# Fourier Analysis and Image Processing


Earl F. Glynn

Scientific Programmer

Bioinformatics

Stowers Institute for Medical Research


14 Feb 2007

# Fourier Analysis and Image Processing

- History
- Periodic Signals
- Fourier Analysis
  - Fourier Series
  - Fourier Transform
  - Discrete Fourier Transform (DFT)
  - Fast Fourier Transform (FFT)
- 2D FFT and Image Processing
  - Spatial Frequency in Images
  - 2D Discrete Fourier Transform
  - 2D FFT Examples
  - Applications of FFTs in Image Processing
- Summary

# History

http://de.wikipedia.org/wiki/Kreiszahl

Jean Baptiste Joseph Fourier
1768-1830

French Mathematician and Physicist

Outlined technique in memoir, *On the Propagation of Heat in Solid Bodies*, which was read to Paris Institute on 21 Dec 1807. Controversial then: Laplace and Lagrange objected to what is now Fourier series: *"... his analysis ... leaves something to be desired on the score of generality and even rigour..." (from report awarding Fourier math prize in 1811)*

In *La Theorie Analytique de la Chaleur (Analytic Theory of Heat)* (1822) Fourier
- developed the theory of the series known by his name, and
- applied it to the solution of boundary-value problems in partial differential equations.

Sources:
www.me.utexas.edu/~me339/Bios/fourier.html and www-gap.dcs.st-and.ac.uk/~history/Biographies/Fourier.html

# Periodic Signals

A continuous-time signal x(t) is ***periodic*** if:

$$x(t + T) = x(t)$$

***Fundamental period***, $T_0$, of x(t) is smallest T satisfying above equation.

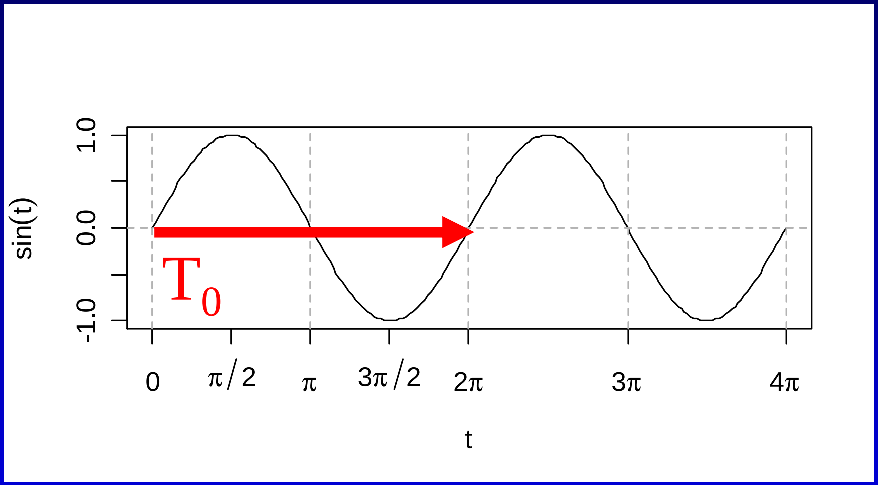
***Fundamental frequency***: $f_0 = 1/T_0$

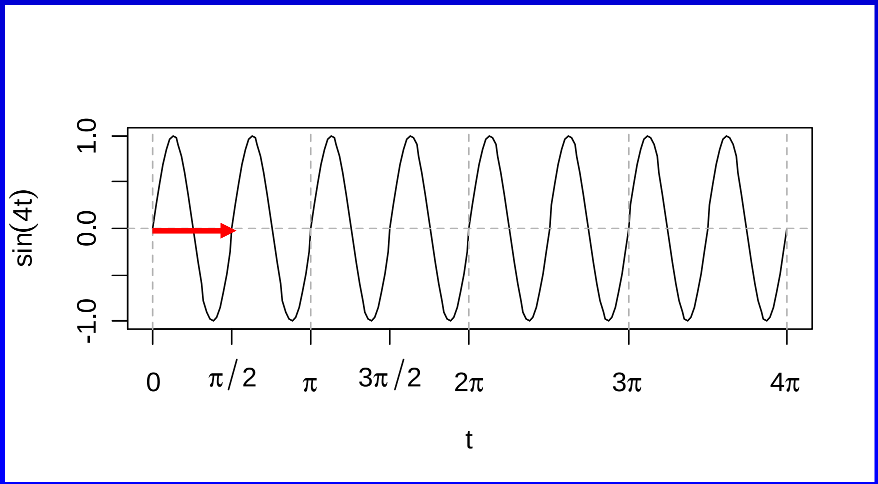
***Fundamental angular frequency***: $\omega_0 = 2\pi/T_0 = 2\pi f_0$

# Periodic Signals

$$x(t + T) = x(t)$$

***Fundamental frequency***:
$f_0 = 1/T_0$

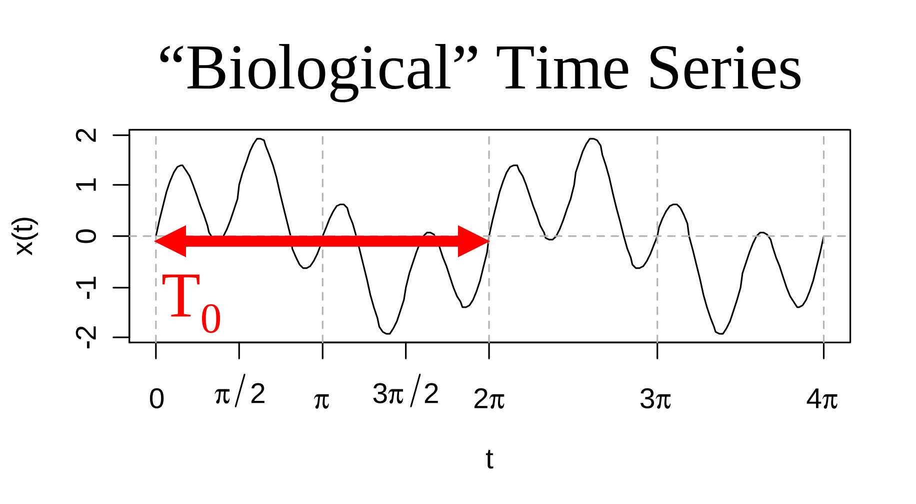



*Harmonics: Integer multiples of frequency of wave*

# Periodic Signals

$$x(t + T) = x(t)$$

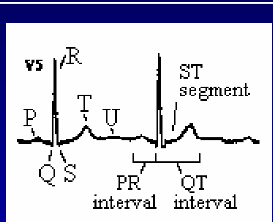





*Biological time series can be quite complex, and will contain noise.*

# Periodic Signals

## Periodicity in Biology and Medicine

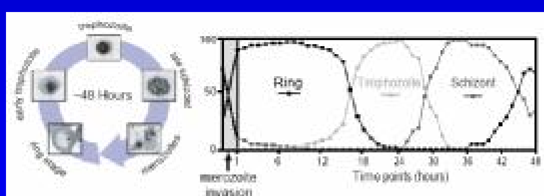



Electrocardiogram (ECG): Measure of the dipole moment caused by depolarization and repolarization of heart muscle cells.

From http://www.ecglibrary.com/norm.html

Somitogenesis: A vertebrate's body plan: a segmented pattern. Segmentation is established during somitogenesis, which is studied by Pourquie Lab.

Photograph taken at Reptile Gardens, Rapid City, SD, June 2003,www.reptile-gardens.com




Intraerythrocytic Developmental Cycle of *Plasmodium falciparum*

From Bozdech, et al, Fig. 1A, *PLoS Biology*, Vol 1, No 1, Oct 2003, p 3.

X-Ray Computerized Tomography. Tomogram ("slice") produced by 2D FFT of digitally filtered x-ray data.

From www.csun.edu/~jwadams/Image_Processing.pdf#search=%22fft%20medical%20 image%20processing%22

# Fourier Analysis

- Fourier Series
  Expansion of continuous function into weighted sum of sines and cosines, or weighted sum of complex exponentials.
- Fourier Transform
  Maps one function to another: continuous-to-continuous mapping. An integral transform.
- Discrete Fourier Transform (DFT)
  Approximation to Fourier integral.  Maps discrete vector to another discrete vector.  Can be viewed as a matrix operator.
- Fast Fourier Transform (FFT)
  Special computational algorithm for DFT.

