CS559: Computer Graphics

Lecture 3: Digital Image Representation

Li Zhang

Spring 2008

Image Representation

- Images
  - Something that represents a *pattern of light* that will be *perceived by something*
- Computer representation
  - Sampled

From http://www.unl.edu/dpilson/sunflower.html

Image Representation

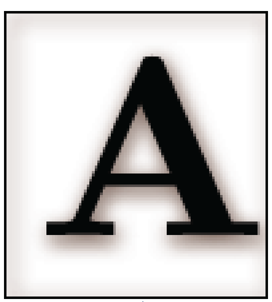
- Images
  - Something that represents a *pattern of light* that will be *perceived by something*
- Computer representation
  - Sampled
  - Object based →

Sampled

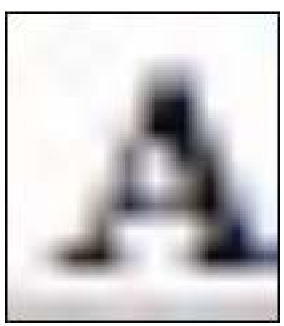
from James O'Brien

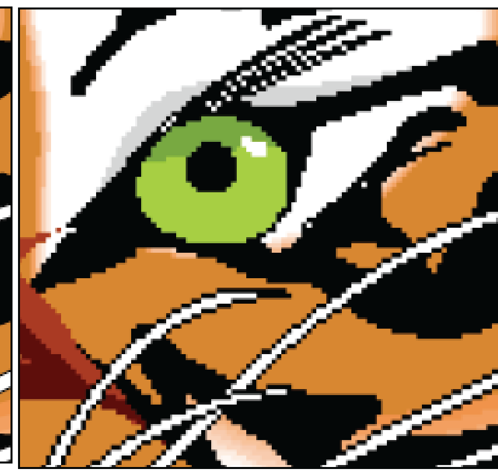
Image Representation

- Images
  - Something that represents a *pattern of light* that will be *perceived by something*
- Computer representation
  - Sampled
  - Object based →



PS Type one font





Sampled

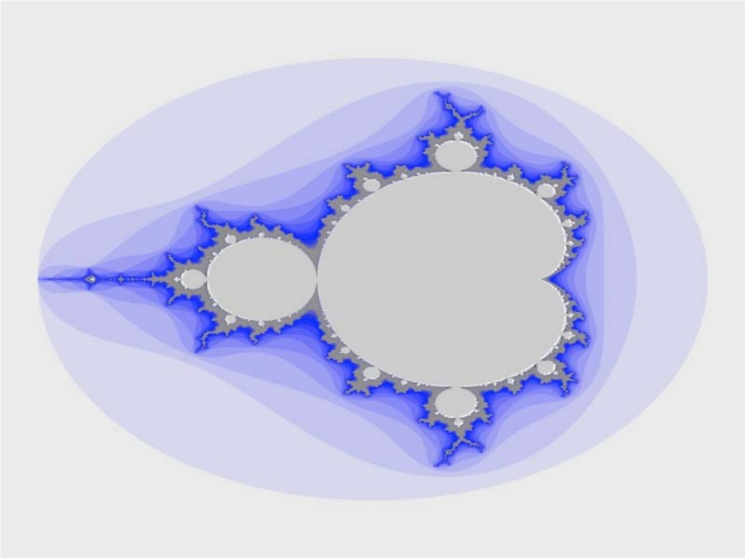
from James O'Brien

Image Representation

- Images
  - Something that represents a *pattern of light* that will be *perceived by something*
- Computer representation
  - Sampled
  - Object based
  - Functional ➡

f(z) = z*z+c
where z is a complex number

Set of c such that
{f(0), f(f(0), f(f(f(0)), …} is bounded



Mandelbrot Fractal Plot by Vincent Stahl

http://upload.wikimedia.org/wikipedia/en/c/ce/Mandelbrot_zoom.gif

Image Representation

- Images
  - Something that represents a *pattern of light* that will be *perceived by something*
- Computer representation
  - Sampled
  - Object based
  - Functional ⟹

Image Representation

- Images
  - Something that represents a *pattern of light* that will be *perceived by something*
- Computer representation
  - Sampled
  - Object based
  - Functional

