3D Computer Graphics

Rasterization II

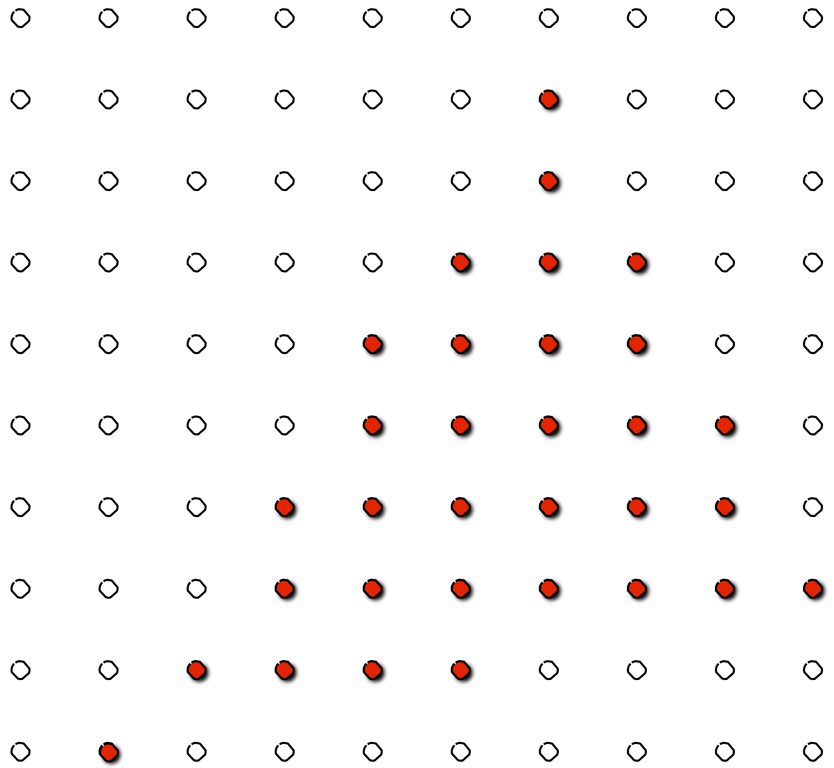
Dr. Zhigang Deng

UNIVERSITY of HOUSTON DEPARTMENT OF COMPUTER SCIENCE

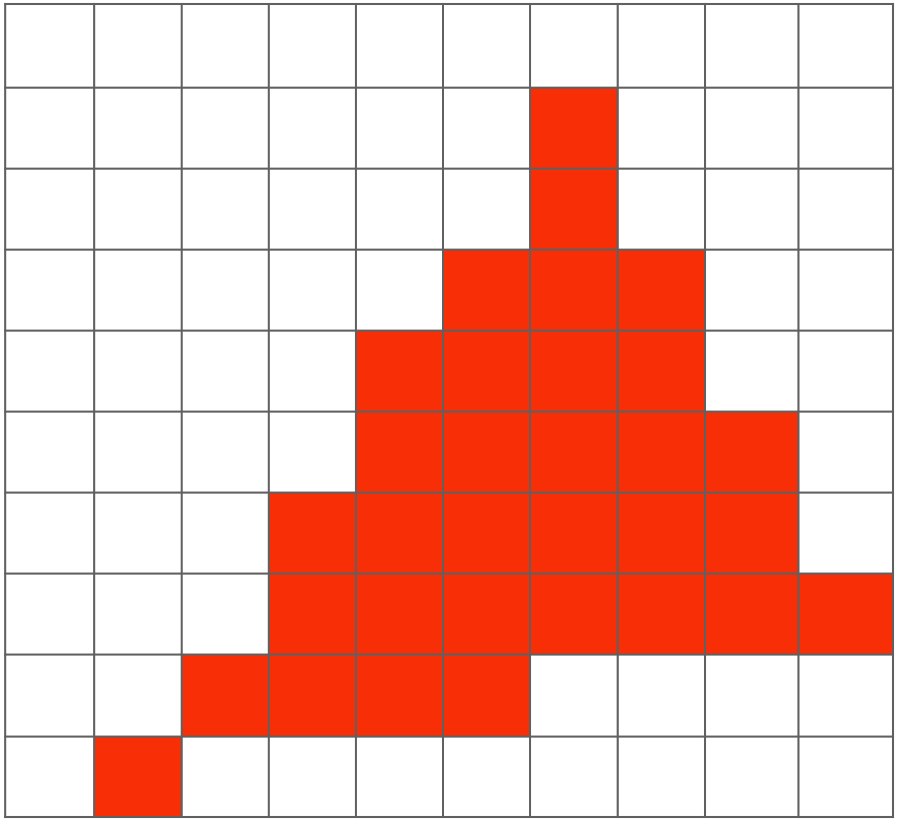
Today

- Antialiasing
  - Sampling theory
  - Antialiasing in practice
- Visibility / occlusion
  - Z-buffering

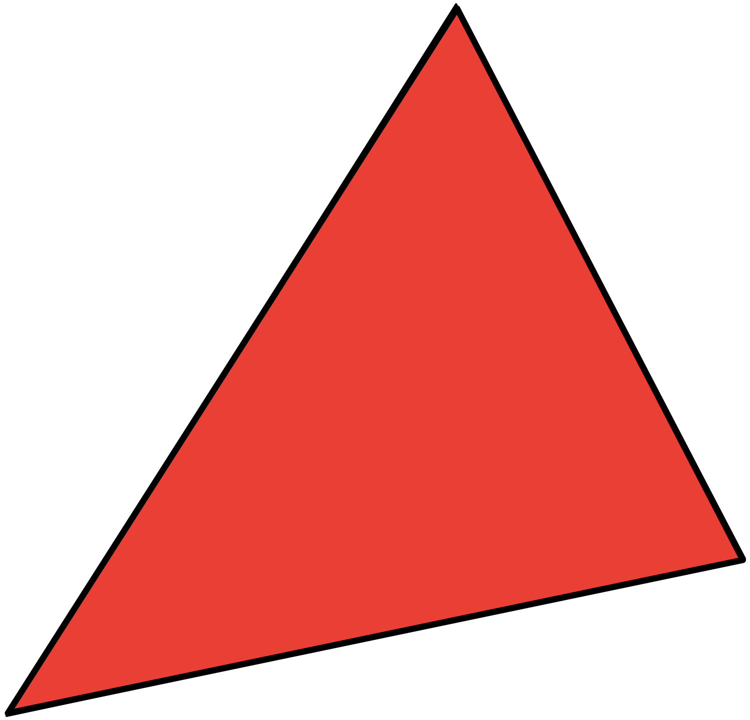
Recap: Testing in/out ¸ at pixels' centers



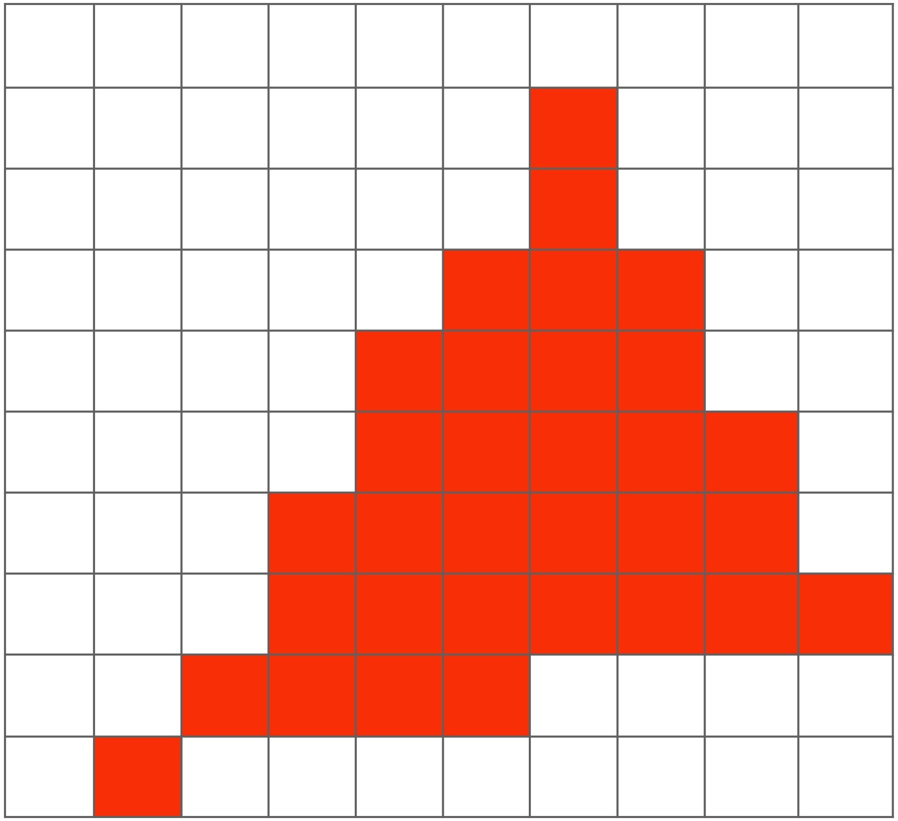
Pixels are uniformly-colored squares



Compare: The Continuous Triangle Function

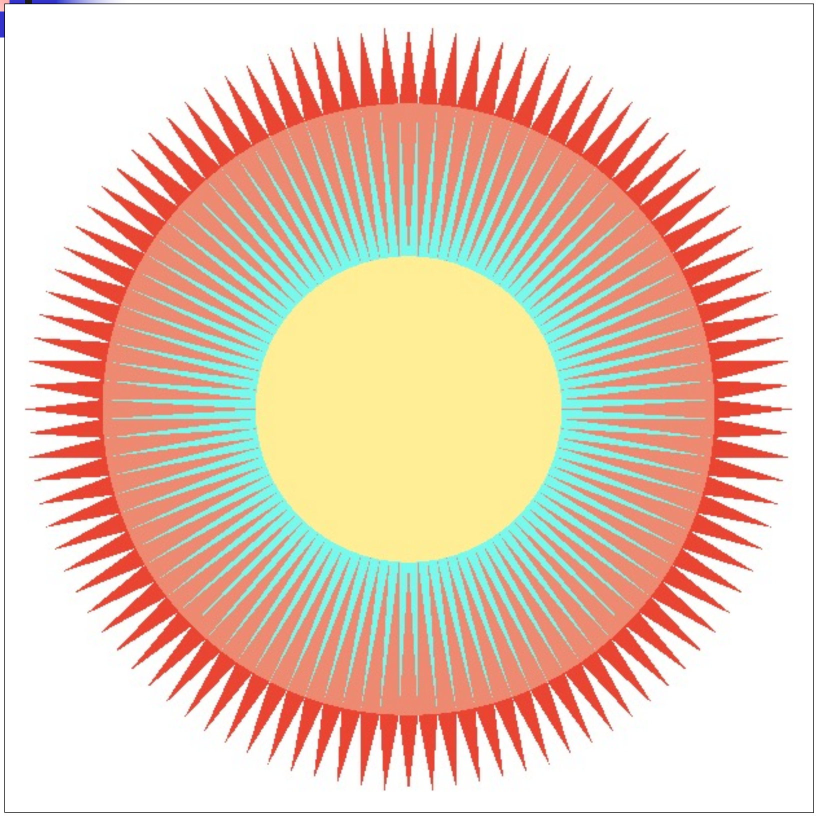


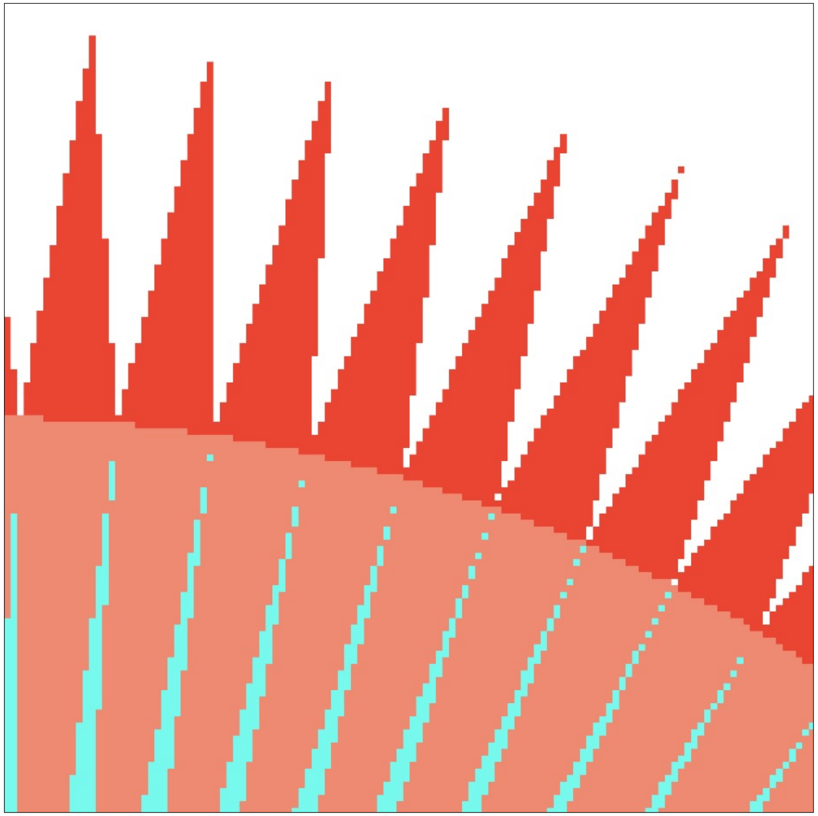
What's Wrong With This Picture?



Jaggies!

Aliasing



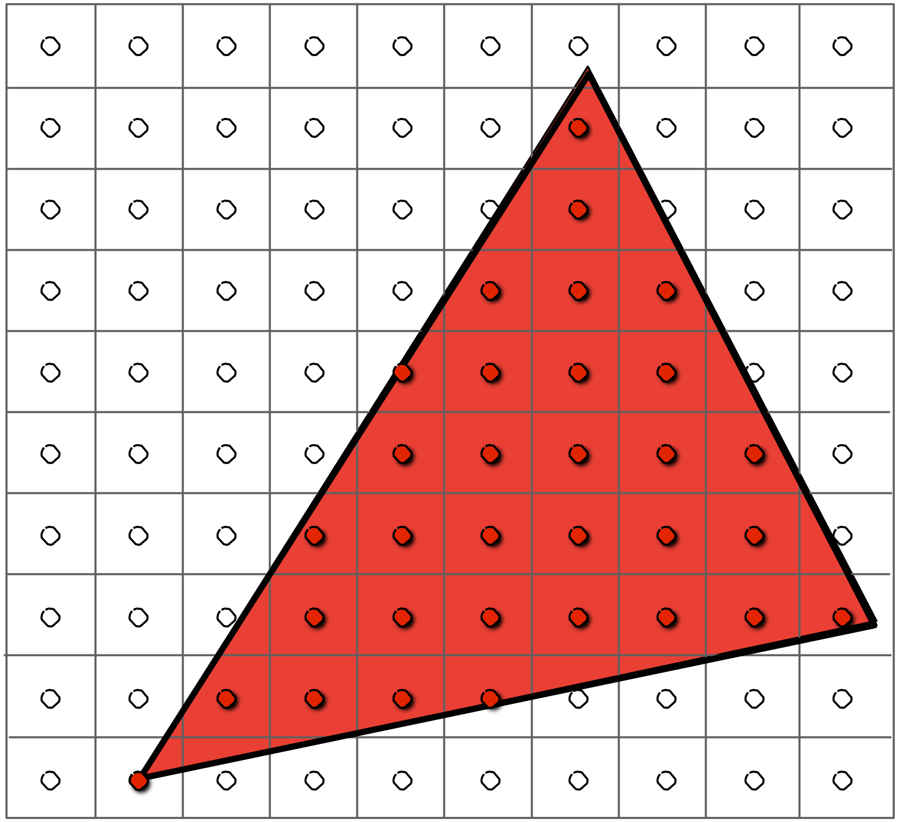


Is this the best we can do?

