Lecture 4: Sampling and Convolution

Yuhao Zhu

http://yuhaozhu.com
yzhu@rochester.edu

CSC 259/459, Fall 2024
Computer Imaging & Graphics

Logistics

- Written assignment 1 is up and is due Sept. 11 11:30 AM.
- You can work in groups of 2.
- Final project due date is 12/16.
- Start thinking and talking to me about your final project idea.

The Roadmap

Theoretical Preliminaries

Human Visual Systems

Color in Nature, Arts, Tech
(a.k.a., the birth, life, and death of light)

Digital Camera Imaging

Modeling and Rendering

Applications

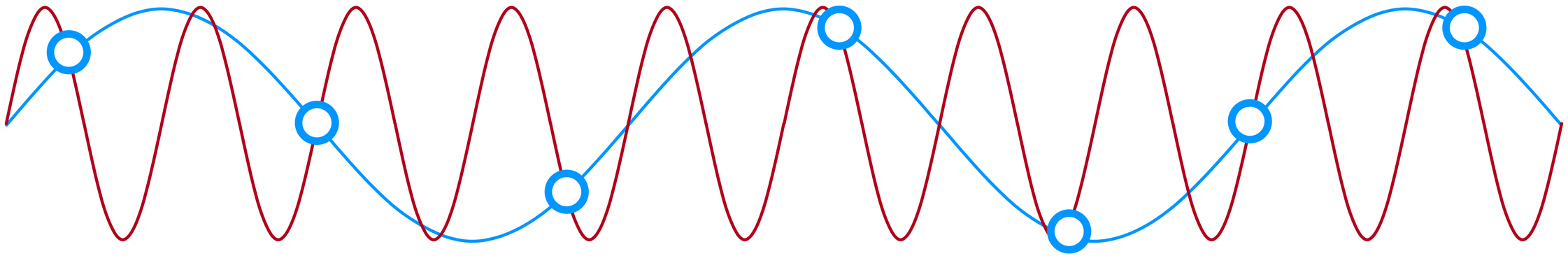
Geometric Transformations

Fourier Series & Transforms

Sampling & Reconstruction

Signal Sampling and Reconstruction

- Given just a few sparse samples, can we always reconstruct the underlying continuous signal?
- If not carefully sampled, reconstructed signals will be aliased: high frequencies signals masquerading as low-frequency signals.

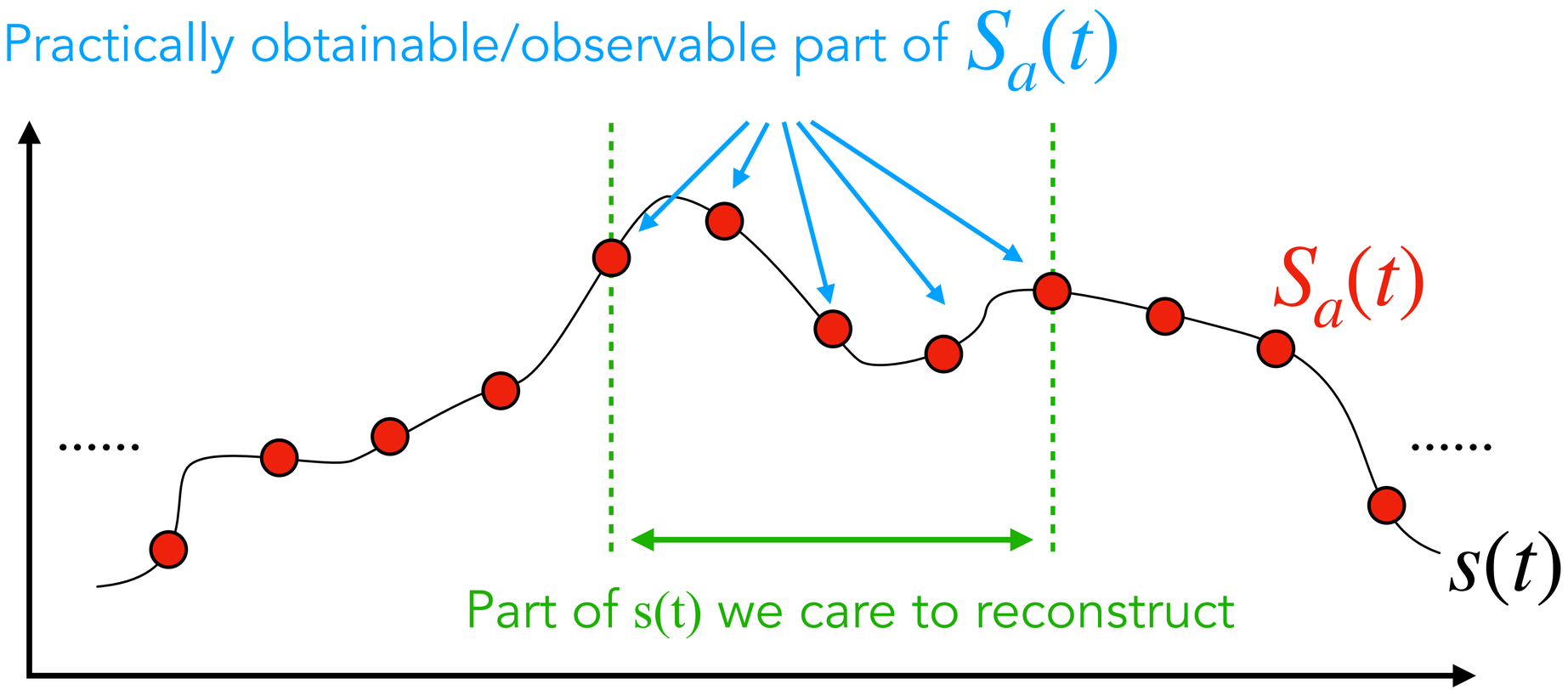


Aliasing

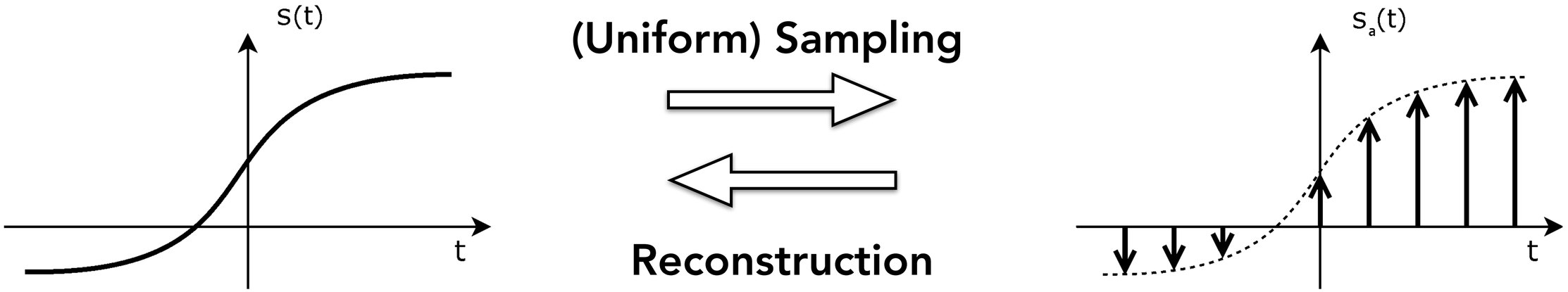
https://svi.nl/AntiAliasing https://distance.ufhealth.org/preparing-for-lecture-capture/ https://www.youtube.com/watch?v=Neh2biiex1A

Fourier Theory of Sampling and Reconstruction

Problem Setup



Problem Setup



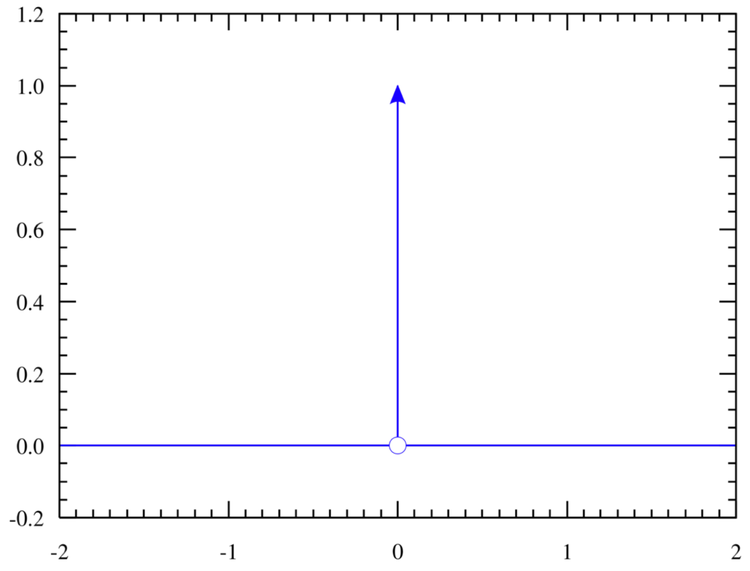
- s(t) is an unknown, continuous function with an infinite support (i.e., amenable to Fourier transform). We uniformly sample it and the goal is to reconstruct s(t) from the samples.
  - In practice we care only about reconstructing s(t) for a particular range and we could only obtain a finite number of samples (e.g., the world is infinite but a camera samples only a small region and we care about reconstructing the scene in that small region).
  - Defining s(t) this way makes theoretical derivation easier.

Impulse and Impulse Train Function

Impuse/Delta/Dirac Delta Function

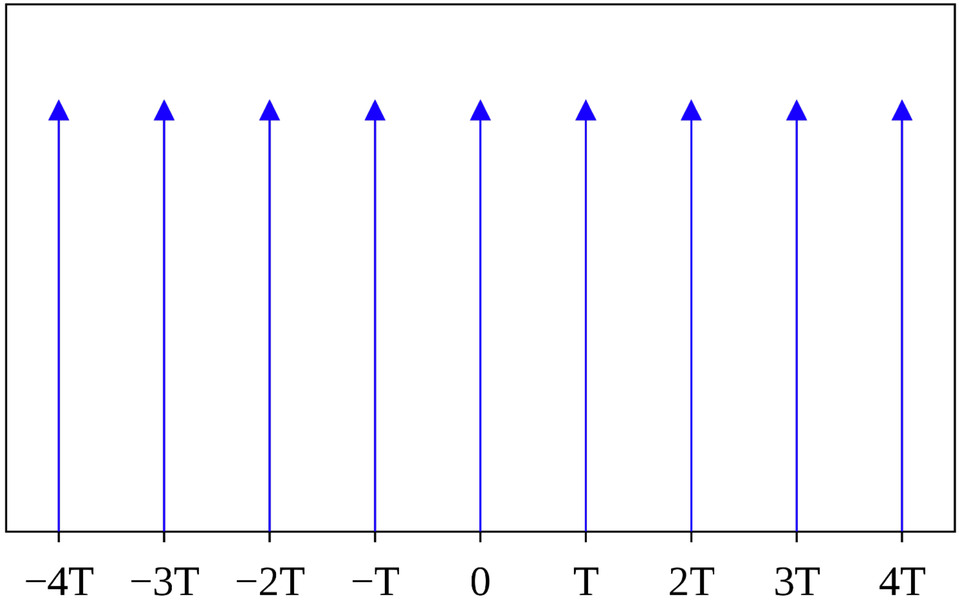
("cannot be expressed as a standard mathematical function, but instead is generally thought of as the limit of a unit area box function centered at the origin with width approaching 0.")

$$\delta(t) = \begin{cases} 0, & t \neq 0, \\ \textbf{something s.t.} \int_{-\infty}^{\infty} \delta(t) = 1, & t = 0. \end{cases}$$



Impulse Train/Dirac Comb/Shah Function (with a period of T and frequency of 1/T)

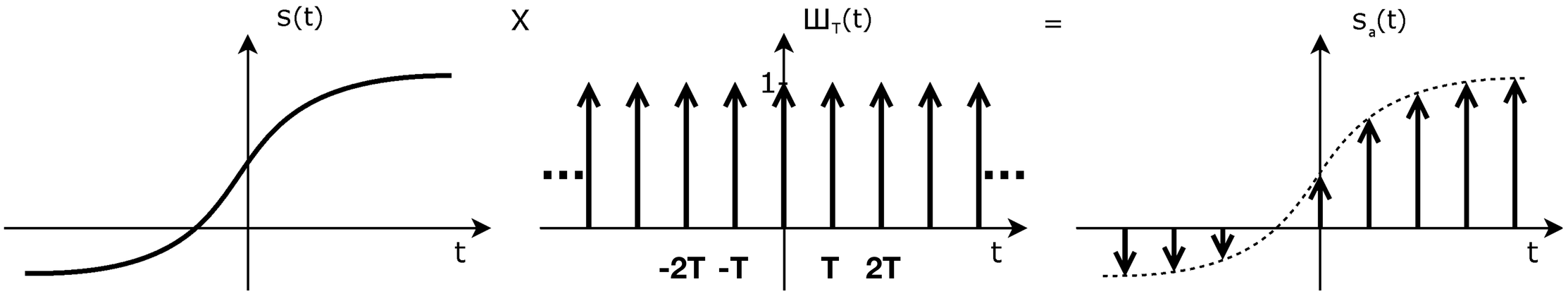
$$III(t) = \sum_{i=-\infty}^{\infty} \delta(t - iT)$$



https://en.wikipedia.org/wiki/Dirac_comb

Mathematically Express Sampling

Unknown continuous signal **Impulse Train** **Discrete samples**

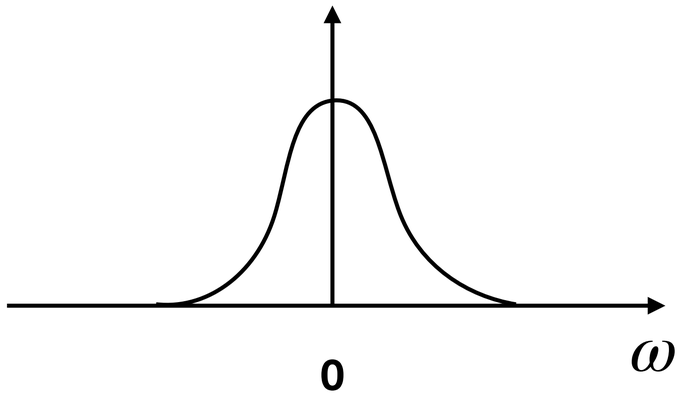


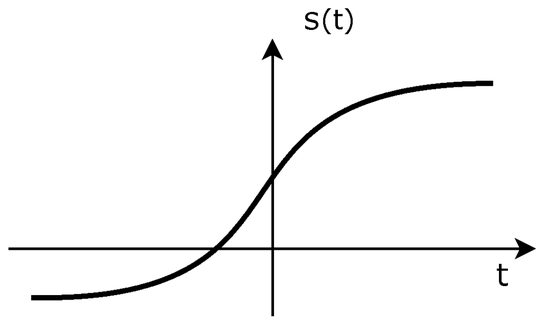
$$S_a(t) = III_T(t)s(t)$$

$s_a(t)$ is defined over a continuous domain; its value is 0 except at integer multiples of **T**.

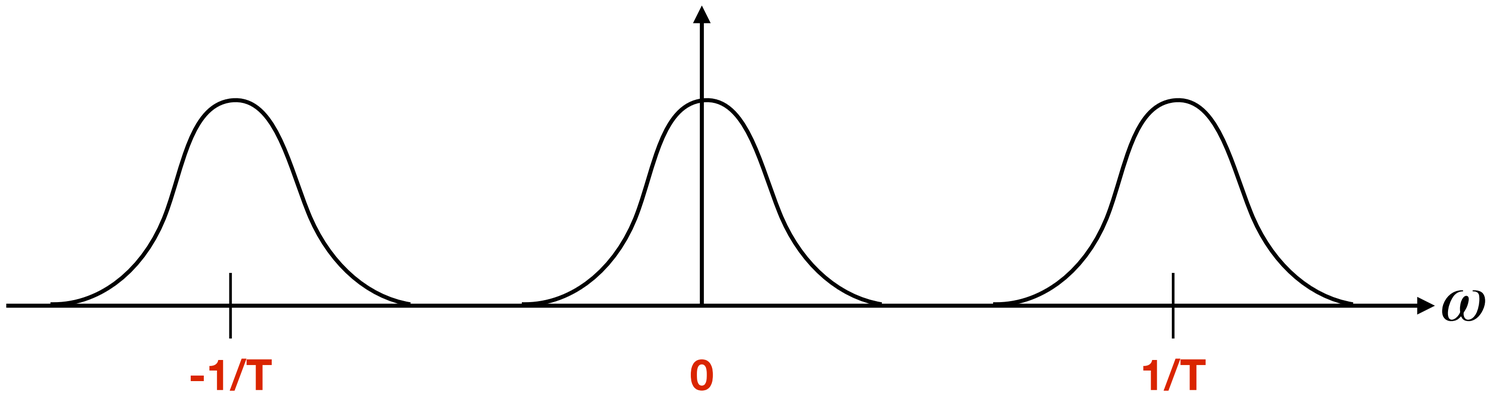
Fourier Transform a Sampled Signal

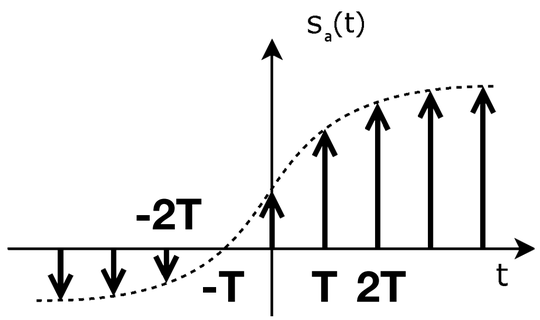
$$F(\omega) = \mathcal{F}(s(t))$$





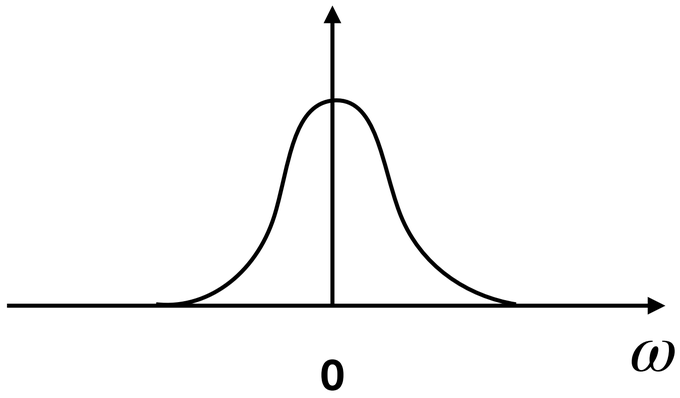
$$\mathcal{F}(III_T(t)s(t))$$



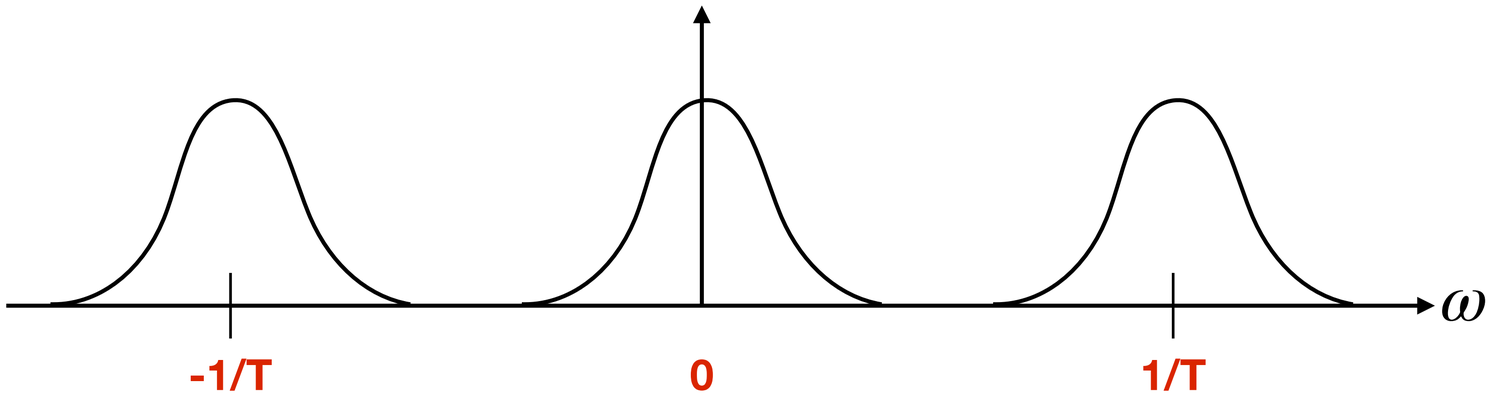


Fourier Transform a Sampled Signal

$$F(\omega) = \mathcal{F}(s(t))$$



$$\mathcal{F}(III_T(t)s(t))$$

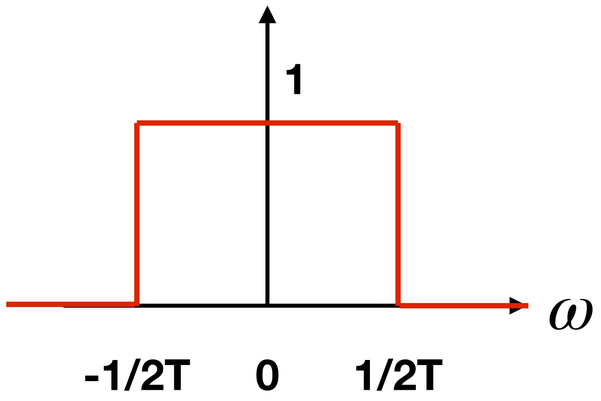


$$\mathcal{F}(III_T(t)s(t)) = \sum_{i=-\infty}^{\infty} F(\omega - i\frac{1}{T})$$

The Fourier Transform of the sampled signal is the sum of infinite copies of the Fourier Transform of the original signal, with spacing of **1/T.**