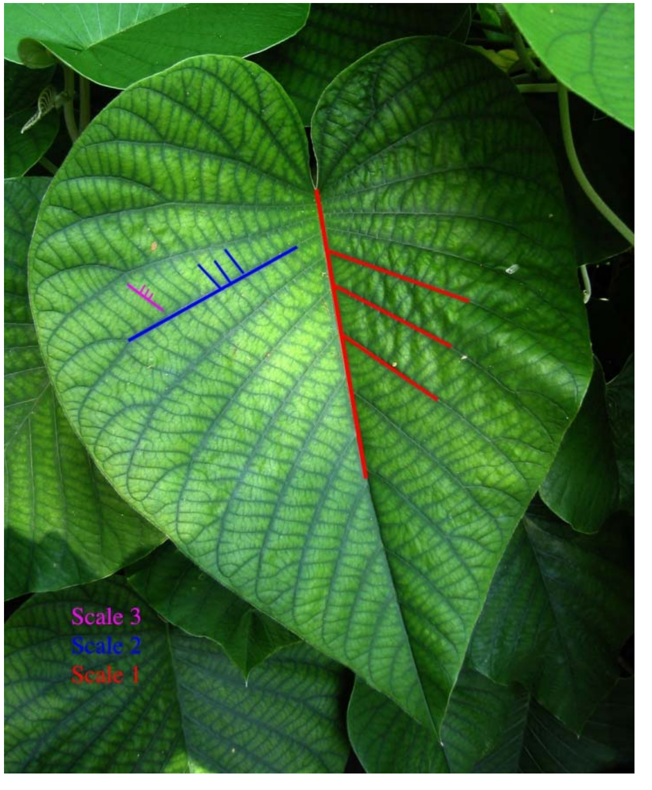
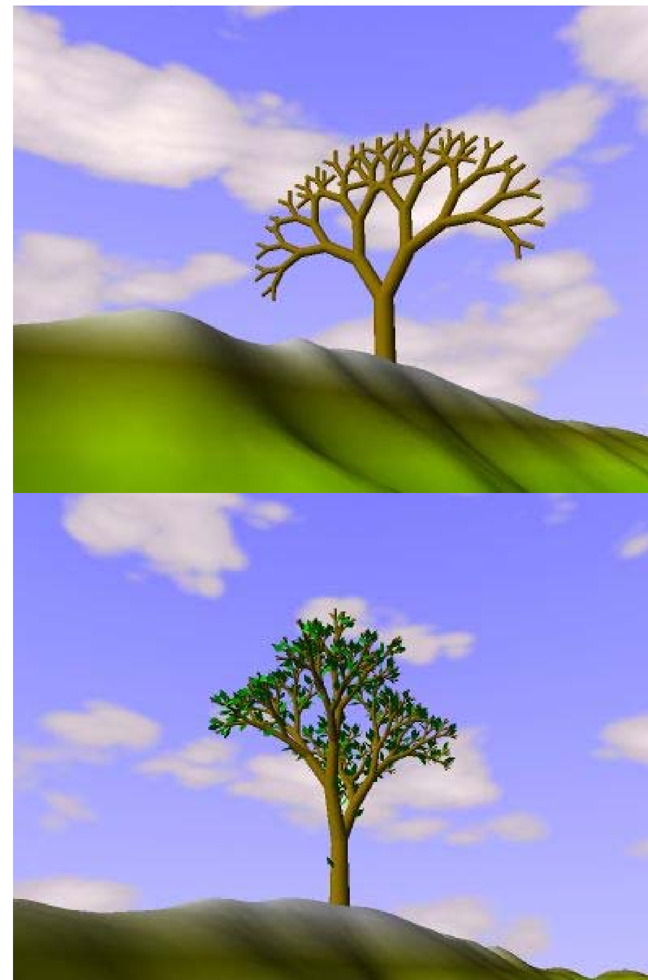


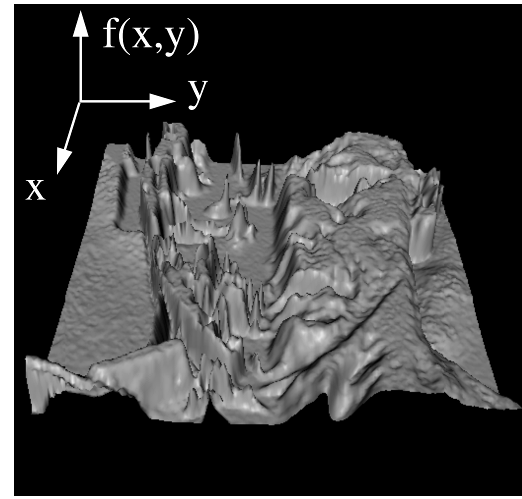
Image Representation

- Images
  - Something that represents a *pattern of light* that will be *perceived by something*
- Computer representation
  - Sampled
  - Object based
  - Functional ➡