Slide courtesy of Prof. Ren Ng, UC Berkeley

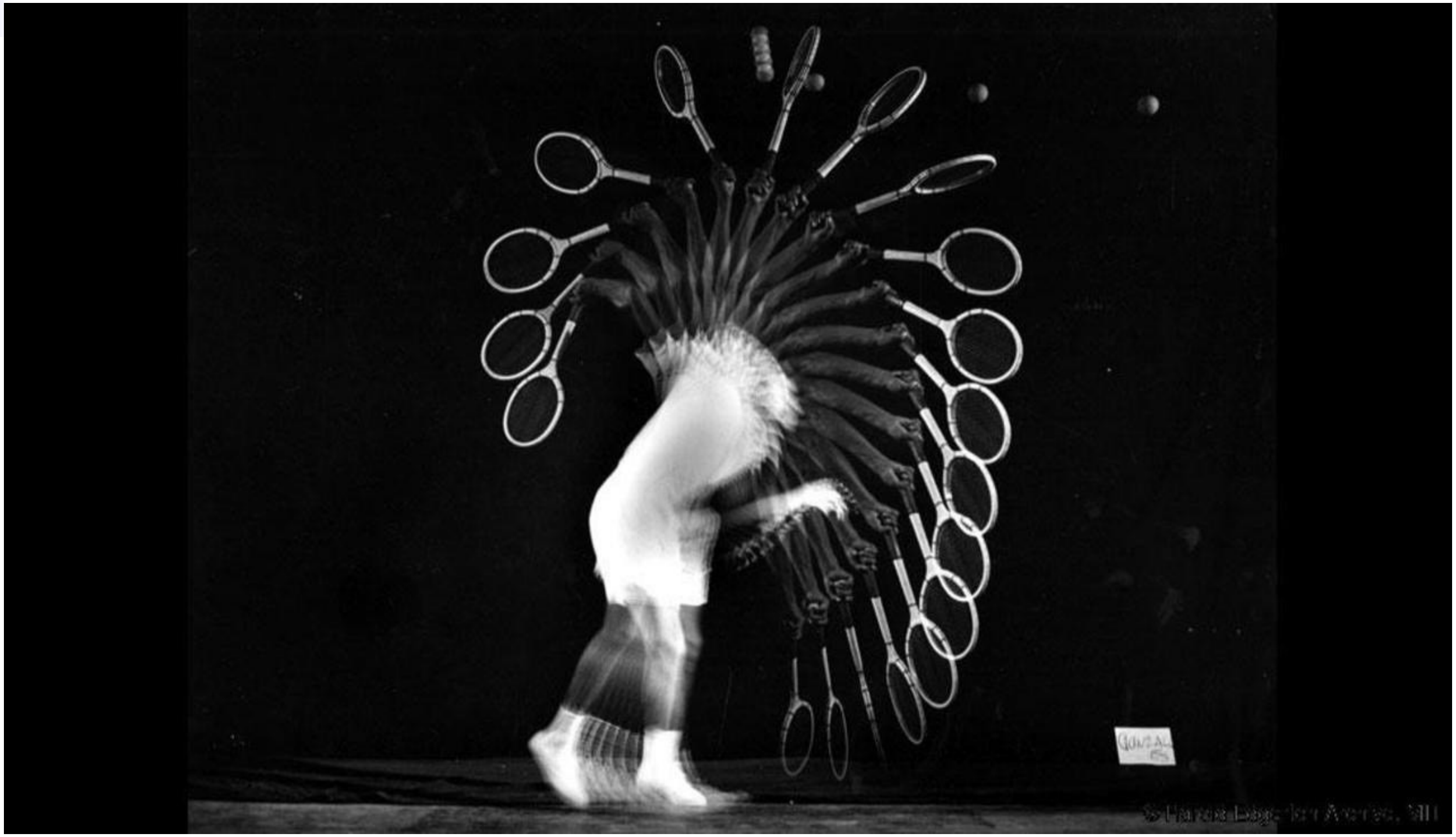
Sampling is Ubiquitous in Computer Graphics

Rasterization = Sample 2D Positions



Photograph = Sample Image Sensor Plane

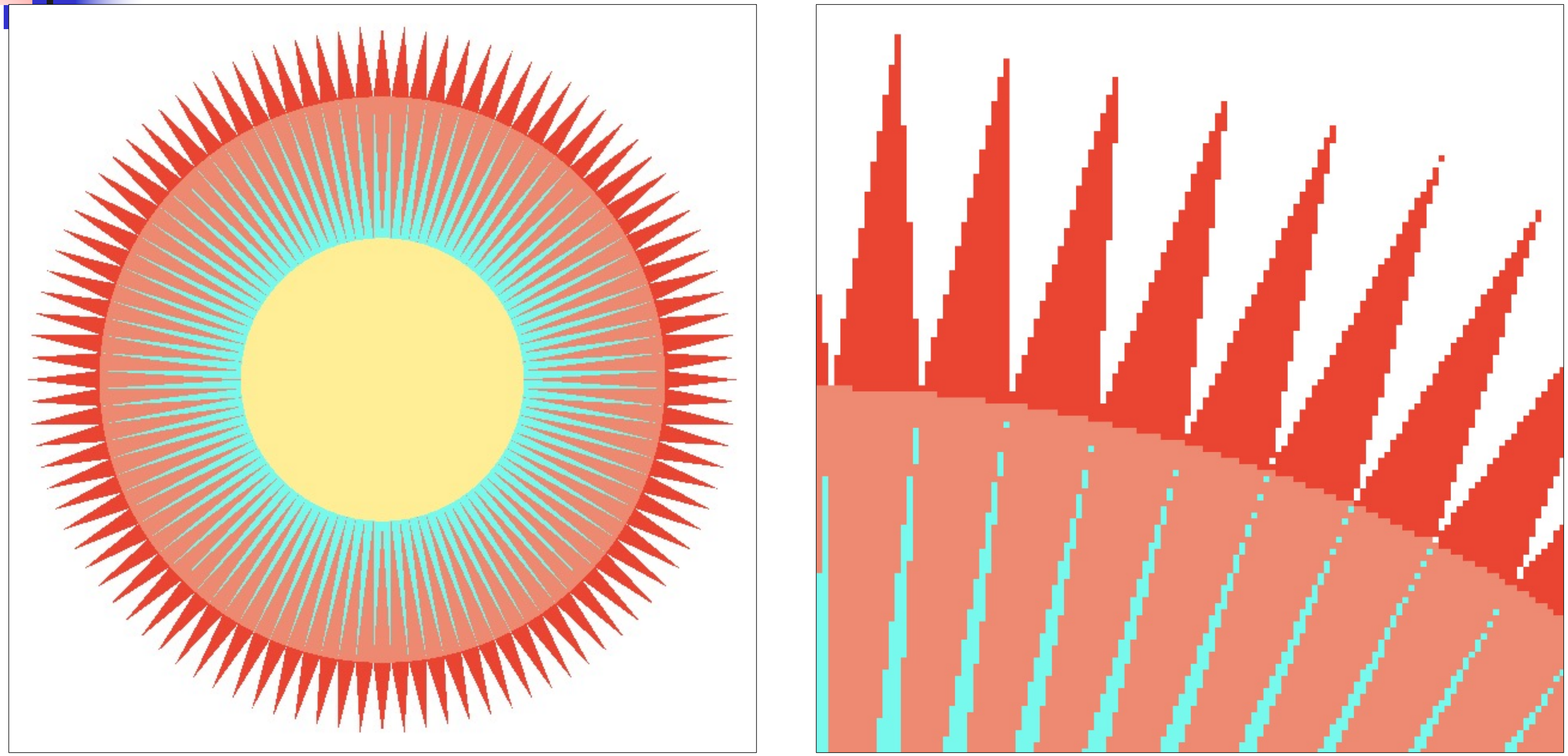
Video = Sample Time



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

lystit.com

Skip odd rows and columns

Wagon Wheel Illusion (False Motion)

https://www.youtube.com/watch?v=QOwzkND_ooU

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

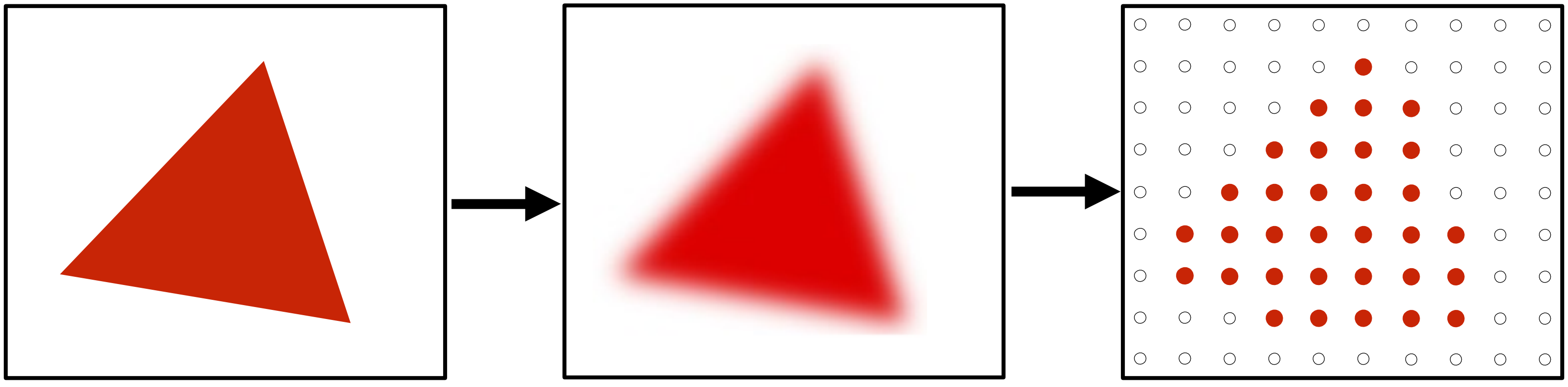
- Signals are changing too fast (high frequency), but sampled too slowly

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



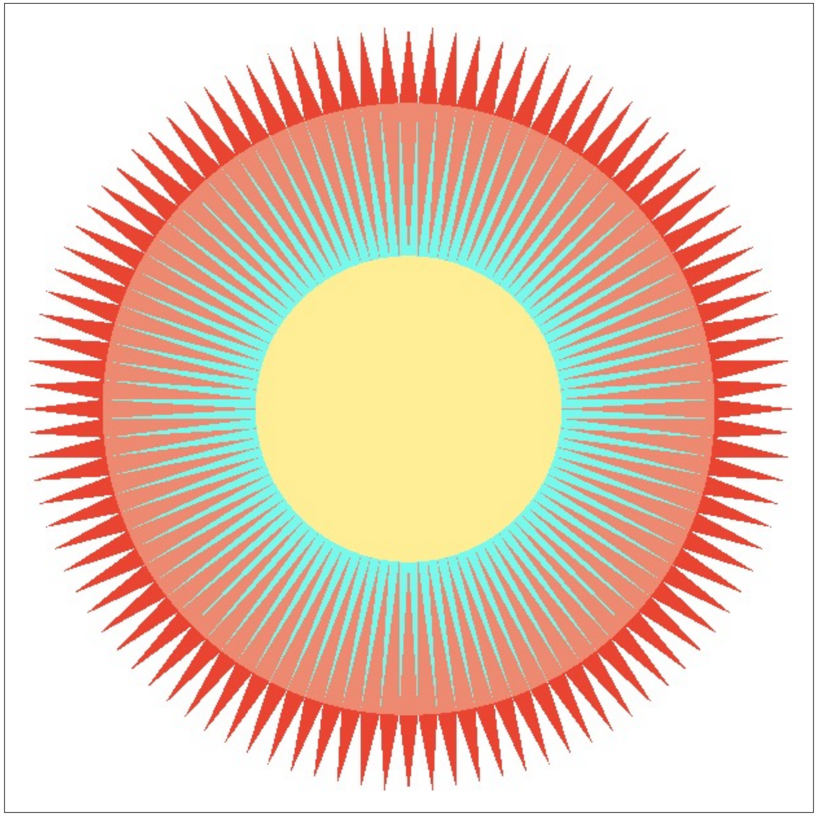
Pre-Filter

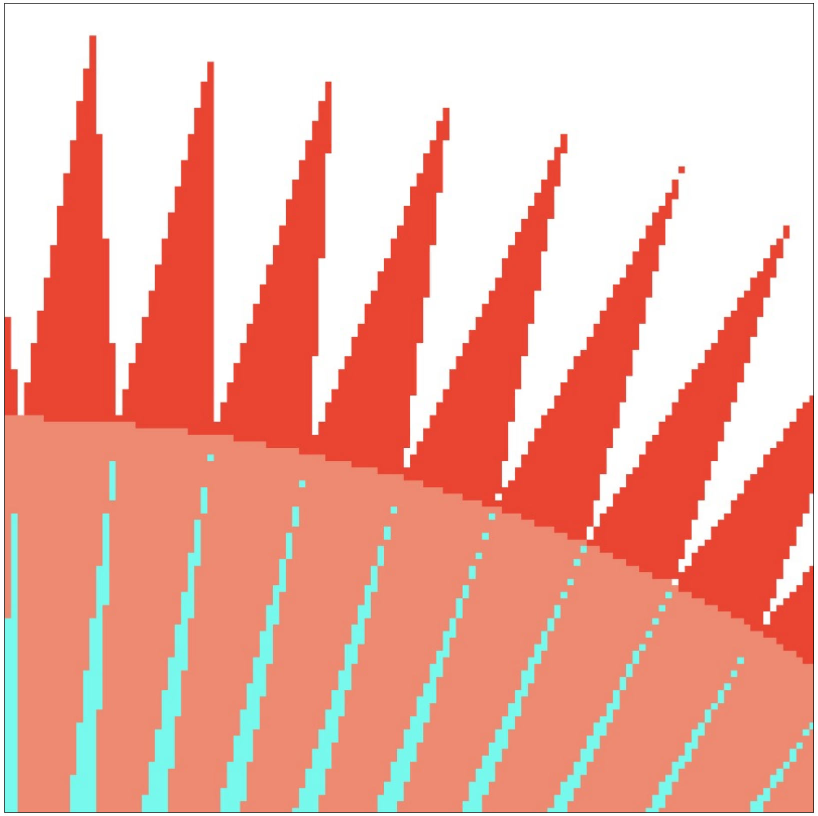
(remove frequencies above Nyquist)

Sample

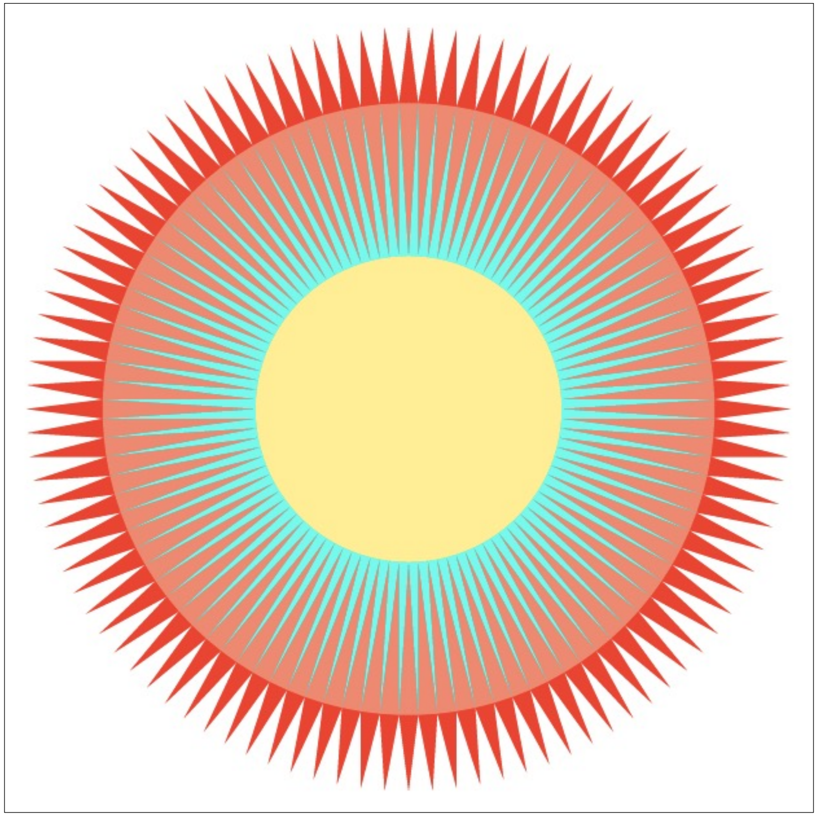
Note antialiased edges in rasterized triangle where pixel values take intermediate values