# Fourier Series

## Trigonometric Fourier Series

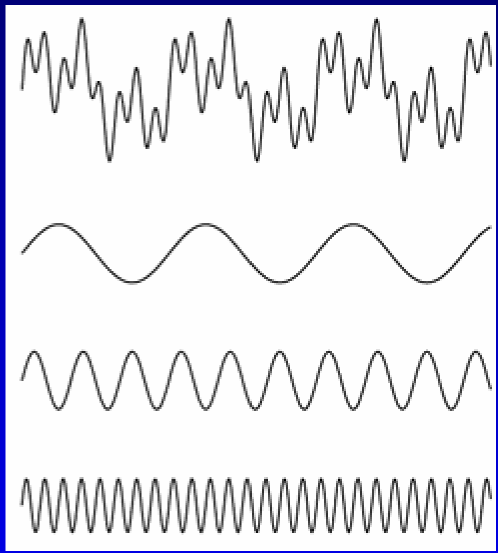
Expansion of continuous function into weighted sum of sines and cosines.

$$x(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty}\left[a_k \cdot \cos(kw_0 t) + b_k \cdot \sin(kw_0 t)\right]$$

$$a_k = \frac{2}{T_0}\int_{T_0} x(t) \cdot \cos(kw_0 t)\, dt$$

$$b_k = \frac{2}{T_0}\int_{T_0} x(t) \cdot \sin(kw_0 t)\, dt$$

$$w_0 = \frac{2p}{T_0} = 2p\ f_0$$



www.science.org.au/nova/029/029img/wave1.gif

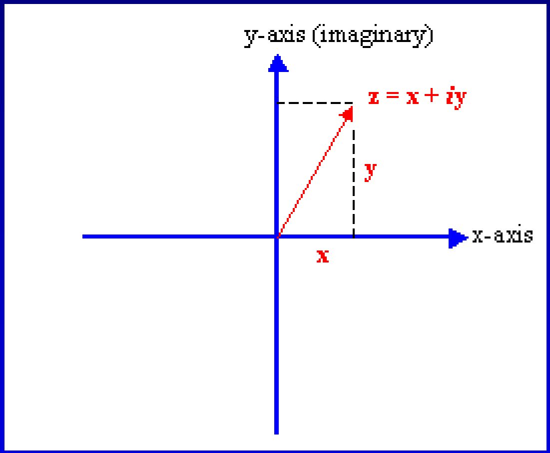
If x(t) is even, i.e., x(-t) = x(t) like cosine, then $b_k = 0$.
If x(t) is odd, i.e., x(-t) = -x(t) like sine, then $a_k = 0$.

Source: *Schaum's Theory and Problems: Signals and Systems*, Hwei P. Hsu, 1995, pp. 211-213

# Complex Math Review

Solutions to $x^2 = -1$: $x = \sqrt{-1} = \pm i$

Complex Plane




$r = |z| = \sqrt{x^2 + y^2}$

$\theta = \arctan(y/x)$

$x = r \cdot \cos\theta$

$y = r \cdot \sin\theta$

$abs(z) = |z| = \sqrt{x^2 + y^2}$

$u = a + ib = r_1 \cdot e^{i\theta_1} = r_1[\cos\theta_1 + i\sin\theta_1] = r_1 \cdot cis\,\theta_1$

$u^* = a - ib$

Operators: +, -, *, /

$u + v = (a + ib) + (c + id) = (a + c) + i(b + d)$

$u \times v = (a + ib)(c + id) = (ac - bd) + i(ad + bc)$

Euler's Formula: $e^{i\theta} = \cos\theta + i\sin\theta$

DeMoivre's Theorem:

$z = x + iy = r \cdot e^{i\theta} = r[\cos\theta + i\sin\theta] = r \cdot cis\,\theta$

$z^n = \left(r \cdot e^{i\theta}\right)^n = r^n \cdot e^{in\theta} = r^n(\cos n\theta + i \cdot \sin n\theta)$

# Fourier Series

## Complex Exponential Fourier Series

Expansion of continuous function into weighted sum of complex exponentials.

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{i \cdot k \cdot W_0 \cdot t}$$

$$c_k = \frac{1}{T_0} \int_{T_0} x(t) \cdot e^{-i \cdot k \cdot W_0 \cdot t} \, dt$$

$$w_0 = \frac{2p}{T_0} = 2p \; f_0$$

Notes:

- If x(t) is real, $c_{-k} = c_k^*$.
- For k = 0, $c_k$ = average value of x(t) over one period.
- $a_0/2 = c_0$; $a_k = c_k + c_{-k}$; $b_k = i \cdot (c_k - c_{-k})$

Source *Schaum's Theory and Problems: Signals and Systems*, Hwei P. Hsu, 1995, pp. 211-213

# Fourier Series

## Complex Exponential Fourier Series

$$c_k = \frac{1}{T_0} \int_{T_0} x(t) \cdot e^{-i \cdot k \cdot \omega_0 \cdot t} \, dt$$

Coefficients can be written as product:

$$c_k = |c_k| \cdot e^{i f_k}$$

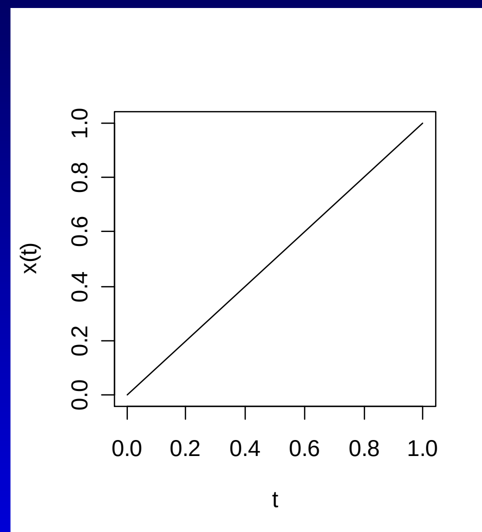
- $c_k$ are known as the spectral coefficients of x(t)
- Plot of $|c_k|$ versus angular frequency $\omega$ is the amplitude spectrum.
- Plot of $\varphi_k$ versus angular frequency is the phase spectrum.
- With discrete Fourier frequencies, $k \cdot \omega_0$, both are discrete spectra.

Source: *Schaum's Theory and Problems: Signals and Systems*, Hwei P. Hsu, 1995, pp. 211-213

$$x(t) = 2\left(\sin t - \frac{\sin 2t}{2} + \frac{\sin 3t}{3} - \dots\right)$$

# Fourier Series

Given: x(t) = t




Fourier Series:

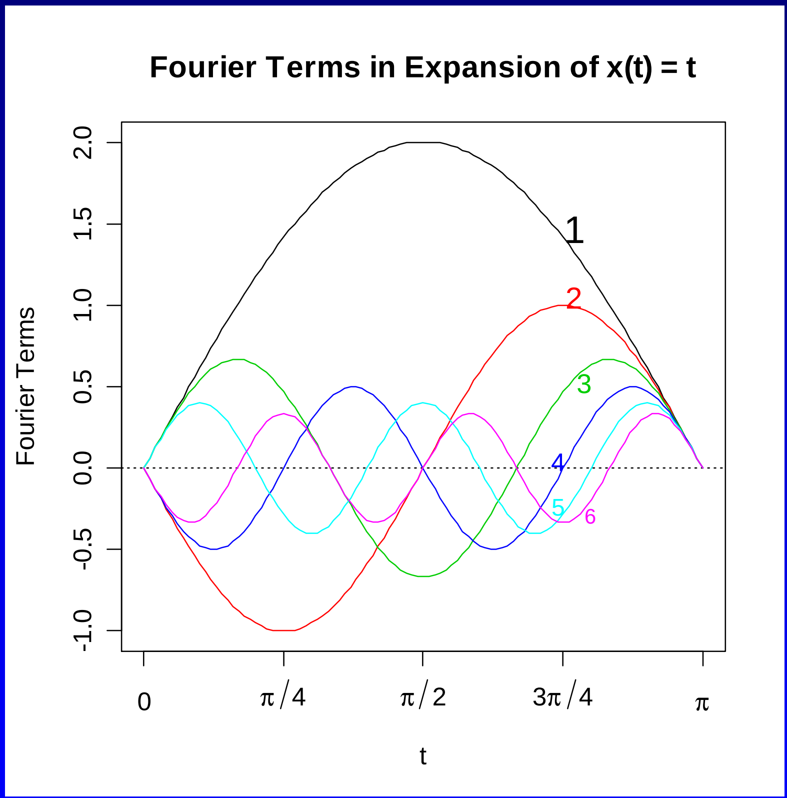
$$x(t) = 2\left(\sin t - \frac{\sin 2t}{2} + \frac{\sin 3t}{3} - \dots\right)$$