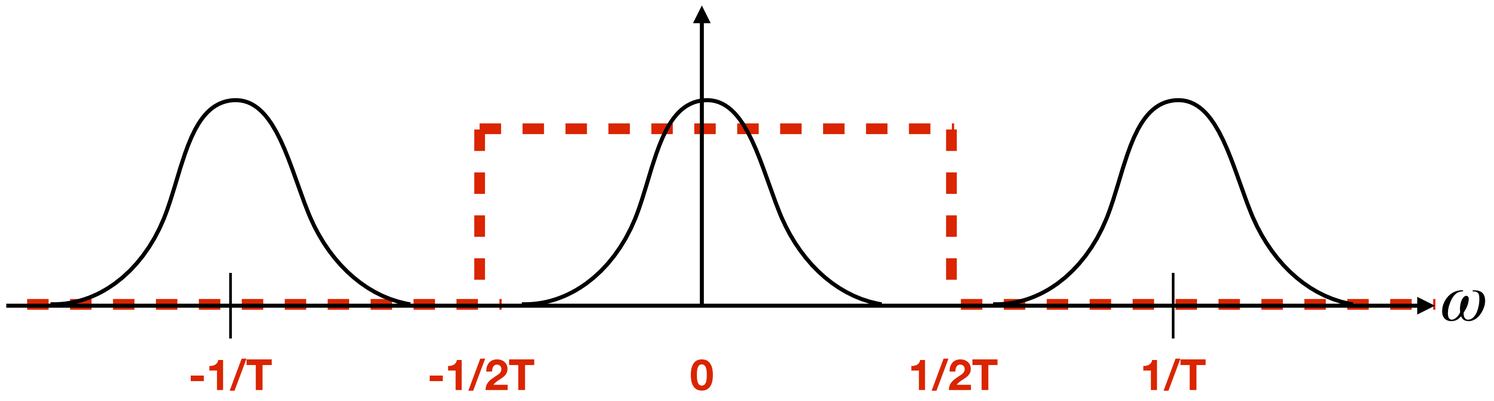
Extracting the Original Spectrum

Box Function



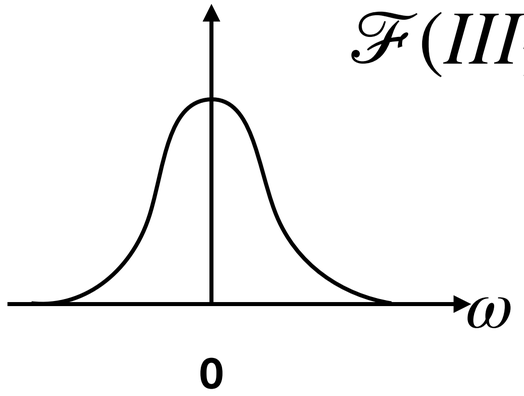
X

$\mathcal{F}(III_T(t)s(t))$



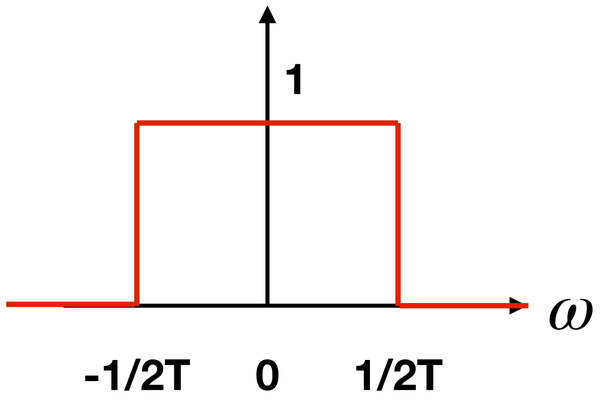
$$B(\omega) = \begin{cases} 1, & |\omega| < \frac{1}{2T}, \\ 0, & \textbf{otherwise}. \end{cases}$$

$$\mathcal{F}(III_T(t)s(t))B(\omega) = F(\omega)$$



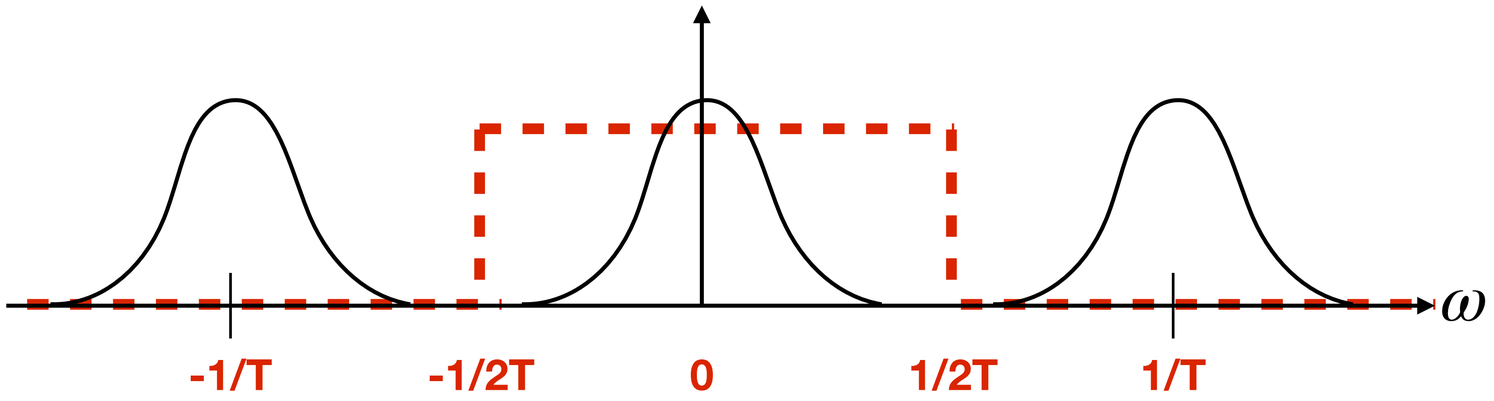
Reconstructing the Original Signal

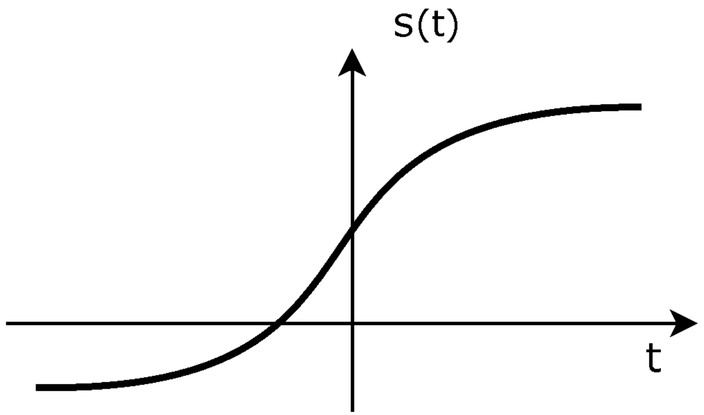
Box Function



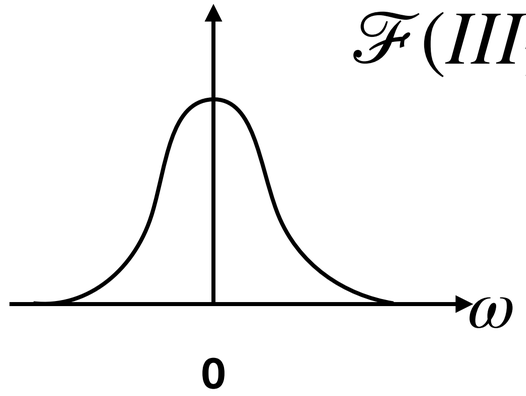
X

$\mathcal{F}(III_T(t)s(t))$



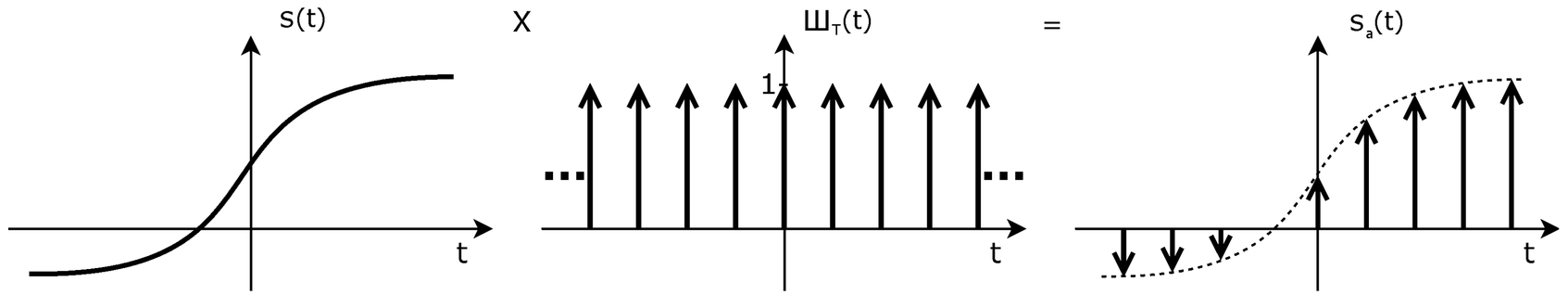


Inverse Fourier Transform



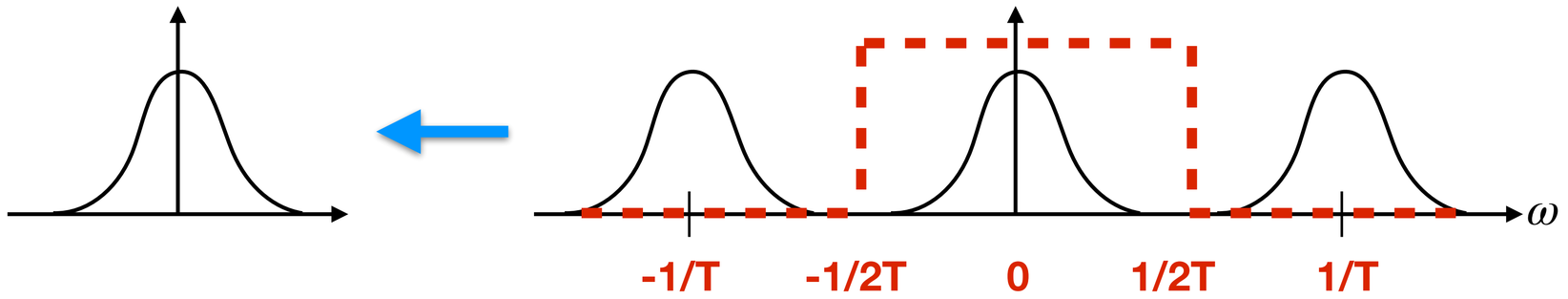
$\mathcal{F}(III_T(t)s(t))B(\omega) = F(\omega)$

Ideal Signal Reconstruction Summary



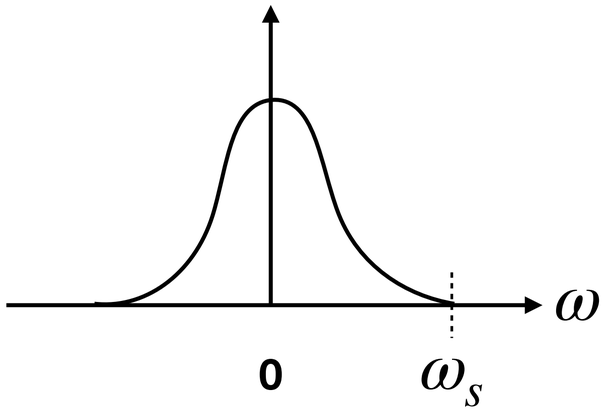
The reconstruction equation

$$s(t) = \mathcal{F}^{-1}[\ \mathcal{F}(S_a(t))B_{1/2T}(\omega)\ ]$$

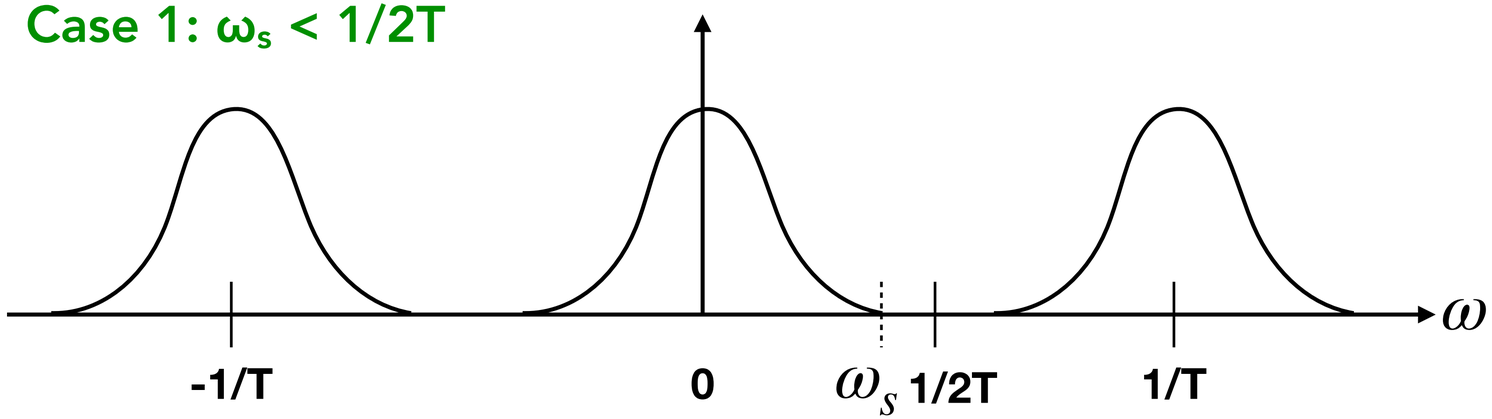


Aliasing

$$\mathcal{F}(S_a(t)) = \sum_{i=-\infty}^{\infty} F(\omega - i\frac{1}{T})$$



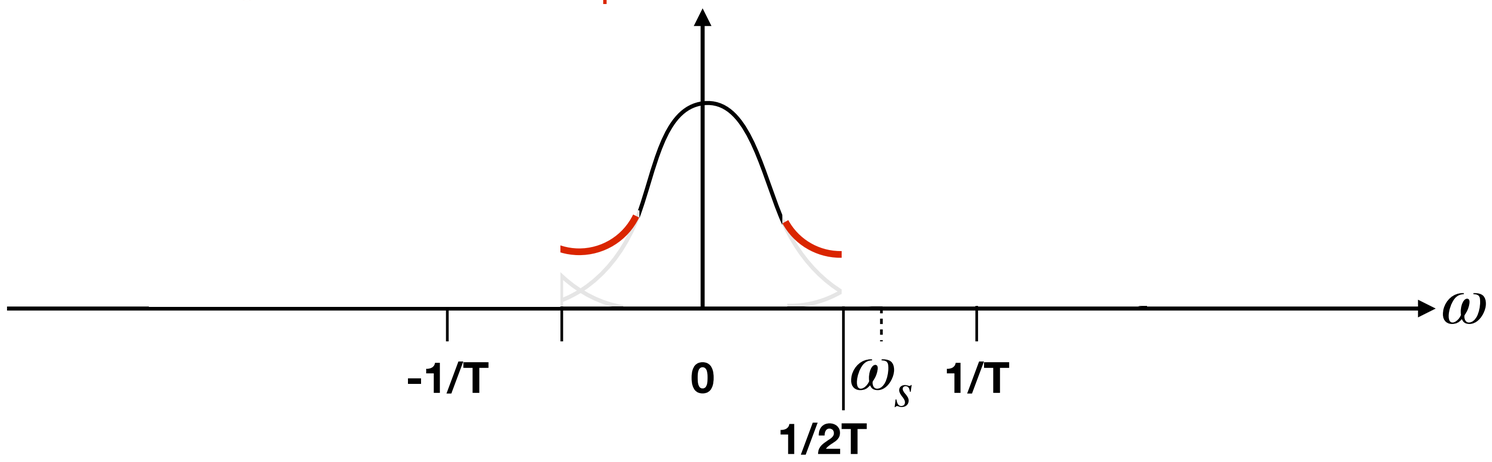
Case 1: ω_s < 1/2T



ω_s is the maximum frequency of the spectrum.
F(**ω**) = 0 for all |**ω**| > **ω_s**.

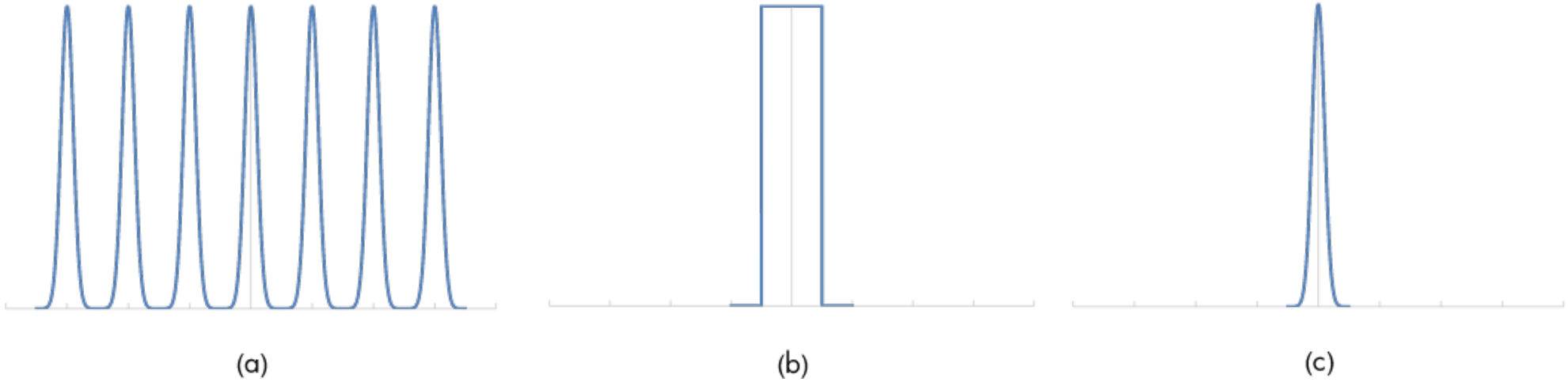
The spectrum of $S_a(t)$ doesn't look like the sum of infinite copies of the original spectrum, which can never be extracted now!

Case 2: ω_s >= 1/2T

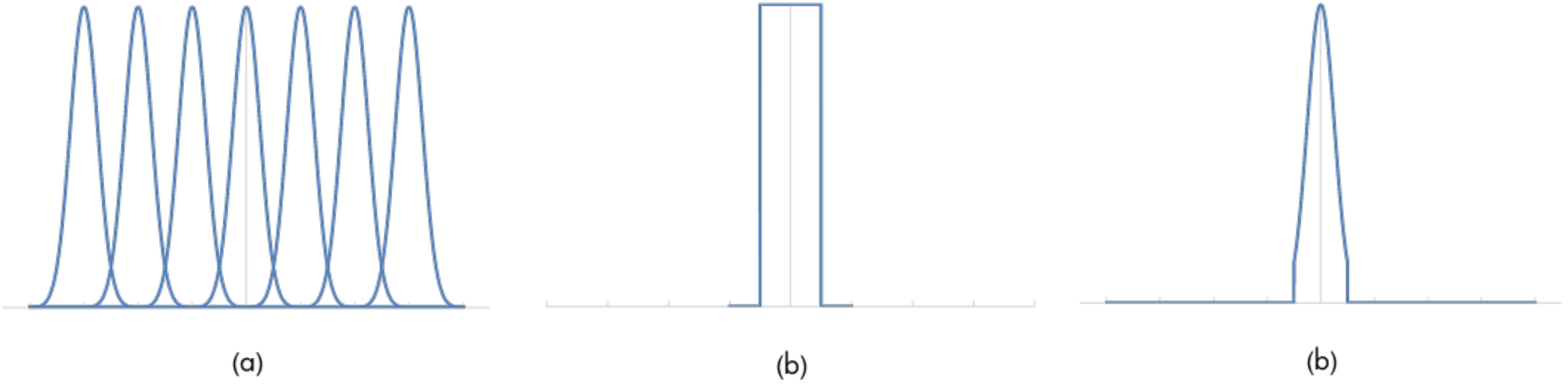


Another Aliasing Example

No Aliasing



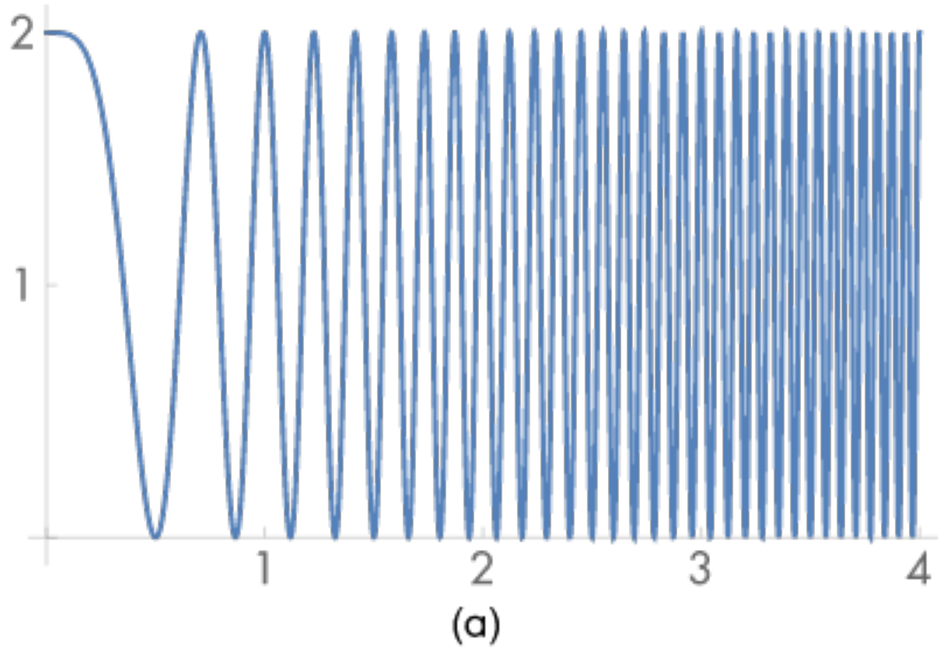
Aliased



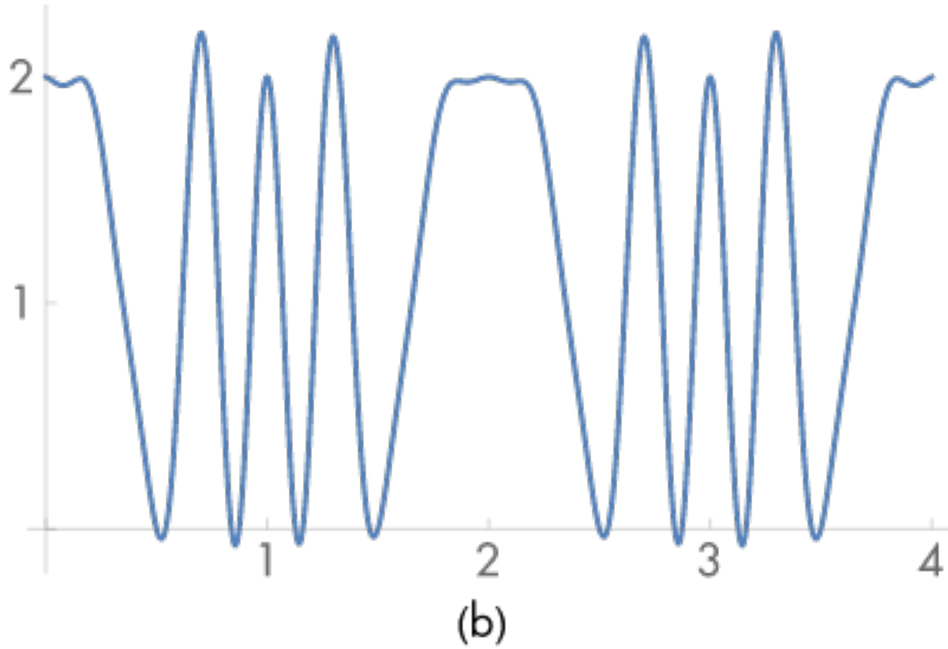
https://pbr-book.org/3ed-2018/Sampling_and_Reconstruction/Sampling_Theory

