

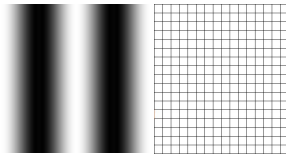
# Objectives

- Using sines and cosines to reconstruct a signal
- The Fourier Transform
- Frequency Domains for a Signal
- Three properties of Convolution relating to Fourier Transform

## Image Spectra for Beginners

### Image Representation

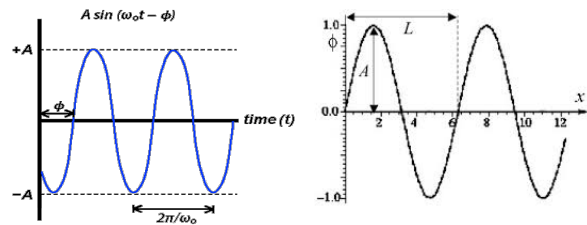
- Reviews: Viewed as pixel intensities varied between
  - 0, 255
- Now we are to see how we to model
  - **Detail** and **contrast** in images by using sine waves.
  - Fine detail is high frequency
  - Contrast is coarse grain detail and low frequency



Basic Principle: Fourier theory states that any signal, in our case visual images, can be expressed as a sum of a series of sinusoids

[http://gsimaging.com/ccd\\_noise\\_interpret\\_ffts.html](http://gsimaging.com/ccd_noise_interpret_ffts.html)  
<http://cns-alumni.bu.edu/~slehar/fourier/fourier.html>

### Sine Waves

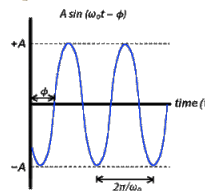
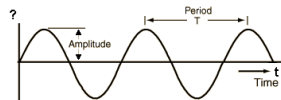
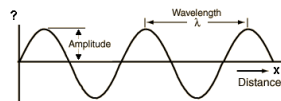


- Variables: The variable of a sine function can be a time variable or a spatial variable:
- $Y(t) = A \sin(\omega t + p)$  -- time variable, t. (e.g., sound, pressure waves)
  - $Y(t) = A \sin(kx + p)$  -- spatial variable x. (e.g., water waves)

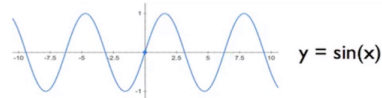
[https://en.wikipedia.org/wiki/Sine\\_wave](https://en.wikipedia.org/wiki/Sine_wave)

### Specifying a Sine Wave (1D)

- **Direction**
  - Normally we see waves that are represented a traveling in the positive x-direction, but a sine wave can move in any direction.
- **Wavelength ( $\lambda$ )** distance traveled in one cycle.
  - Period (sec/cycle), or
  - Frequency, f (cycles/sec)
    - How often (e.g., Hz)
- **Amplitude (A)**
- **Phase Shift ( $\phi$ )**
- **Vertical Shift**

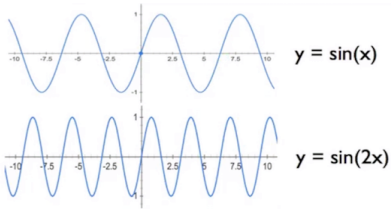


### Adding Sine Waves

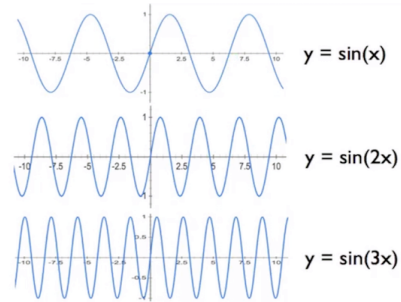


[https://en.wikipedia.org/wiki/Sine\\_wave](https://en.wikipedia.org/wiki/Sine_wave)