L-system tree
http://gamedev.cs.cmu.edu/graphics1/lab4.php

Image as a discreet function



Represented by a matrix:

j →

i ↓

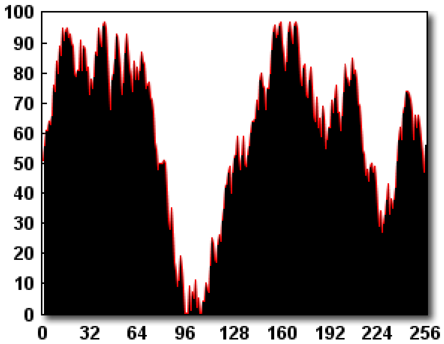
62	79	23	119	120	105	4	0
10	10	9	62	12	78	34	0
10	58	197	46	46	0	0	48
176	135	5	188	191	68	0	49
2	1	1	29	26	37	0	77
0	89	144	147	187	102	62	208
255	252	0	166	123	62	0	31
166	63	127	17	1	0	99	30

Q1: How many discrete samples are needed to represent the original continuous function?

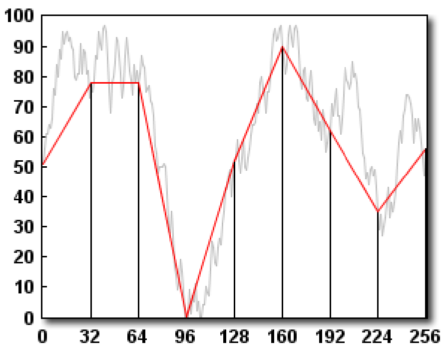
Q2: How to reconstruct the continuous function from the samples?

Sampling a continuous function (1D)