Reconstruction with Aliasing

Original Signal



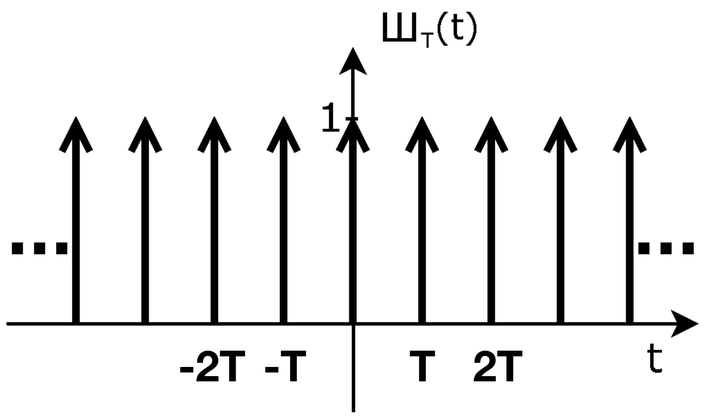
Reconstructed Signal

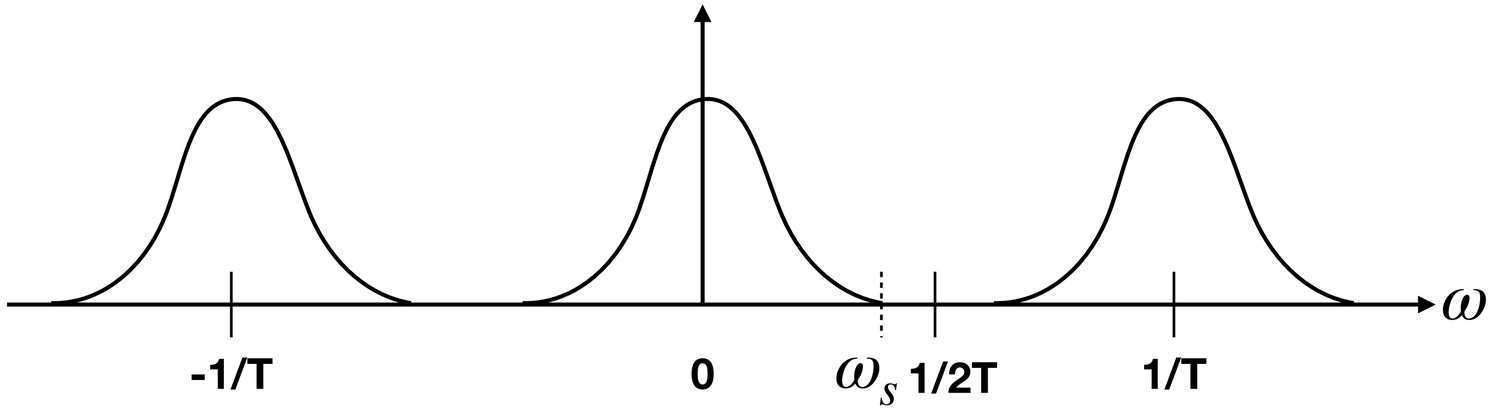


High-frequency information in the original signal is lost and shows up as low-frequency errors.

https://pbr-book.org/3ed-2018/Sampling_and_Reconstruction/Sampling_Theory

How To Guarantee Alias-Free Reconstruction?





Sampling period: **T**
Sampling frequency f_s: **1/T**

$$\omega_s < \frac{1}{2T} = \boxed{\frac{f_s}{2}}$$

Nyquist frequency

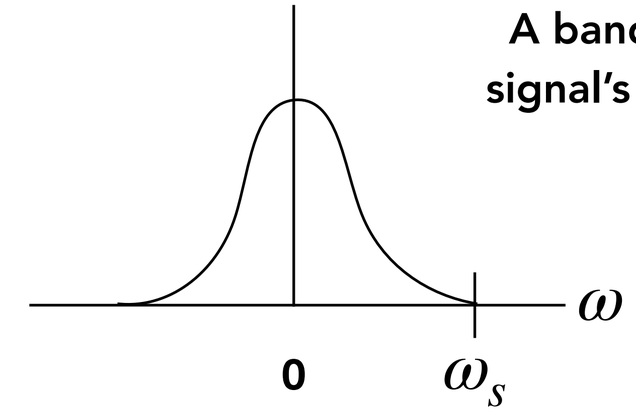
$$f_s > \boxed{2\omega_s}$$

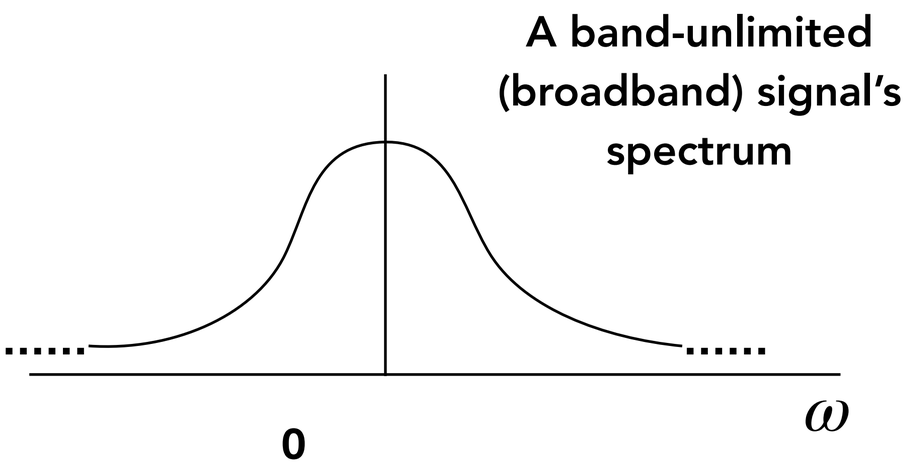
Nyquist rate

Nyquist–Shannon sampling theorem: if a signal contains no frequency higher than ω_s, it can be perfectly reconstructed from sampling it at a rate higher than $2\omega_s$.

https://en.wikipedia.org/wiki/Nyquist_frequency

Band-Limited Signal

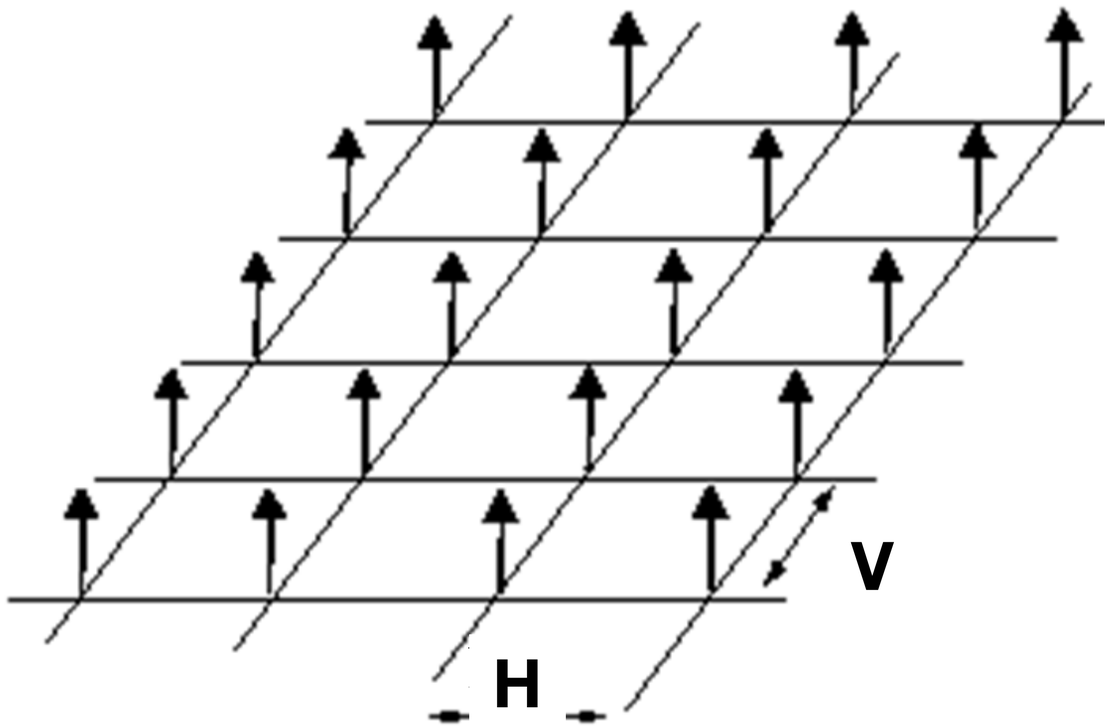




- Nyquist-Shannon sampling theorem is only useful if the original signal **does have** a maximum frequency ω_s. Otherwise the sampling rate would have to be infinite.
- Band-limited signal: a signal where there exists a frequency ω_s such that $F(\omega) = 0$ for all $|\omega| > \omega_s$.
- Most real-world signals are "broad-band" signals; they are **not** band-limited.

2D Sampling

“Brush” Function



2D sampling theorem: if a signal contains no horizontal frequency higher than ω_u and no vertical frequency higher than ω_v, it can be completely reconstructed from sampling it at a horizontally rate higher than $2\omega_u$ and a vertical rate higher than $2\omega_v$.

The horizontal sampling distance $H < 1/(2\ \omega_u)$
The vertical sampling distance $V < 1/(2\ \omega_v)$

https://www.di.univr.it/documenti/Occorrenzalns/matdid/matdid346761.pdf

Anti-Aliasing Techniques

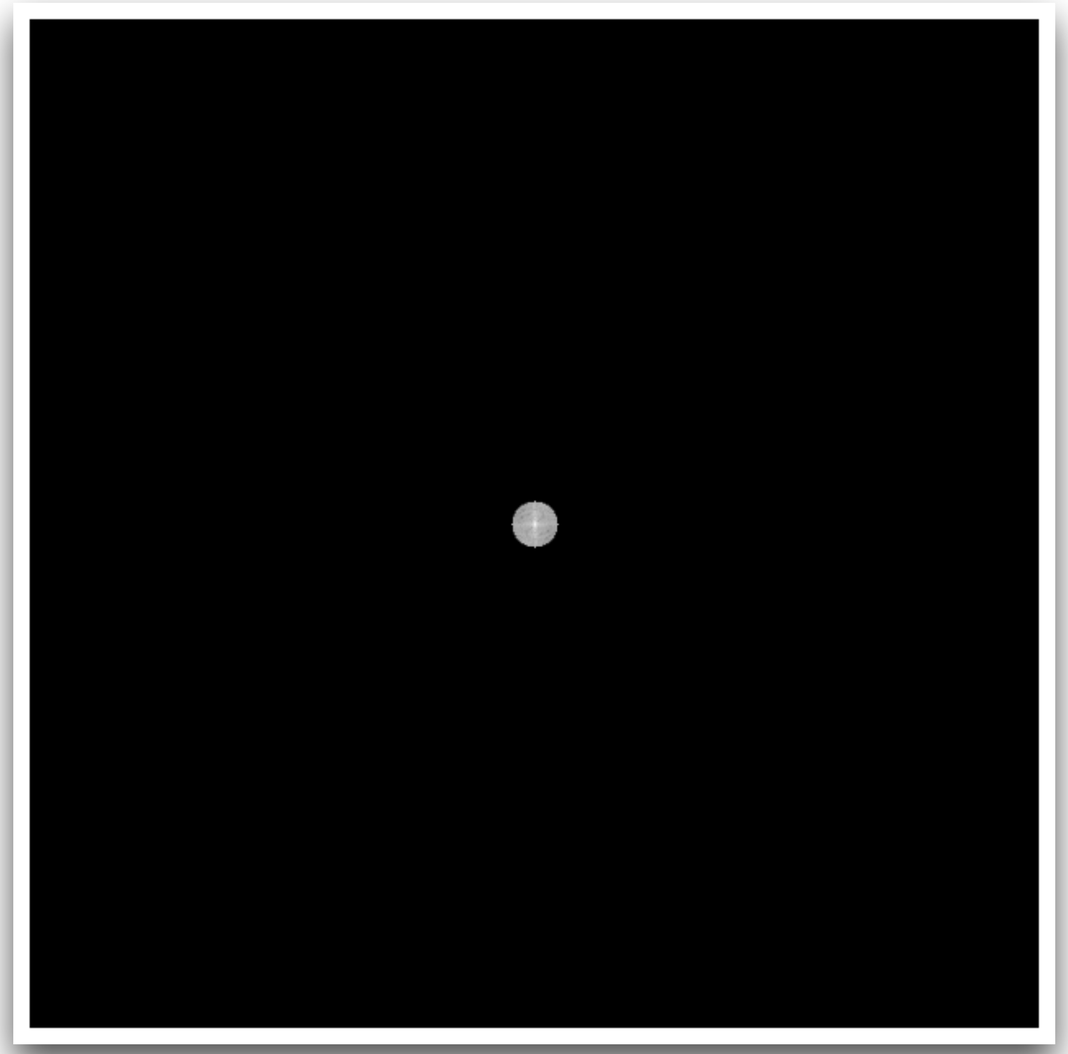
Insufficient Sampling and Reconstruction

https://svi.nl/AntiAliasing https://distance.ufhealth.org/preparing-for-lecture-capture/

Anti-Aliasing By Pre-Filtering

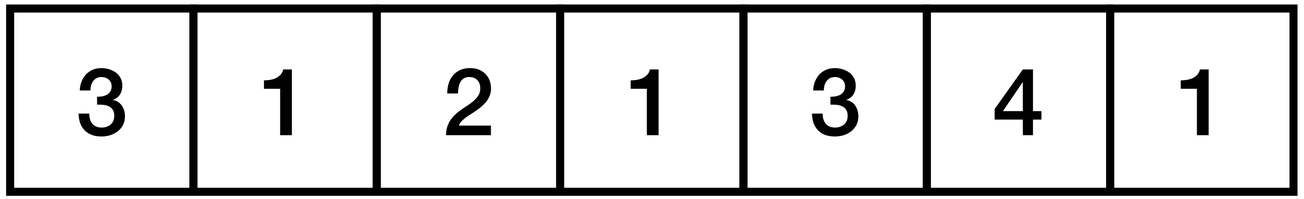
- If we can only sample at a rate of f_{sample}, pre-filter the signal to remove the frequency higher than $f_{sample}/2$.
- Then sample; won't see aliasing, but the reconstructed signal is blurred.
- Blur is more acceptable visually than aliasing.