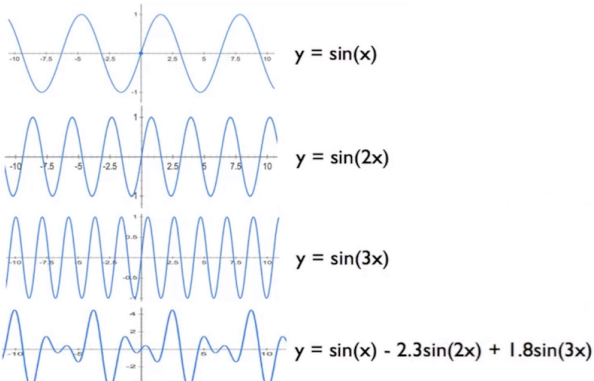
## Adding Sine Waves



## Adding Sine Waves

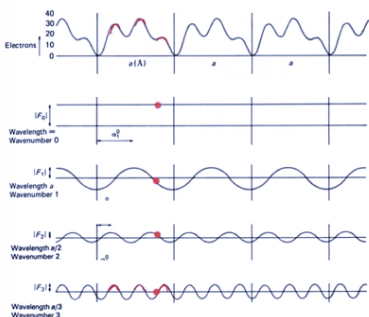


## Adding Sine Waves



- We can also do the opposite
  - Take a complex wave and take its sums apart.

## Method



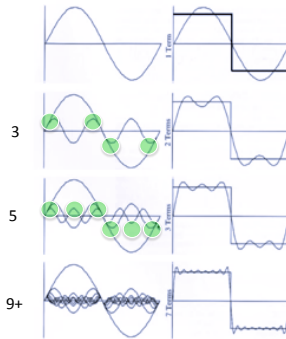
1. **0<sup>th</sup> Wave form:** infinite Wave length (average value)
2. **1<sup>st</sup> Wave Form:** Fundamental: Wave Length is the same as the Complex Wave Form
3. **The rest:**  $1/2, 1/3, \dots$

- A Fourier transform, or a Fourier decomposition transforms a Complex Wave Form
  - Any complex wave form can be decompose it into its separate sine waves.

For each wave form that we add we also need to figure out if they occur at a phase shift, but for simplicity we skipped that here.

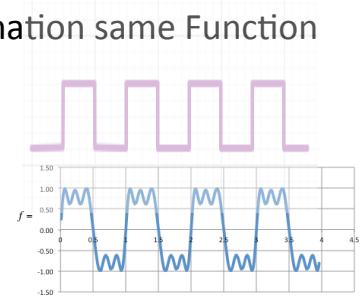
## Step or Square Function Approximation

- Transformation may not be exact it may depend on how many terms you use
  - “resolution”
- Example Overview
  - A step function may need infinite number to be correct.
  - Or impulse function



## In Excel Approximation same Function

- Approximated in Excel (actually only odd waveforms in the square function)
- How many terms to approximate? 7 or 9?
  - The higher the more precise

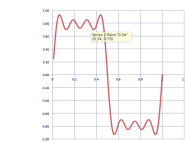
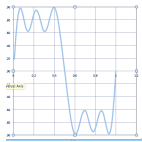


$$f(x) = \sin x + \frac{1}{3} \sin 3x + \frac{1}{5} \sin 5x + \frac{1}{7} \sin 7x$$

$$f(x) = \sin x + \frac{1}{3} \sin 3x + \frac{1}{5} \sin 5x + \frac{1}{7} \sin 7x + \frac{1}{9} \sin 9x$$

<http://mathworld.wolfram.com/FourierSeriesSquareWave.html>  
[CP-excel-sine.xlsx](#) (see schedule page exercise duplicate this in python / openCV, matplotlib).

## Excel Square Function Approximation



$$f(x) = \sin x + \frac{1}{3} \sin 3x + \frac{1}{5} \sin 5x + \frac{1}{7} \sin 7x$$

$$f(x) = \sin x + \frac{1}{3} \sin 3x + \frac{1}{5} \sin 5x + \frac{1}{7} \sin 7x + \frac{1}{9} \sin 9x$$

- Each additional sine wave that we like to add to would have two more oscillations within the period, so odd numbers each time, and we guess we may need to add an infinite number of these waves.

