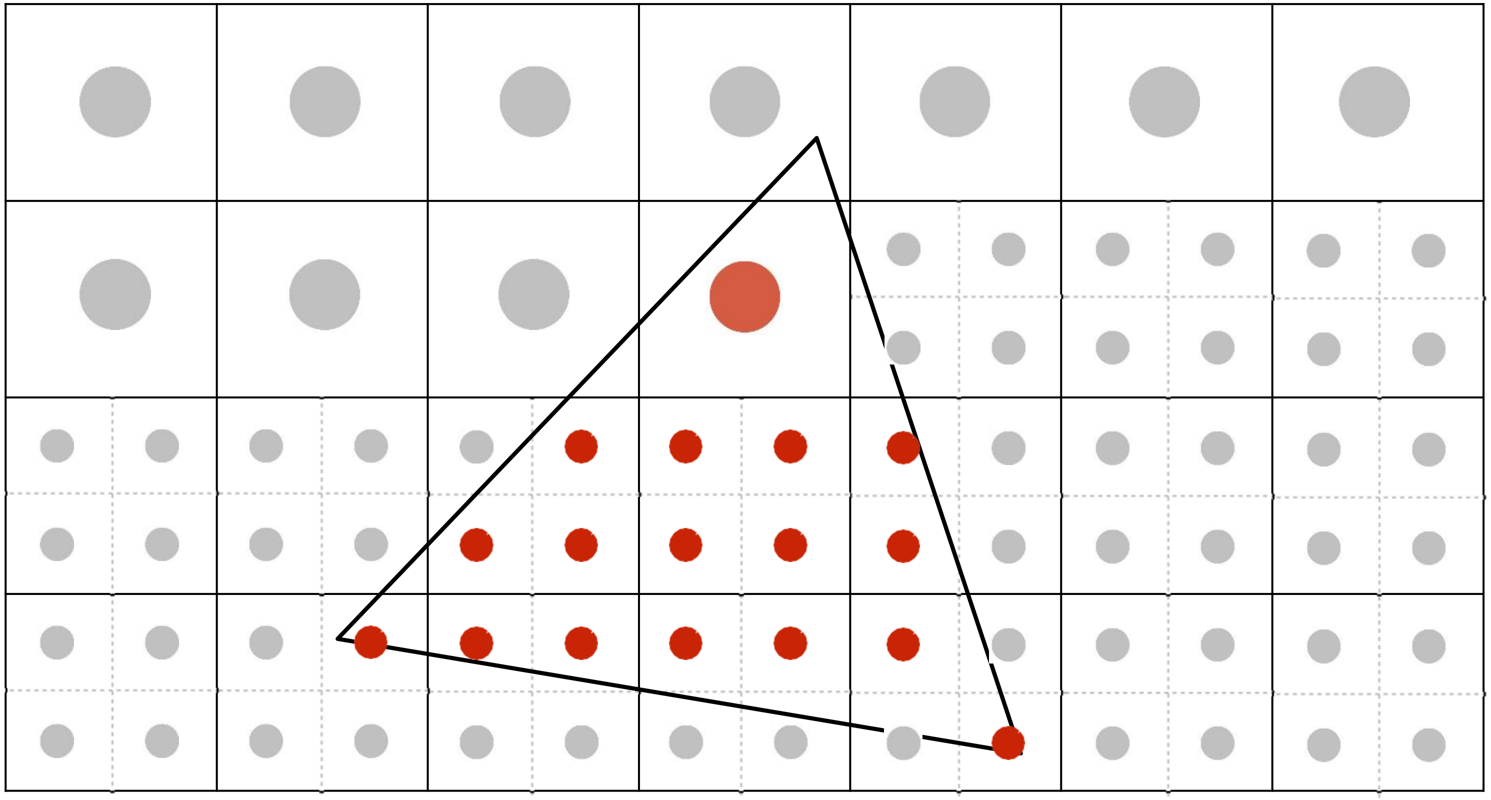
Average the NxN samples “inside” each pixel.