Recall: Low-Pass Filtering

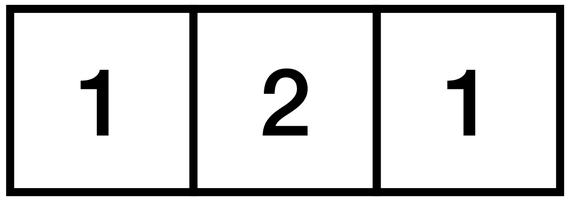


1D Discrete Convolution

1D Discrete Signal

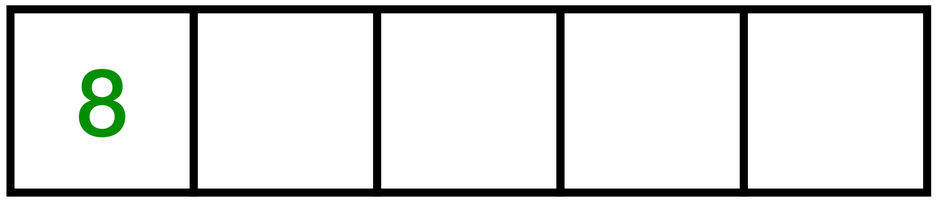


Filter/Kernel



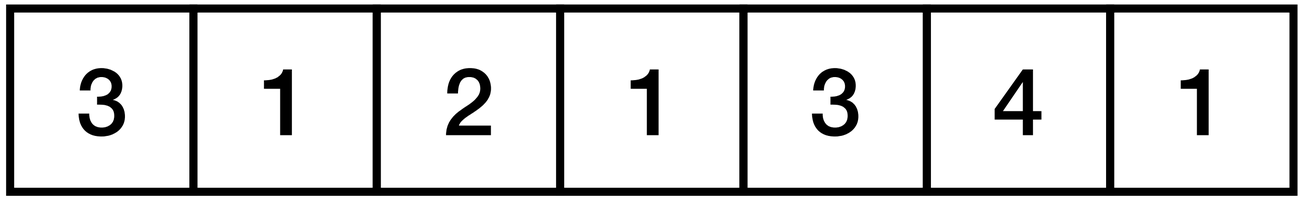
Convolution 3x1 + 1x2 + 2x1

Filtered Signal

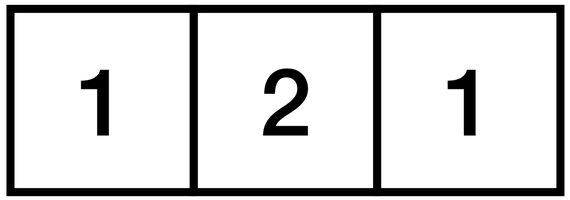


1D Discrete Convolution

1D Discrete Signal

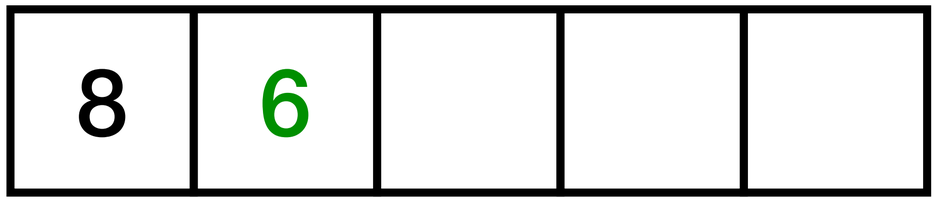


Filter/Kernel



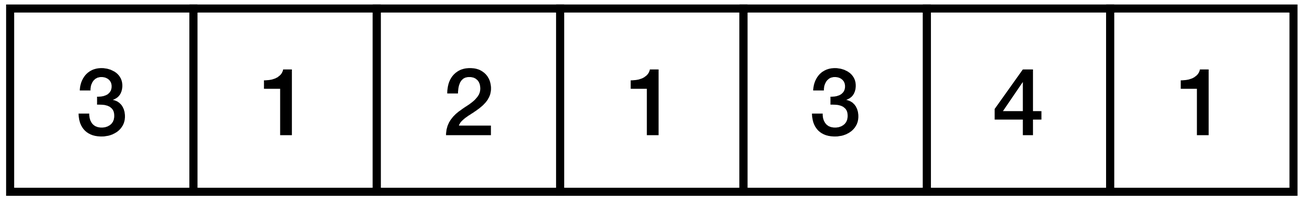
Convolution 1x1 + 2x2 + 1x1

Filtered Signal

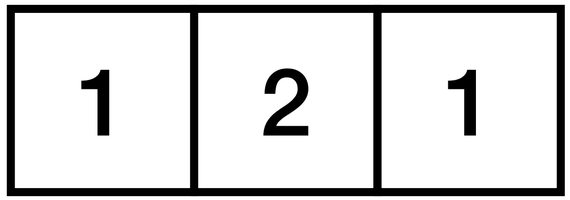


1D Discrete Convolution

1D Discrete Signal



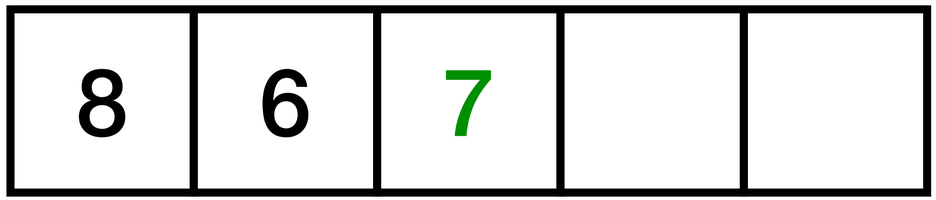
Filter/Kernel



Convolution

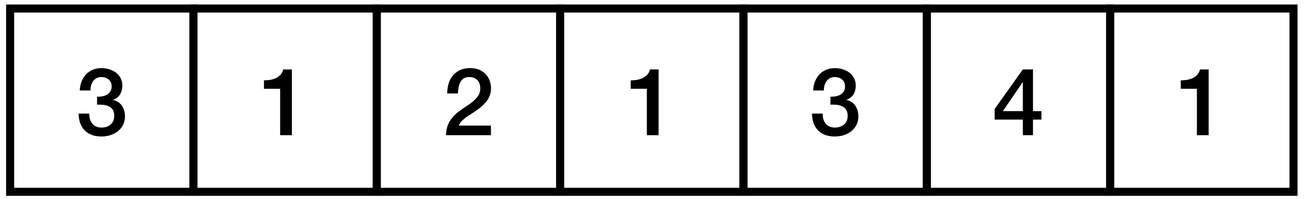
2x1 + 1x2 + 3x1

Filtered Signal

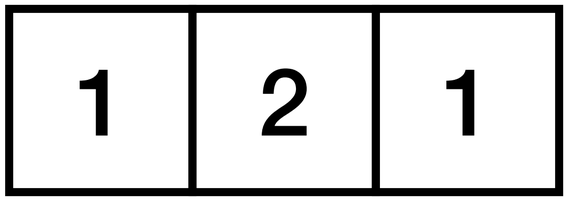


1D Discrete Convolution

1D Discrete Signal



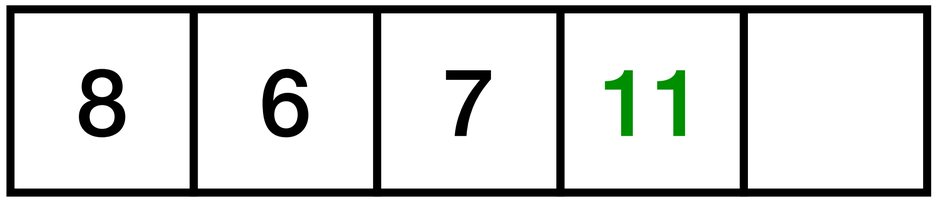
Filter/Kernel



Convolution

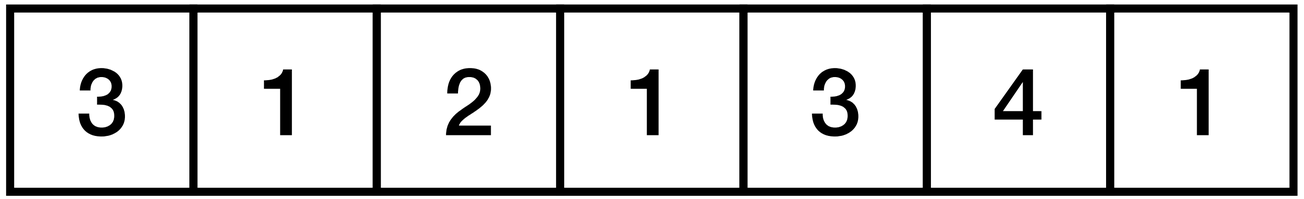
1x1 + 3x2 + 4x1

Filtered Signal

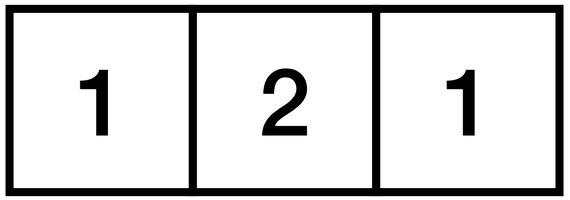


1D Discrete Convolution

1D Discrete Signal



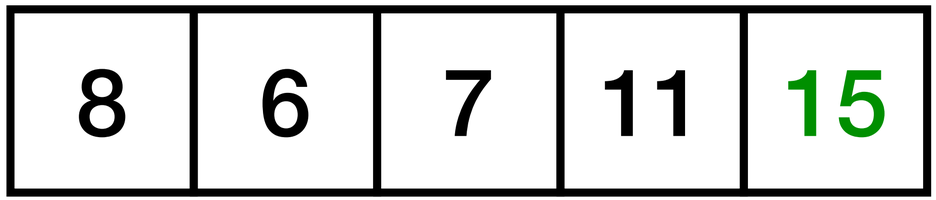
Filter/Kernel



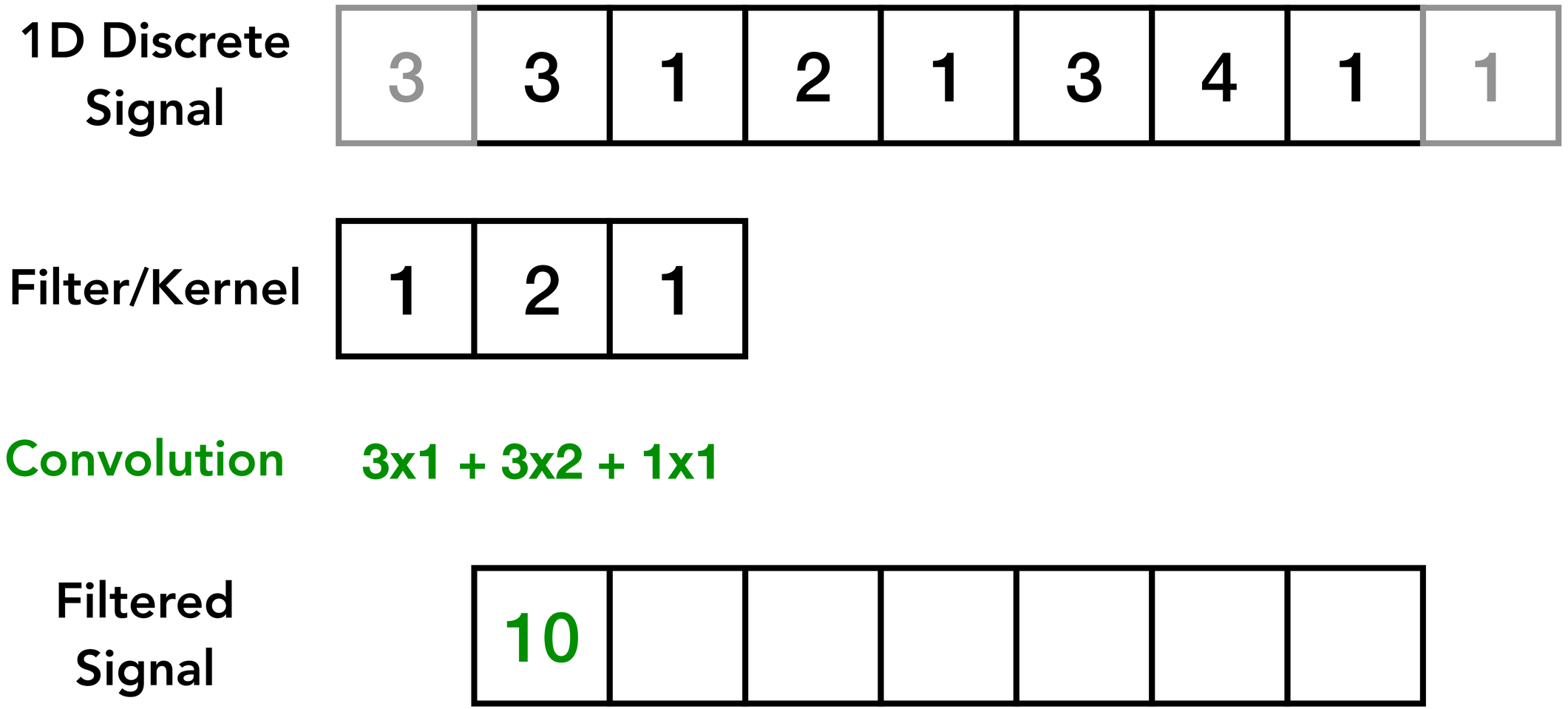
Convolution

3x1 + 4x2 + 4x1

Filtered Signal



1D Discrete Convolution w/ Padding



1D Discrete Convolution w/ Padding

1D Discrete Signal

3	3	1	2	1	3	4	1	1

Filter/Kernel

1	2	1

Convolution

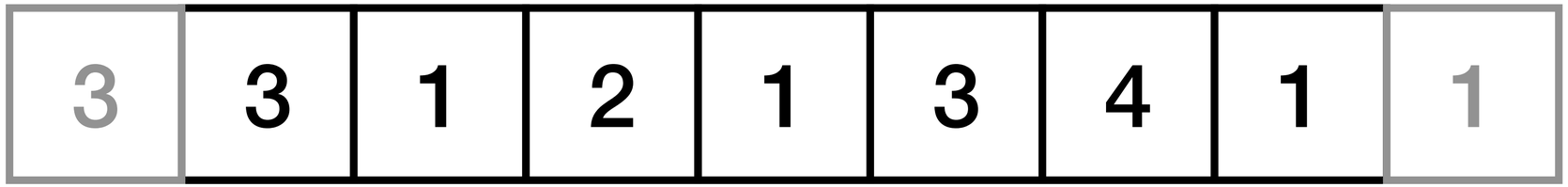
3x1 + 1x2 + 2x1

Filtered Signal

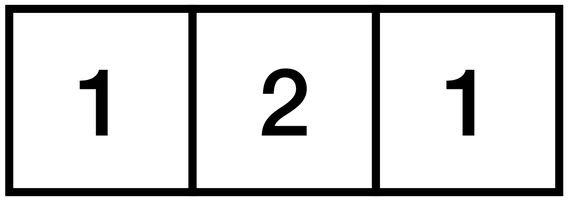
10	7					

1D Discrete Convolution w/ Padding

1D Discrete Signal



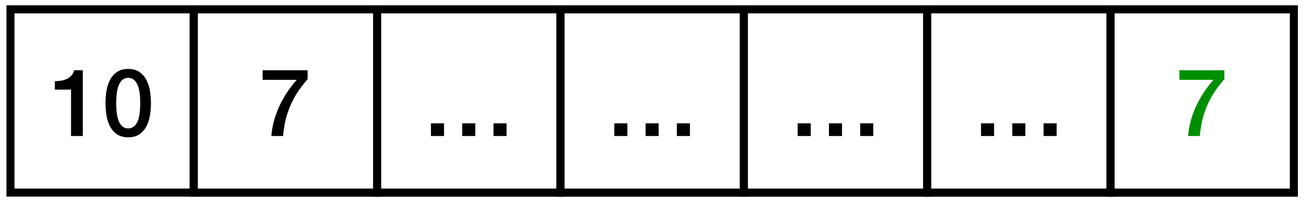
Filter/Kernel



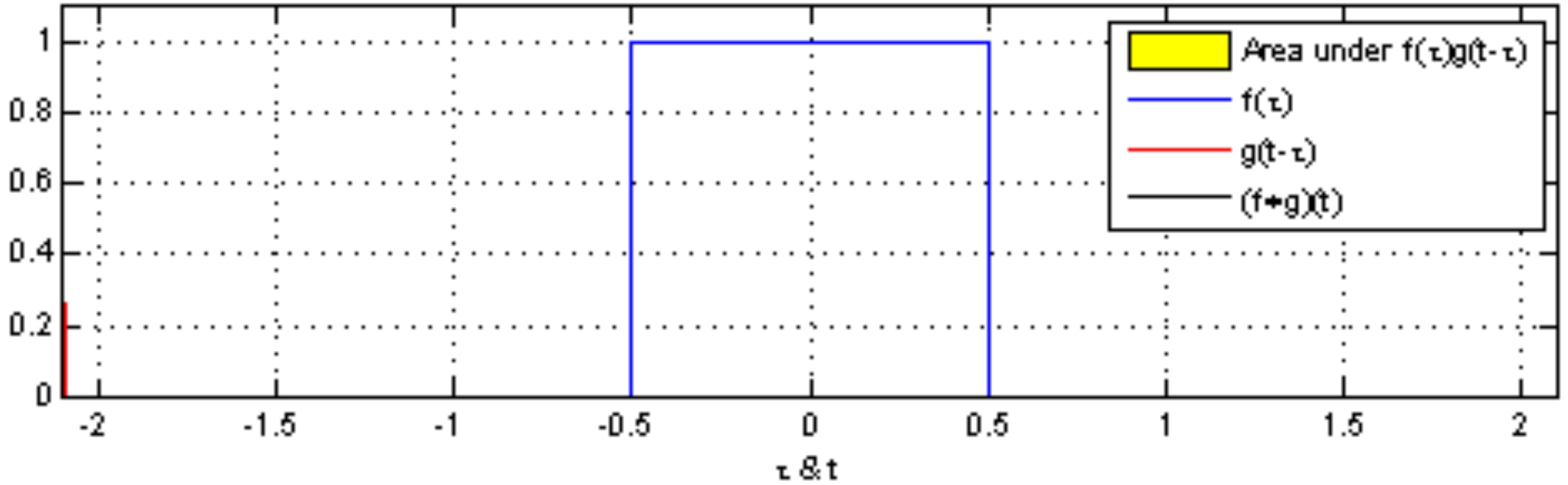
Convolution

4x1 + 1x2 + 1x1

Filtered Signal



1D Continuous Convolution Visualization



https://en.wikipedia.org/wiki/Convolution

2D Discrete Convolution

2D Discrete Signal

3	1	2	1	3	4	1
2	4	0	1	10	2	0
0	2	4	21	9	1	14
34	5	4	7	8	90	34
54	6	8	9	13	36	4
6	8	14	2	4	8	52
32	14	54	3	6	8	0

2D Filter

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Filtered Signal

2D Discrete Convolution

2D Discrete Signal

3	1	2	1	3	4	1
2	4	0	1	10	2	0
0	2	4	21	9	1	14
34	5	4	7	8	90	34
54	6	8	9	13	36	4
6	8	14	2	4	8	52
32	14	54	3	6	8	0

2D Filter

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Filtered Signal

2				

2D Discrete Convolution

2D Discrete Signal

3	1	2	1	3	4	1
2	4	0	1	10	2	0
0	2	4	21	9	1	14
34	5	4	7	8	90	34
54	6	8	9	13	36	4
6	8	14	2	4	8	52
32	14	54	3	6	8	0

2D Filter

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Filtered Signal

2	4			

Box (Mean/Moving Average) Filter

2D Discrete Signal

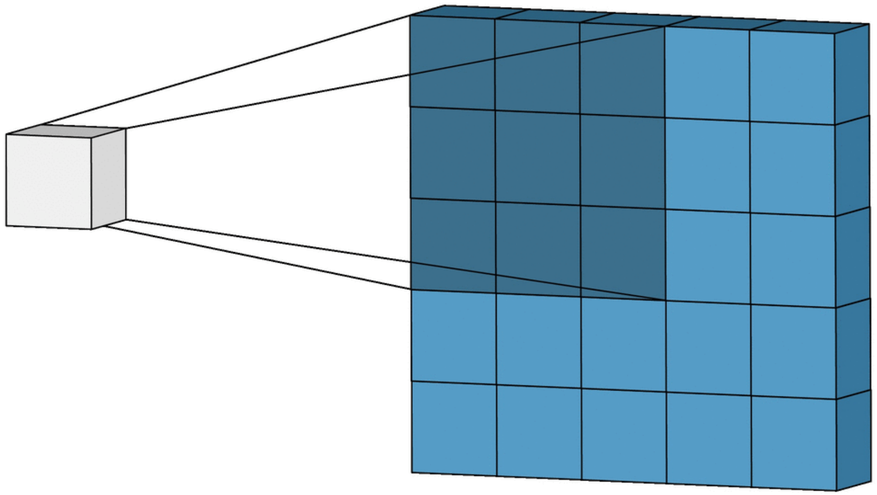
3	1	2	1	3	4	1
2	4	0	1	10	2	0
0	2	4	21	9	1	14
34	5	4	7	8	90	34
54	6	8	9	13	36	4
6	8	14	2	4	8	52
32	14	54	3	6	8	0

Box Filter

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

A box filter makes pixels more similar to its neighbors, i.e., blur.

Visualization



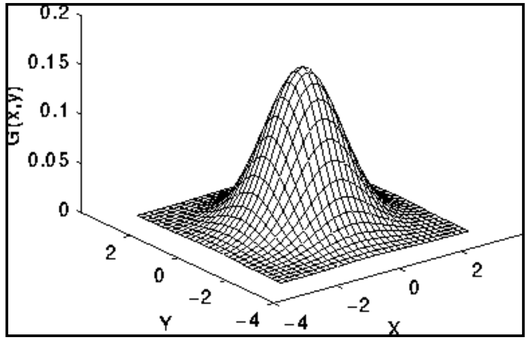
2D Box Filter

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

* Output size is smaller than input size without padding.

Gaussian Filter

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



2D Gaussian distribution

$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

A sample 2D Gaussian kernel with mean [0, 0] and σ=1

A Gaussian filter also averages neighboring pixels, but gives more weight to closer neighbors. It's still a low-pass filter.

https://homepages.inf.ed.ac.uk/rbf/HIPR2/gsmooth.htm

Convolution

1D Discrete Convolution

$$f[x] \star g[x] = \sum_{k=-\infty}^{k=\infty} f[k]g[x-k]$$

1D Continuous Convolution

$$f(x) \star g(x) = \int_{-\infty}^{\infty} f(\tau)g(x-\tau)d\tau$$

2D Discrete Convolution

$$f[x,y] \star g[x,y] = \sum_{i=-\infty}^{i=\infty} \sum_{j=-\infty}^{j=\infty} f[i,j]g[x-i,y-j]$$

2D Continuous Convolution

$$f(x) \star g(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\tau,\eta)g(x-\tau,y-\eta)d\tau d\eta$$

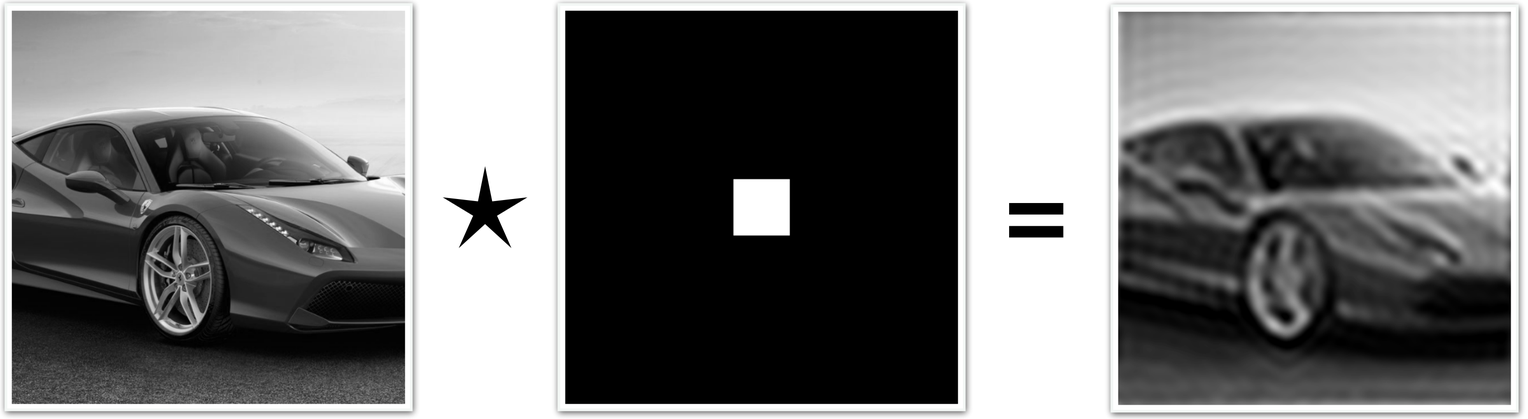
Convolution Theorem

$$\mathcal{F}(f \star g) = \mathcal{F}(f)\mathcal{F}(g) \qquad f \star g = \mathcal{F}^{-1}(\mathcal{F}(f)\mathcal{F}(g))$$

- Spatial domain convolution is equivalent to frequency domain multiplication, and vice versa.
- Why useful in signal sampling and reconstruction?
  - All we observe are a finite number of samples over a small domain in f, so