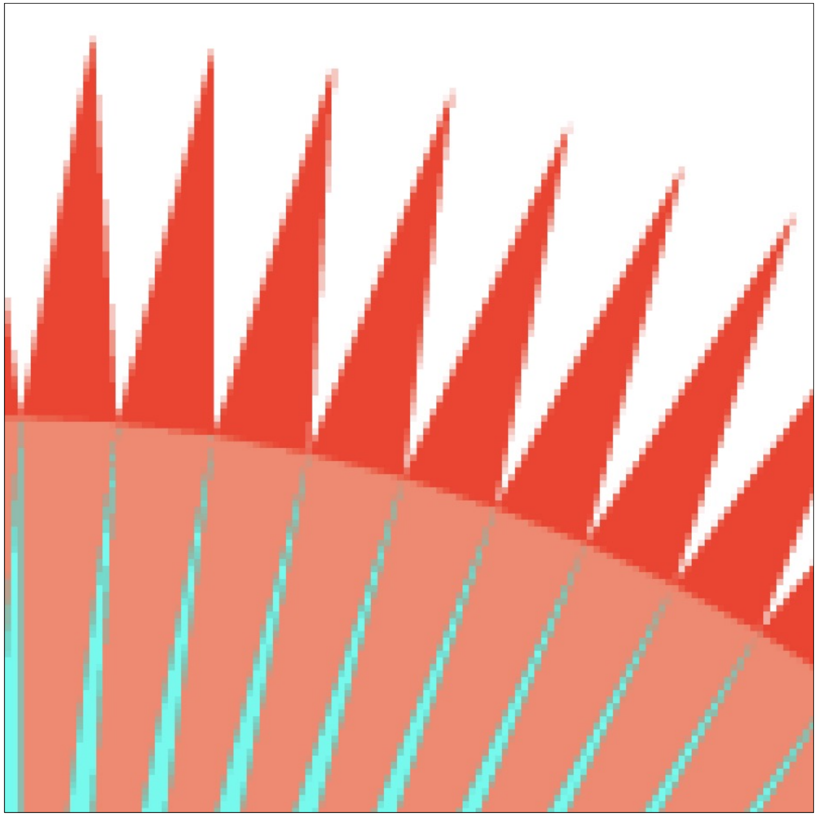
Point Sampling



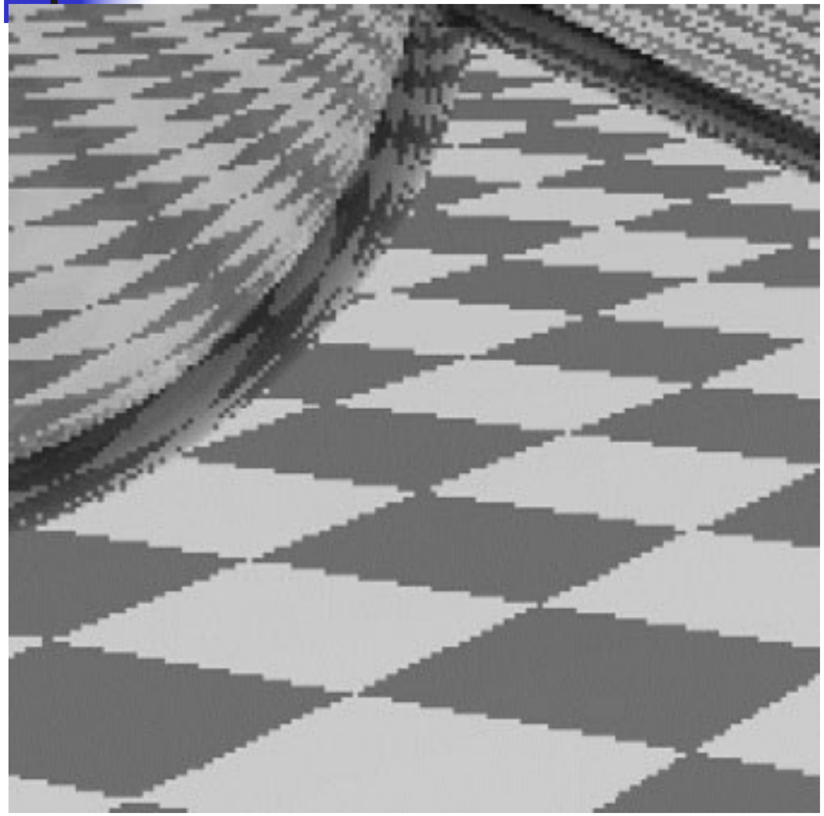


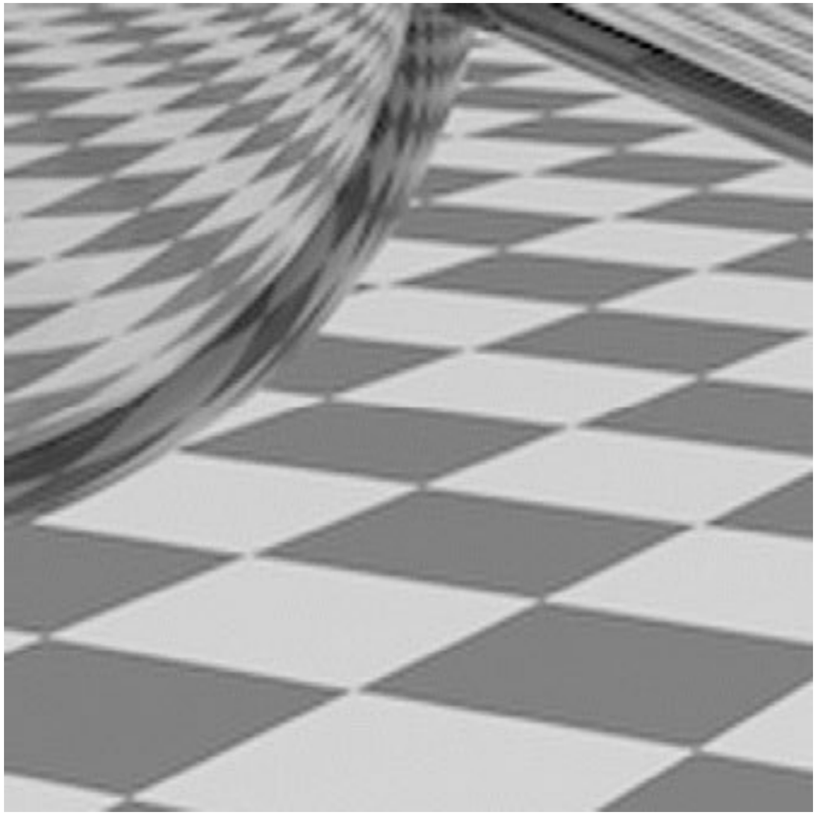
Antialiasing



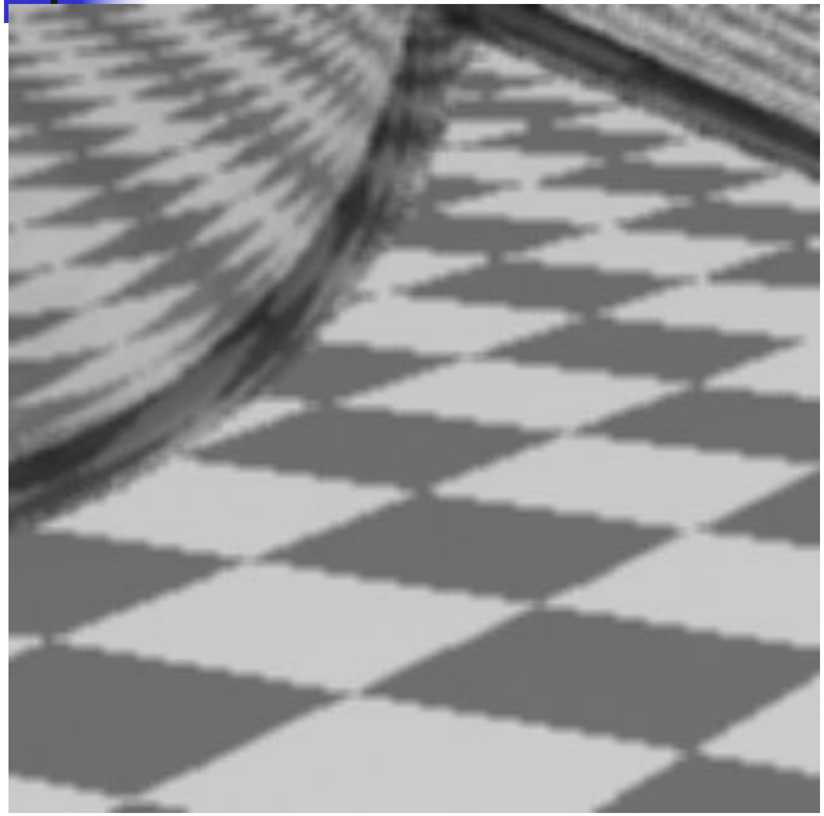


Point Sampling vs Antialiasing

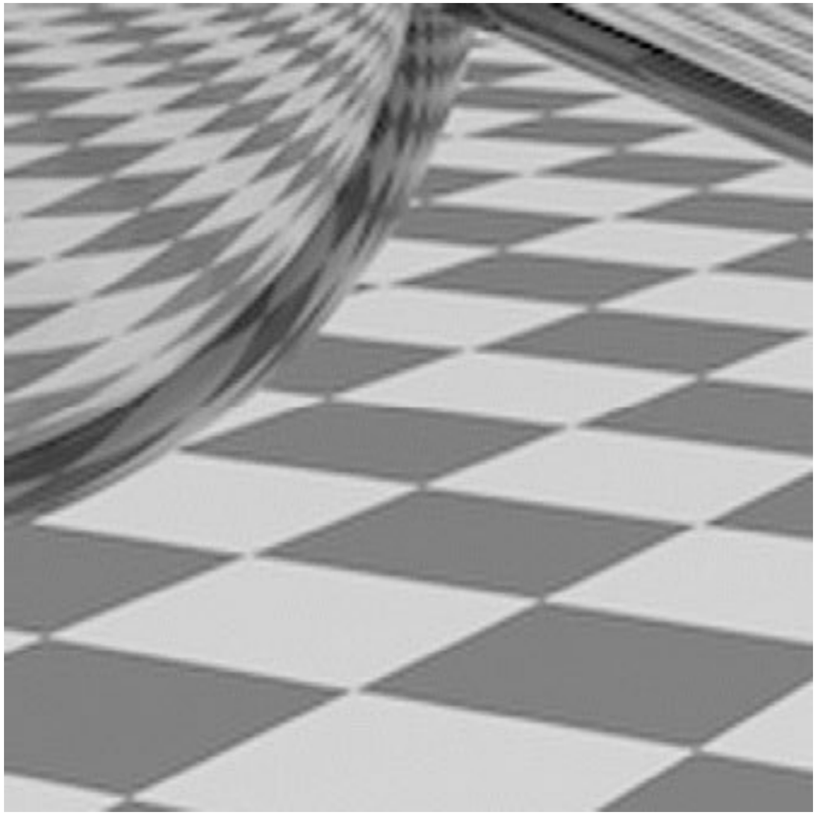




Antialiasing vs Blurred Aliasing



(Sample then filter, WRONG!)



(Filter then sample)

But why?

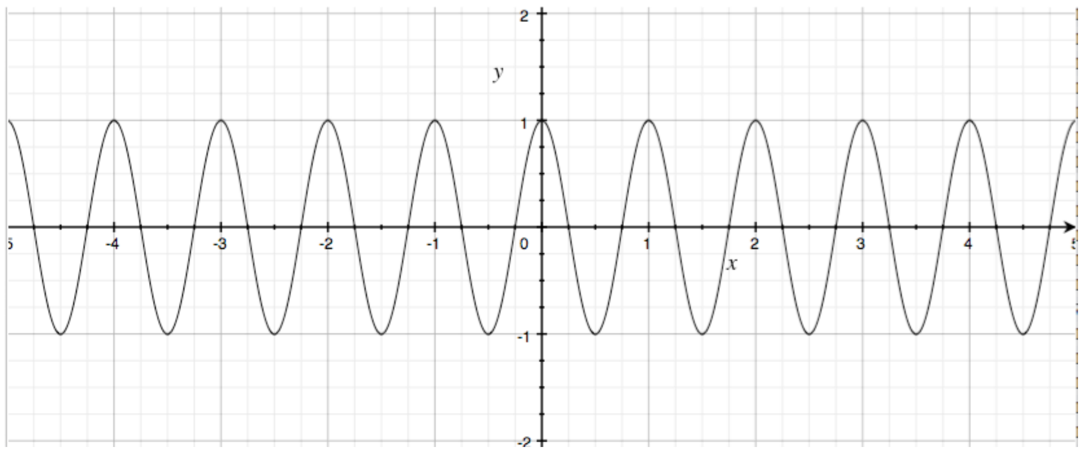
1. Why undersampling introduces aliasing?
2. Why pre-filtering then sampling can do antialiasing?

Let's dig into fundamental reasons

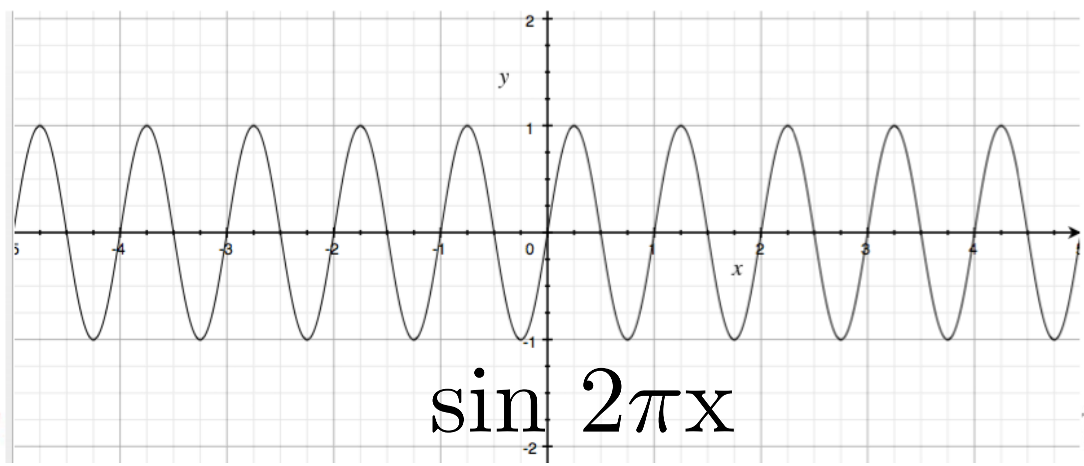
And look at how to implement antialiased rasterization

Frequency Domain

Sines and Cosines



$\cos 2\pi x$

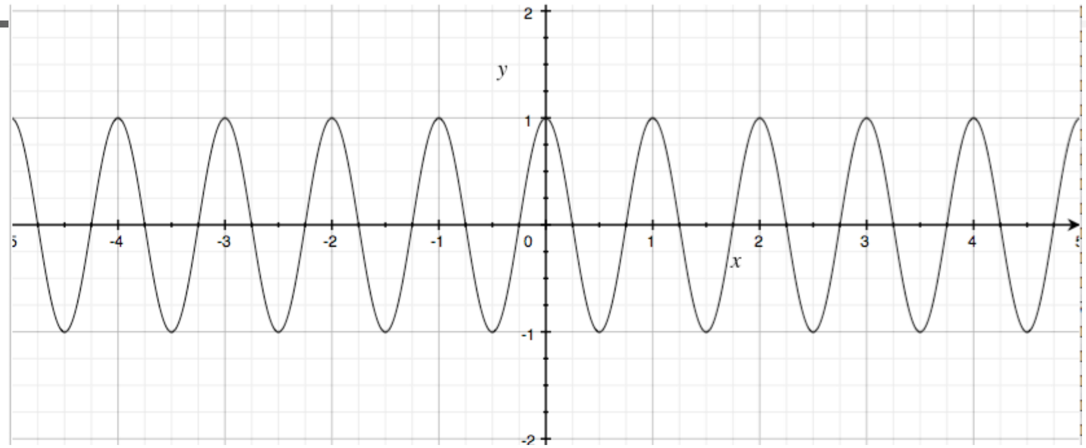


$\sin 2\pi x$

Frequencies

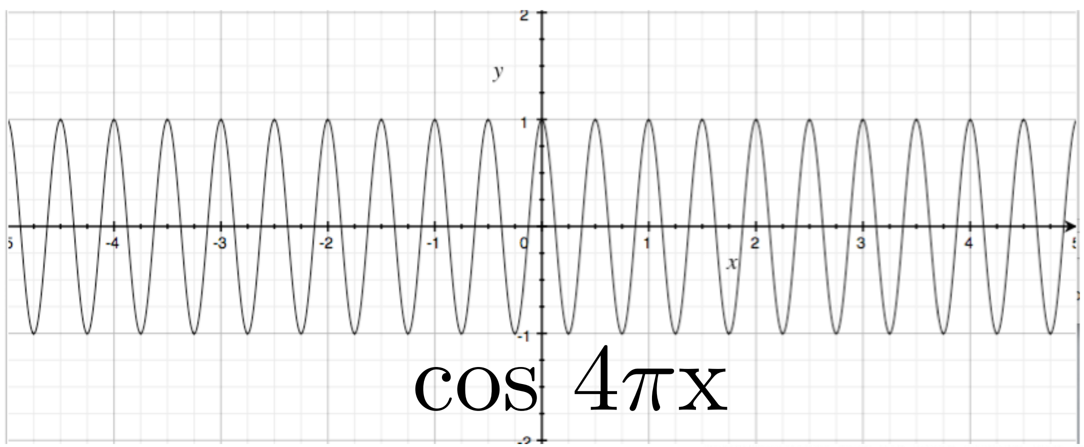
$$f = \frac{1}{T}$$

$$\cos 2\pi fx$$



$$\cos 2\pi x$$

$$f = 1$$

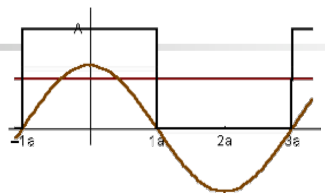


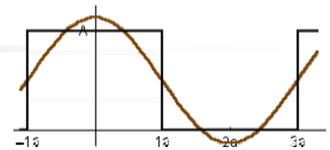
$$\cos 4\pi x$$

$$f = 2$$

Fourier Transform

Represent a function as a weighted sum of sines and cosines



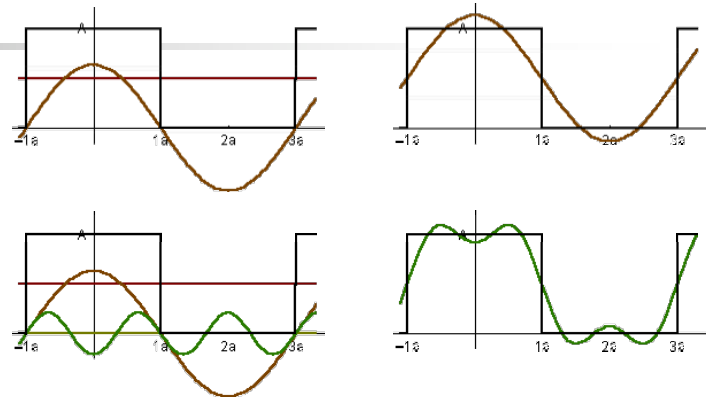


Joseph Fourier 1768 - 1830

$$f(x) = \frac{A}{2} + \frac{2A\cos(t\boldsymbol{\omega})}{\boldsymbol{\pi}} + \cdots$$