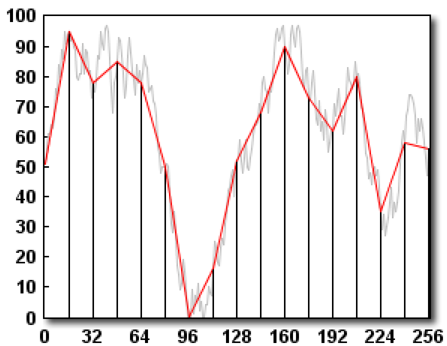
Continuous Function



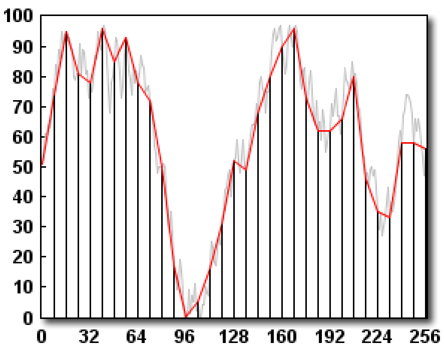
Discrete Samples



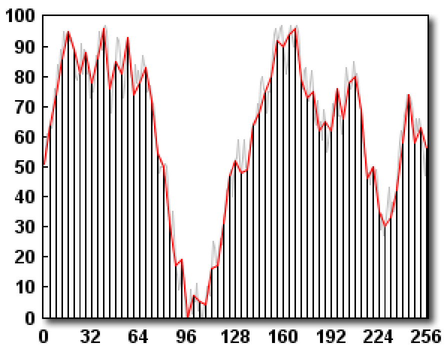
Sampling Period T = 32



Sampling Period T = 16



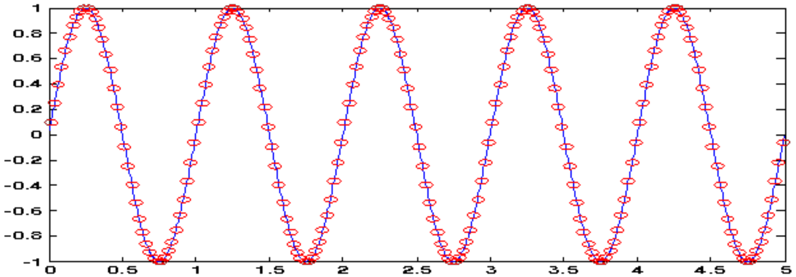
Sampling Period T = 8



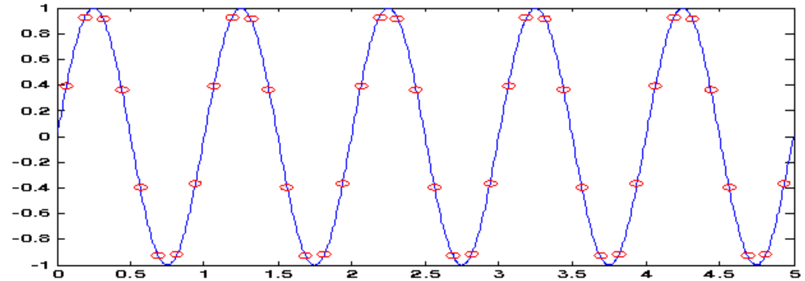
Sampling Period T = 4

The denser the better, but what's the minimum requirement?

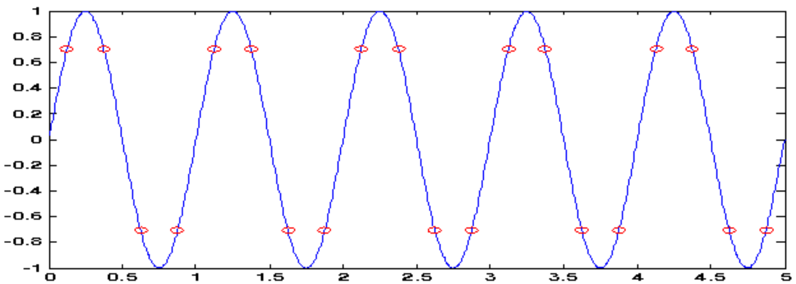
Consider a simple case – sine wave



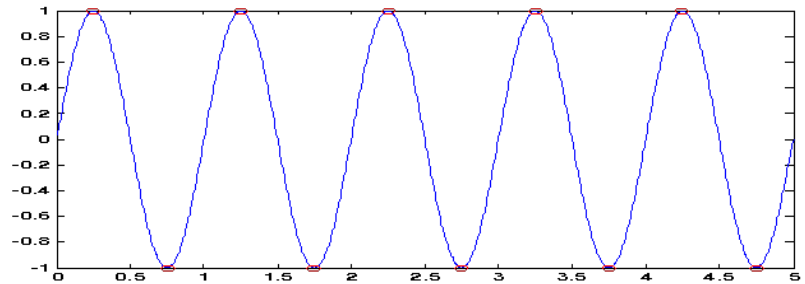
40 samples per period



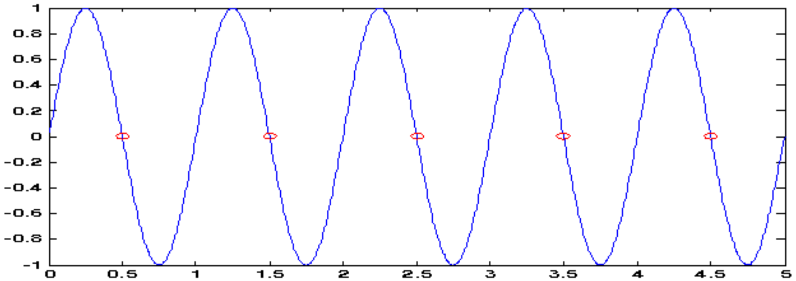
8 samples per period



4 samples per period



2 samples per period



1 samples per period

Intuitively, each period should have at least 2 samples to represent the up-and-down shape of sine wave.

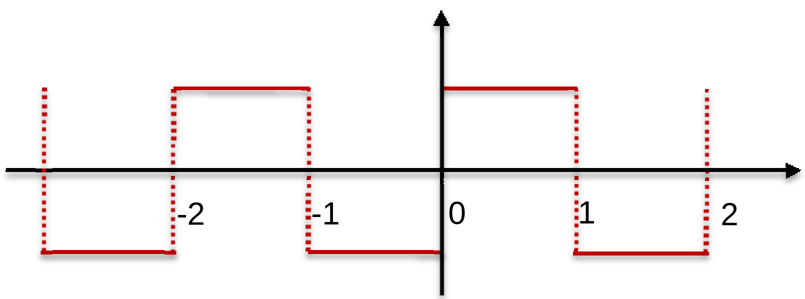
Theoretically, it can be proved that if we have more than 2 samples per period, we can recover the sine wave from the samples.

How about general functions?