Convolution Theorem

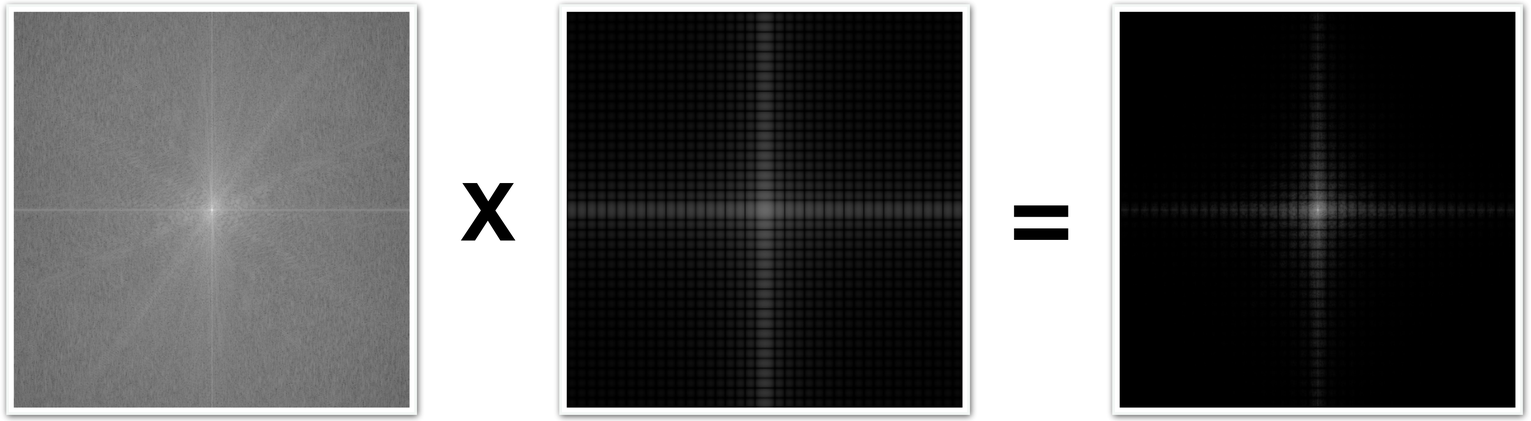
Spatial Domain



Fourier Transform

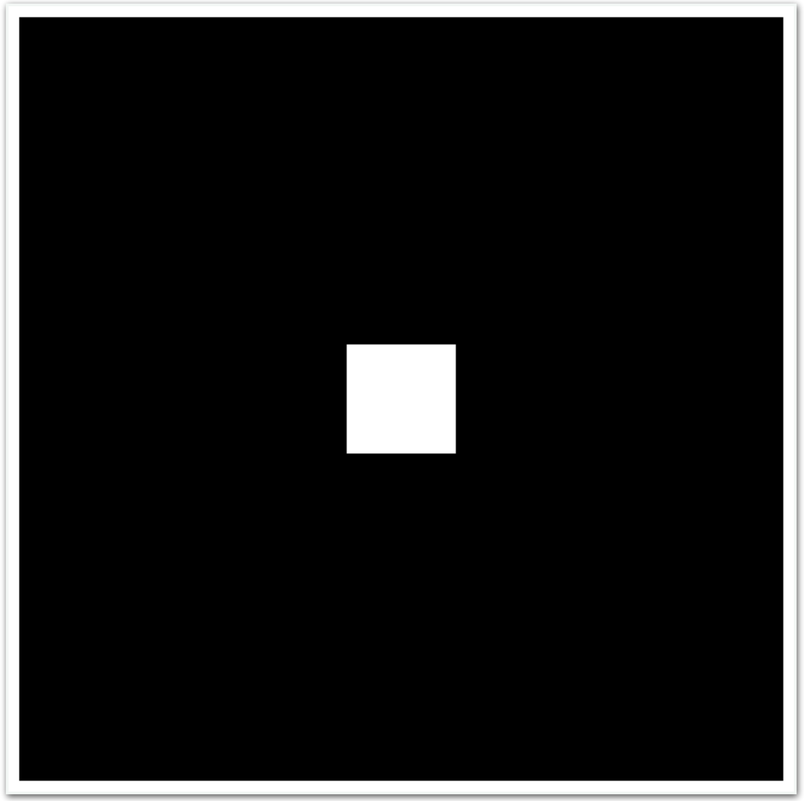
Inv. Fourier Transform

Frequency Domain

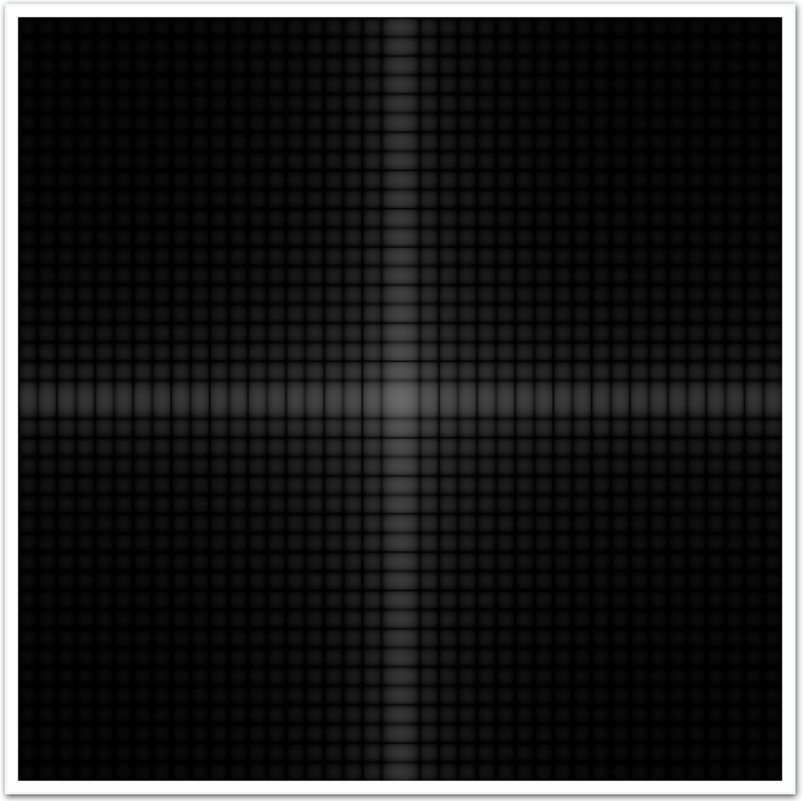


Box Filter in Spatial and Frequency Domains

Multiplying with this spectrum
attenuates high-frequency components.



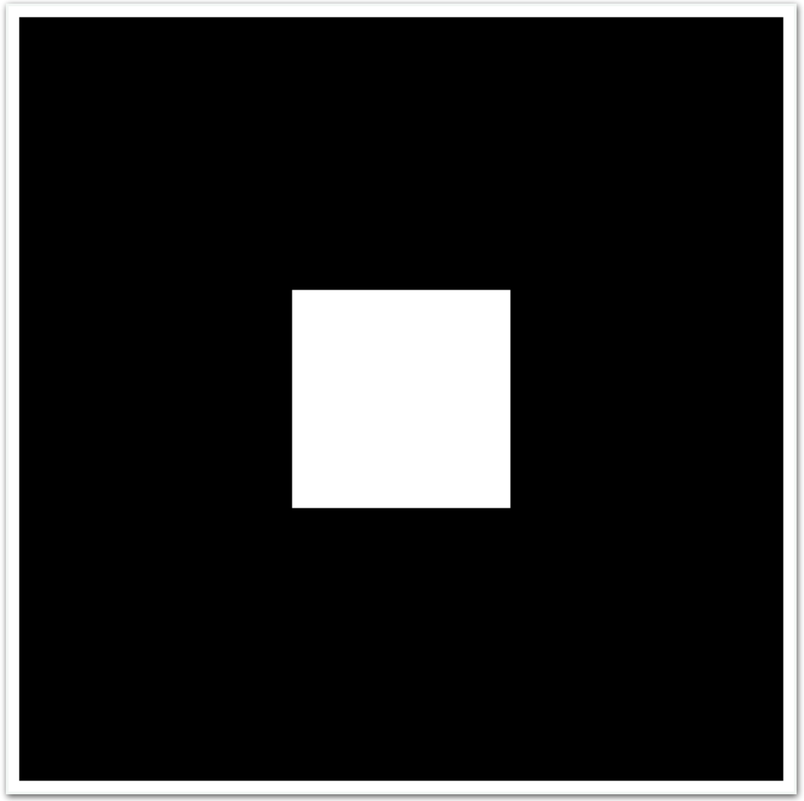
Spatial Domain



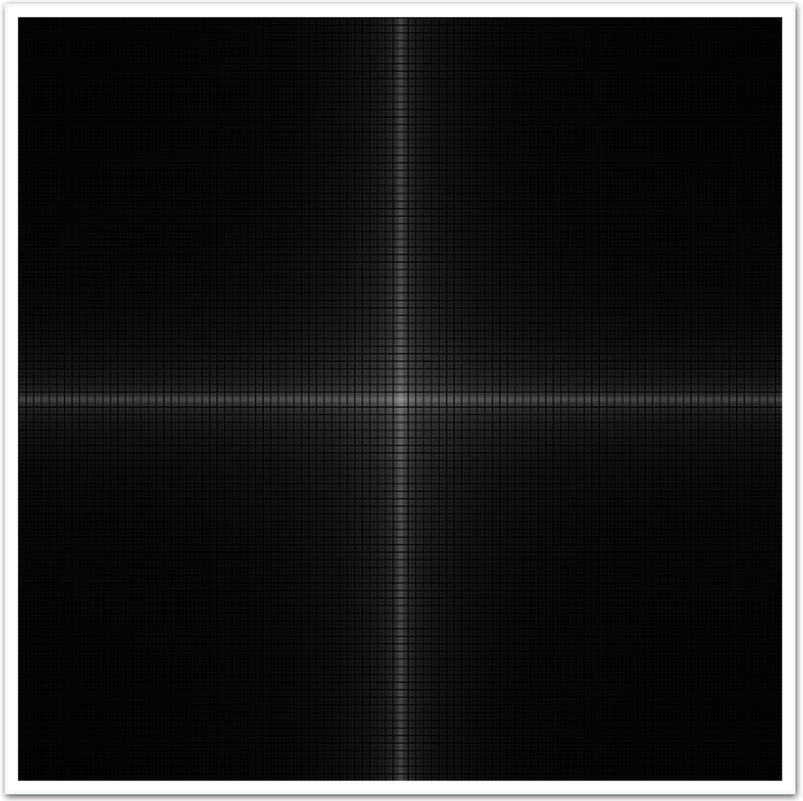
Frequency Domain

Box Filter in Spatial and Frequency Domains

Wider box attenuates high frequencies even more (averaging over a larger window)

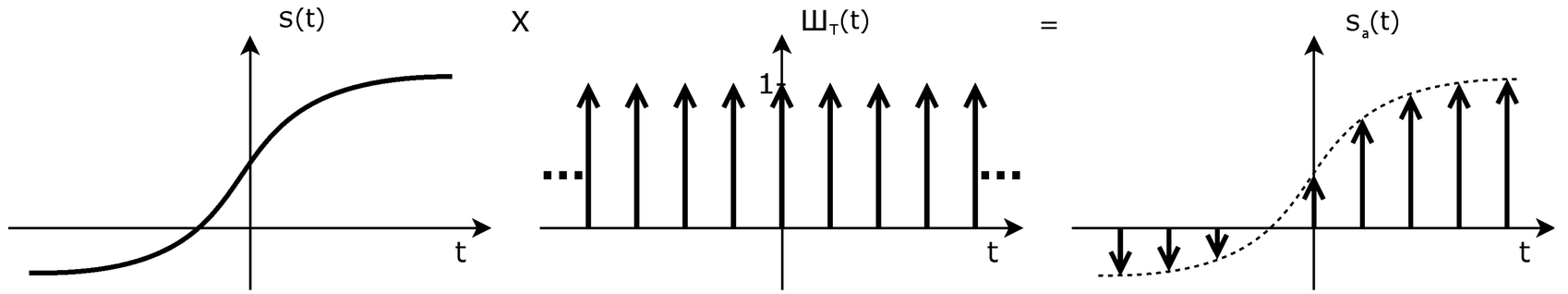


Spatial Domain



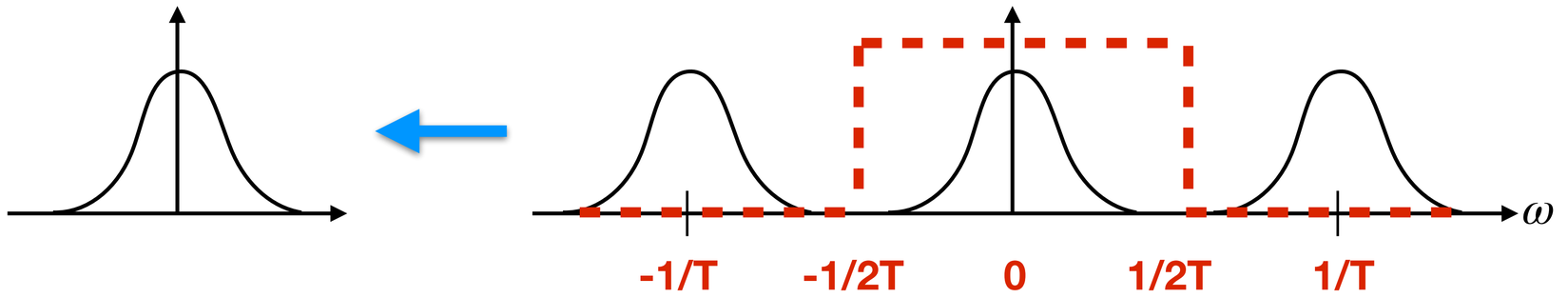
Frequency Domain

Revisiting Ideal Signal Reconstruction



The reconstruction equation

$$s(t) = \mathcal{F}^{-1}[\ \mathcal{F}(S_a(t))B_{1/2T}(\omega)\ ]$$



Signal Reconstruction == Convolution

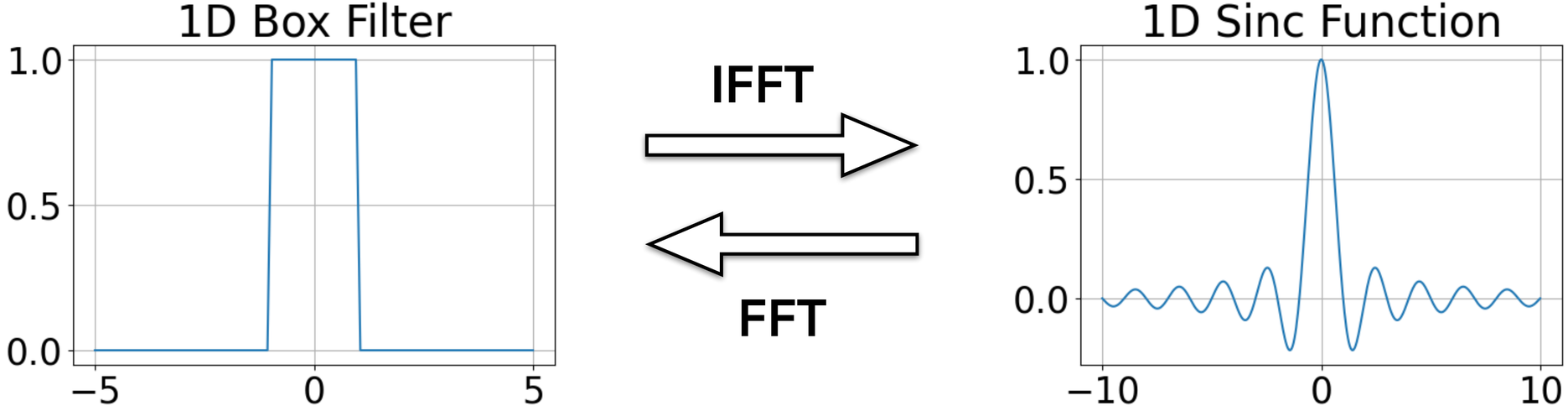
Reconstructed signal

Sampled signal

$$f \star g = \mathcal{F}^{-1}(\mathcal{F}(f)\mathcal{F}(g))$$

$$s(t) = \mathcal{F}^{-1}[\ \mathcal{F}(S_a(t))B(\omega)\ ]$$

$$s(t) = \mathcal{F}^{-1}[\ \mathcal{F}(S_a(t))\mathcal{F}(sinc(t))\ ] = S_a(t) \star sinc(t)$$



A Few Practical Notes

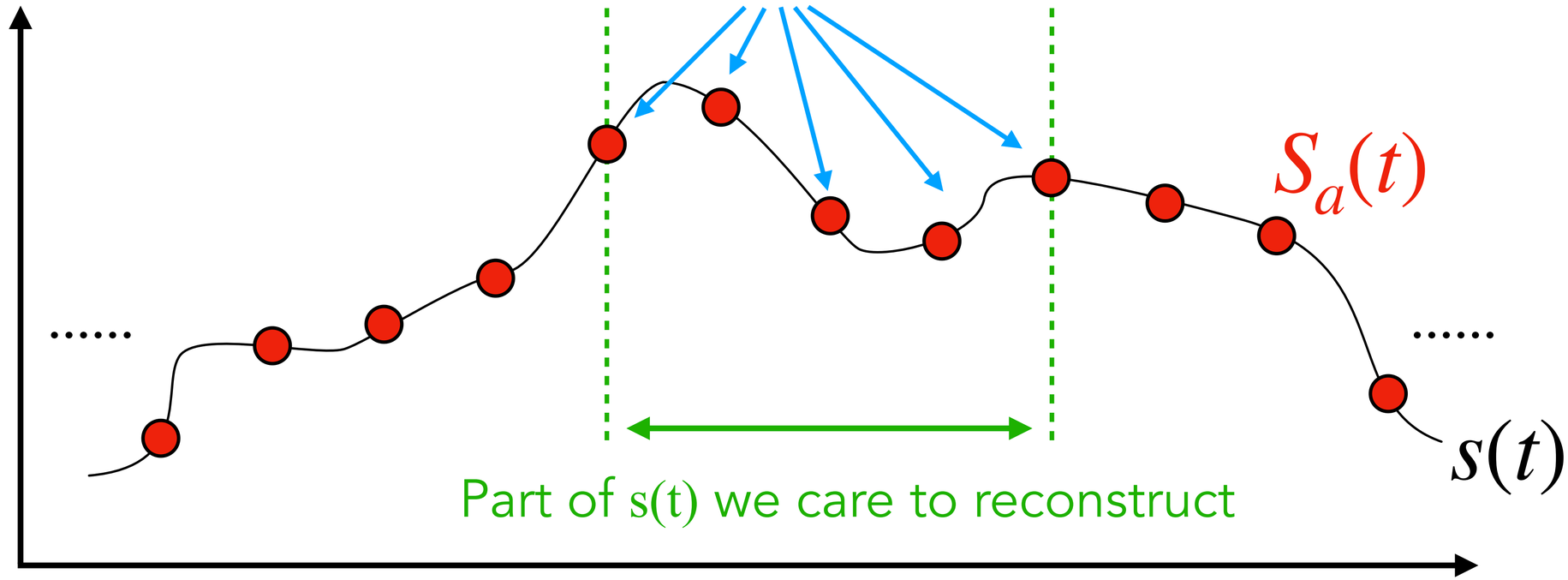
Reconstructed signal **Sampled signal**

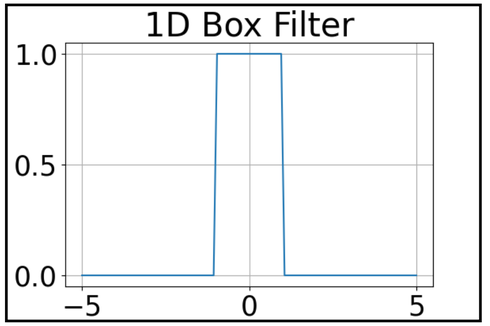
$$s(t) = \mathcal{F}^{-1}[\ \mathcal{F}(S_a(t))B(\omega)\ ] = S_a(t) \star sinc(t)$$

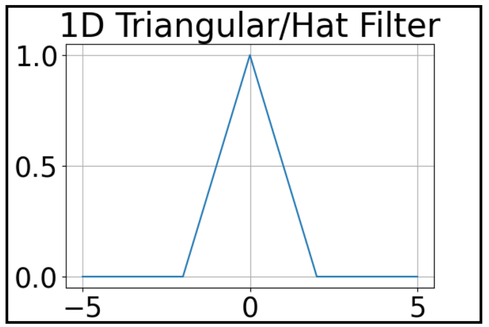
- Is it a bit surprising that we can do this even without knowing the full $S_a(t)$?
  - Answer: we can just assume the missing part of $S_a(t)$, e.g., all 0s or $S_a(t)$ being periodic. Each will give a different reconstructed signal.
- Even if we know $S_a(t)$, ideal reconstruction is computationally inefficient
  - $S_a(t)$ usually doesn't have an analytical form so its Fourier transform needs to be calculated numerically using all the samples in $S_a(t)$
  - ***sinc***(t) has infinite support so we again must use all $S_a(t)$ samples if we use convolution

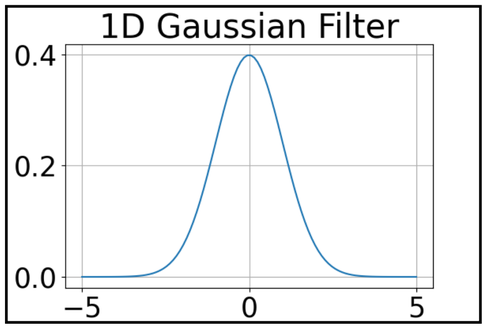
Practically We Prefer Filters with a Finite Support

Practically obtainable/observable part of $S_a(t)$









* Gaussian filter still has infinite support, but it decays exponentially so we can cut it off