Fourier Transform

Represent a function as a weighted sum of sines and cosines



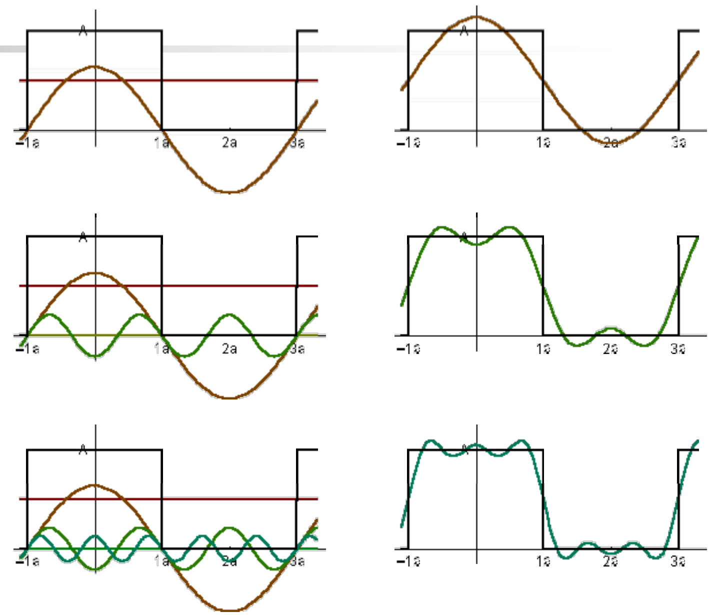
Joseph Fourier 1768 - 1830

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

Fourier Transform

Represent a function as a weighted sum of sines and cosines

Joseph Fourier 1768 - 1830

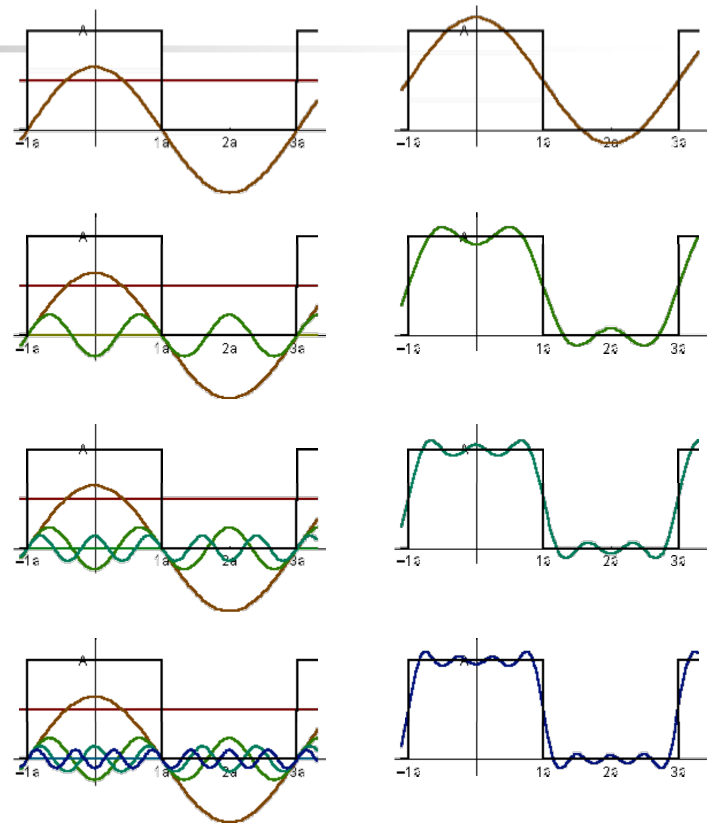


$$f(x) = \frac{A}{2} + \frac{2A\cos(t\boldsymbol{\omega})}{\pi} - \frac{2A\cos(3t\boldsymbol{\omega})}{3\pi} + \frac{2A\cos(5t\boldsymbol{\omega})}{5\pi}$$

Fourier Transform

Represent a function as a weighted sum of sines and cosines

Joseph Fourier 1768 - 1830



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

Fourier Transform Decomposes A Signal Into Frequencies

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

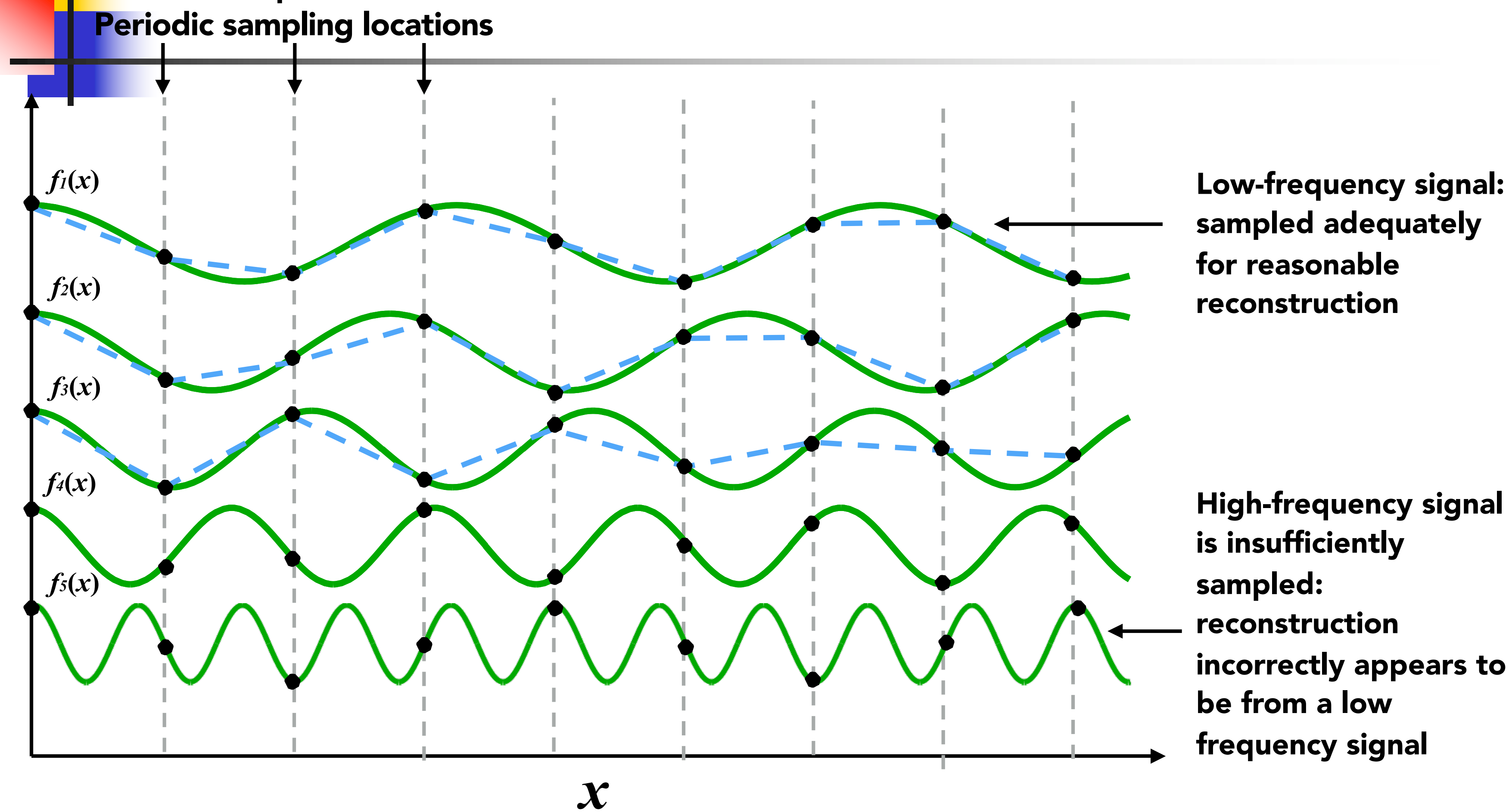
spatial domain → Fourier transform → frequency domain

spatial domain ← Inverse transform ← frequency domain

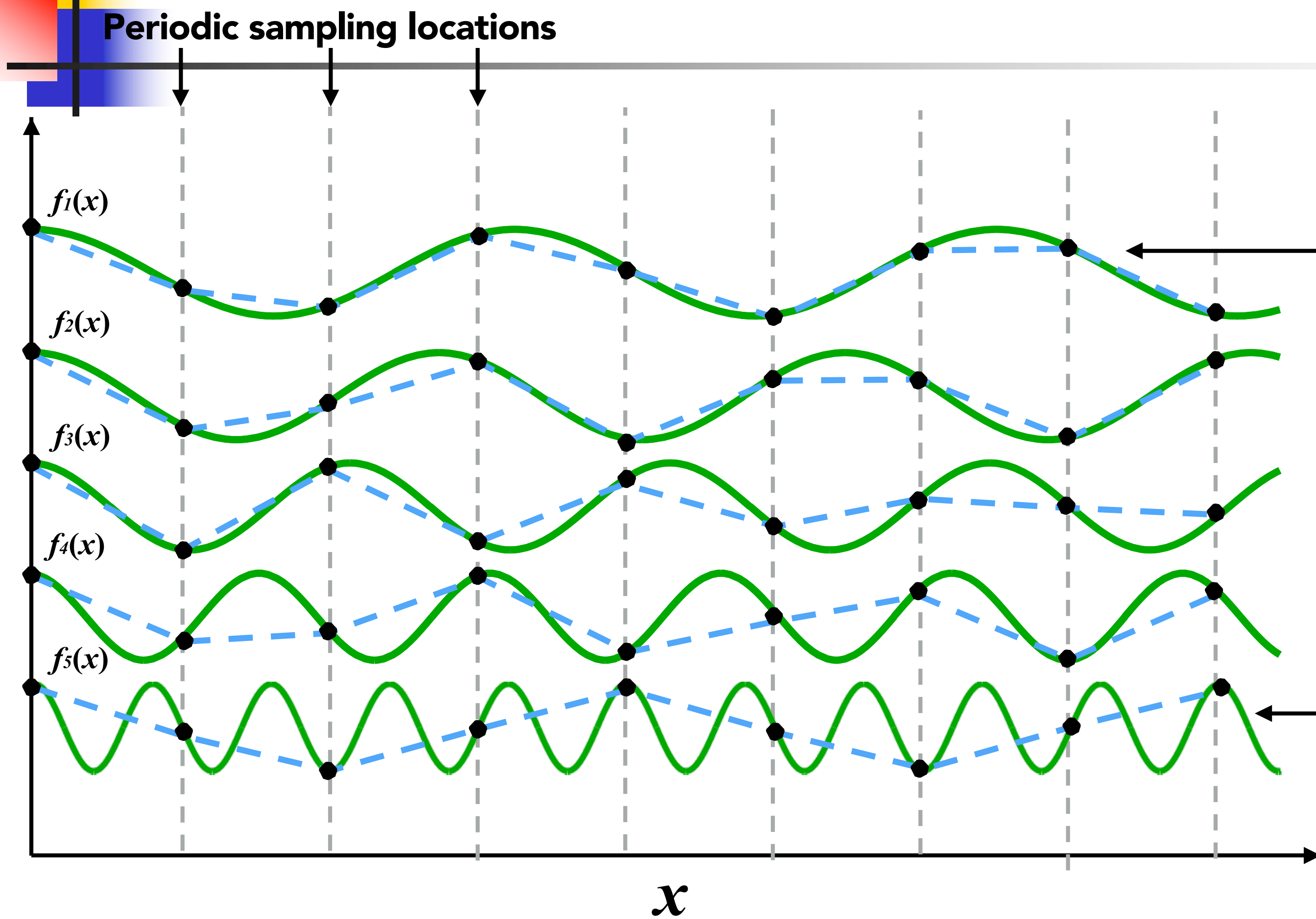
$$f(x) = \int_{-\infty}^{\infty} F(\omega) e^{2\pi i \omega x} d\omega$$

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

Higher Frequencies Need Faster Sampling



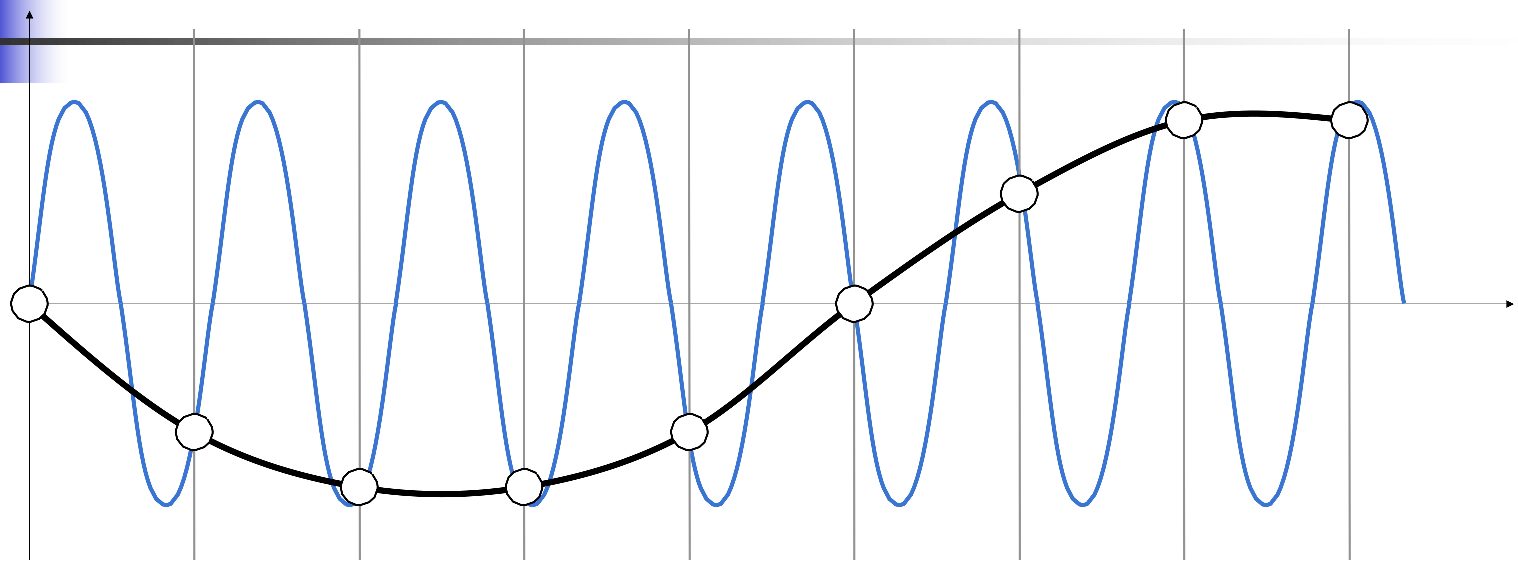
Higher Frequencies Need Faster Sampling



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

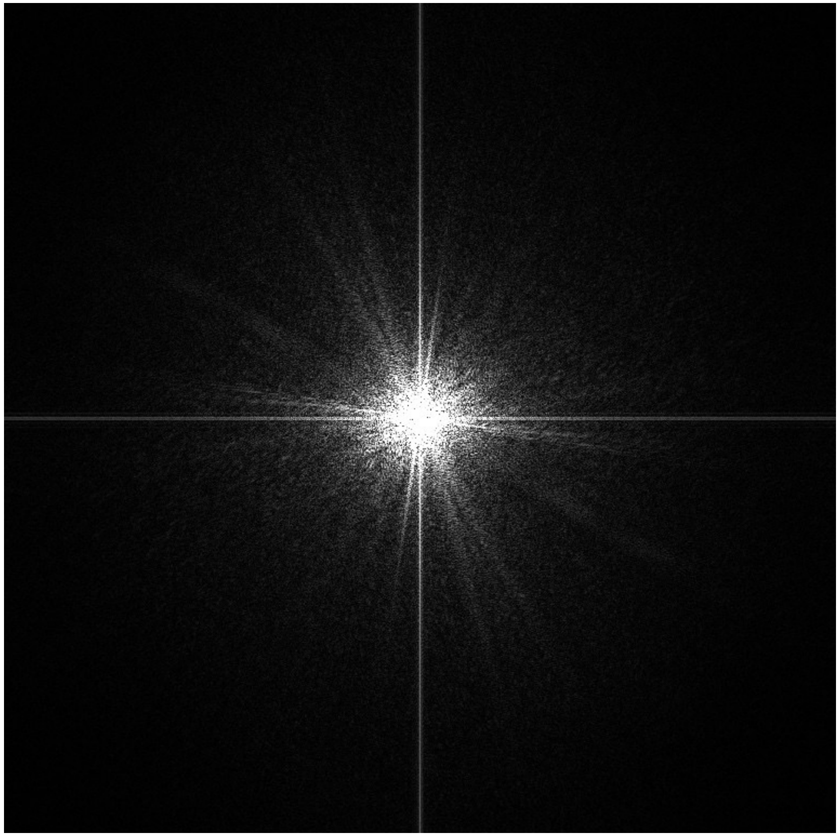


High-frequency signal is insufficiently sampled: samples erroneously appear to be from a low-frequency signal

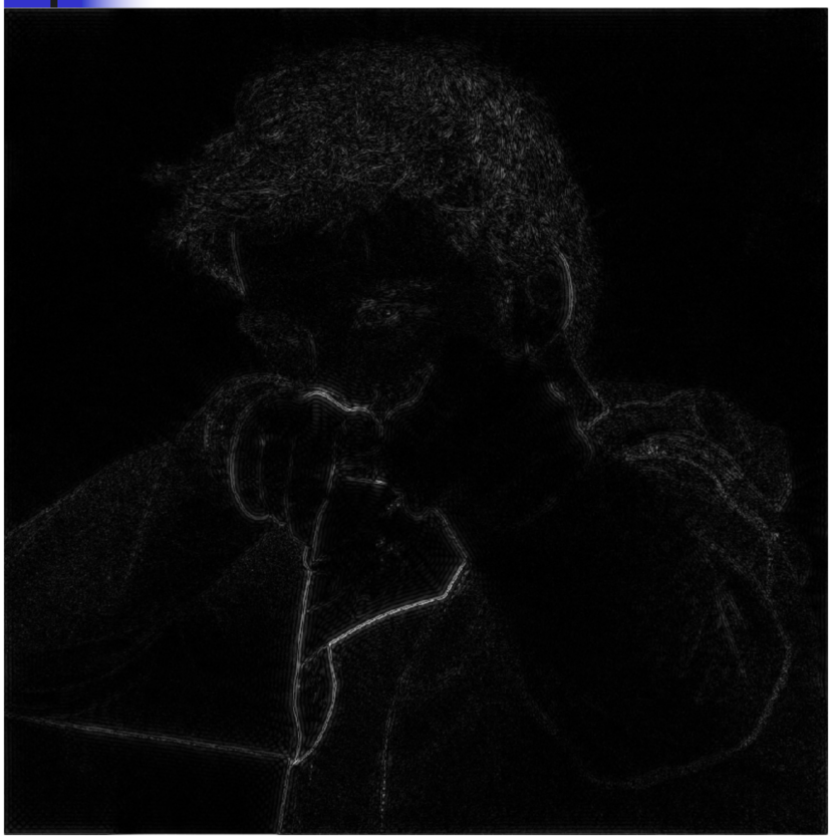
Two frequencies that are indistinguishable at a given sampling rate are called "aliases" (*misidentification of a signal frequency, Introducing distortion or error*).