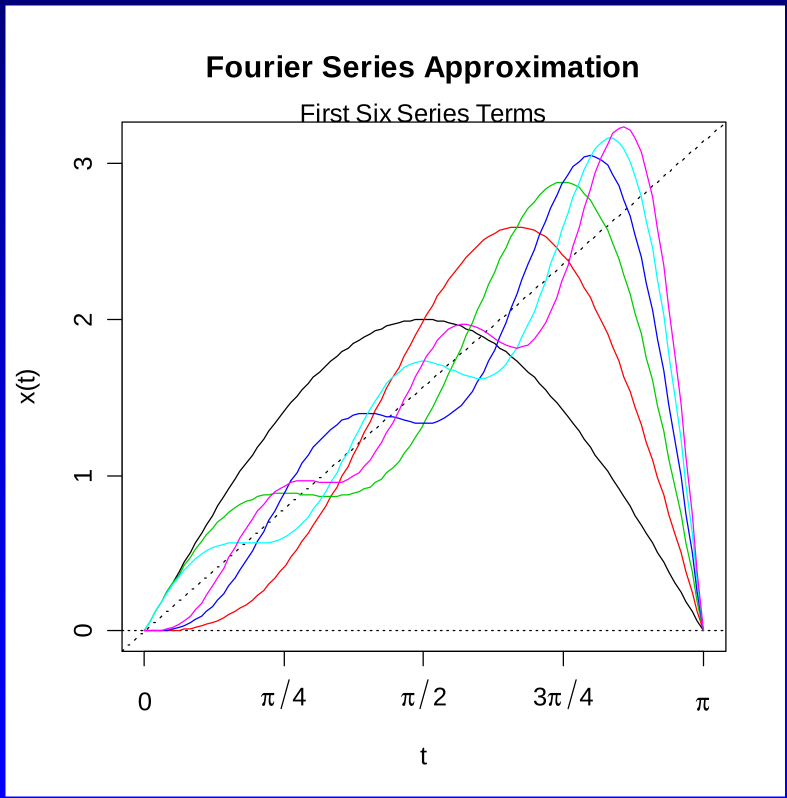
*Approximate any function as truncated Fourier series*

# Fourier Series

Given: x(t) = t

Fourier Series: $x(t) = 2\left(\sin t - \frac{\sin 2t}{2} + \frac{\sin 3t}{3} - \ldots\right)$



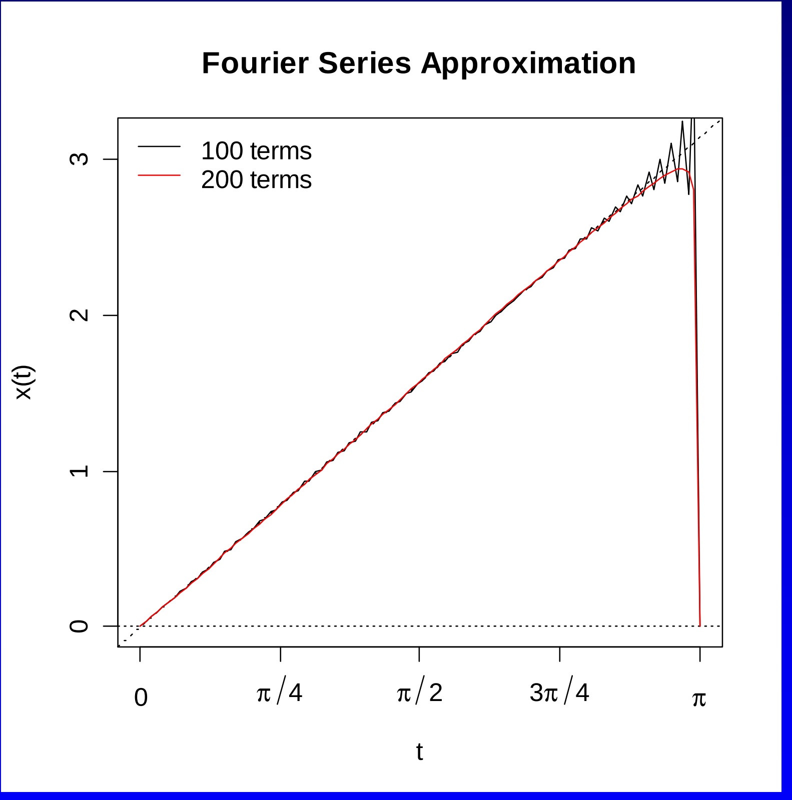

*Approximate any function as truncated Fourier series*

$x(t)=2\left(\sin t-\frac{\sin 2t}{2}+\frac{\sin 3t}{3}-...\right)$

# Fourier Series

Given: x(t) = t

Fourier Series: $$x(t)=2\left(\sin t-\frac{\sin 2t}{2}+\frac{\sin 3t}{3}-...\right)$$

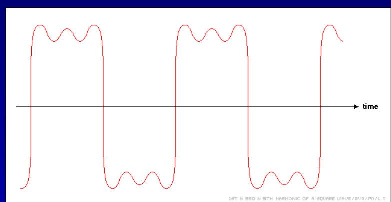



*Approximate any function as truncated Fourier series*

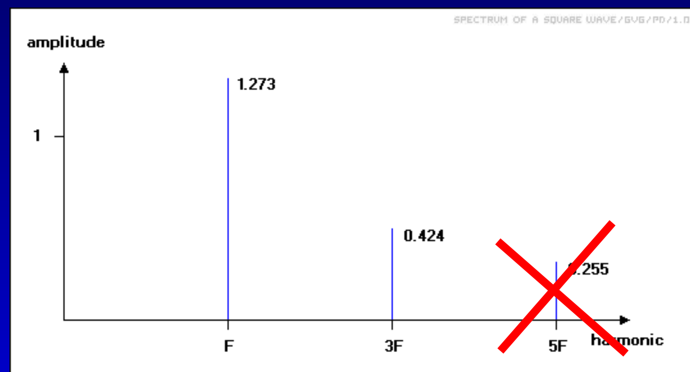
# Fourier Series

## Noise Removal

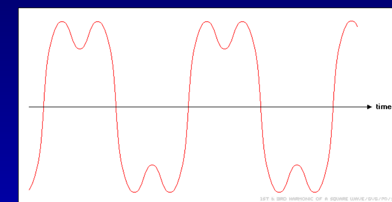
Time Series

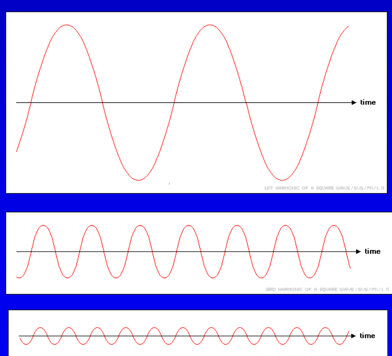


Periodogram



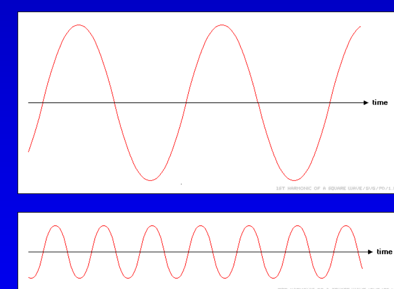

Filtered Series





Decomposition

Remove
High Frequency
Component



0

Reconstruction

*"Remove" high frequency noise by zeroing a term in series expansion*

# Fourier Transform

Maps one function to another: continuous-to-continuous mapping.