## Fourier Decomposition

$$f(x) = \frac{A_0}{2} + \sum_{m=1}^{\infty} A_m \cos\left(\frac{2\pi mx}{\lambda}\right) + \sum_{m=1}^{\infty} B_m \sin\left(\frac{2\pi mx}{\lambda}\right)$$

$$A_m = \frac{2}{\lambda} \int_0^{\lambda} f(x) \cos\left(\frac{2\pi mx}{\lambda}\right) dx$$

$$B_m = \frac{2}{\lambda} \int_0^{\lambda} f(x) \sin\left(\frac{2\pi mx}{\lambda}\right) dx$$

- Any periodic function  $f(x)$  can be decomposed into a series of sine (and also cosine) waves (we will focus in the sine term).
  - Question: What is the phase shifts and amplitude of those waves?
  - Note: The infinite wave is defined by the first term ( $A_0/2$ ).
- To get a feel of the arguments inside sine:
  - $m=1$ : ( $x: 0 \rightarrow \lambda$ ) where  $\lambda$  is the period of complex wave form.
    - Sin will go from  $0 \rightarrow 2\pi$  creating 1 oscillation across its “box”
  - $m=2$ : ( $x: 0 \rightarrow \lambda$ ) arguments of
    - sin goes from  $0 \rightarrow 4\pi$ , creating 2 oscillations within ‘box’
  - $m=3$  ( $x: 0 \rightarrow \lambda$ )
    - Sin’s arguments will go from  $0 \rightarrow 6\pi$  creating 3 oscillation across the box.

Hecht and Ganesan, Optics, 2008 Ch 7 pg 288 and Boas, Mathematical Methods, 2007

- Amplitude** determines how **dominate** the frequency is in the original wave form.
  - How much does the wave form contribute to the complex form.
  - So Amplitude determines the **weight** of the simpler form
- Phase Shift** (sliding the wave from forward or backwards by a phase shift  $\Phi_m$ )

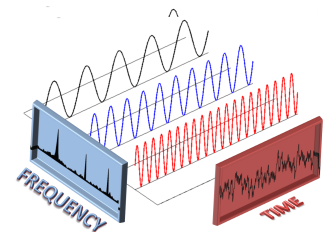
$$B_m \sin\left(\frac{2\pi mx}{\lambda} + \phi_m\right)$$

- Instead of **adding  $\Phi_m$  into the sine term** we can add another term cosine of same wave length as the sine term.

$$f(x) = \frac{A_0}{2} + \sum_{m=1}^{\infty} A_m \cos\left(\frac{2\pi mx}{\lambda}\right) + \sum_{m=1}^{\infty} B_m \sin\left(\frac{2\pi mx}{\lambda}\right)$$

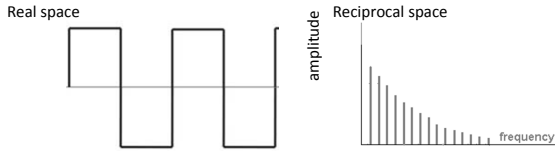
## Result

- We end up with sine waves of different frequencies ranging from:
  - From Course to Fine wave forms.



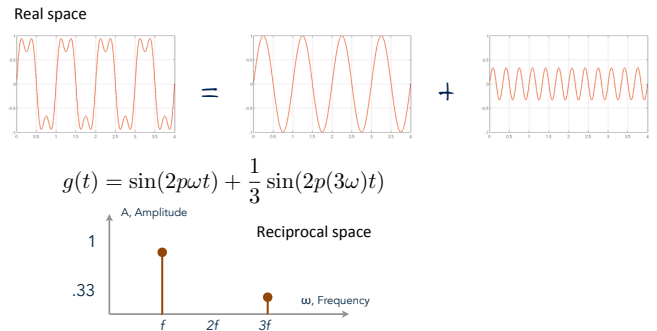
# Amplitude Measure

- We could plot the 'dominance' of each of the frequency of the waves, i.e., how much each wave form contributes,
  - Example: Square Wave:

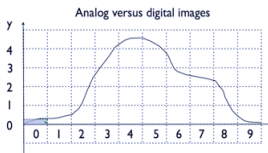


- Frequency Spectra, or spatial frequency.
- Not that higher frequency waves have less amplitude are less dominant.

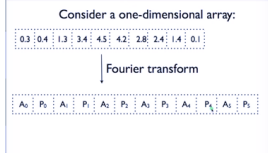
# Example 2.