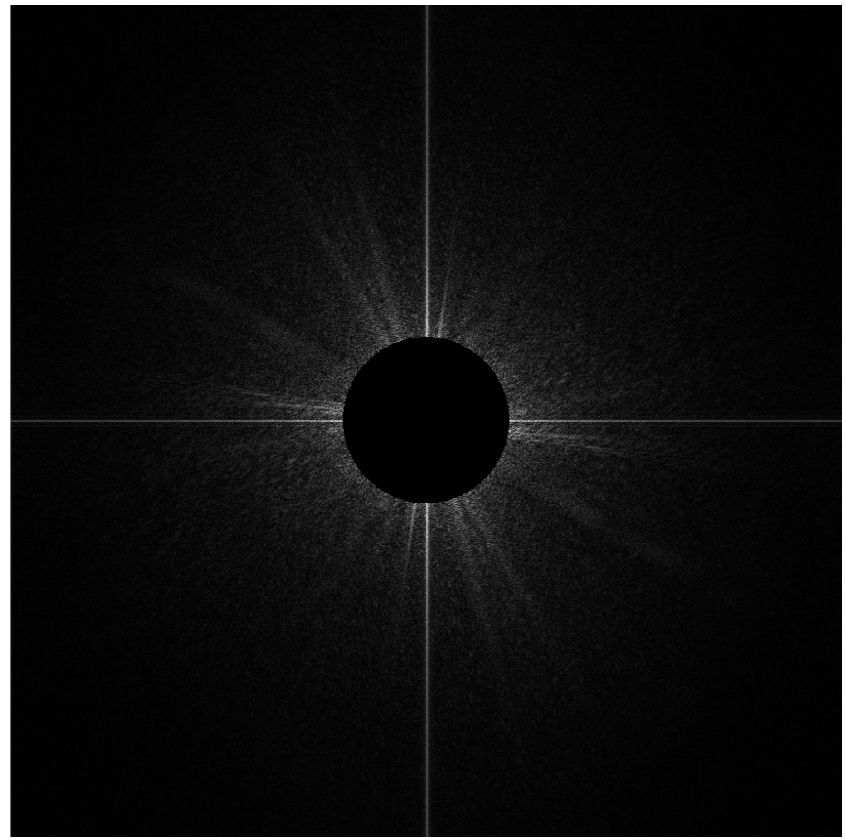
Filtering = Getting rid of certain frequency contents

Visualizing Image Frequency Content



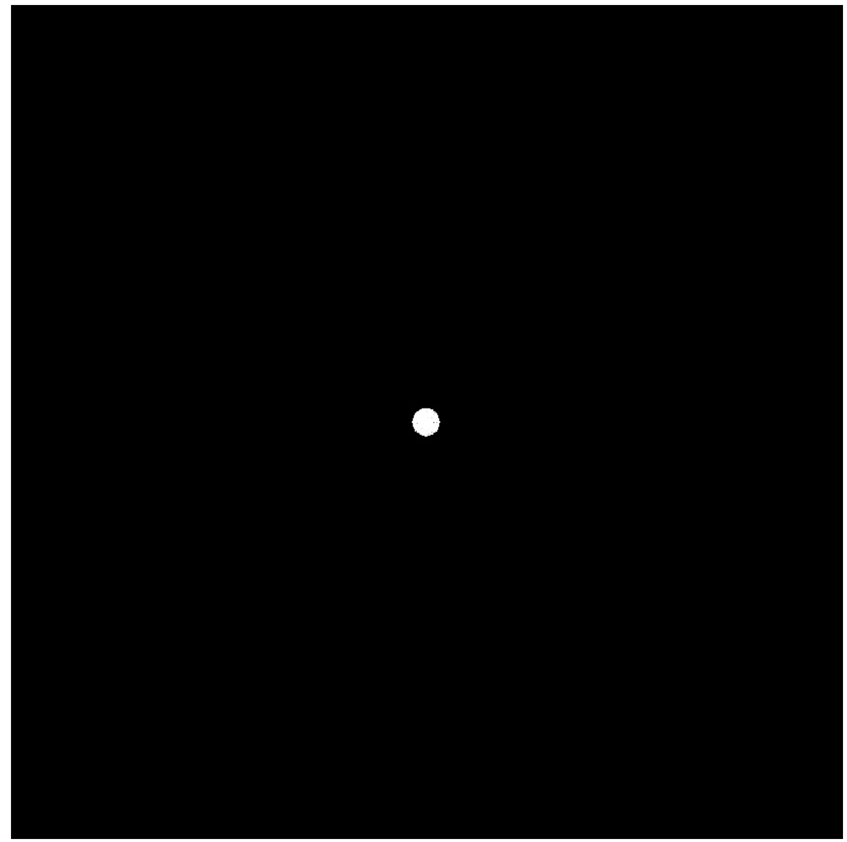
Filter Out Low Frequencies Only (Edges)





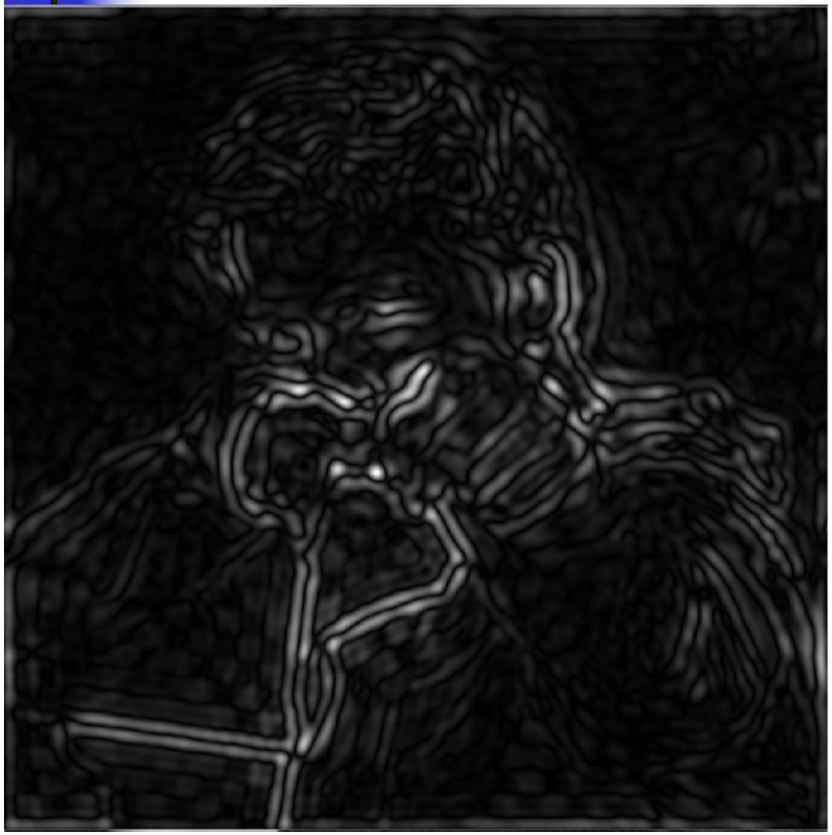
High-pass filter

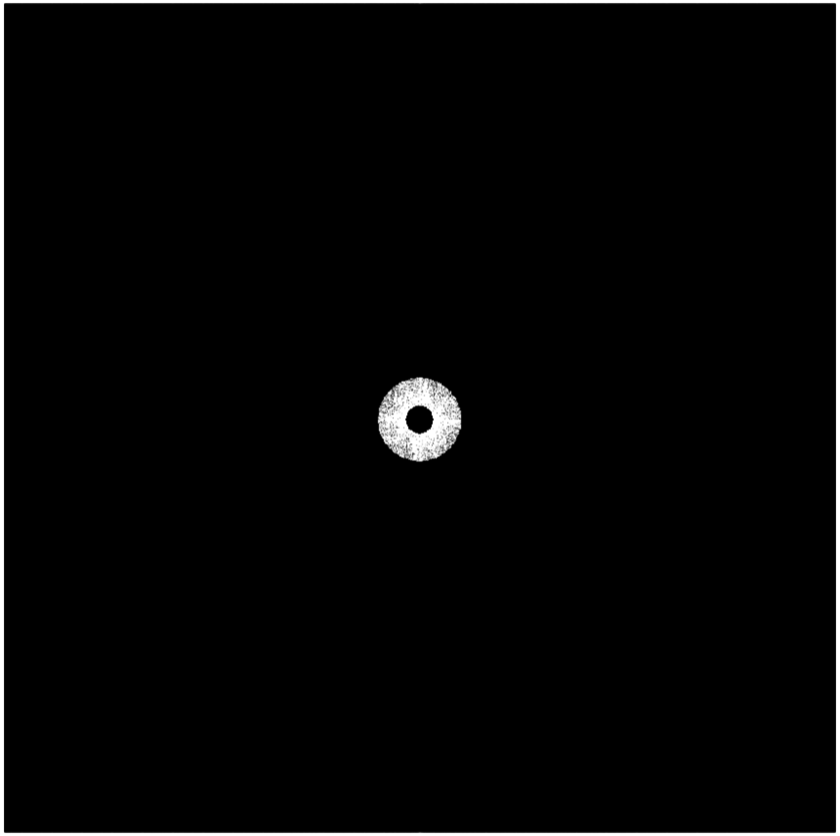
Filter Out High Frequencies (Blur)



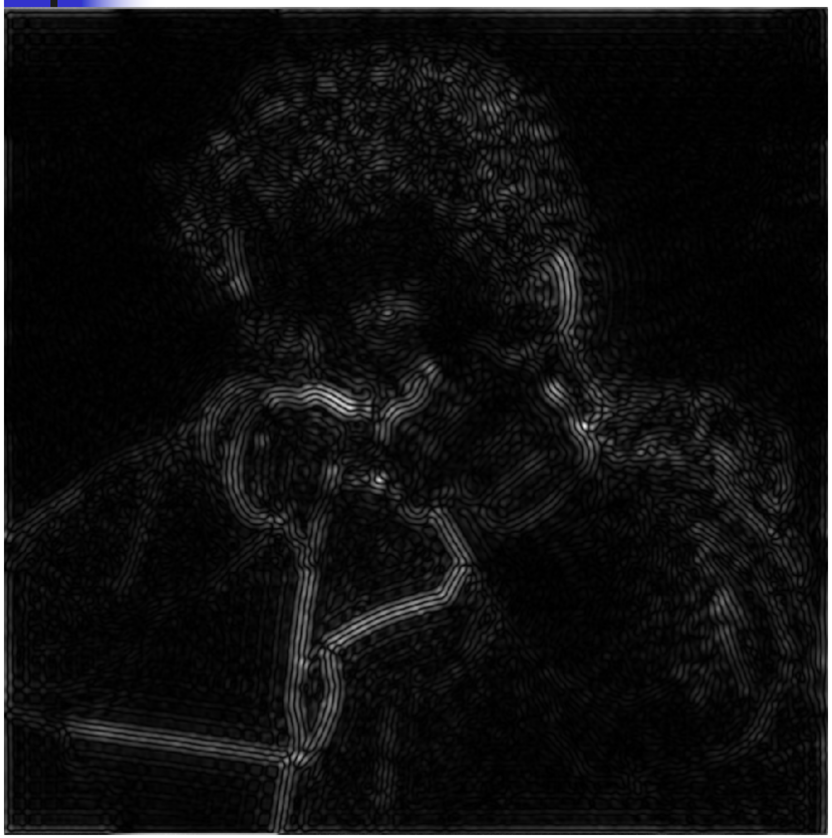
Low-pass filter

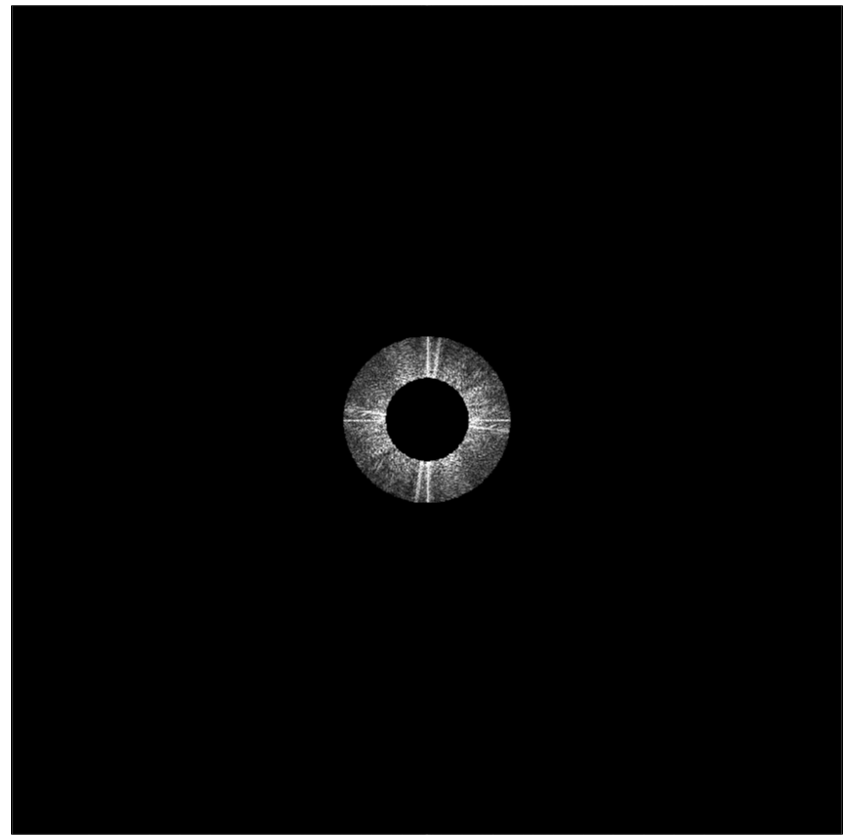
Filter Out Low and High Frequencies





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

Signal

1	3	5	3	7	1	3	8	6	4

Filter

1/4	1/2	1/4

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

Result

	3								

Convolution

Signal

1	3	5	3	7	1	3	8	6	4

Filter

1/4	1/2	1/4

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

Result

	3	4							

Convolution Theorem

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

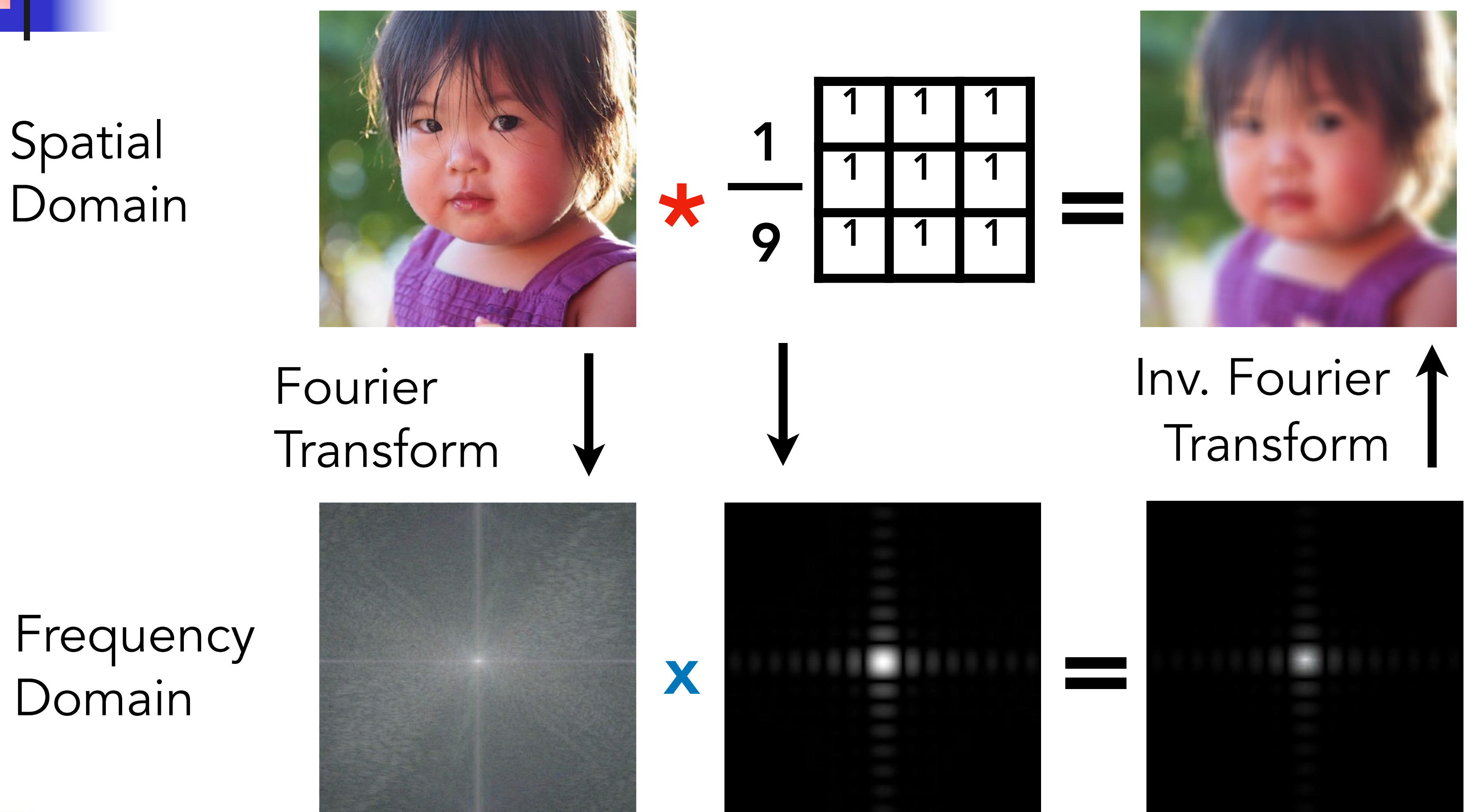
Option 1:

- Filter by convolution in the spatial domain

Option 2:

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

Convolution Theorem

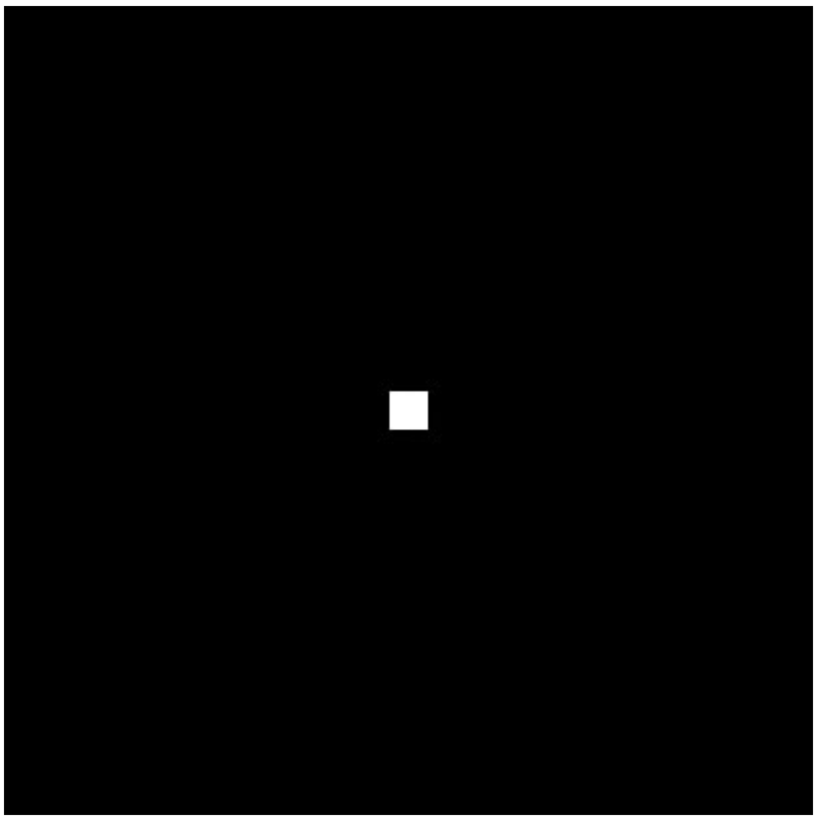


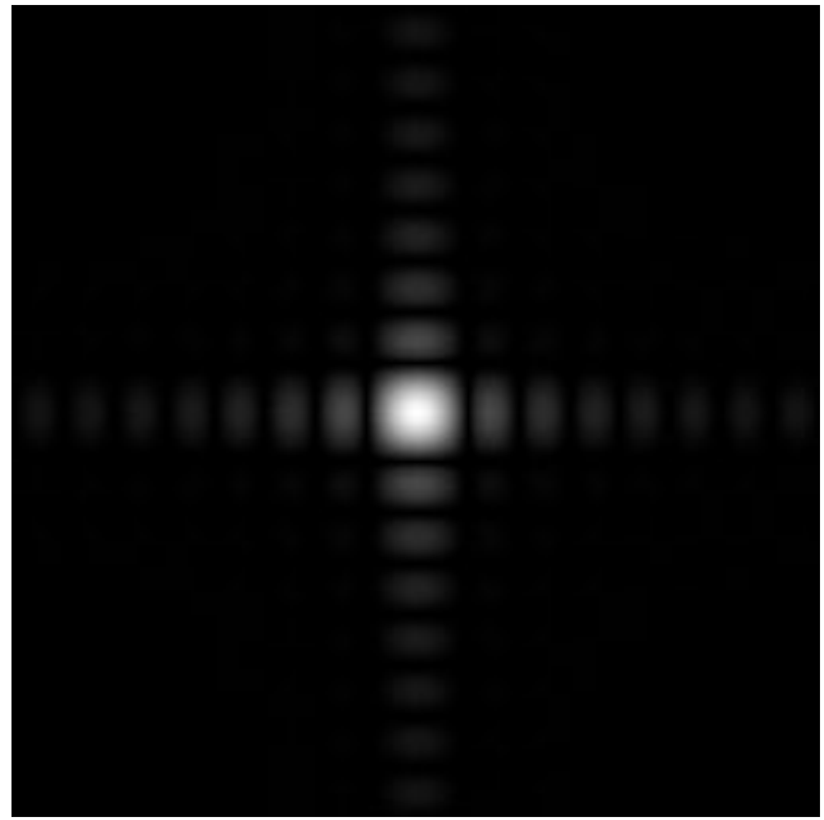
Box Filter

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

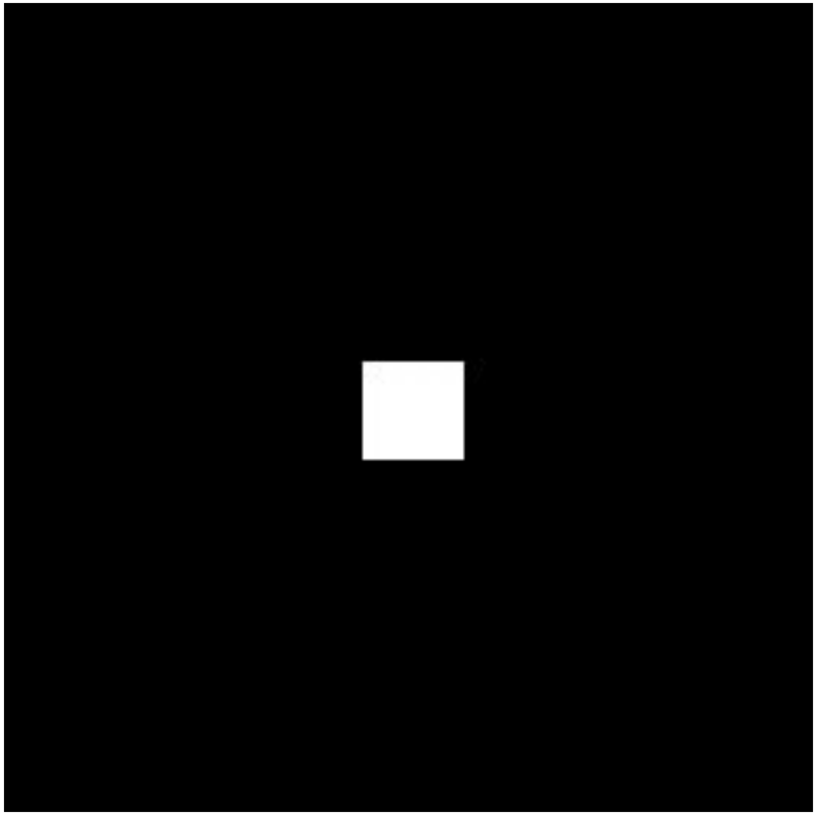
Example: 3x3 box filter

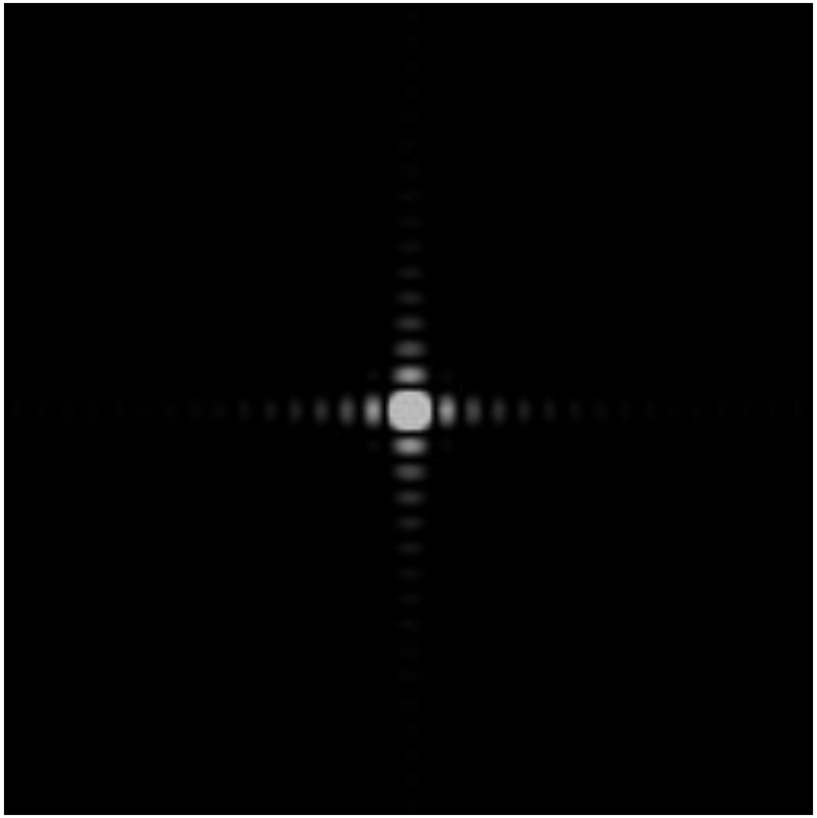
Box Function = "Low Pass" Filter





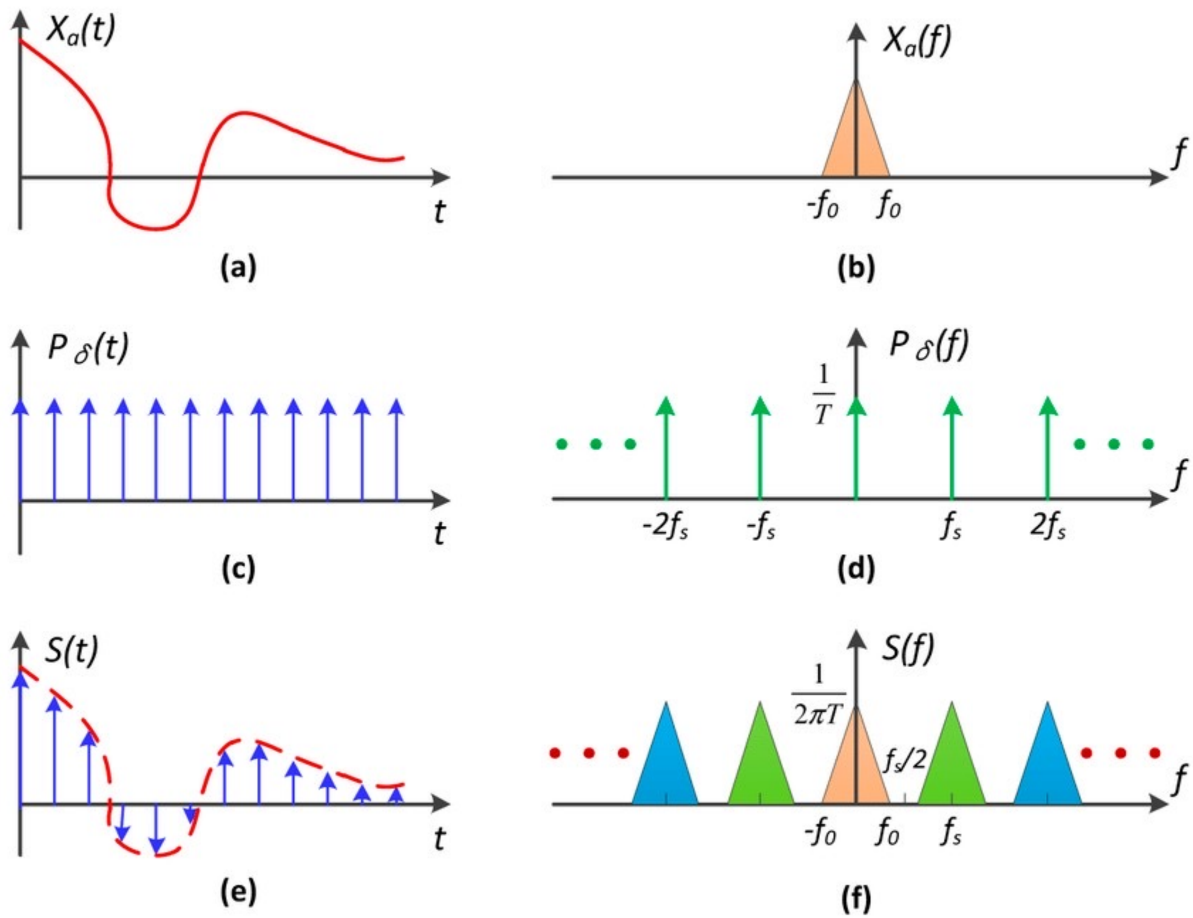
Wider Filter Kernel = Lower Frequencies





Sampling = Repeating Frequency Contents

Sampling = Repeating Frequency Contents

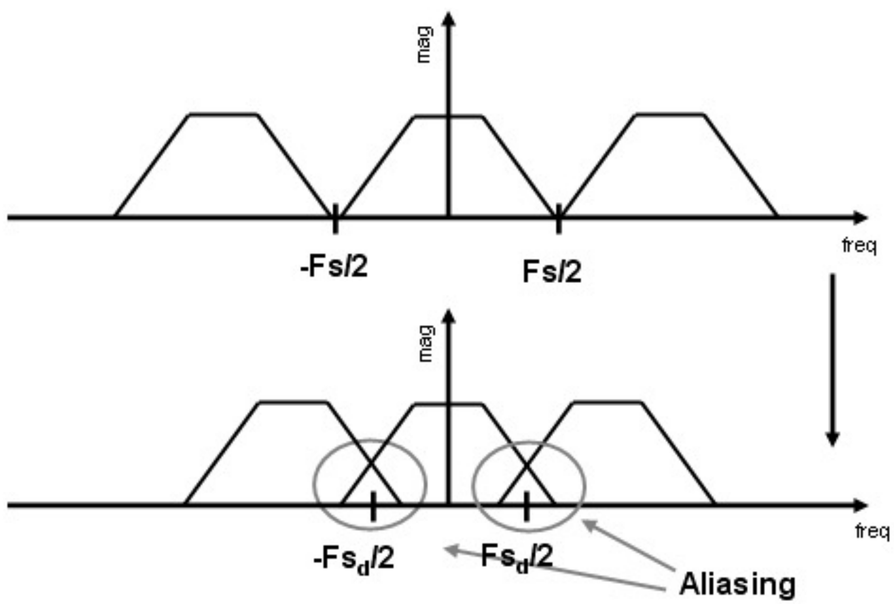


https://www.researchgate.net/figure/The-evolution-of-sampling-theorem-a-The-time-domain-of-the-band-limited-signal-and-b_fig5_301556095