**Fourier transform of x(t) is X(ω):**
(converts from time space to frequency space)

$$X(w) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{\infty} x(t) \cdot e^{-iwt}\, dt$$

**Fourier inverse transform of X(ω) recovers x(t):**
(converts from frequency space to time space)

$$x(t) = \mathcal{F}^{-1}\{X(w)\} = \frac{1}{2p} \int_{-\infty}^{\infty} X(w) \cdot e^{iwt}\, dw$$

**x(t) and X(ω) form a Fourier transform pair: x(t) « X(ω)**

*The Fourier Transform is a special case of the Laplace Transform, s = i ·ω*



Source: *Schaum's Theory and Problems: Signals and Systems*, Hwei P. Hsu, 1995, pp. 214-218

# Fourier Transform

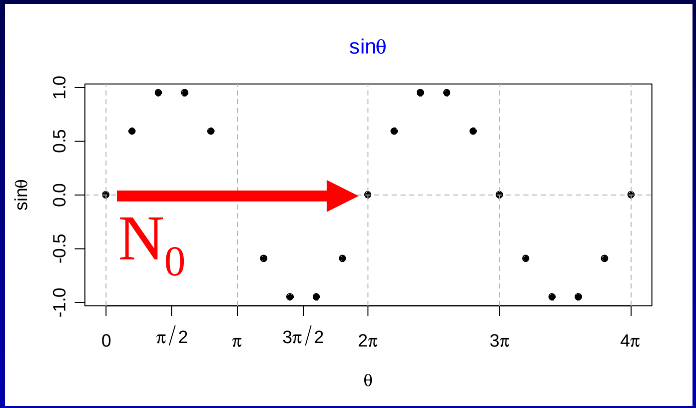
## Properties of the Fourier Transform

| | Signal | Fourier transform unitary, angular frequency | Fourier transform unitary, ordinary frequency | Remarks |
|---|---|---|---|---|
| | $g(t) \equiv \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} G(\omega) e^{i\omega t} d\omega$ | $G(\omega) \equiv \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(t) e^{-i\omega t} dt$ | $G(f) \equiv \int_{-\infty}^{\infty} g(t) e^{-i2\pi f t} dt$ | |
| 1 | $a \cdot g(t) + b \cdot h(t)$ | $a \cdot G(\omega) + b \cdot H(\omega)$ | $a \cdot G(f) + b \cdot H(f)$ | Linearity |
| 2 | $g(t-a)$ | $e^{-ia\omega} G(\omega)$ | $e^{-i2\pi af} G(f)$ | Shift in time domain |
| 3 | $e^{iat} g(t)$ | $G(\omega - a)$ | $G\left(f - \frac{a}{2\pi}\right)$ | Shift in frequency domain, dual of 2 |
| 4 | $g(at)$ | $\frac{1}{\|a\|} G\left(\frac{\omega}{a}\right)$ | $\frac{1}{\|a\|} G\left(\frac{f}{a}\right)$ | If $\|a\|$ is large, then $g(at)$ is concentrated around 0 and $\frac{1}{\|a\|} G\left(\frac{\omega}{a}\right)$ spreads out and flattens |
| 5 | $G(t)$ | $g(-\omega)$ | $g(-f)$ | Duality property of the Fourier transform. Results from swapping "dummy" variables of $t$ and $\omega$. |
| 6 | $\frac{d^n g(t)}{dt^n}$ | $(i\omega)^n G(\omega)$ | $(i2\pi f)^n G(f)$ | Generalized derivative property of the Fourier transform |
| 7 | $t^n g(t)$ | $i^n \frac{d^n G(\omega)}{d\omega^n}$ | $\left(\frac{i}{2\pi}\right)^n \frac{d^n G(f)}{df^n}$ | This is the dual to 6 |
| 8 | $(g * h)(t)$ | $\sqrt{2\pi} G(\omega) H(\omega)$ | $G(f) H(f)$ | $g * h$ denotes the convolution of $g$ and $h$ — this rule is the convolution theorem |
| 9 | $g(t)h(t)$ | $\frac{(G * H)(\omega)}{\sqrt{2\pi}}$ | $(G * H)(f)$ | This is the dual of 8 |

From http://en.wikipedia.org/wiki/Continuous_Fourier_transform

Also see Schaum's Theory and Problems: Signals and Systems, Hwei P. Hsu, 1995, pp. 219-223

# Discrete Time Signal




A discrete-time signal x[n] is ***periodic*** if:

$$x[n + N] = x[n]$$

***Fundamental period***, $N_0$, of x[n] is smallest integer N satisfying above equation.

***Fundamental angular frequency***: $\Omega_0 = 2\pi/N_0$

# Discrete Fourier Transform (DFT)

Given discrete time sequence, x[n], n = 0, 1, …, N-1

**Discrete Fourier Transform (DFT)**

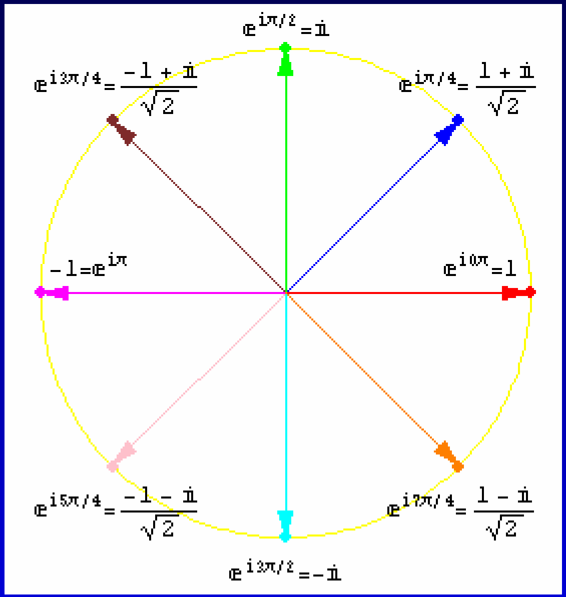
$$X[k] = \mathrm{DFT}\{x[n]\} = \sum_{n=0}^{N-1} x[n] \cdot e^{-i(2pkn/N)}$$

k = 0, 1, …, N-1

$N^{th}$ root of unity

**Inverse Discrete Fourier Transform (IDFT)**

$$x[n] = \mathrm{IDFT}\{X[k]\} = \frac{1}{N} \sum_{n=0}^{N-1} X[k] \cdot e^{i(2pkn/N)}$$




**The Eight Eighth Roots of Unity**
http://math.fullerton.edu/mathews/c2003/ComplexAlgebraRevisitedMod.html

- One-to-one correspondence between x[n] and X[k]
- DFT closely related to discrete Fourier series and the Fourier Transform
- DFT is ideal for computer manipulation
- Share many of the same properties as Fourier Transform
- Multiplier (1/N) can be used in DFT or IDFT. Sometimes 1/SQRT(N) used in both.

Sources: *Schaum's Theory and Problems: Signals and Systems*, Hwei P. Hsu, 1995, pp. 305
and http://en.wikipedia.org/wiki/Fast_Fourier_transform

# Discrete Fourier Transform (DFT)

$$X[k] = \text{DFT}\{x[n]\} = \sum_{n=0}^{N-1} x[n] \cdot e^{-i(2pkn/N)}$$

k = 0, 1, …, N-1

For N = 4, the DFT becomes:

k

$$\text{k} \downarrow \begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} e^{-0 \cdot ip/2} & e^{-0 \cdot ip/2} & e^{-0 \cdot ip/2} & e^{-0 \cdot ip/2} \\ e^{-0 \cdot ip/2} & e^{-1 \cdot ip/2} & e^{-2 \cdot ip/2} & e^{-3 \cdot ip/2} \\ e^{-0 \cdot ip/2} & e^{-2 \cdot ip/2} & e^{-4 \cdot ip/2} & e^{-6 \cdot ip/2} \\ e^{-0 \cdot ip/2} & e^{-3 \cdot ip/2} & e^{-6 \cdot ip/2} & e^{-9ip/2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \downarrow \text{n}$$

$$\begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

# Discrete Fourier Transform (DFT)

For N = 4, the DFT is:

$$\begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

x = [1, 0, 1, 0] X = [2, 0, 2, 0]

x = [0, 3, 0, 3] X = [6, 0, -6, 0]

x = [1, 1, 1, 1] X = [4, 0, 0, 0]

x = [0, 0, 0, 0] X = [0, 0, 0, 0]

x = [0, 0, 1, 1] X = [2, -1+i, 0, -1-i]