# Fourier Transform of a Digital Image



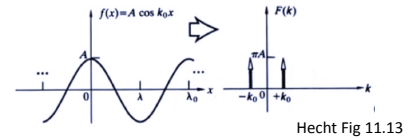
- Example: 10x10 image
- Fourier → Returns 5 Sine Waves + DC
- Last Term : Nyquist: Oscillates 5 times across box w/ 10 values (Up in one pixel, down in the other) so it is 1/2 of all pixel values across an image (x in this example).
- Highest frequency that is present in image.



- $A_0$  = amplitude of "DC" component
- $A_1$  = amplitude of "fundamental" frequency (one wavelength across box)
- $P_1$  = phase of "fundamental" frequency component
- $A_2$  = amplitude of first "harmonic" (two wavelengths across box)
- $P_2$  = phase of first harmonic
- $A_3$  = amplitude of second harmonic
- $P_3$  = phase of second harmonic
- etc.
- $A_5$  = amplitude of "Nyquist" frequency component

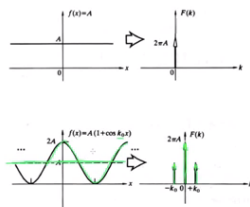
# Caveats

- Be Aware:
  - The y-axis is Spectra Space is
  - typically:
    - Amplitude Squared, Intensity or the Power, not just simple Amplitude.
- Direction (forward/backward) of sine waves (in an image is not detectable)
  - So we indicate both -1, and 1 frequencies, it is only one wave but we don't know which wave is present.



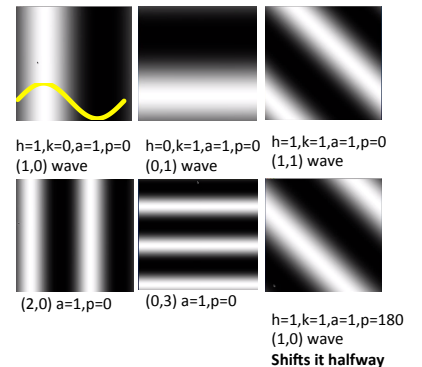
# Another Example: 1D Space

- Lifted Cos Function above X axis.
  - Indicates 2 components are present, the DC function that lifts the wave up or down.
  - And a cosine wave super imposed



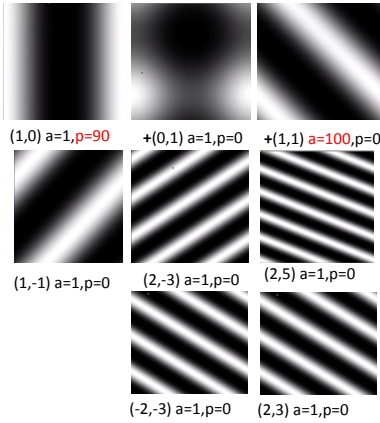
# 2D Space Examples of Sine Waves

- 800x800 sine wave
- Parameter (h, k)
  - Miller indices
  - h # oscillations along x, and
  - k # oscillations (along y)
  - Degrees
- Note the 0,0 coordinate is lower left so not a typical image.

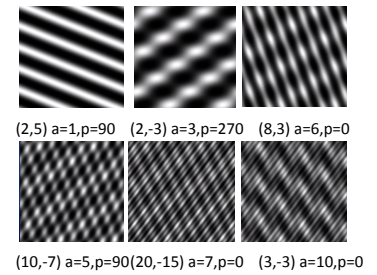


## More 2D Waves & A 1 Combination

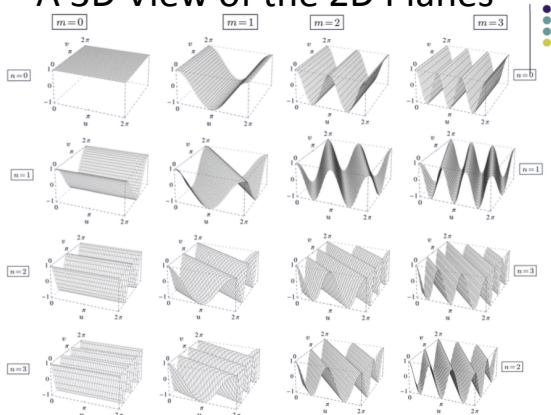
- Adding 0,1 do the 1,0 to the right interferes.
- -h or -k can change the direction the way crests are headed.
- If both are the same they look the same.