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

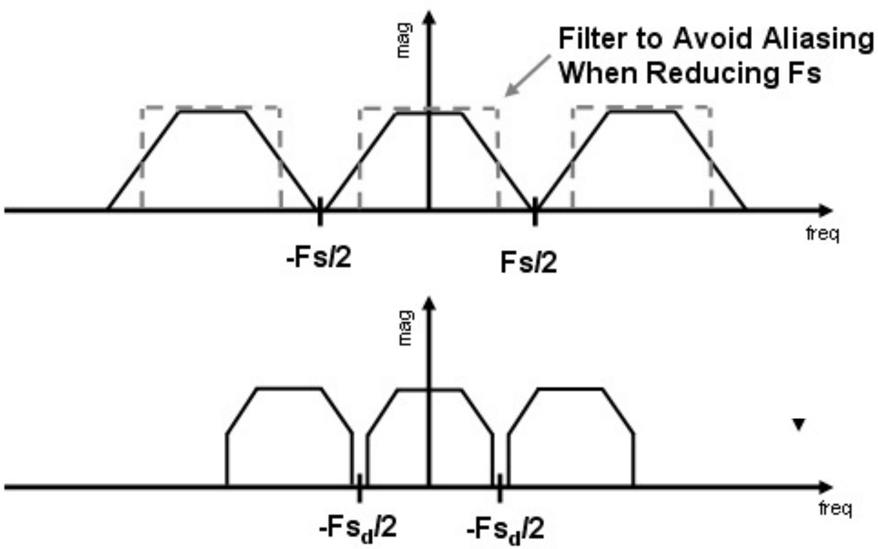
Option 2: Antialiasing

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

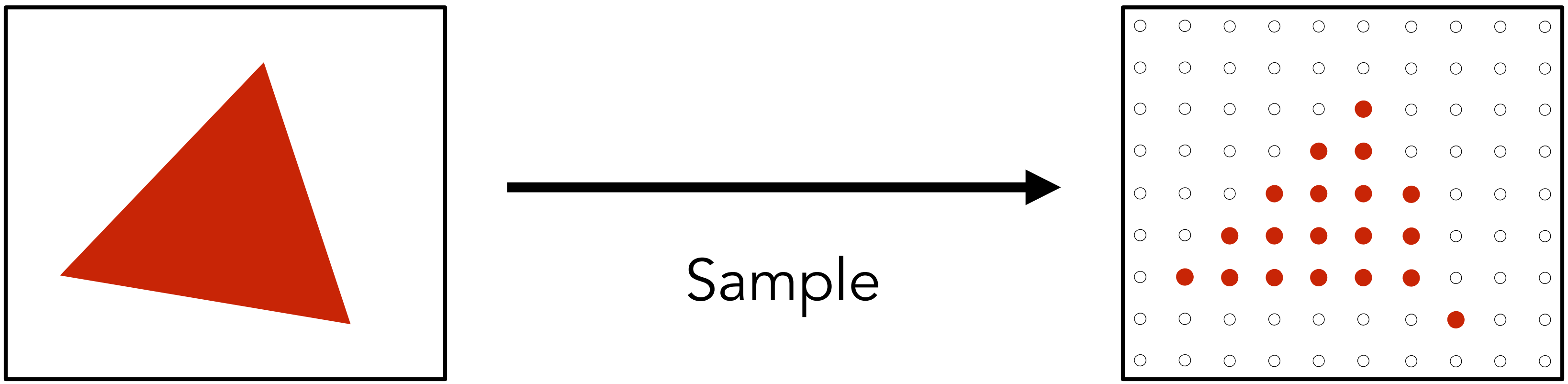
Antialiasing = Limiting, then repeating

Filtering

Then sparse sampling

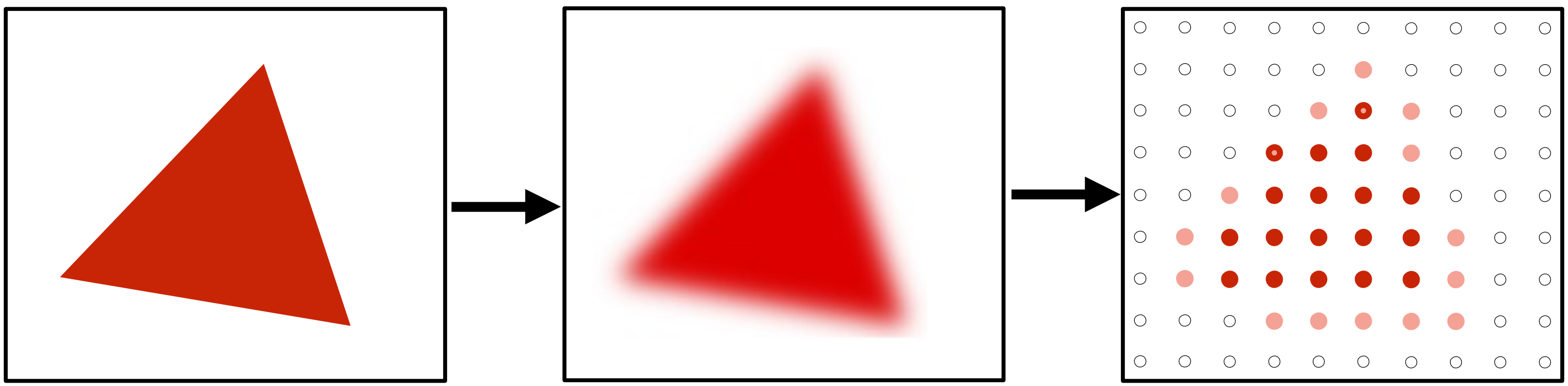


Regular Sampling



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

Antialiased Sampling



Pre-Filter

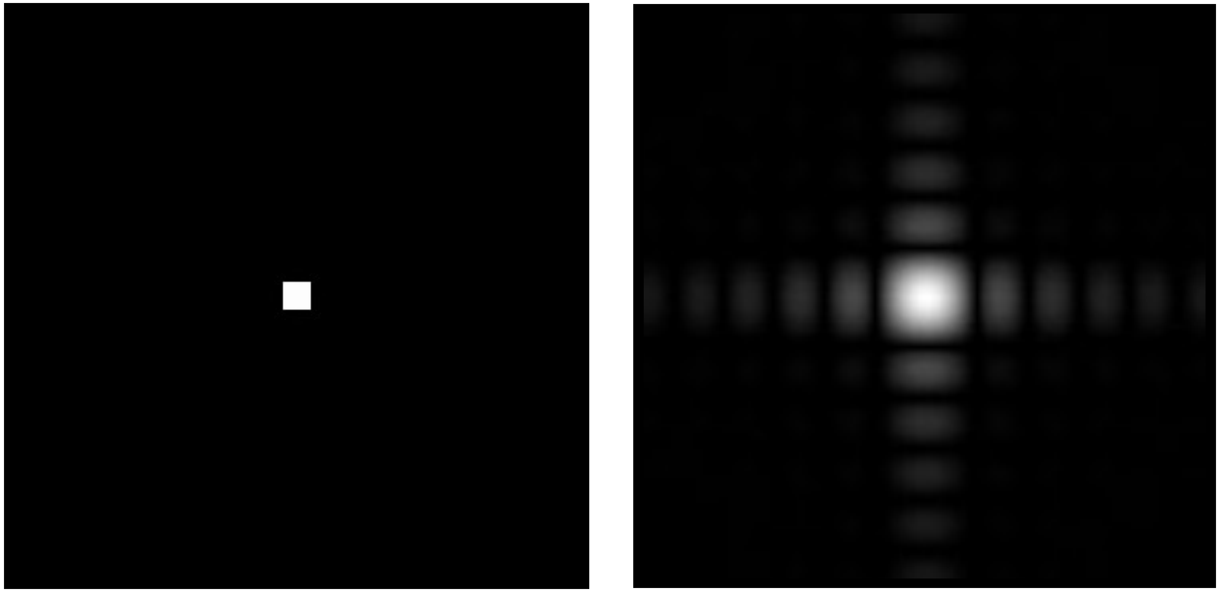
(remove frequencies above Nyquist)

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:

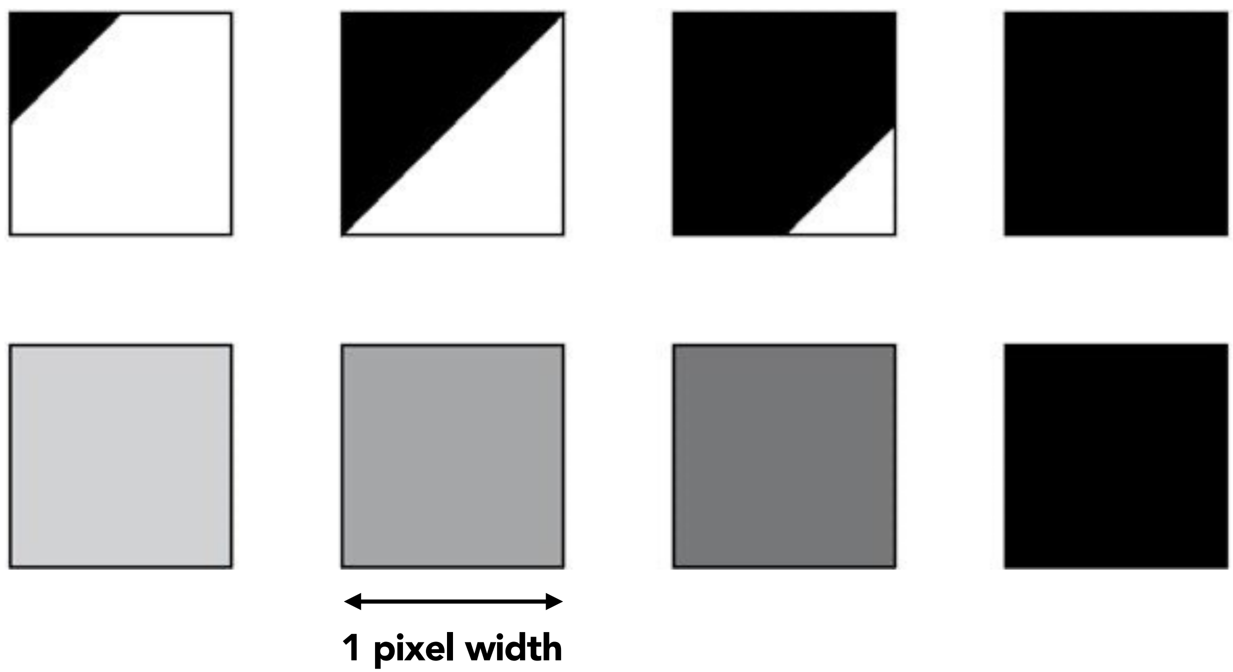
- **Convolve** f(x,y) by a 1-pixel box-blur
  - Recall: convolving = filtering = averaging
- **Then sample** at every pixel's center

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.

Original

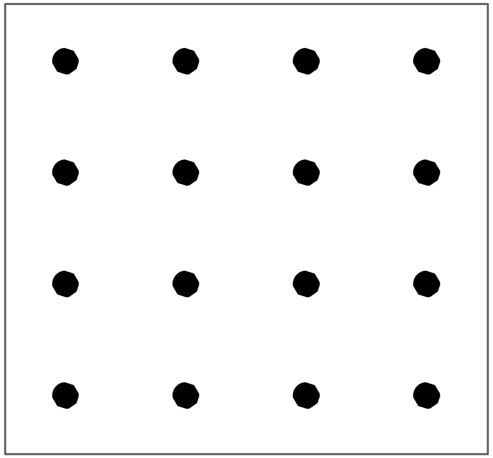
Filtered



Antialiasing By Supersampling (MSAA)

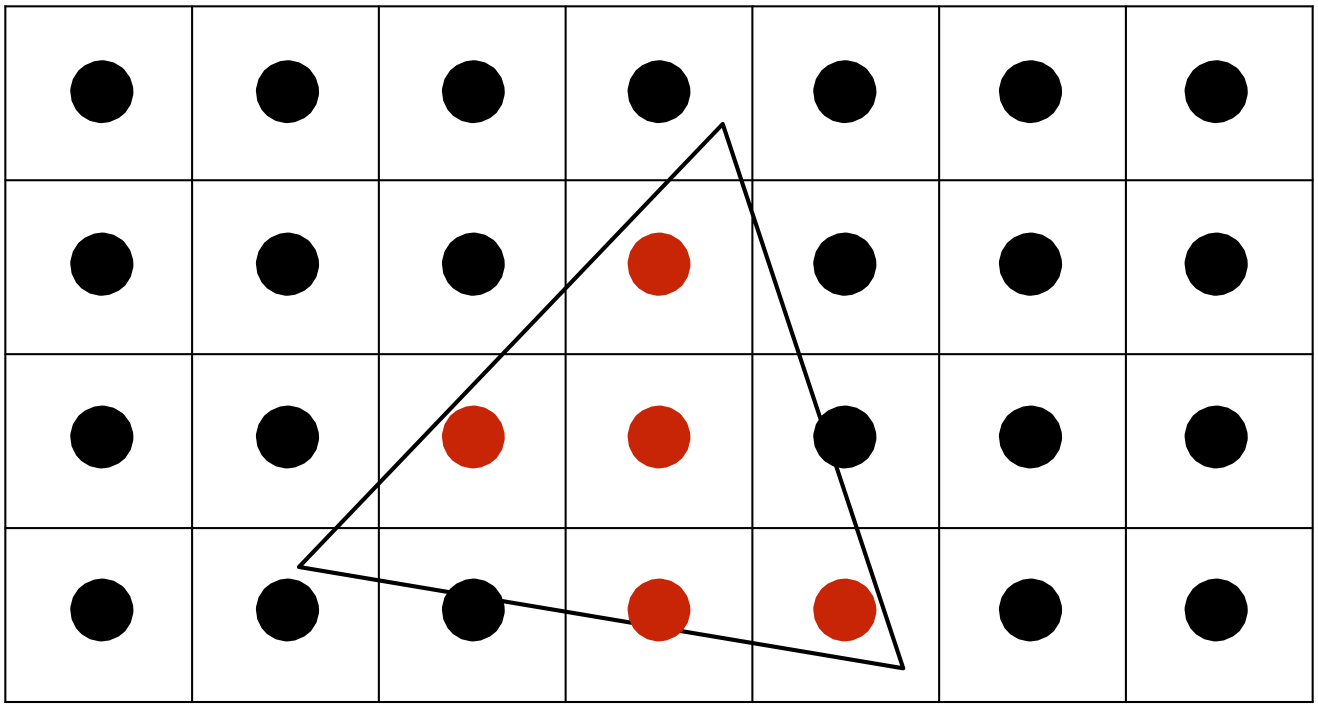
Supersampling

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



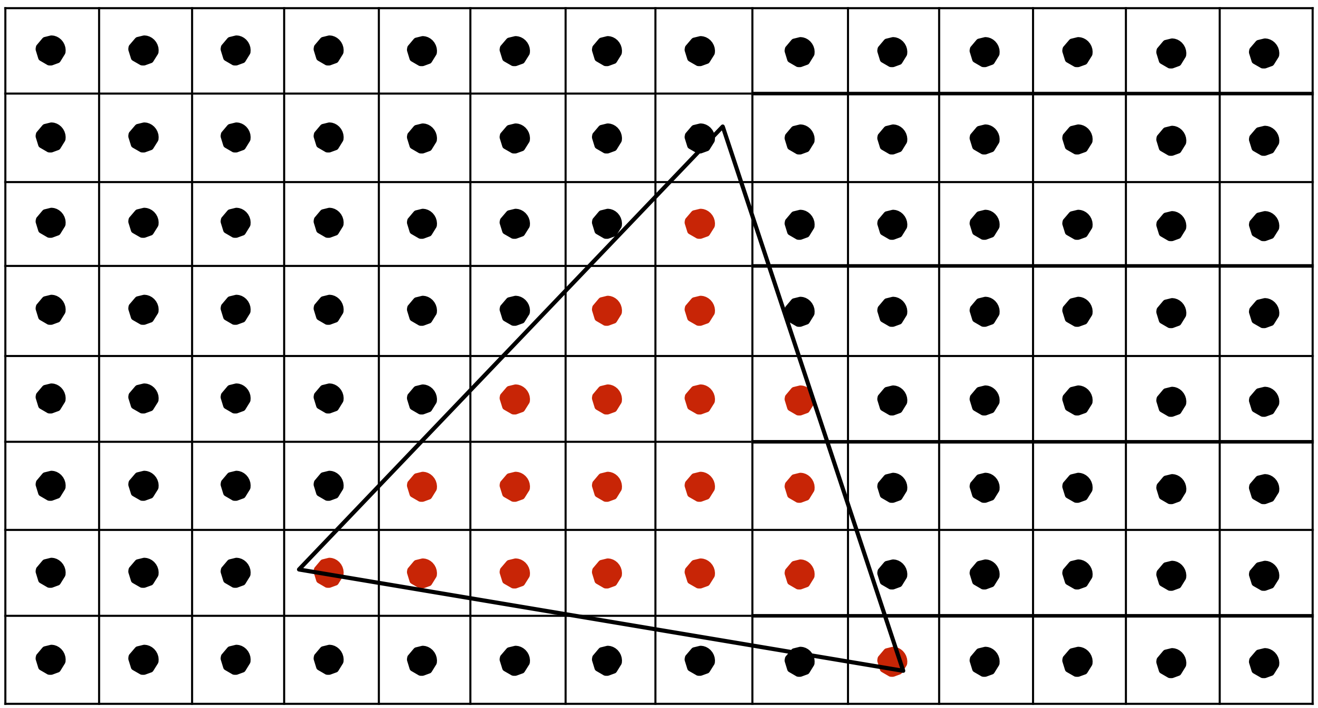
4x4 supersampling

Point Sampling: One Sample Per Pixel



Supersampling: Step 1

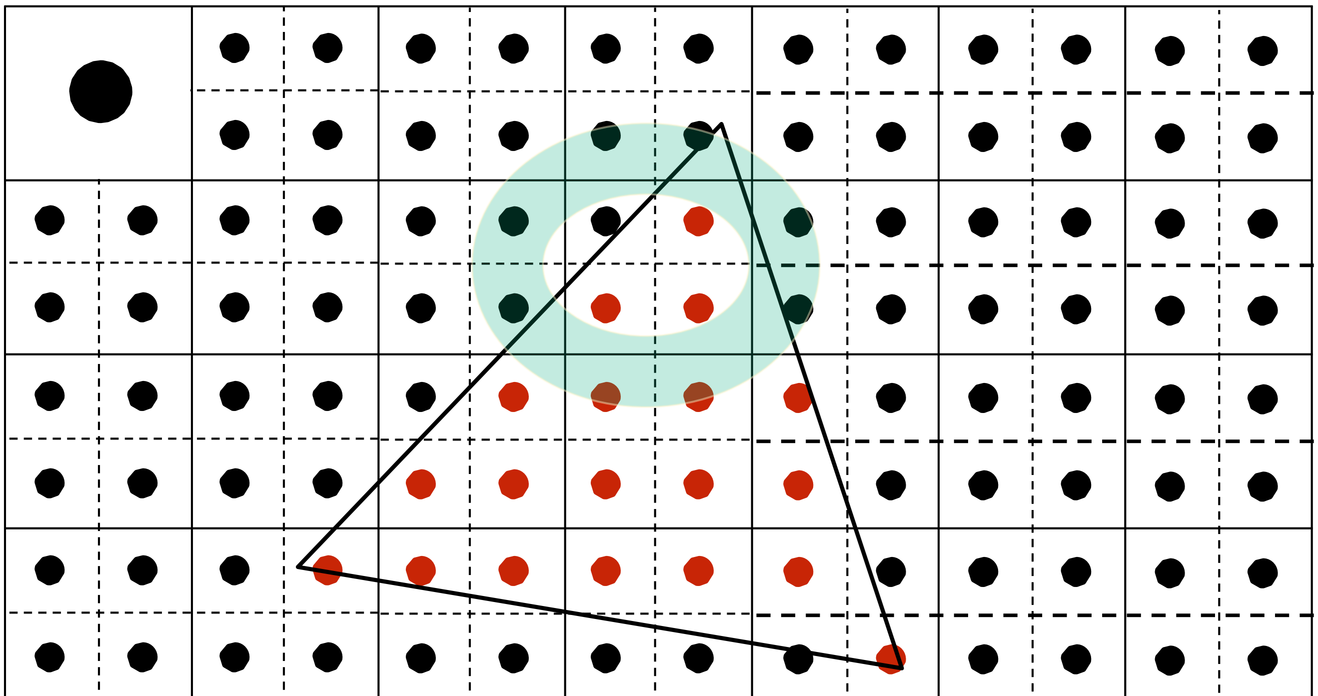
Take NxN samples in each pixel.