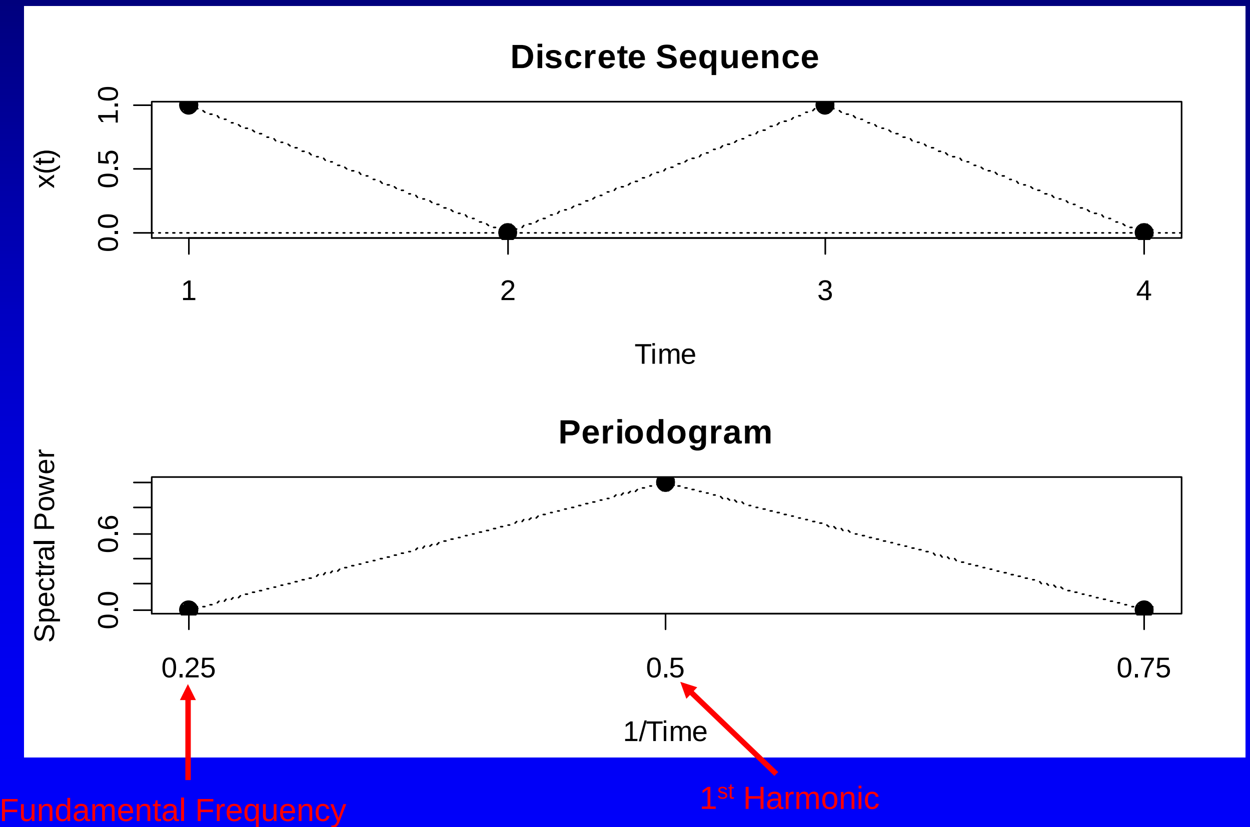
x = [1, 1, 0, 0] X = [2, 1- i, 0, 1+i]

*X[0]/N = mean*

# Discrete Fourier Transform (DFT)

x = [1, 0, 1, 0]     DFT(x) = [2, 0,  2, 0]

x = [0, 1, 0, 1]     DFT(x) = [2, 0, -2, 0]



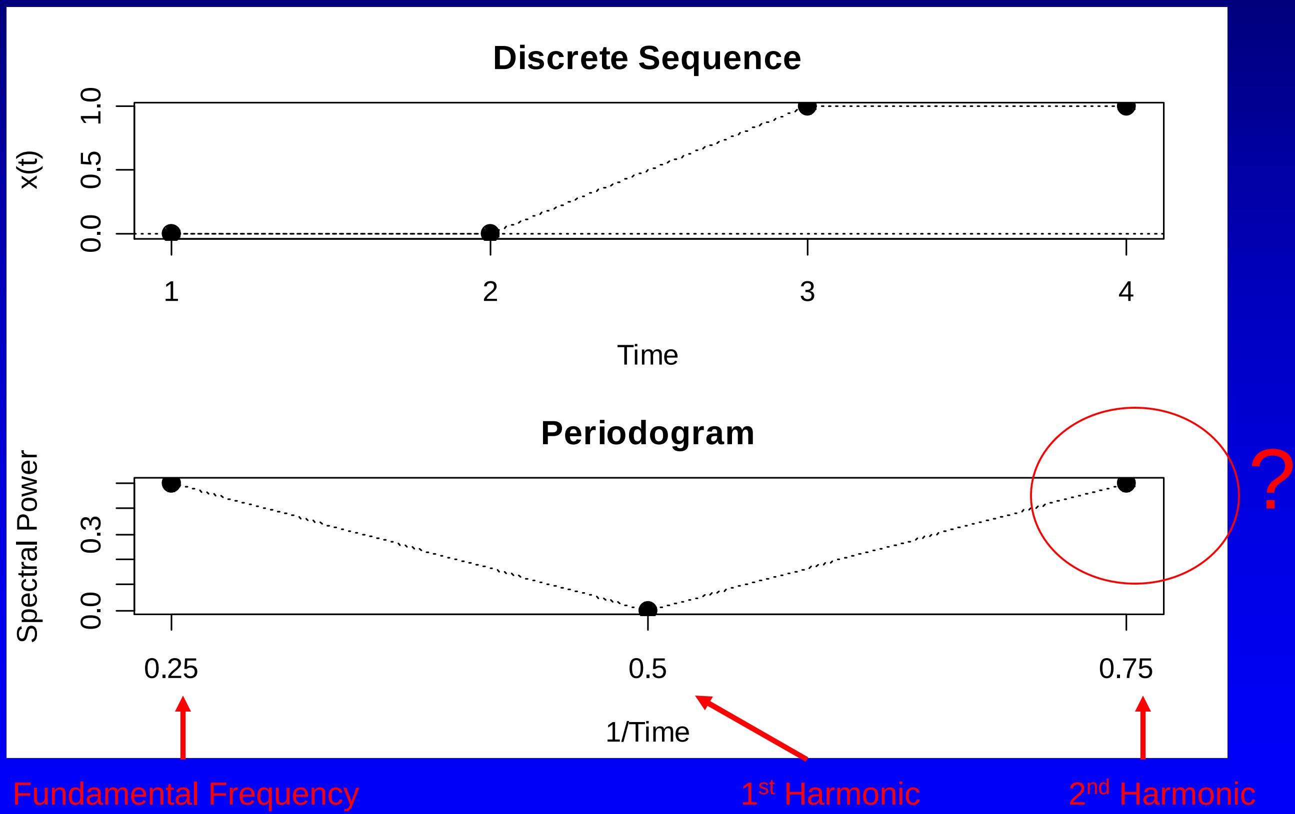

*Periodogram = |DFT(x)|² / N*

*(excluding first term, which is the mean)*

# Discrete Fourier Transform (DFT)

x = [0, 0, 1, 1]     X = [2, -1+i, 0, -1-i]

x = [1, 1, 0, 0]     X = [2,  1- i, 0,  1+i]