## Adding Many 2D Waves

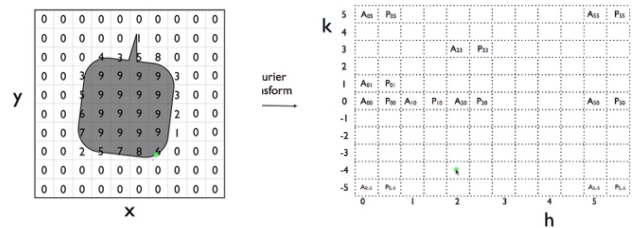


## A 3D View of the 2D Planes



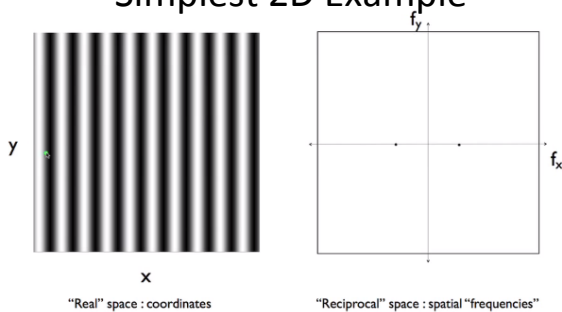
<http://web.cs.wpi.edu/~emmanuel/>

## 2D Image and Transformations



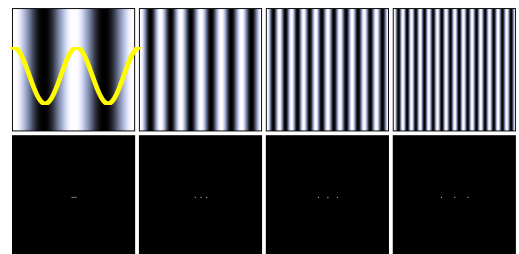
- Pixel intensities 0 → 9 with 10 pixels across
- Send Image to a 2D Fourier Transform Routine
  - Returns a matrix of **Amplitudes** and **Phase Shifts**

## Simplest 2D Example



- We only plot the amplitude
  - Across the spatial indexes along the x and y axis.
  - H-10, k=0 Wave
  - We plot intensity square (and their mirrors)

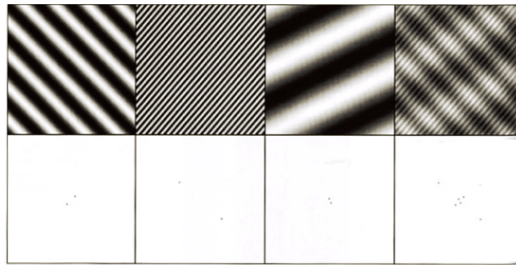
## 2D Fourier Transforms



- 2, 6, 10, 14 - Across X (says 1, 3, 5, 7) on page, but there are 2 cycles shown)
- Take Home: Message: Finer Grain Detail Dots are Further Apart, and
- Coarser Grain Closer together (contrast)

<http://cns-alumni.bu.edu/~slehar/fourier/fourier.html>

## More Examples



- And more:

[http://qsimaging.com/ccd\\_noise\\_interpret\\_ffts.html](http://qsimaging.com/ccd_noise_interpret_ffts.html)

## Practical: High Pass Filter

- High (Low) Pass Filter processing (e.g., finding details in your image)
  - **Fourier Transform** to the Frequency Domain
    - HPF Pass only the details (the high frequencies)
  - **Inverse Fourier transform** to observe just the details in the image

## Complex Exponential Form

- **Fourier series** :  $f(x) = \frac{A_0}{2} + \sum_{m=1}^{\infty} A_m \cos\left(\frac{2\pi mx}{\lambda}\right) + \sum_{m=1}^{\infty} B_m \sin\left(\frac{2\pi mx}{\lambda}\right)$

$$\omega = 2\pi / \lambda$$

$$f(t) = \frac{A_0}{2} + \sum_{m=1}^{\infty} A_m \cos(m\omega t) + B_m \sin(m\omega t)$$