Averaging down

Supersampling: Step 2

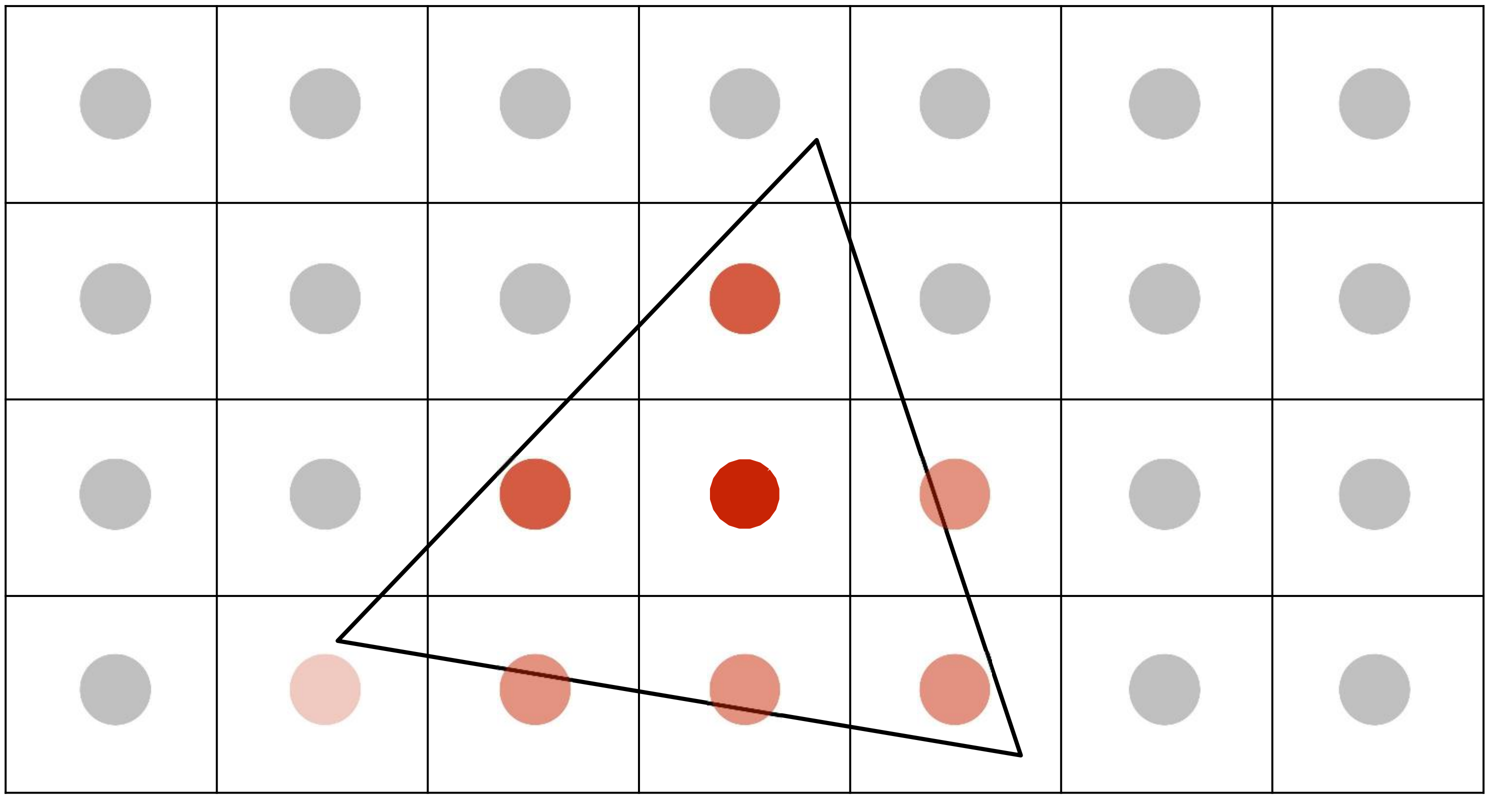
Average the NxN samples “inside” each pixel.