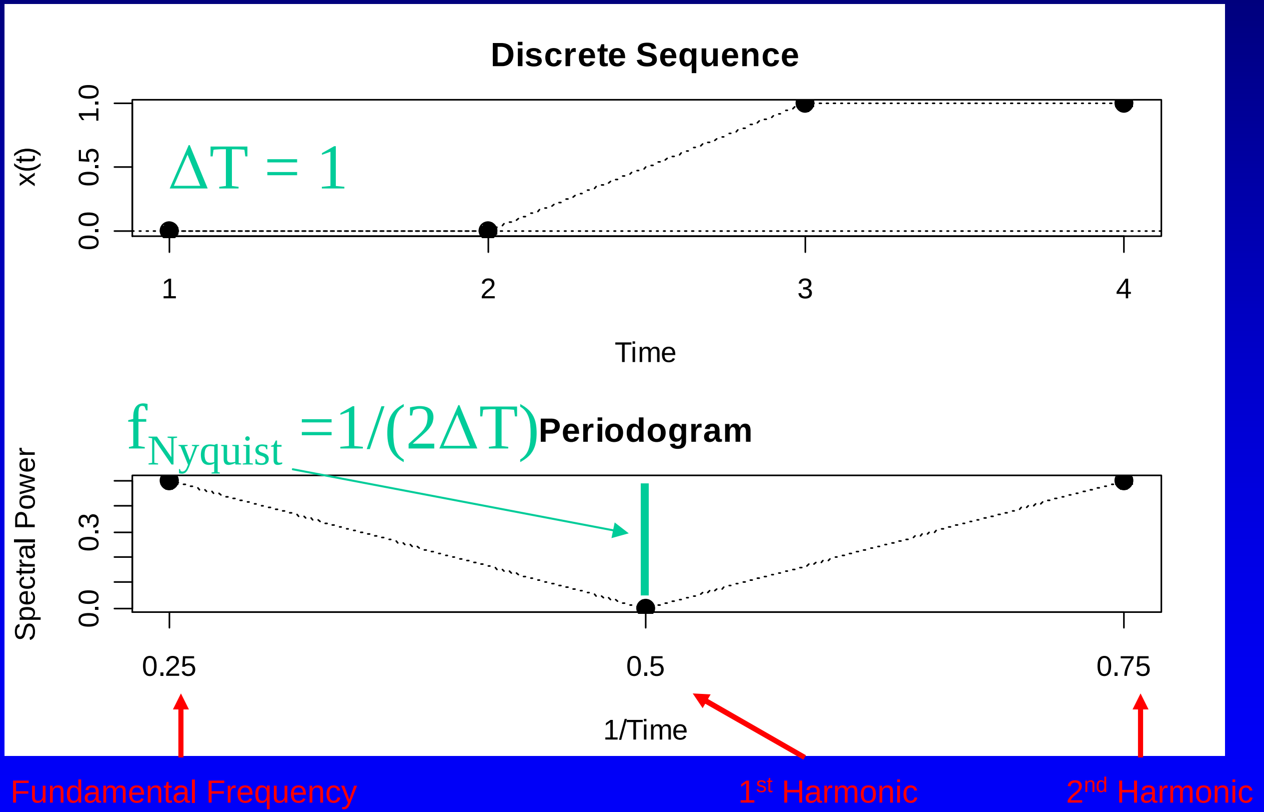



*Why so much spectral "power" in 2nd Harmonic?*

# Discrete Fourier Transform (DFT)

x = [0, 0, 1, 1]  X = [2, -1+i, 0, -1-i]

x = [1, 1, 0, 0]  X = [2, 1- i, 0, 1+i]

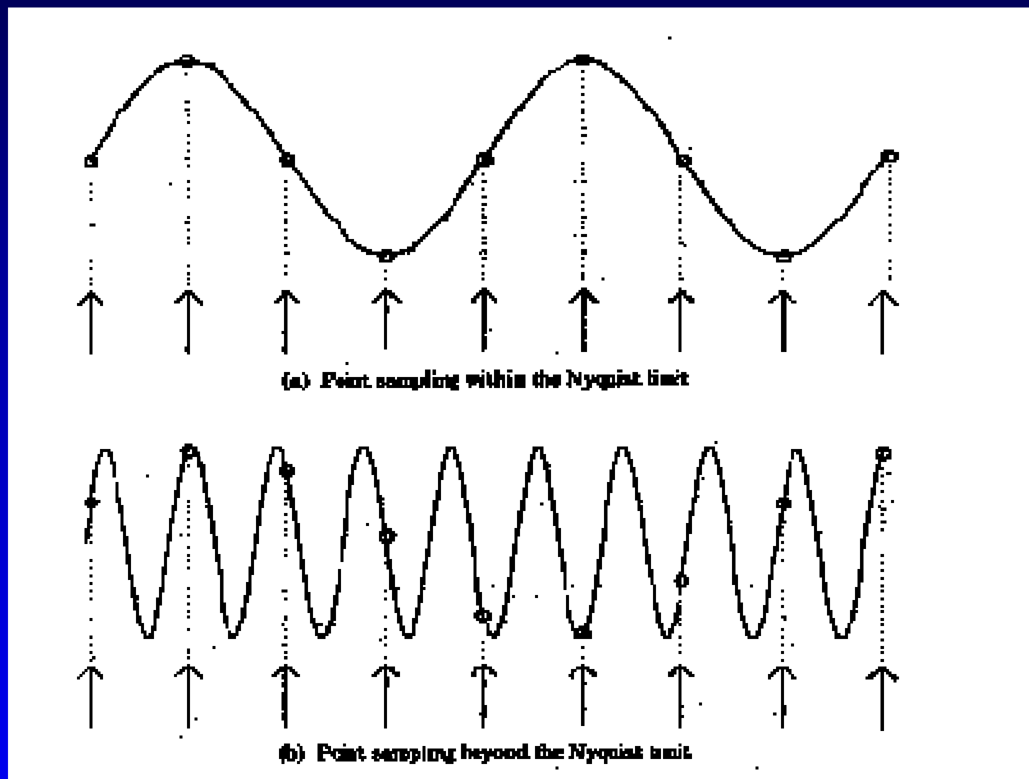



*Nyquist frequency is a consequence of Shannon Sampling Theorem*



Also see: http://en.wikipedia.org/wiki/Nyquist-Shannon_sampling_theorem

# Sampling and Aliasing



(a) Point sampling within the Nyquist limit

(b) Point sampling beyond the Nyquist limit

The top signal is sampled at the Nyquist limit and is not aliased.
The bottom signal is sampled beyond the Nyquist limit and is aliased.
*Aliasing occurs when higher frequencies are folded into lower frequencies.*



From: http://www.siggraph.org/education/materials/HyperGraph/aliasing/alias3.htm

# Fast Fourier Transform (FFT)

**Discrete Fourier Transform (DFT)**

$$X[k] = \mathrm{DFT}\{x[n]\} = \sum_{n=0}^{N-1} x[n] \cdot e^{-i(2pkn/N)}$$

k = 0, 1, …, N-1

- The FFT is a computationally efficient algorithm to compute the Discrete Fourier Transform and its inverse.
- Evaluating the sum above directly would take $O(N^2)$ arithmetic operations.
- The FFT algorithm reduces the computational burden to $O(N \log N)$ arithmetic operations.
- FFT requires the number of data points to be a power of 2 (usually 0 padding is used to make this true)
- FFT requires evenly-spaced time series

Source: http://en.wikipedia.org/wiki/Fast_Fourier_transform

# Fast Fourier Transform (FFT)

## What's the "Trick" to the Speedup?

**Discrete Fourier Transform (DFT)**

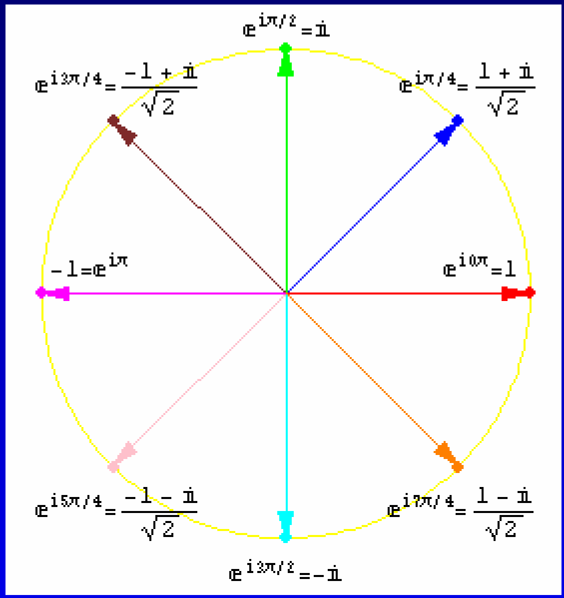
$$X[k] = \text{DFT}\{x[n]\} = \sum_{n=0}^{N-1} x[n] \cdot e^{-i(2pkn/N)}$$

k = 0, 1, …, N-1

Use "Divide & Conquer" by splitting polynomial evaluation into "even" and "odd" parts, recursively:

$p(x) = p_0x^0 + p_1x^1$
Split: $p(x) = p_{even} + p_{odd}$

$p(x) = p_0x^0 + x \cdot p_1x^0$




**The Eight Eighth Roots of Unity**
http://math.fullerton.edu/mathews/c2003/ComplexAlgebraRevisitedMod.html

# Fast Fourier Transform (FFT)

## Software

www.fftw.org

FFTW is a C subroutine library for computing the discrete Fourier transform (DFT) in one or more dimensions, of arbitrary input size

IDL (see Signal Processing Demo for Fourier Filtering)

```
IDL> print, fft( [0,1,0,1] )
(     0.500000,      0.000000)(      0.000000,      0.000000)(     -0.500000,      0.000000)(      0.000000,      0.000000)
```

MatLab: *Signal Processing/Image Processing Toolboxes*

```
>> fft( [0,1,0,1] )
ans =
     2     0    -2     0
```

Mathematica: *Perform symbolic or numerical Fourier analysis*

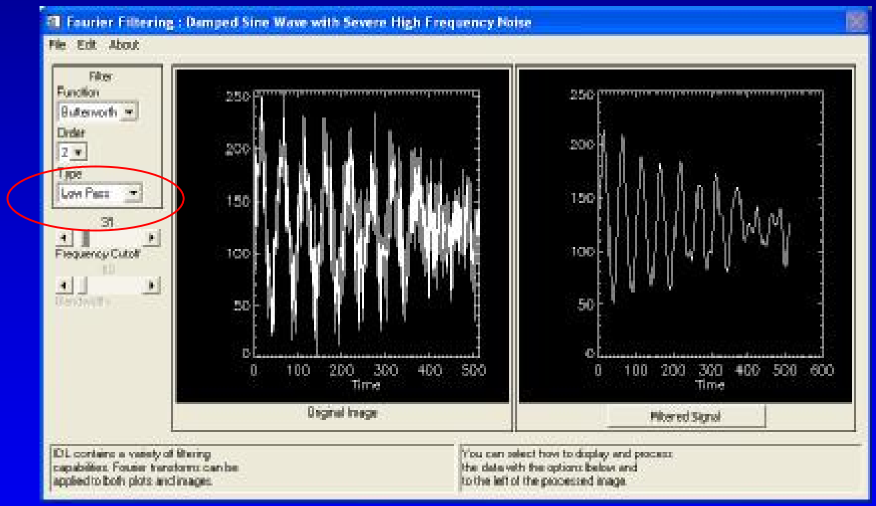
```
In[3]:= Fourier[{0, 1, 0, 1}]

Out[3]= {1. + 0. i, 0. + 0. i, -1. + 0. i, 0. + 0. i}
```

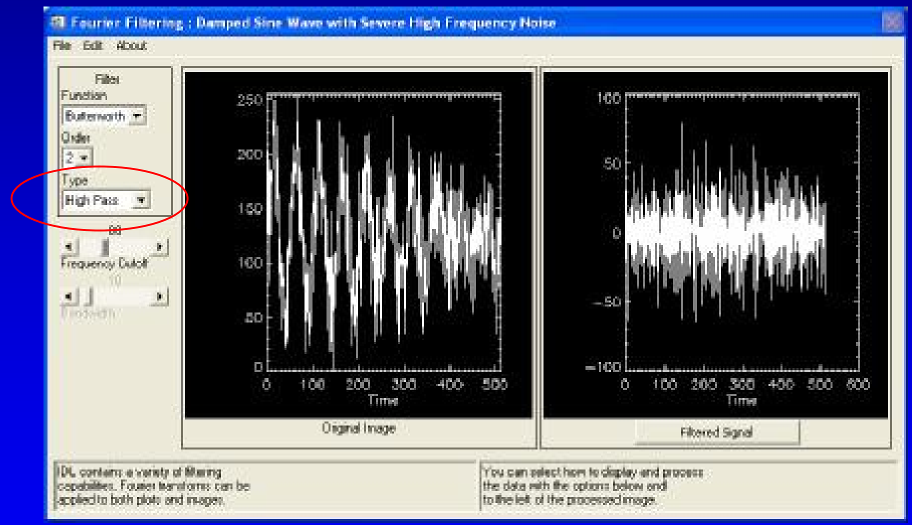
R

```
> fft( c(0,1,0,1) )
[1]  2+0i  0+0i -2+0i  0+0i
```

# Fast Fourier Transform (FFT)

## 1D FFT in IDL Software

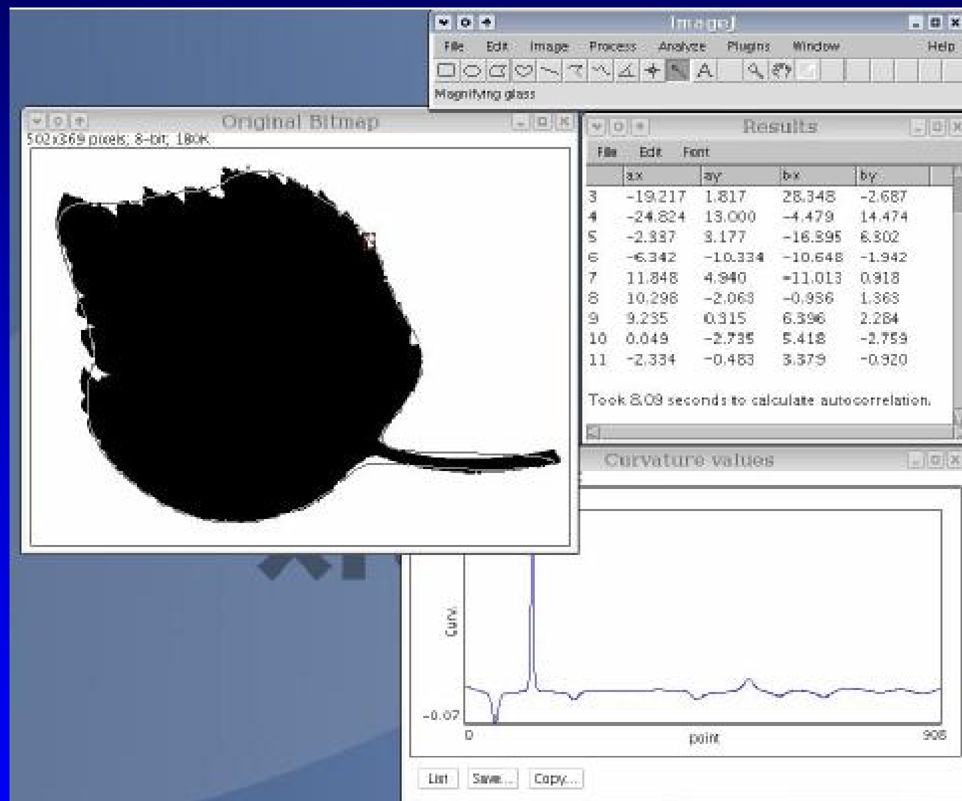



*IDL Run Demo, Data Analysis, Signal Processing, Filtering Demo*

# Fast Fourier Transform (FFT)

## 1D FFT in ImageJ:  Fourier Shape Analysis




Source:  http://imagejdocu.tudor.lu/Members/tboudier/plonearticle.2006-07-12.6904098144/2006-07-14.2969642786/image

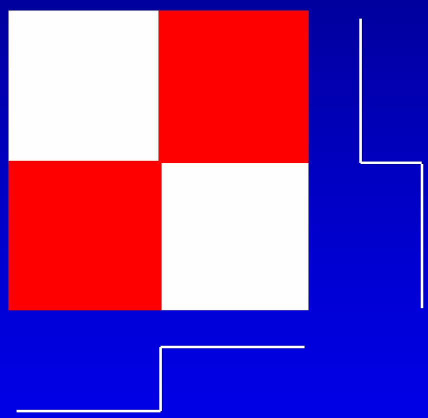
*This is an application of Fourier analysis NOT involving a time series …*

# 2D FFT and Image Processing

- Spatial Frequency in Images
- 2D Discrete Fourier Transform
- 2D FFT Examples
- Applications of FFT
  - Noise Removal
  - Pattern / Texture Recognition
  - Filtering:  Convolution and Deconvolution