- **Complex Form (variants)** easier to manipulate

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{in\omega t}$$

## Fourier Transforms and Inverse

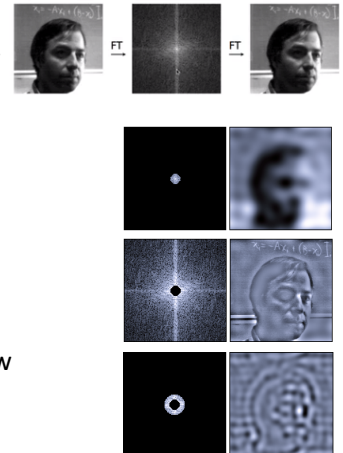
- A **Fourier Transform** decomposes any periodic complex function  $f(x)$  into a weighted sum of sines and cosines  $F(\omega)$ . For every  $\omega$  from 0 to  $\infty$ ,  $F(\omega)$  holds both the **amplitude** and the **phase** ( $\phi$ )

$$f(x) \longrightarrow \text{Fourier Transform} \longrightarrow F(\omega)$$

- and the inverse  $F(\omega) \rightarrow f(x)$ .

$$F(\omega) \longrightarrow \text{Inverse Fourier Transform} \longrightarrow f(x)$$

- **Resolution**
  - Low resolution near origin, low frequency
  - High resolution, high frequency
- **Low Pass Filter**: Only **include** (pass) pixels from middle of Fourier Transform
- **High Pass Filter** : Pass higher frequency waves
- **Band Pass Filter**: Pass frequency that are not low or high frequencies



<http://cns-alumni.bu.edu/~slehar/fourier/fourier.html>

## Convolution Theorem and the Fourier Transform

- **Fourier Transform** of a convolution (\*) of two functions:  $f$ , and  $g$ , is the product of their Fourier Transforms

$$\mathcal{F}\{f * g\} = \mathcal{F}\{f\} \cdot \mathcal{F}\{g\}$$

- **Inverse Fourier Transform** of the product of two Fourier transforms is the convolution of the two inverse Fourier transforms

$$\mathcal{F}^{-1}\{f \cdot g\} = \mathcal{F}^{-1}\{f\} * \mathcal{F}^{-1}\{g\}$$

**Convolution** in the spatial domain is equivalent to **multiplication** in the frequency domain

<http://mathworld.wolfram.com/FourierSeries.html>  
[https://en.wikipedia.org/wiki/Fourier\\_series](https://en.wikipedia.org/wiki/Fourier_series)

[https://en.wikipedia.org/wiki/Convolution\\_theorem](https://en.wikipedia.org/wiki/Convolution_theorem)

## Other Images (and insights)

- <http://cns-alumni.bu.edu/~slehar/fourier/fourier.html>
  - See above for low pass and high pass filters, and do the next exercise at home
- <http://www.cs.unm.edu/~brayer/vision/fourier.html>
  - See discussion of edges and the effect on the frequency spectra

## Exercise/Homework

- Exercise at home:
  - [http://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_imgproc/py\\_transforms/py\\_fourier\\_transform/py\\_fourier\\_transform.html](http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_transforms/py_fourier_transform/py_fourier_transform.html)
  - Work on this tutorial at home
- Read:
  - <http://cns-alumni.bu.edu/~slehar/fourier/fourier.html>
  - <https://www.cs.unm.edu/~brayer/vision/fourier.html>
  - <http://mathworld.wolfram.com/FourierSeries.html>

## Contributions

- Dr. **Grant Jensen**, Caltech, Pasadena, CA
  - <http://jensenlab.caltech.edu>
- Dr. **Mervin Roy**, University of Leicester, UK
- **Hecht** and **Ganesan**, Optics, 2008, Ch 7 & 11
- **Boas**, Mathematical Methods in Physical Sciences, 2007, Ch 8.
- <http://cns-alumni.bu.edu/~slehar/fourier/fourier.html>
- <https://www.cs.unm.edu/~brayer/vision/fourier.html>