Averaging down

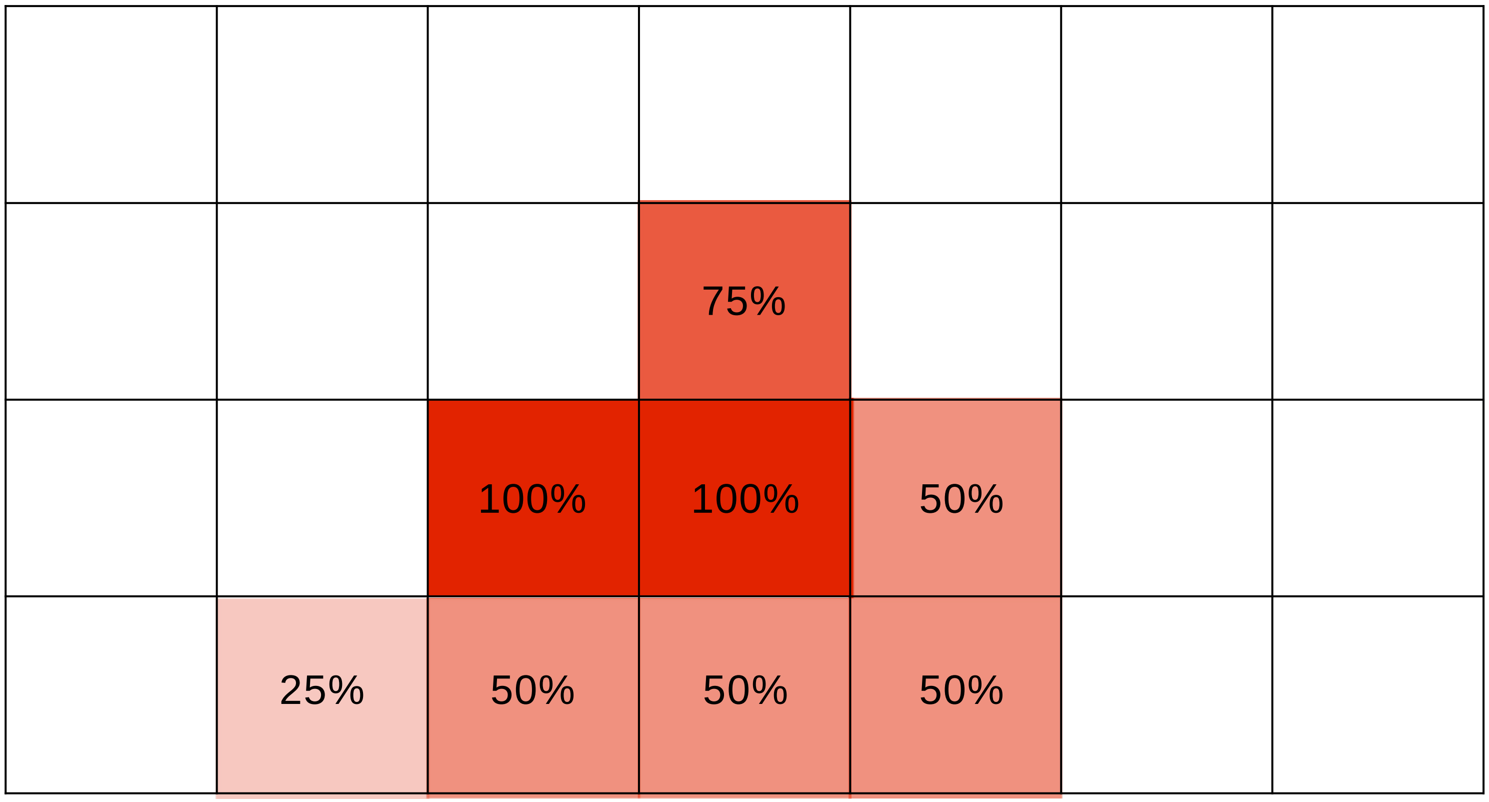
Supersampling: Step 2

Average the NxN samples “inside” each pixel.