Reconstruction by Convoluting with a Box Filter

Samples

3	0	0	2	0	0	5	0	0	6	0	0	4

Filter

1	1	1

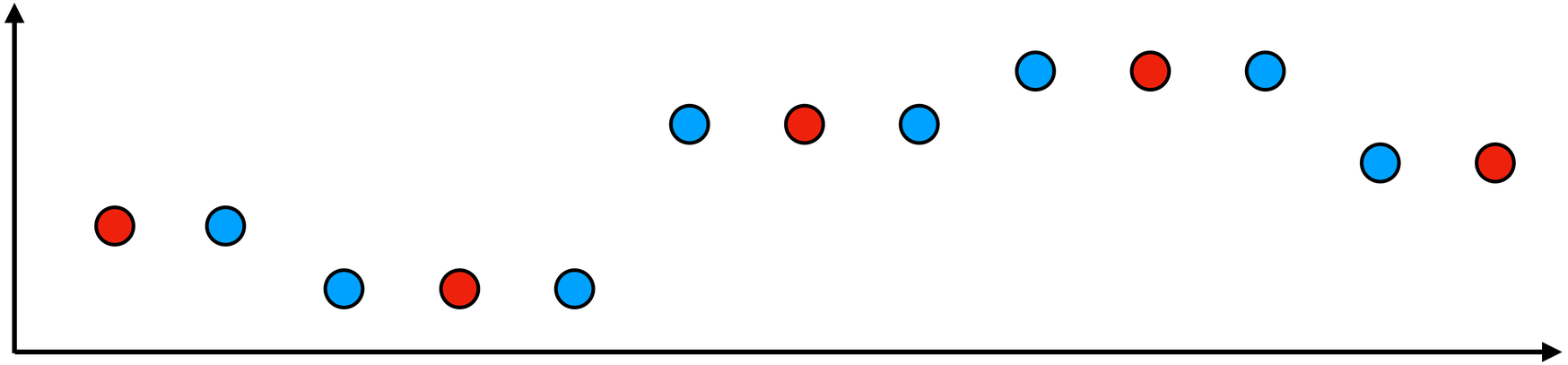
Reconstructed Signal

3	2	2	2	5	5	5	6	6	6	4

== Nearest Neighbor Interpolation

Samples

3	0	0	2	0	0	5	0	0	6	0	0	4



Reconstructed Signal

3	2	2	2	5	5	5	6	6	6	4

Reconstruction by Convoluting with a Hat Filter

Samples

0	3	0	2	0	5	0	5	0	1	0	6	0	4	0

Filter

0	0.5	1	0.5	0

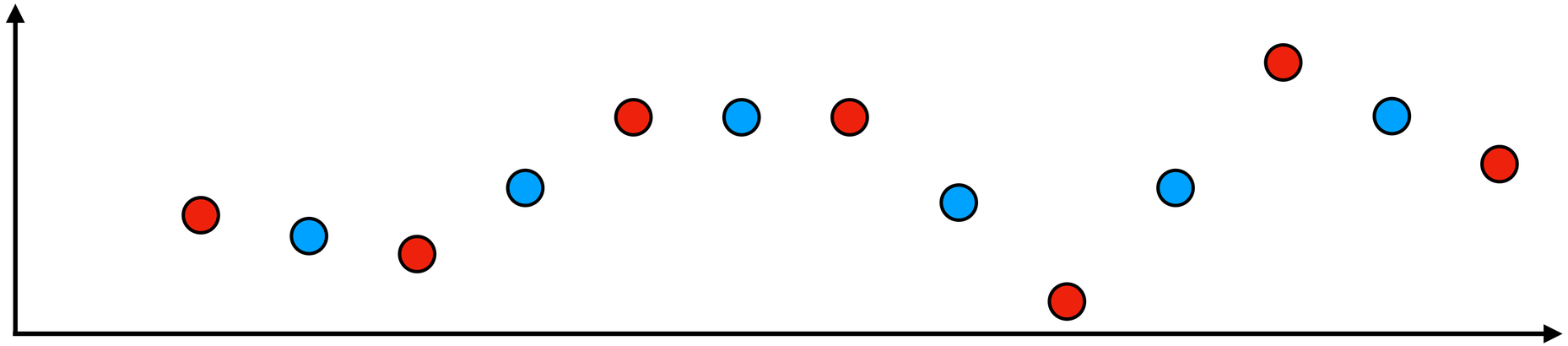
Reconstructed Signal

2.5	2	3.5	5	5	5	3	1	3.5	6	5

== Linear Interpolation

Samples

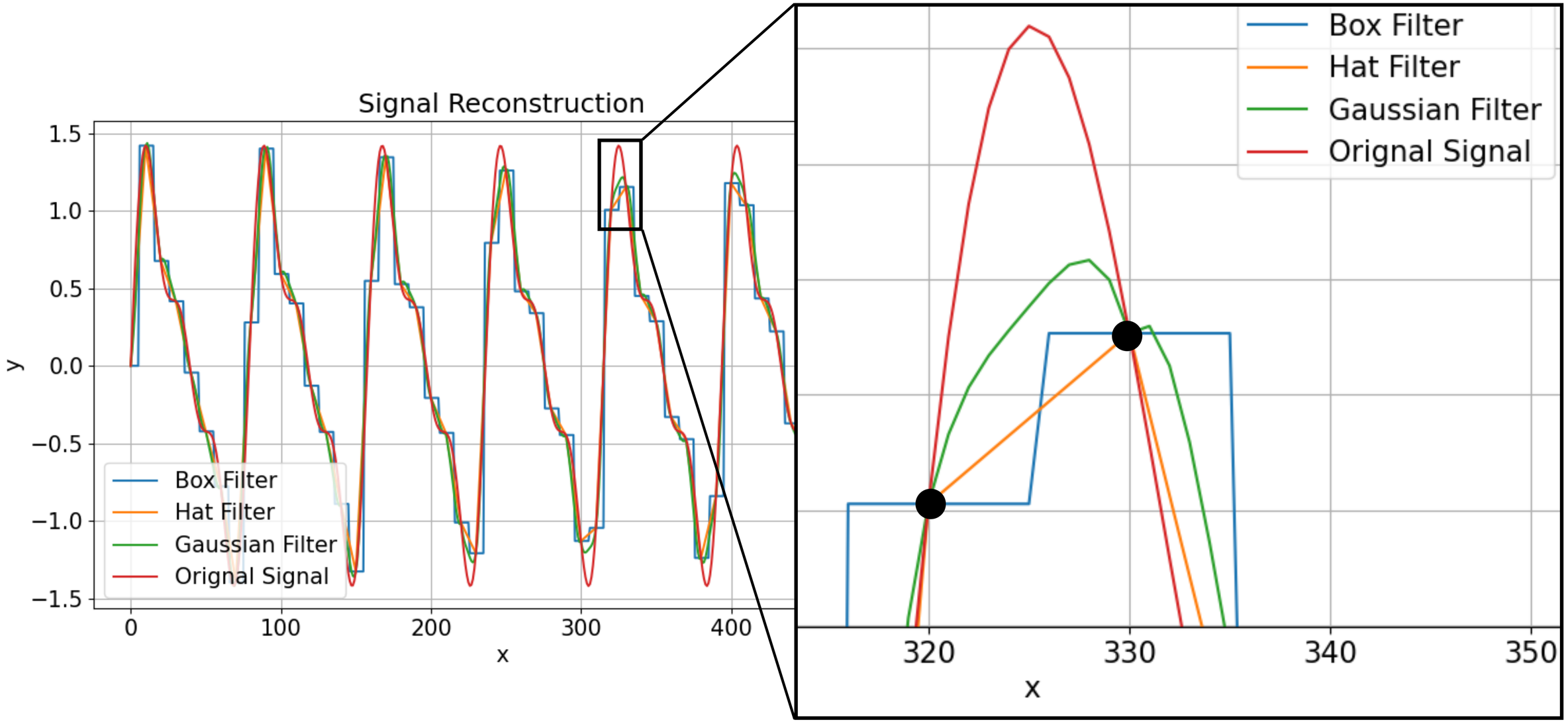
0	3	0	2	0	5	0	5	0	1	0	6	0	4	0



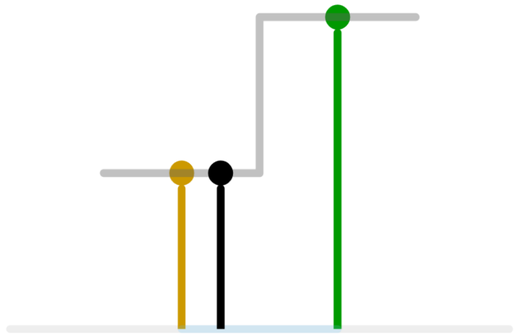
Reconstructed Signal

2.5	2	3.5	5	5	5	3	1	3.5	6	5

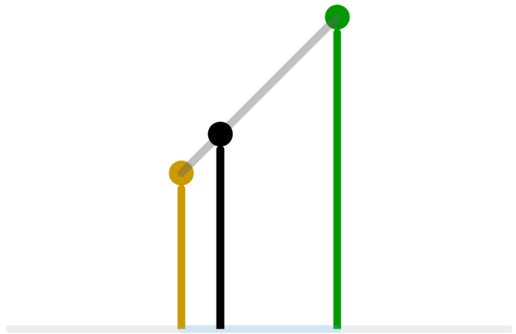
Box vs. Hat vs. Gaussian Filter



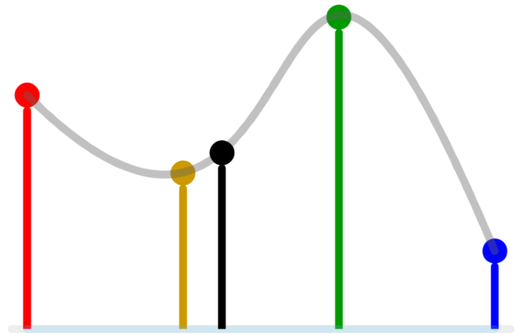
Common Filters (in 2D and 3D)



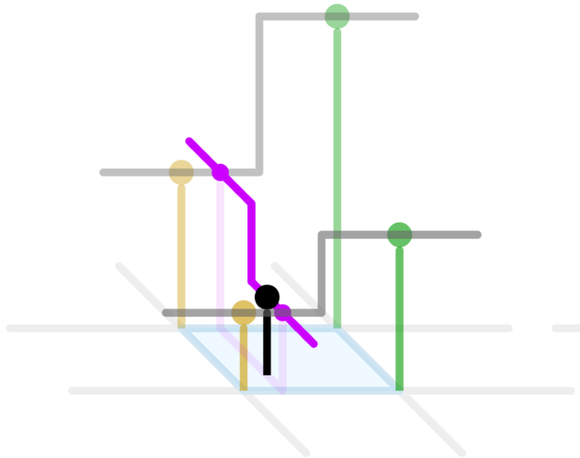
1D nearest-neighbour



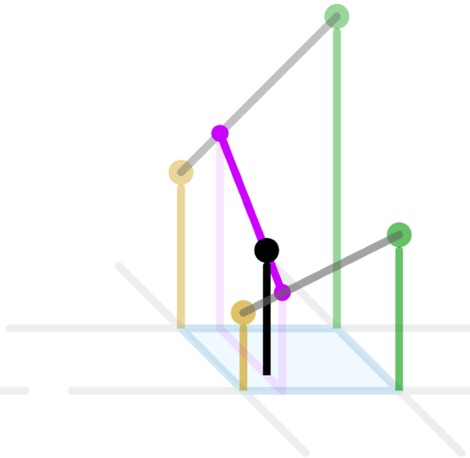
Linear



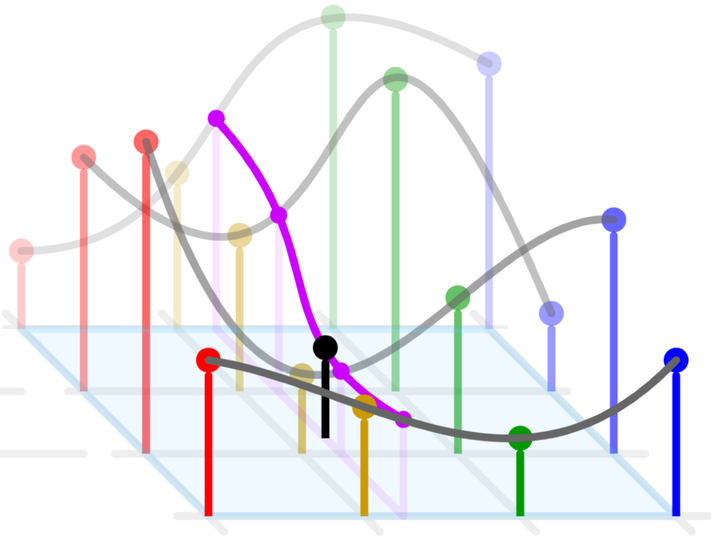
Cubic



2D nearest-neighbour



Bilinear



Bicubic

https://en.wikipedia.org/wiki/Bilinear_interpolation

Anti-Aliased Signal Reconstruction

Anti-aliasing filter
(e.g., a box filter)

Reconstruction filter
(e.g., a sinc filter)

$$s(t) = S_a(t) \star F_1(t) \star F_2(t)$$

- Convolving with multiple filters is equivalent to convolving with one composite filter.
- Most real-world signals are not band-limited, so perfect reconstruction is impossible. So the name of the game is to design the composite filter so that the perceptual quality of the reconstructed signal is acceptable.

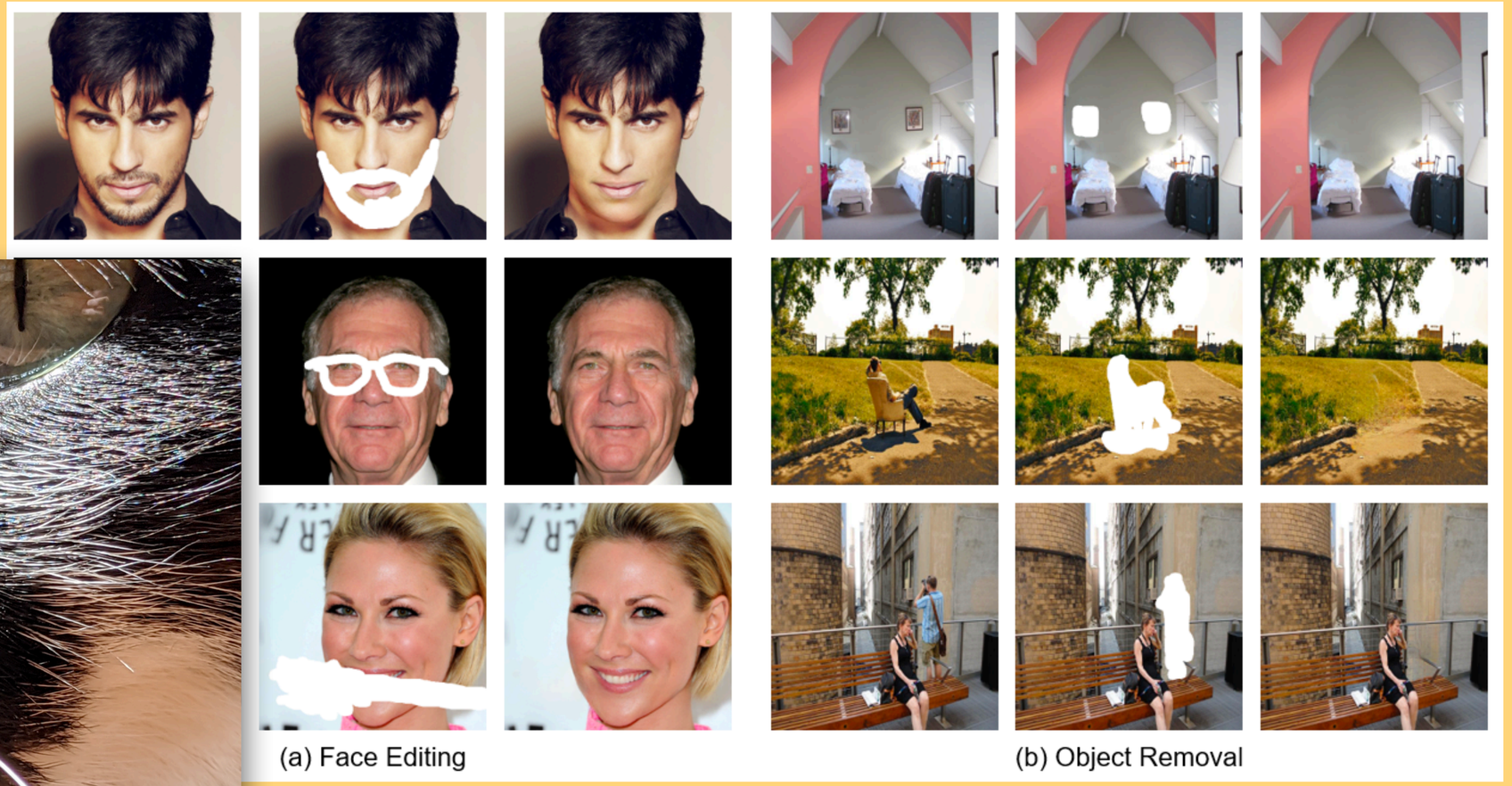
Beating Nyquist-Shannon Sampling Theorem (1)

- Nyquist-Shannon sampling theorem applies to generic signals
- If signals have strong patterns (e.g., mostly ***sparse*** in the Fourier domain), we can sample at a much lower rate than the Nyquist rate but still obtain a good reconstruction
- The technique is called compressive sensing
  - Goal: solving for the most sparse frequency domain representation that, after IFFT, is consistent with the few samples
  - “Single-pixel” camera, CT imaging, etc.
  - Must do random sampling

Beating Nyquist-Shannon Sampling Theorem (2)

- Nyquist-Shannon sampling theorem applies to reconstruction from samples of a single signal without help from **prior** information
- We could also learn from prior data to reconstruct signals
  - Through machine/deep learning
  - Image inpainting, super-resolution

Beating Nyquist-Shannon Sampling Theorem (2)



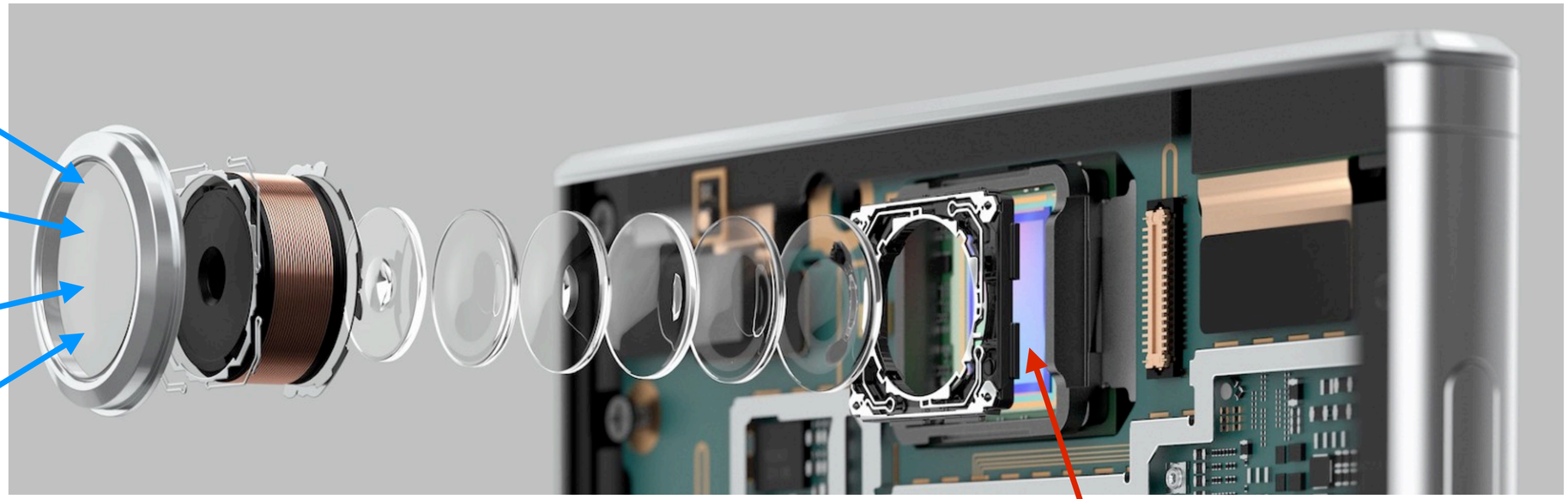
https://www.slashgear.com/google-pixel-4-camera-review-brilliant-and-frustrating-21596272/

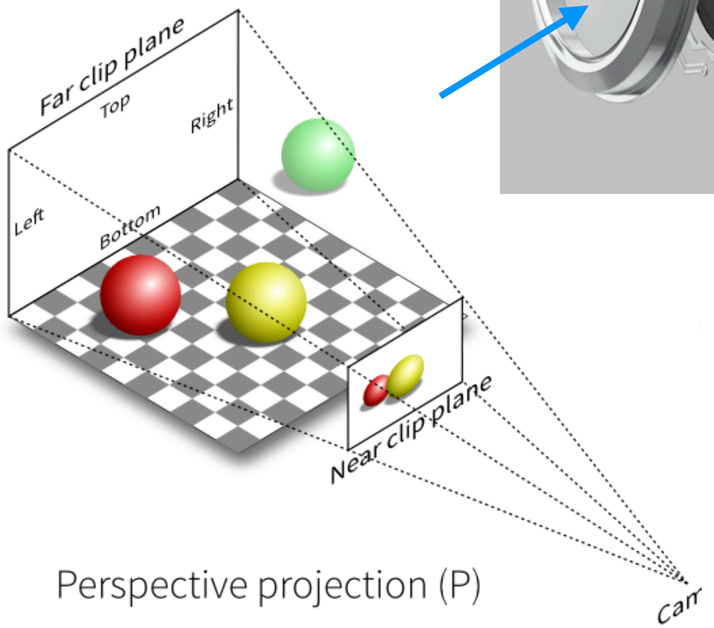
https://pythonawesome.com/aot-gan-for-high-resolution-image-inpainting/

Signal Sampling and Reconstruction in Camera and Display

Optical Signal on the Sensor Plane is 2D Continuous

Lights in the physical scene (3D continuous signal)

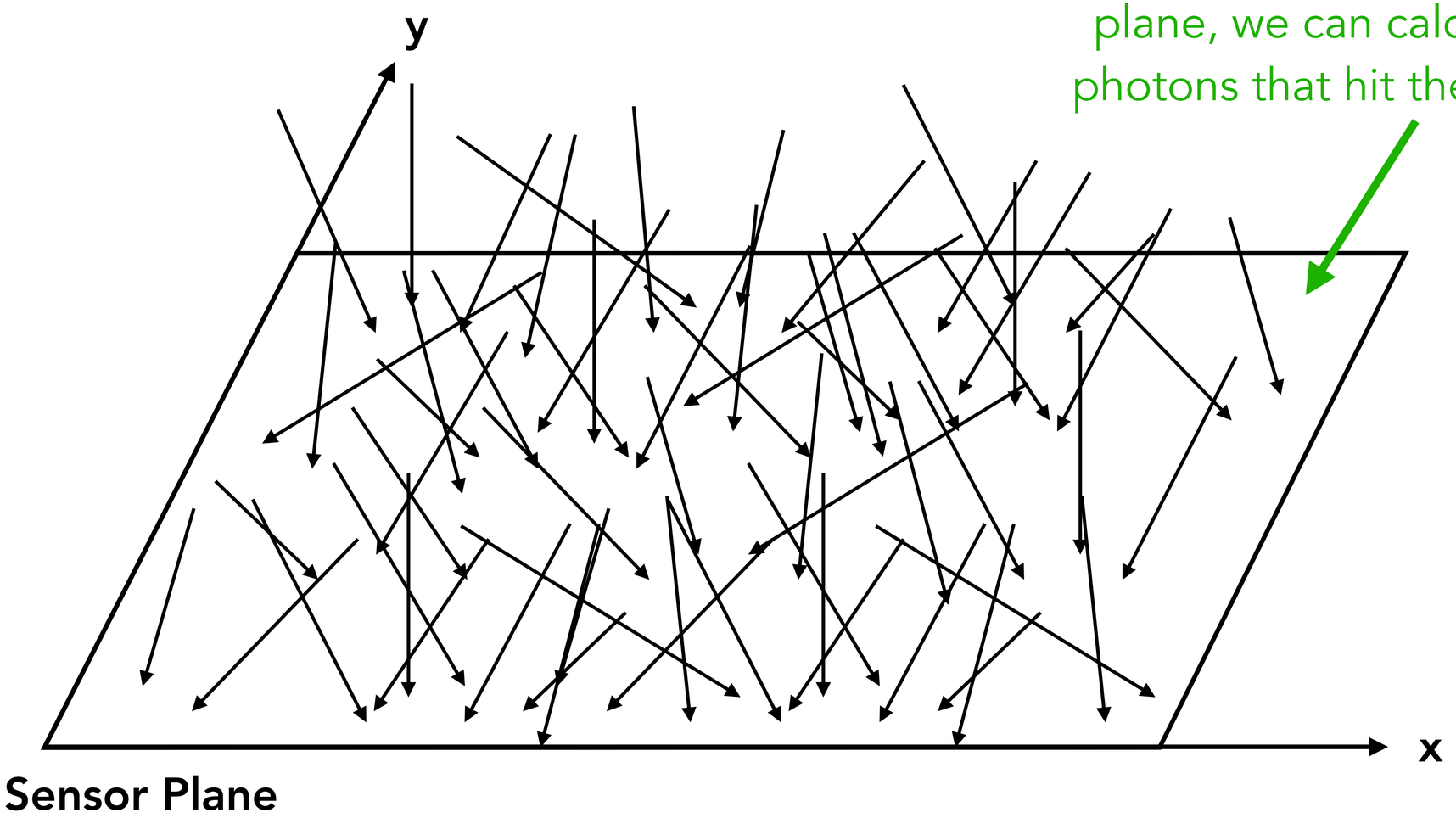




Lights on the sensor plane (2D continuous signal)