2x2 supersampling

Supersampling: Step 2

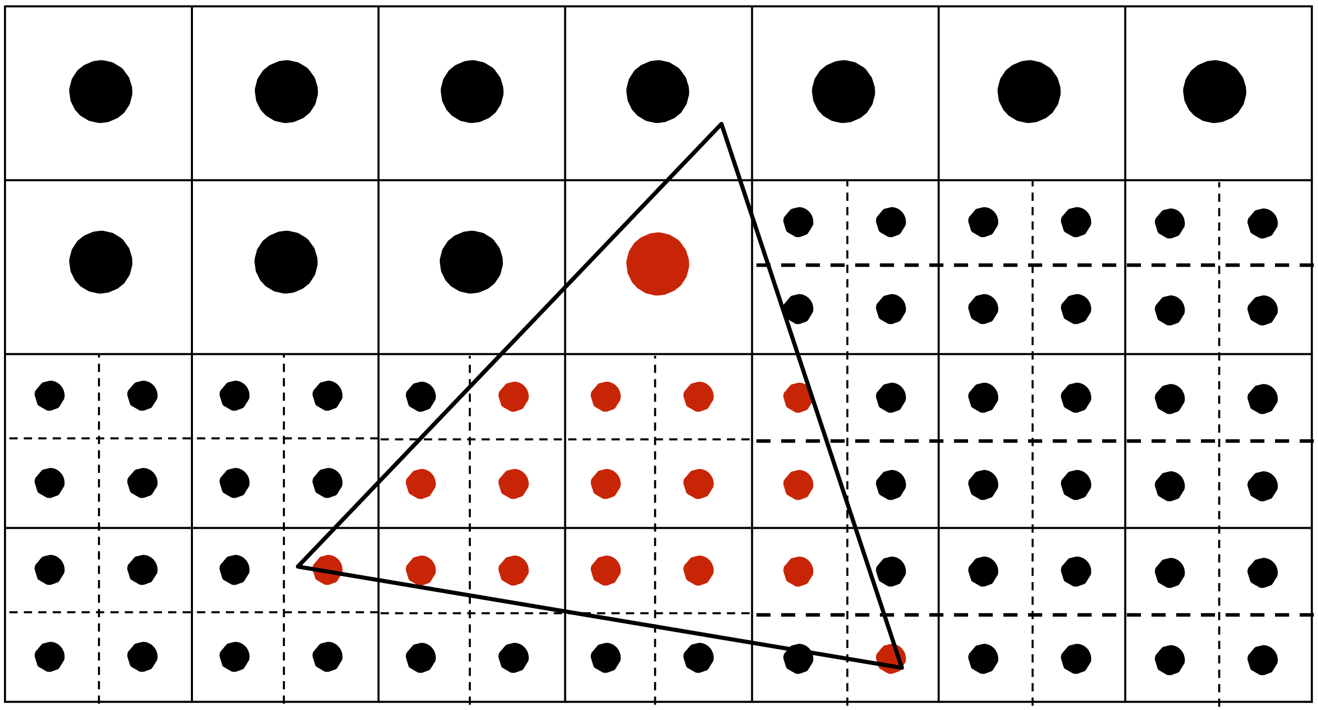
Average the NxN samples “inside” each pixel.



Averaging down

Supersampling: Step 2

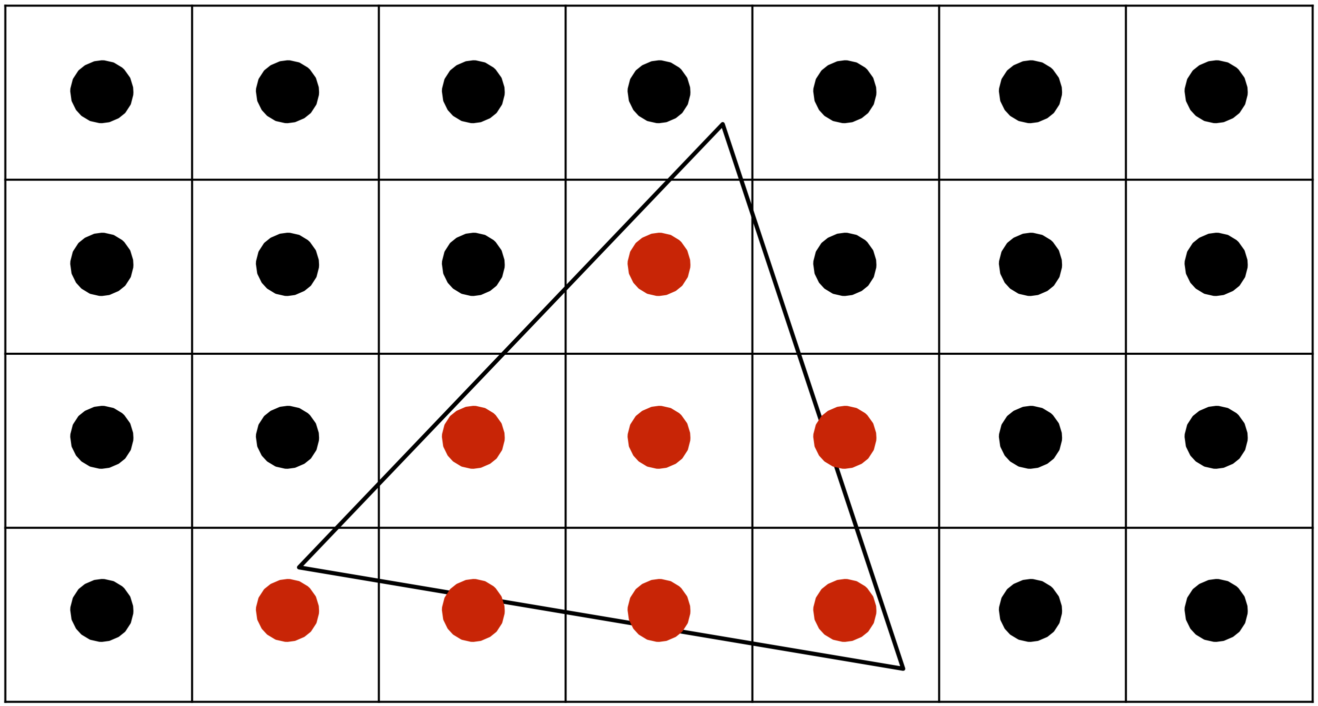
Average the NxN samples “inside” each pixel.



Averaging down

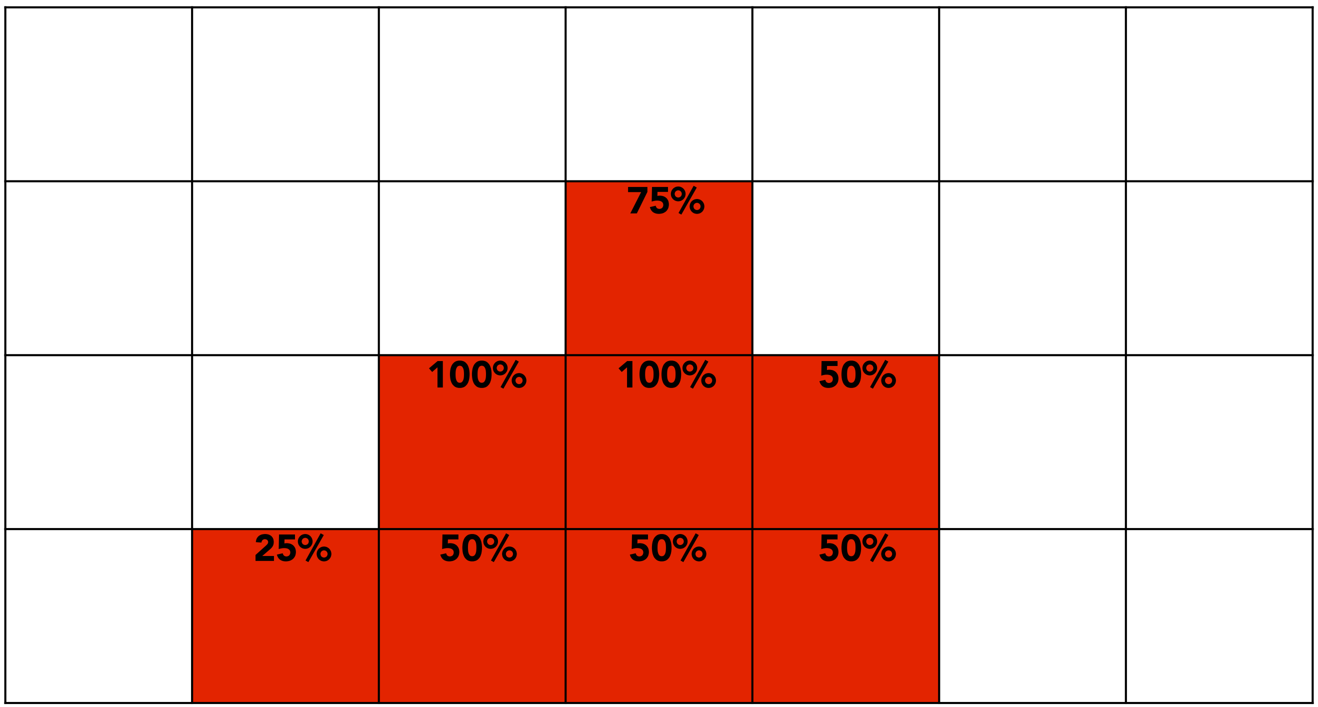
Supersampling: Step 2

Average the NxN samples “inside” each pixel.

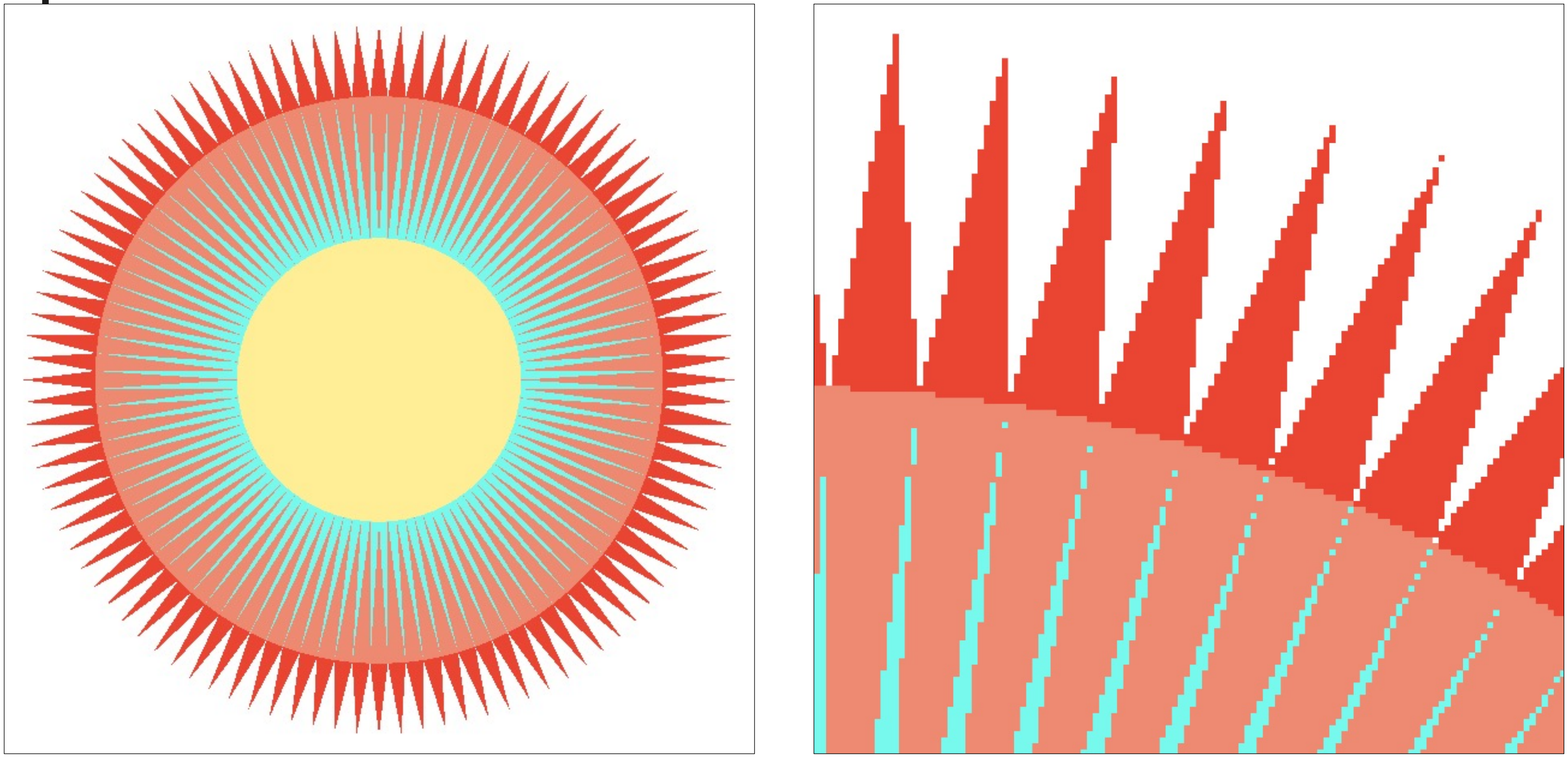


Supersampling: Result

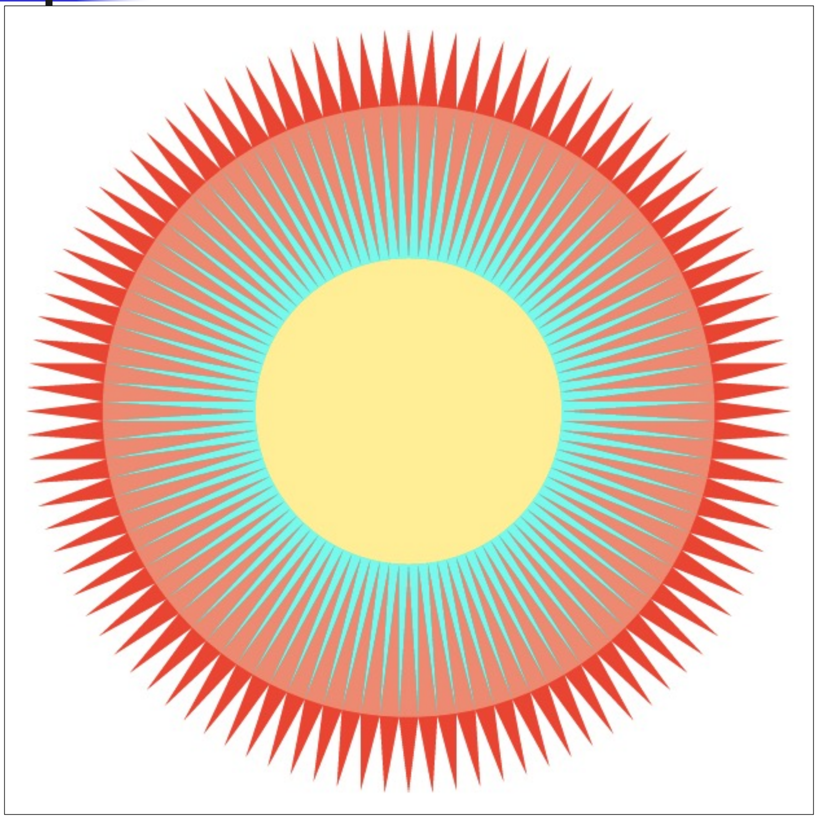
This is the corresponding signal emitted by the display

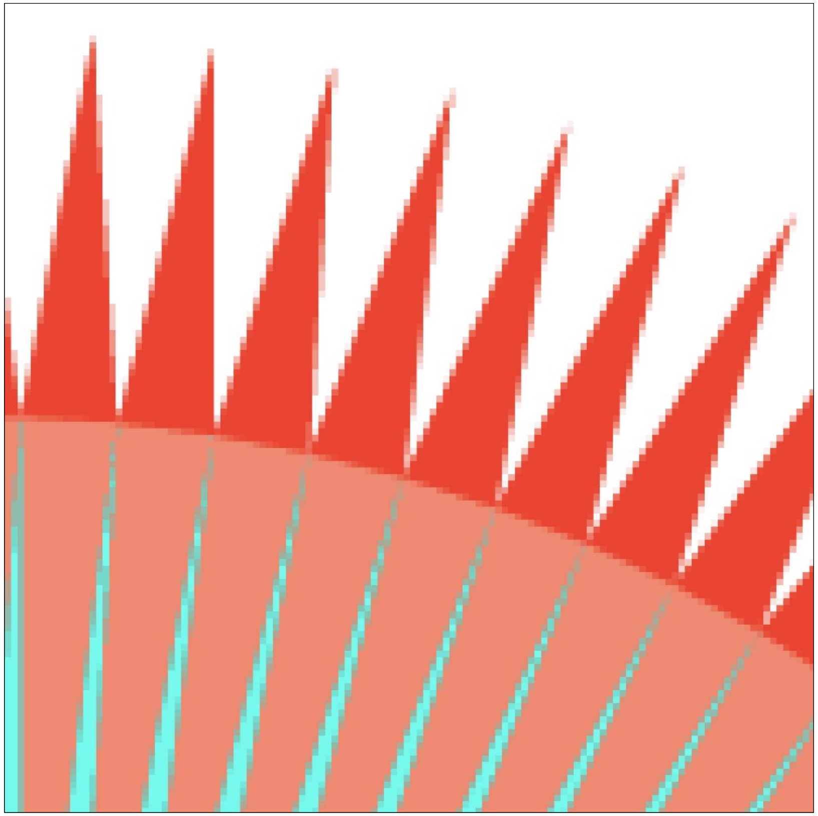


Point Sampling



4x4 Supersampling





Antialiasing Today

No free lunch!

- What's the cost of MSAA?

Milestones

- 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!

(And thank Prof. lingqi Yan (UCSB) for many of the slides!)