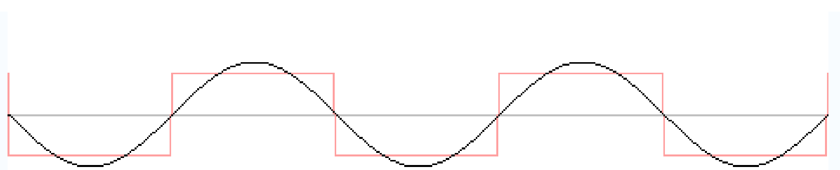
- Idea: represent an arbitrary function using sine waves.

Fourier Series

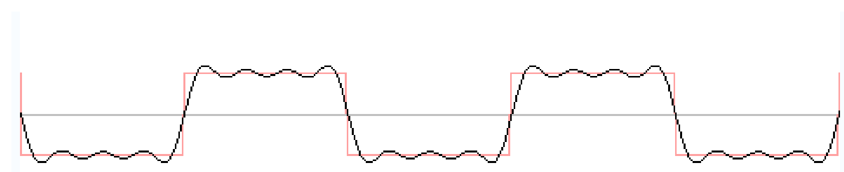
- Any periodic function f(x) can be expressed by summing up a sequence of sine and cosine waves. For example



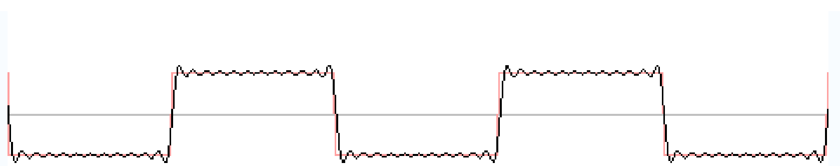
$$= \sum_{n=1,3,4,\ldots}^{\infty} \frac{4\pi}{n} \sin(2\pi n x)$$



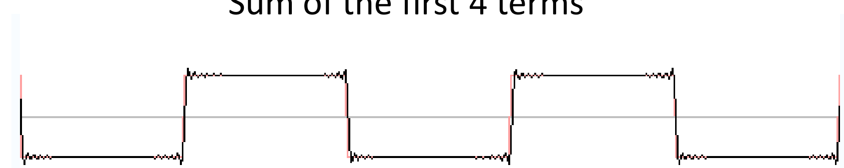
Sum of the first 1 term



Sum of the first 4 terms



Sum of the first 12 terms

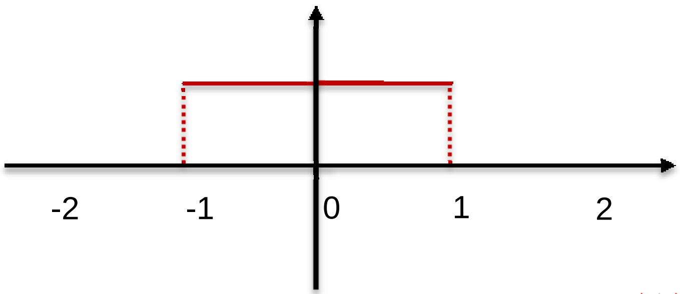


Sum of the first 25 terms

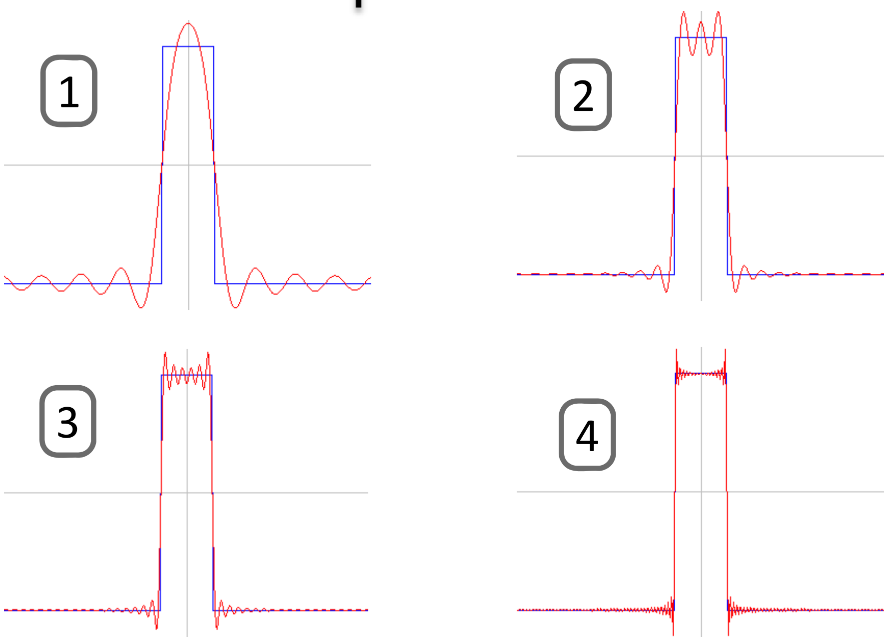
If we approximate the function by a finite number of sine terms, we can sample this function at a sampling frequency that is twice its highest sine wave frequency.