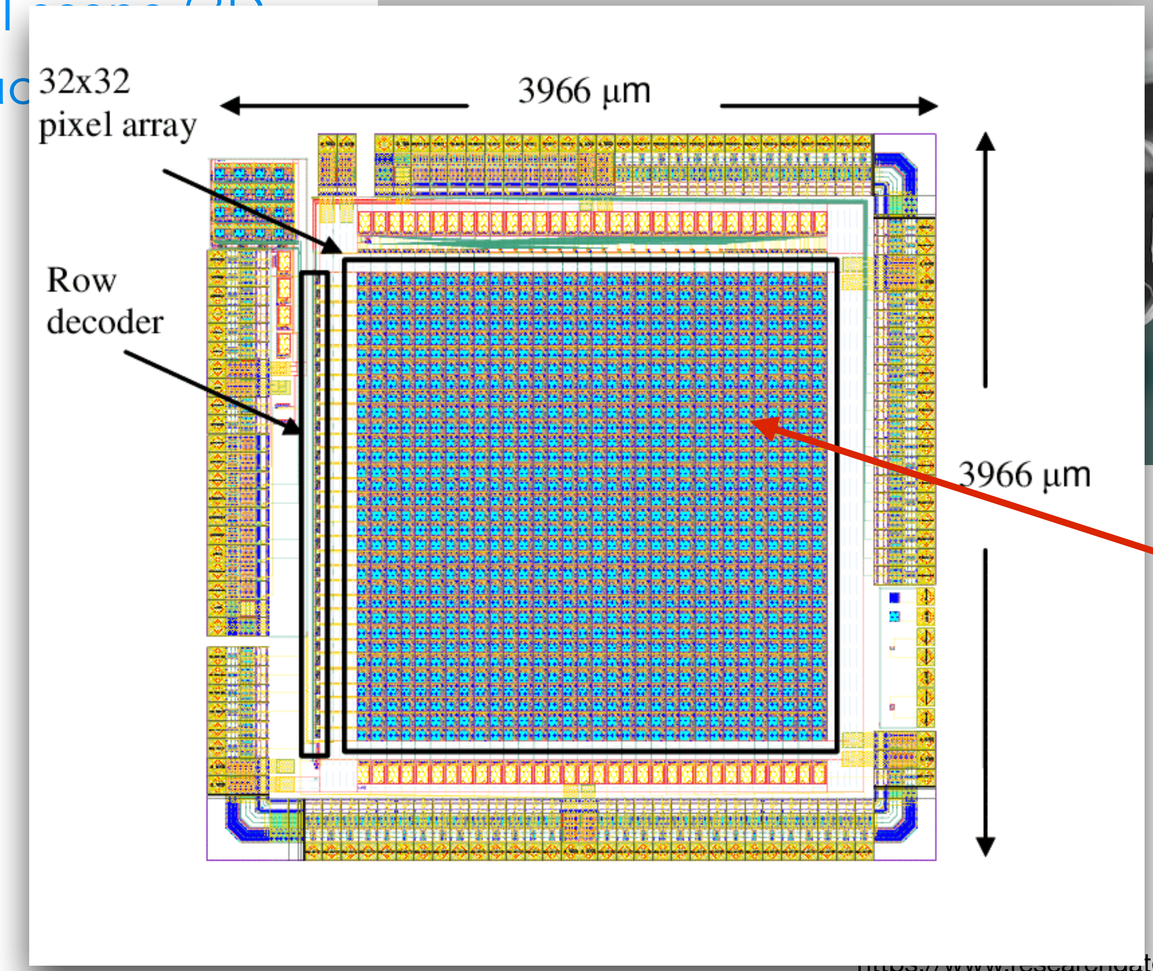
Optical Signal on the Sensor Plane is 2D Continuous

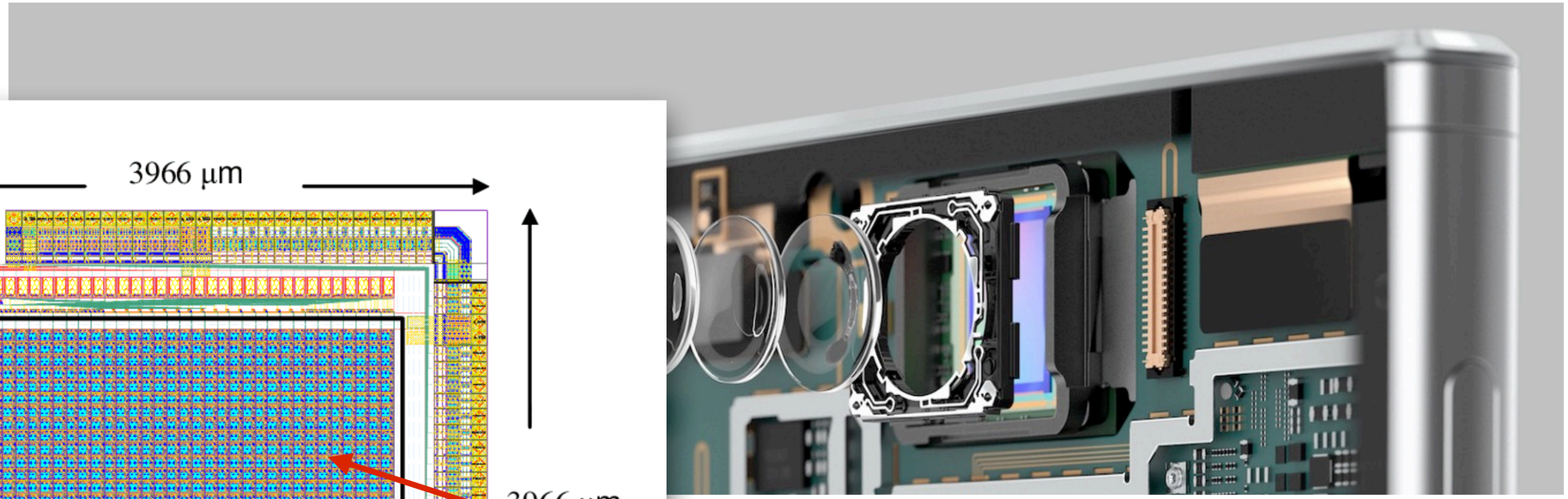
At any position [x, y] on the sensor plane, we can calculate the amount of photons that hit the position (irradiance)



Sensor Filters and Samples the 2D Continuous Signal

Lights in the physical scene (3D continuous signal)





Pixels on the image sensor **filter** and then **sample** the continuous 2D signal

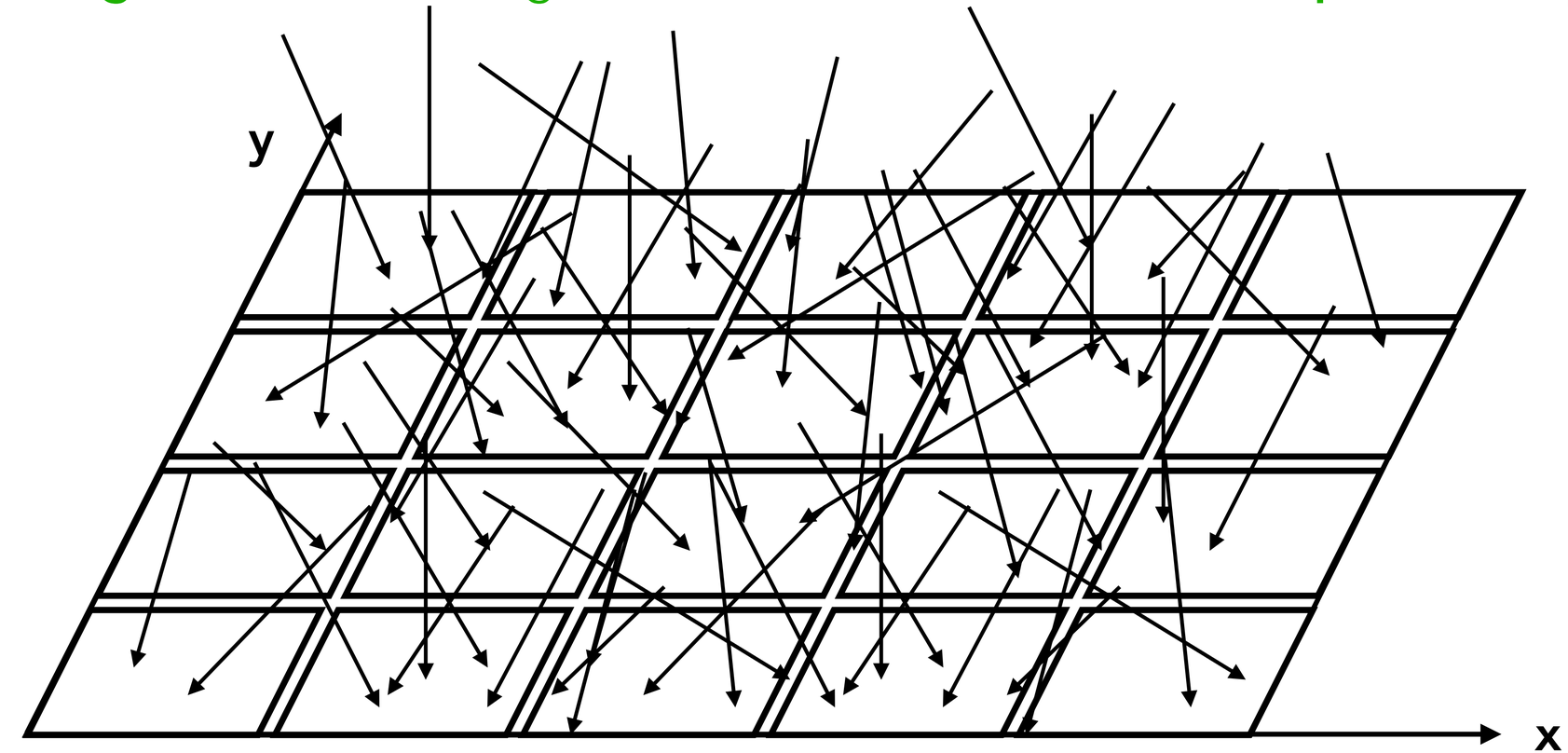
https://www.researchgate.net/publication/243717128_Optical_imaging_and_spectroscopy_of_superficial_tissue

What Do Camera Pixels Do?

integrates signal within each pixel (“counting photons”)

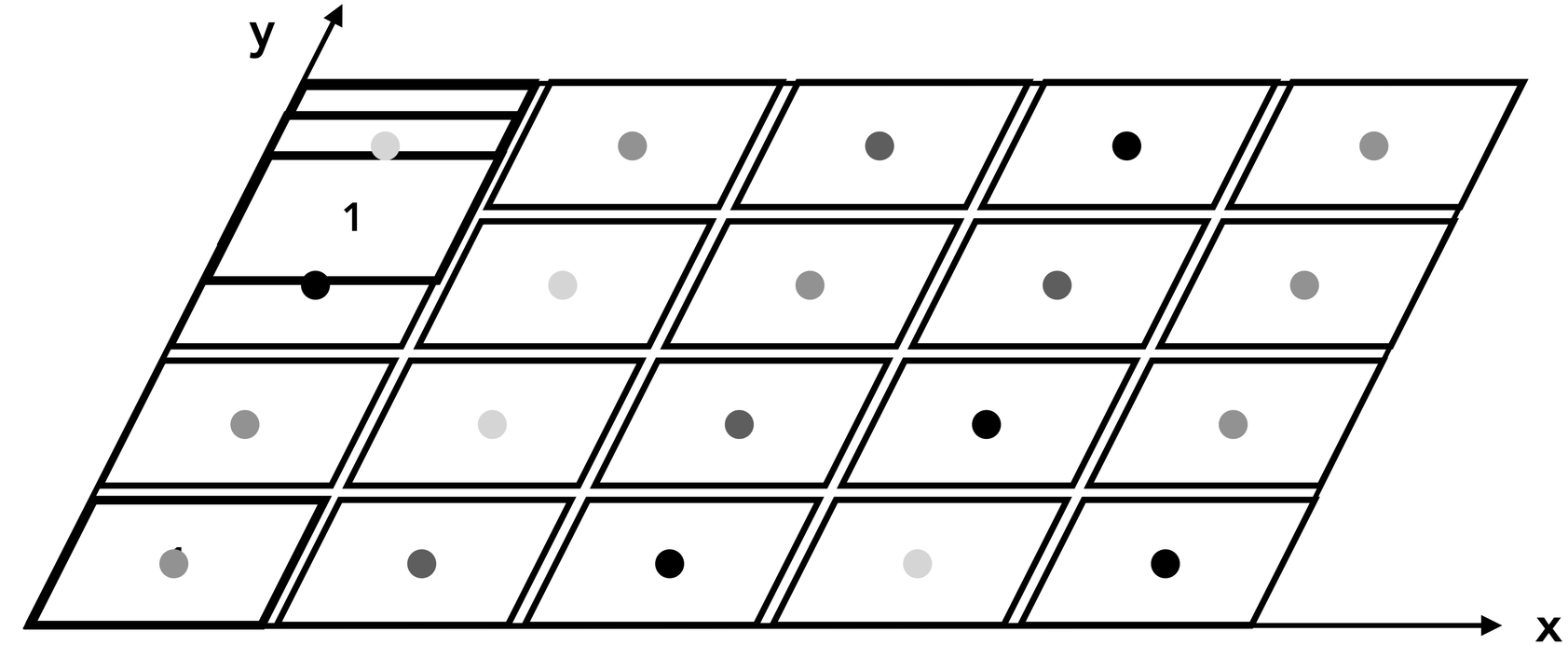
== convolving the continuous signal with a box filter at the pixel positions

== **filtering** the continuous signal with a **box filter** and then **samples** at the pixel positions!



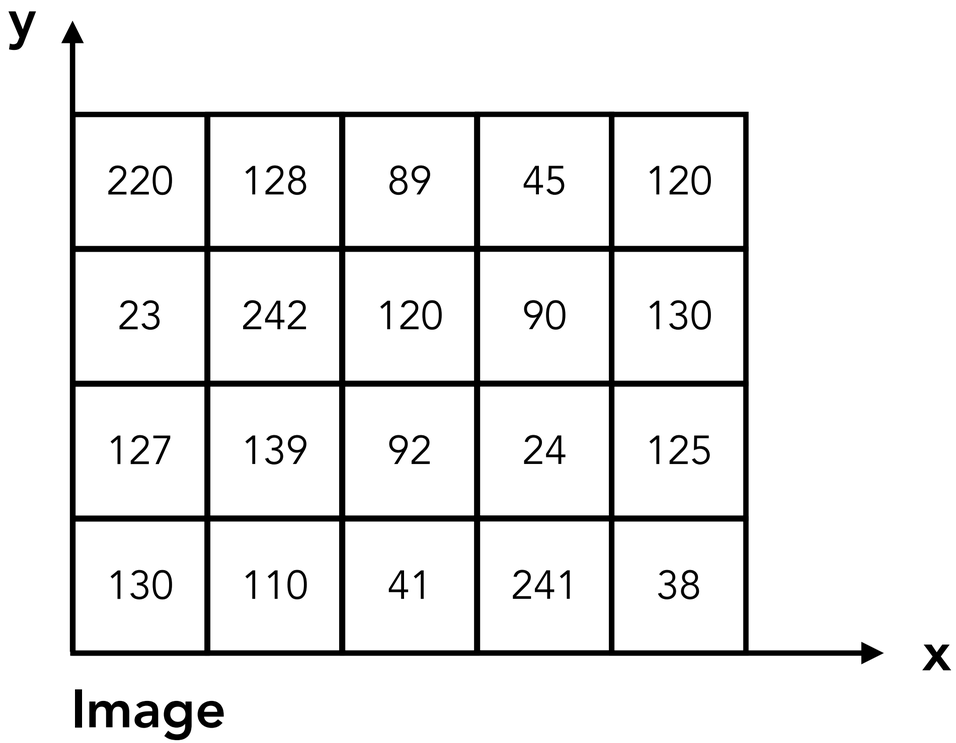
Pixel Array

Continuous Box Filtering + 2D Sampling



Pixel Array

An Image Pixel is a Sample



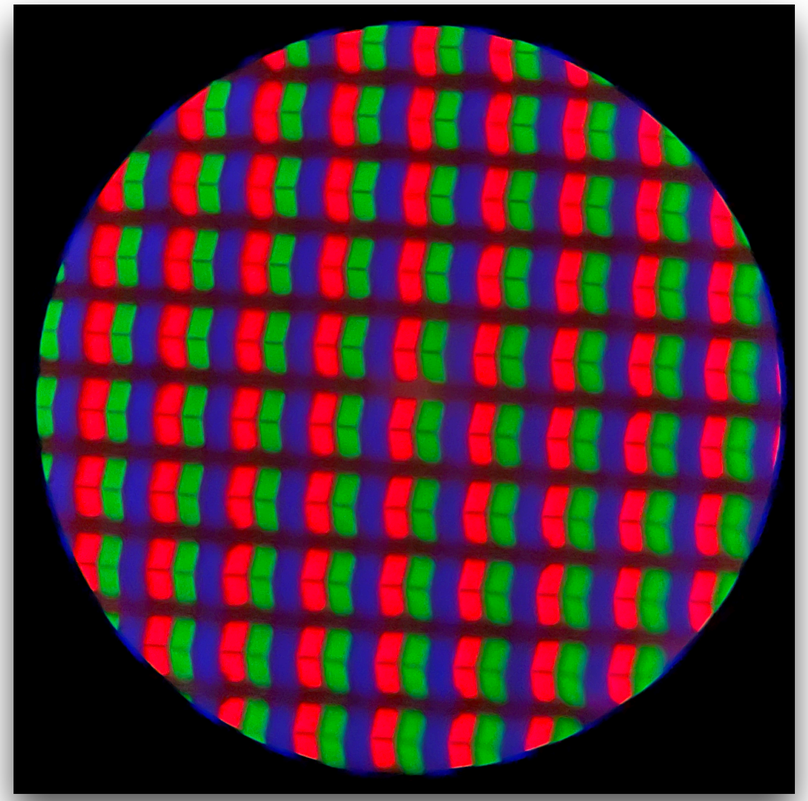
What is an image?

- Each pixel value is a single sample of the filtered 2D continuous signal, representing the integrated optical power over the pixel area.
- A camera sensor pixel has an area, but an image pixel has no area!

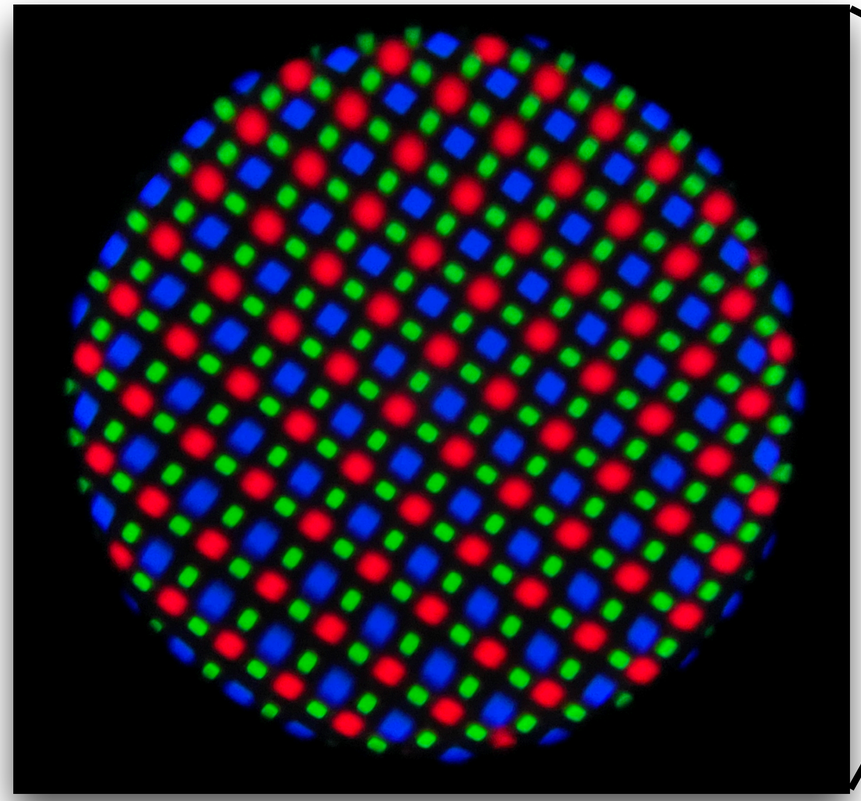
Any image post-processing should keep this in mind.