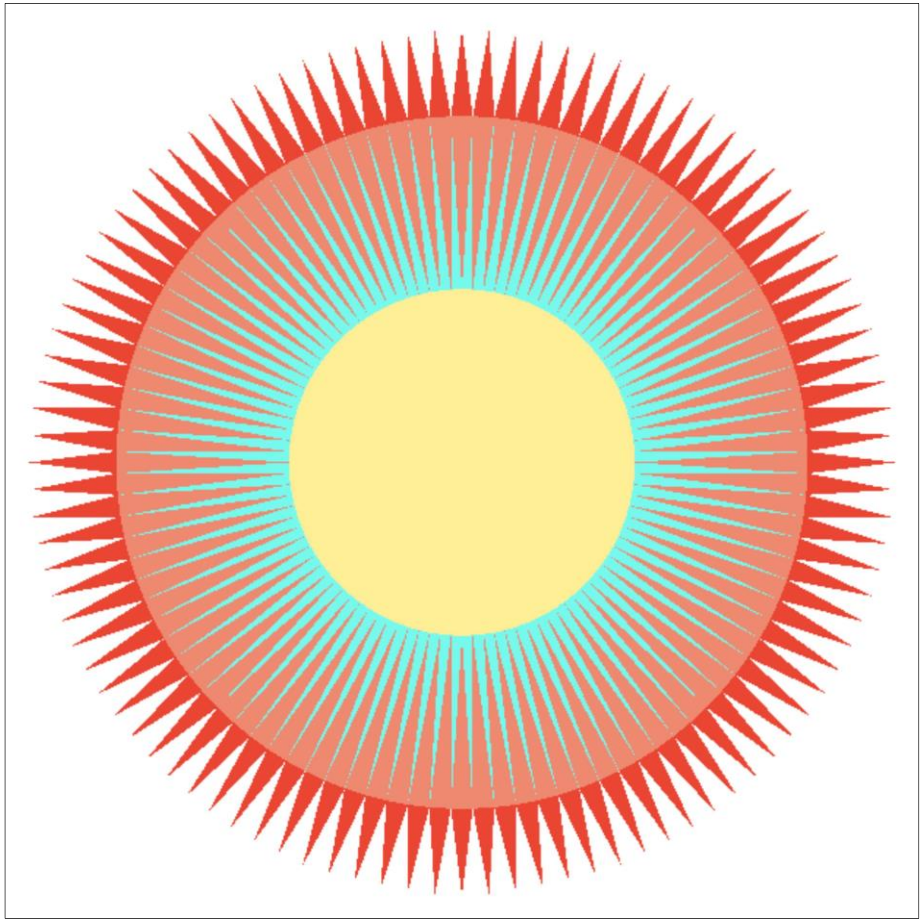


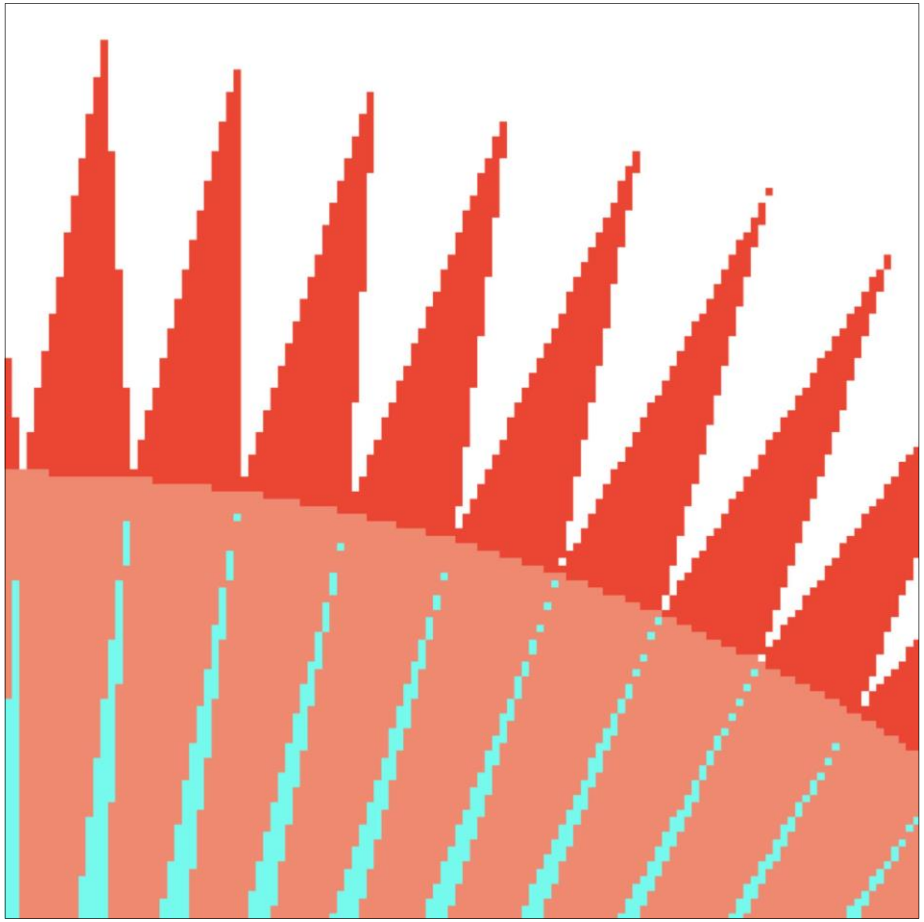
Supersampling: Result

This is the corresponding signal emitted by the display