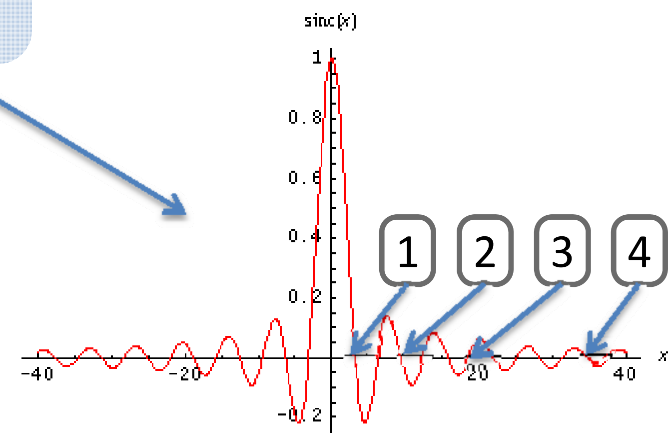
Fourier Transform

- In general, a non periodic function f(x) can be represented as a sum of sin's and cos's, using all frequencies. For example,



$$= \int_{-\infty}^{\infty} \frac{\sin \pi s}{\pi s} \cos(2\pi s x)\, ds$$





Fourier Transform

- In general, a non periodic function f(x) can be represented as a sum of sin's and cos's, using all frequencies.

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

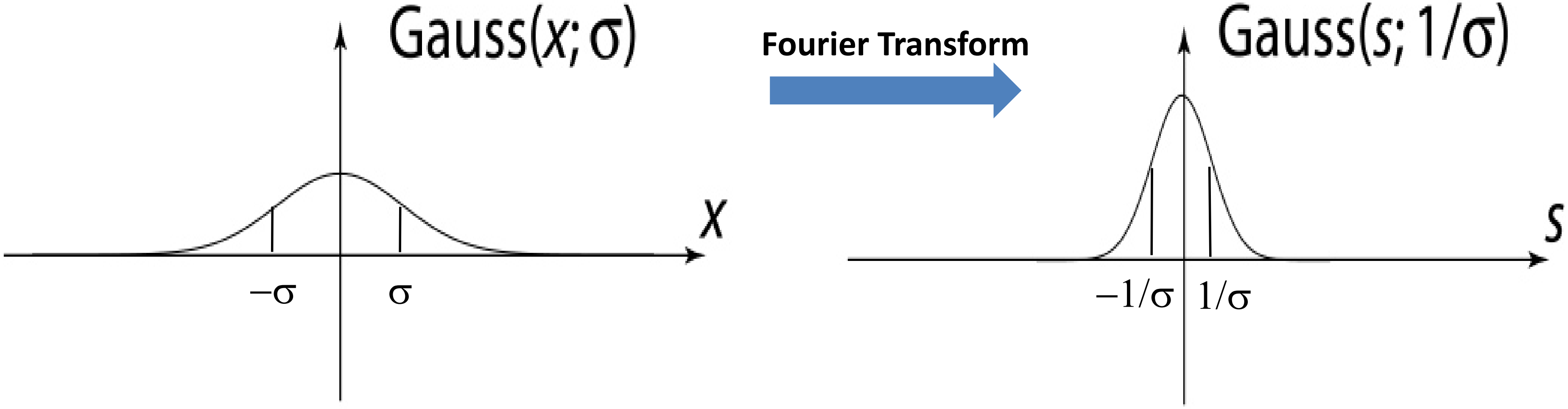
$$e^{i\theta} = \cos\theta + i \sin\theta$$

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

F(s) is the **Fourier Transform** of f(x)

Another example of Fourier Transform

$$Gauss(x;\sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}}$$



The Fourier Transform of a Gauss is still a Gauss

Sampling theorem

- This result is known as the **Sampling Theorem** and is due to Claude Shannon who first discovered it in 1949:

A signal can be reconstructed from its samples without loss of information, if the original signal has no frequencies above ½ the sampling frequency.