Display Reconst. Continuous Signal w/ a Box Filter

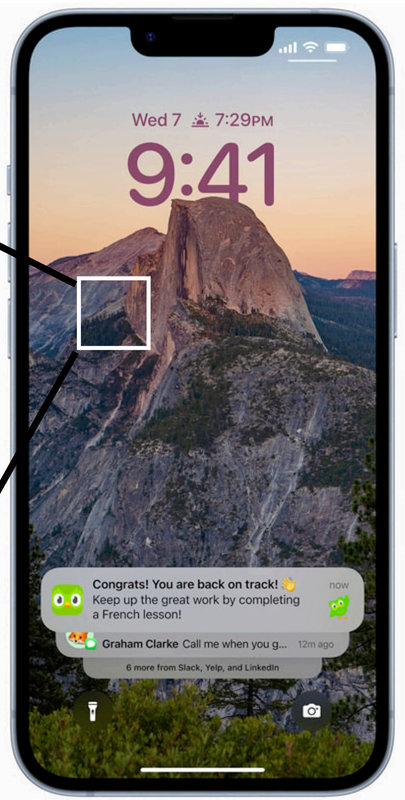
Displays reconstruct continuous 2D signal from 2D samples (pixel values)



iPhone 6



iPhone 11



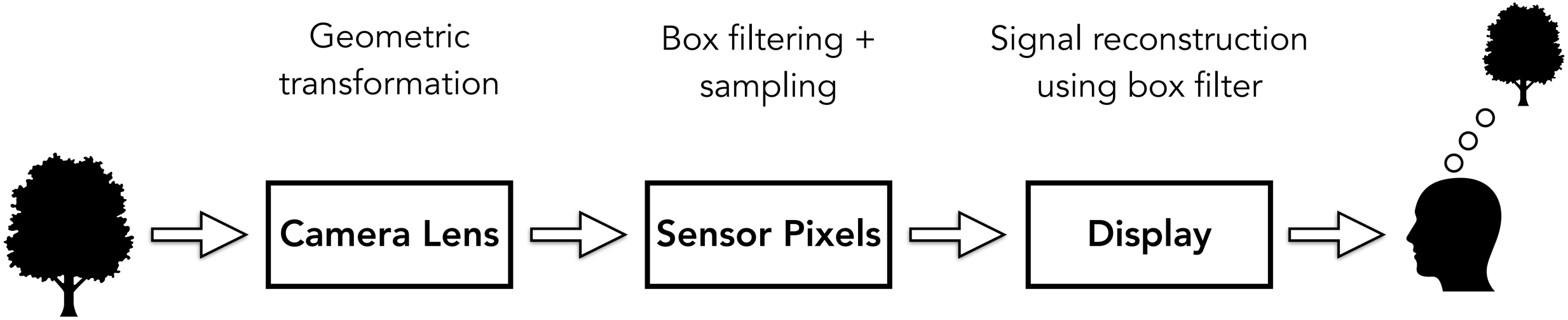
https://www.pinterest.com/pin/866943615545355525/

Sampling and Reconstruction in Camera and Display

Geometric transformation

Box filtering + sampling

Signal reconstruction using box filter



3D continuous signal

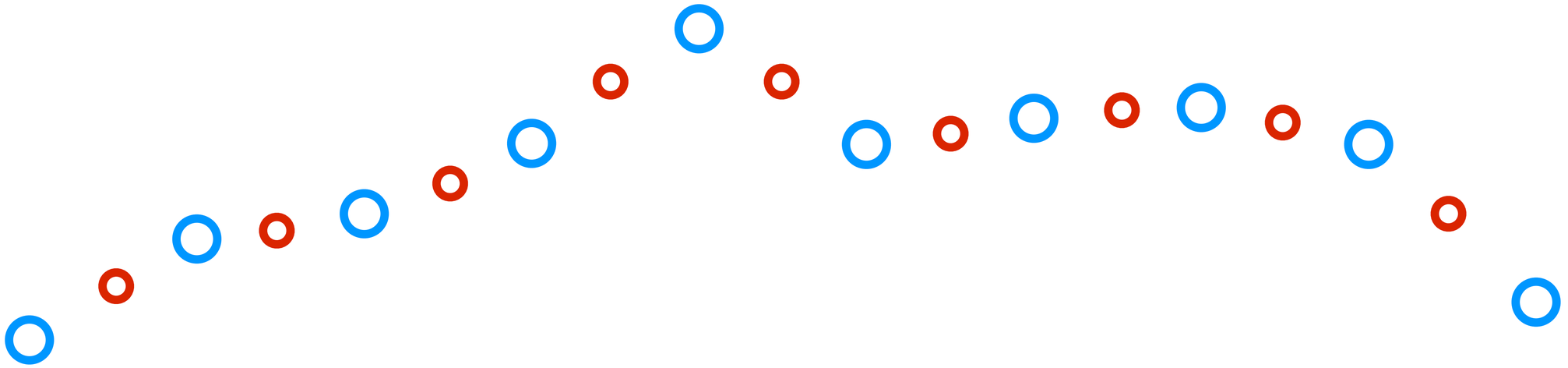
2D continuous signal (on the sensor plane)

2D sampled signal (i.e., image)

Reconstructed 3D continuous signal

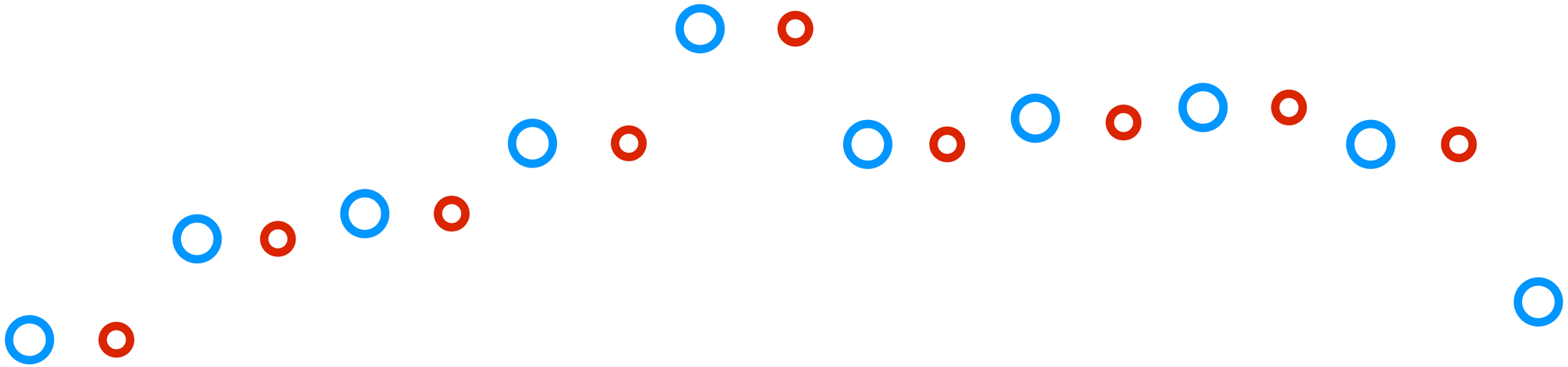
Upsampling and Downsampling

Linear Interpolation (Hat Filter)

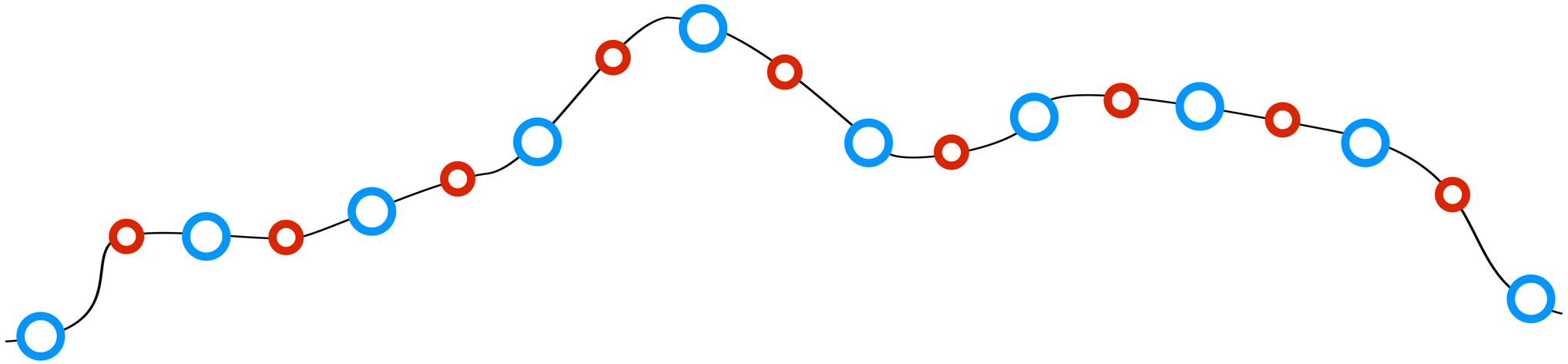


Sample and Hold (Box Filter)

Nearest neighbor is similar

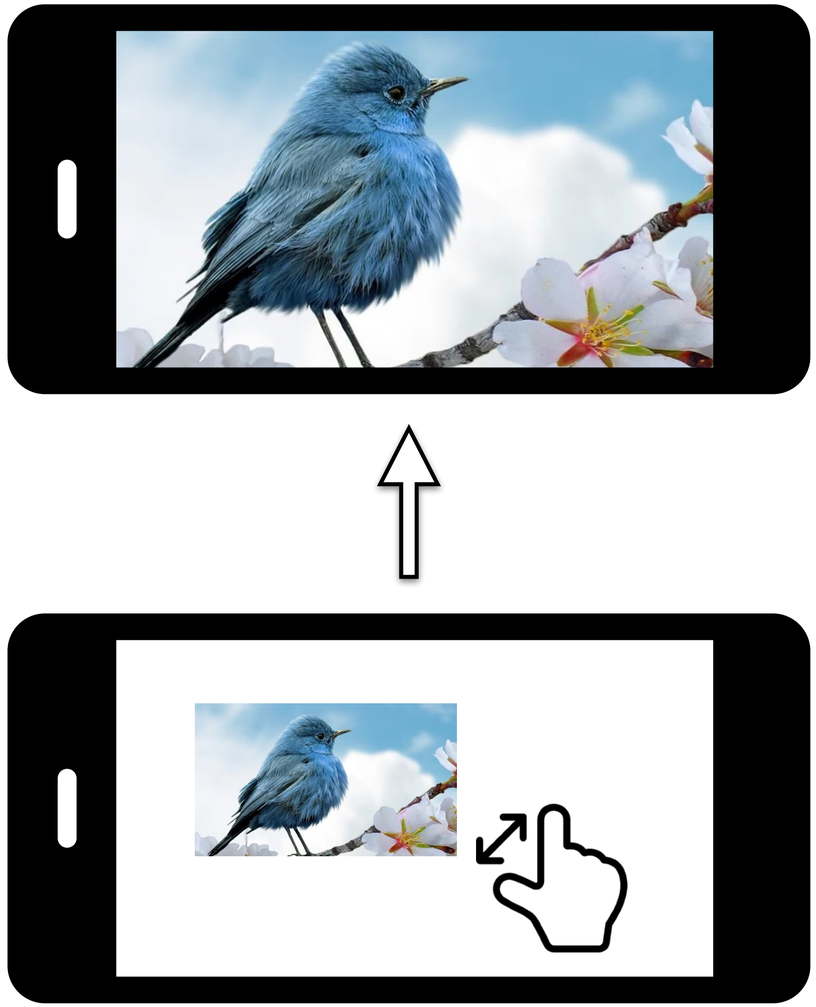


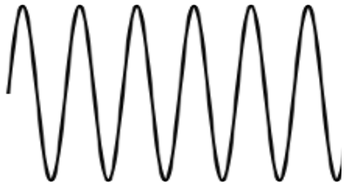
Theoretically Optimal Upsampling

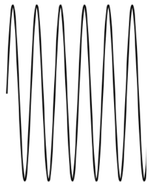


- First reconstruct the underlying continuous signal through a filter
  - With potential anti-aliasing
- And then **resample** at a desired, higher rate
- Mathematically that's equivalent to computing the convolution only at the desired re-sampled points. Filter choice is usually empirical.

Upscaling = Reduce Frequency + Upsampling

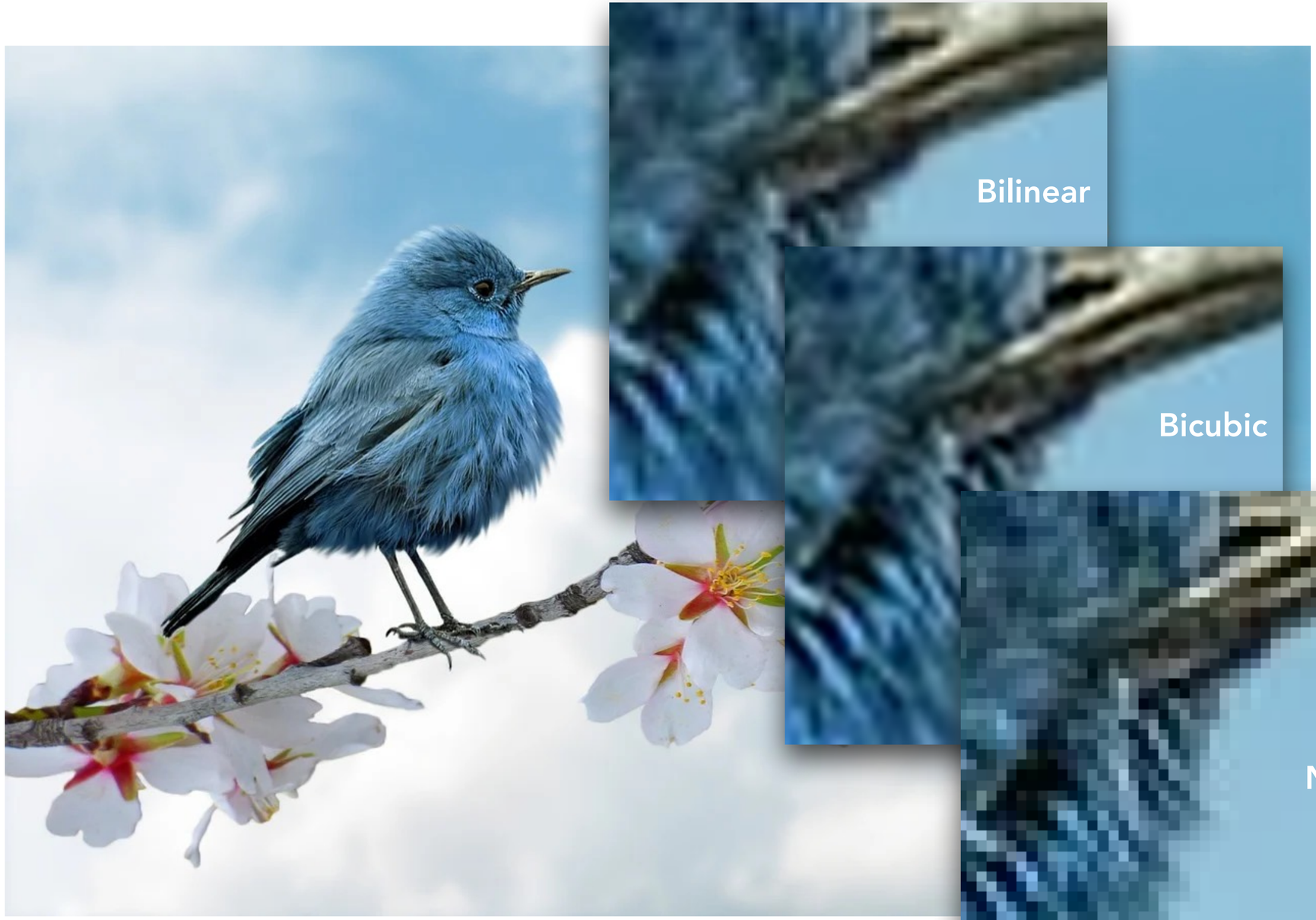


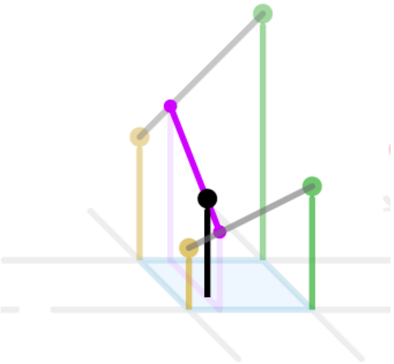




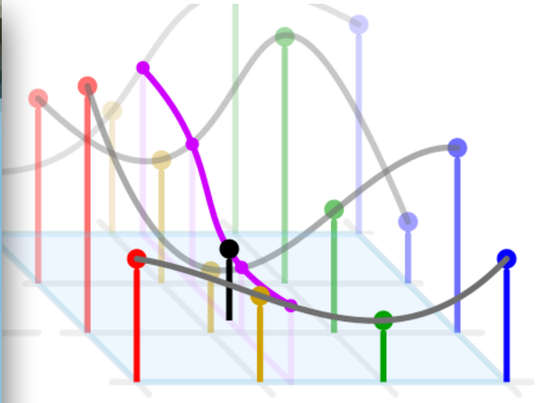
- Theoretically:
  - First “stretch” the underlying signal into a wider domain, which **reduces** the signal frequency
  - Then we sample that lower frequency signal with **more** samples!
  - This makes later reconstruction (e.g., by a display) easier in theory.
- If you want to capture details, take photo closer to an object or use a telephoto lens.

https://pixabay.com/photos/bird-blue-clouds-weather-pen-8788491/





Bilinear

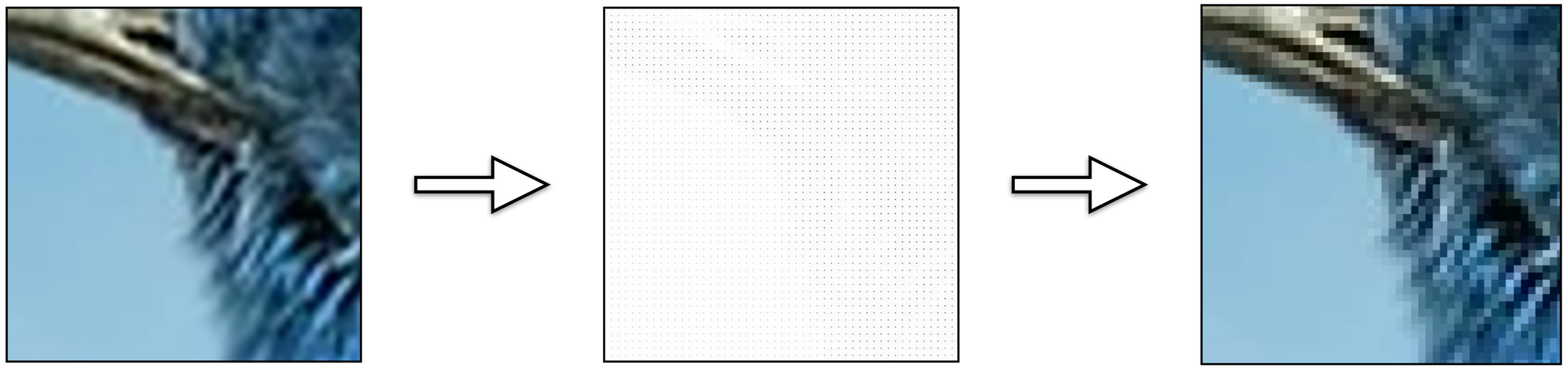


Bicubic

https://pixabay.com/photos/bird-blue-clouds-weather-pen-8788491/ https://en.wikipedia.org/wiki/Bicubic_interpolation

Downsampling

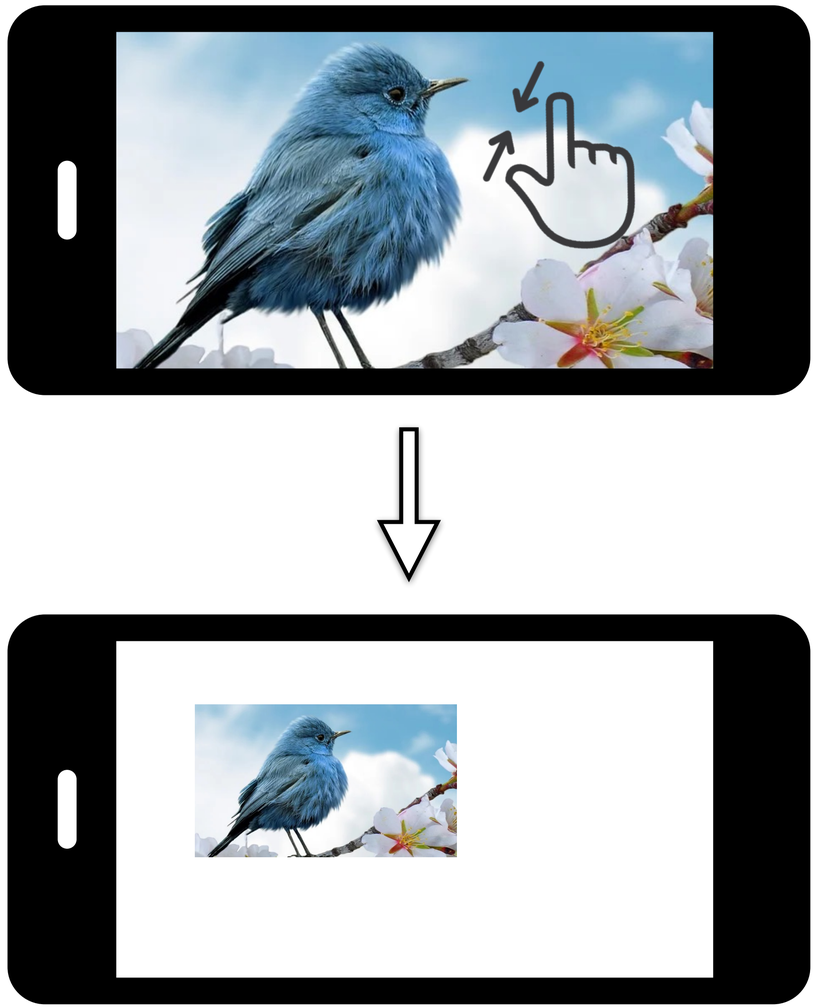
- Simply dropping samples/pixels? Why is that bad?
- Dropping pixels is equivalent to sampling the original continuous signal using a lower rate. It would make later reconstruction harder!

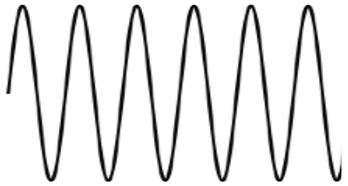


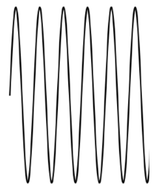
Downsampling by dropping pixels

Upsampling by nearest neighbor/box filter

Downscaling = Increase Frequency + Downsampling







- In theory:
  - First “squeeze” the underlying signal into a narrower domain, which **increases** the signal frequency
  - Then we sample that higher frequency signal with **fewer** samples!
- Now displaying the image at the downsampled resolution
  - Very hard: reconstructing a higher frequency signal from fewer samples.

Key Things to Take Away

- Nyquite-Shannon Sampling Theorem
  - Sample at 2f if f is the maximum frequency in a signal
  - Otherwise aliasing occurs.
- Anti-Aliasing
  - Pre-filter high-frequency components
  - Pre-filtering could be done either in the frequency domain or in the spatial domain (convolution); they are equivalent according to the Convolution Theorem
- Real-world examples:
  - Camera (box filter + sampling) → Image Pixels → Display (box filter reconstruction)
  - Downsampling/upsampling are resampling problems