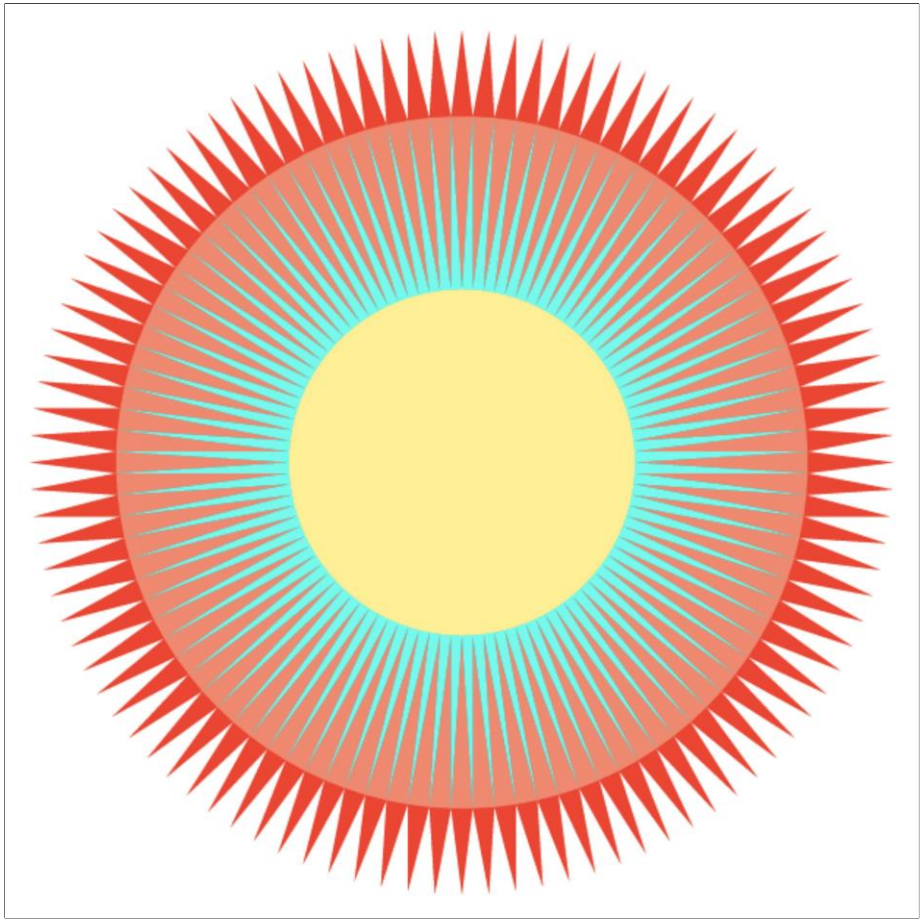


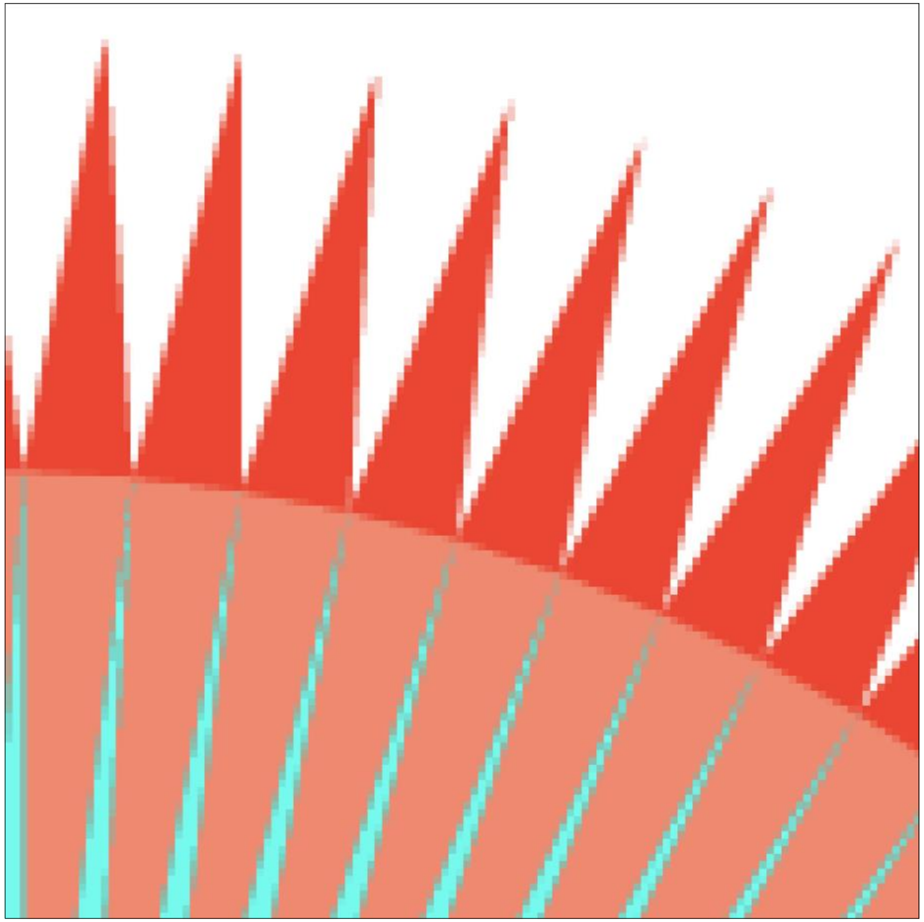
Point Sampling





4x4 Supersampling





Antialiasing Today

No free lunch!

- What's the cost of MSAA?

Milestones (personal idea)

- FXAA (Fast Approximate AA)
- TAA (Temporal AA)

Super resolution / super sampling

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

Thank you!