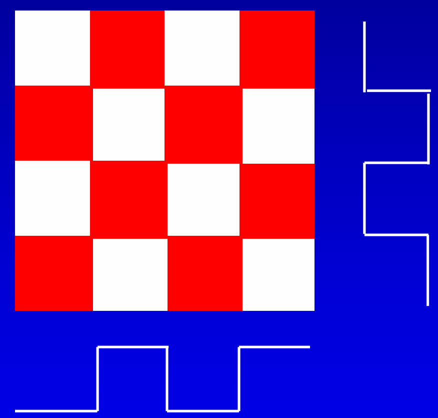
# Spatial Frequency in Images

Frequency = 1

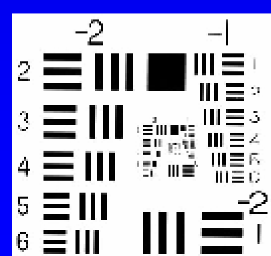


1 Cycle

Frequency = 2

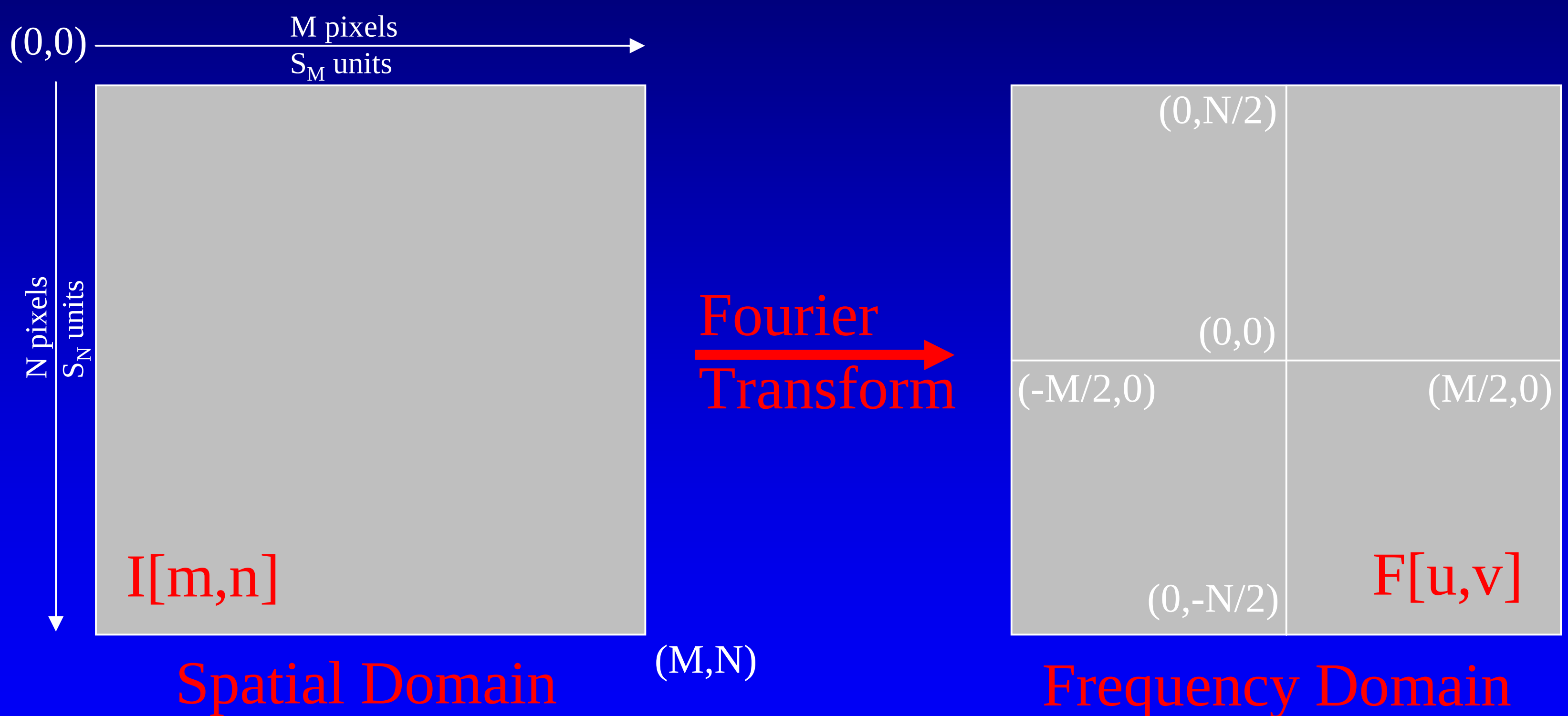


2 Cycles



# 2D Discrete Fourier Transform

$$F[u,v] = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} I[m,n]\cdot e^{-i2p\left(\frac{um}{M}+\frac{vn}{N}\right)}$$

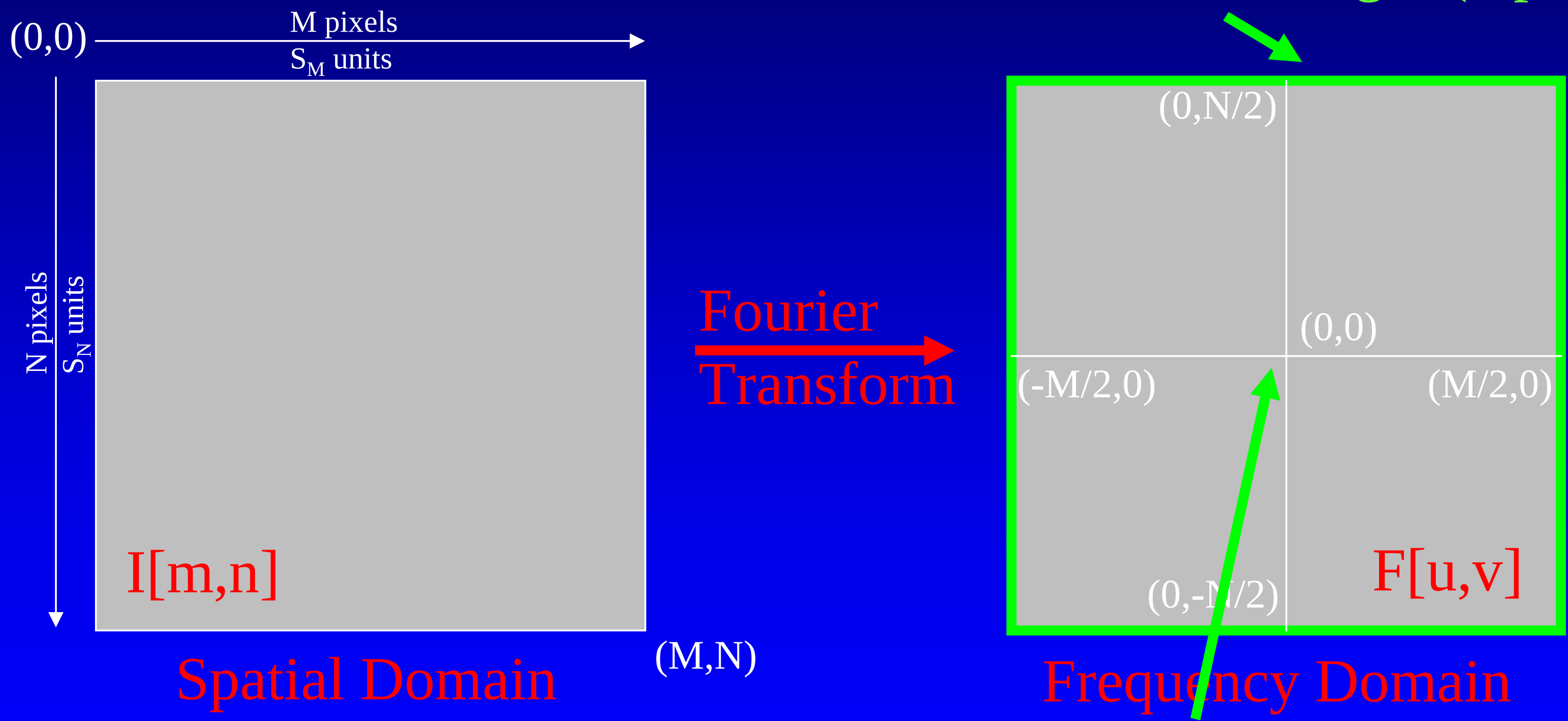



Source: Seul et al, *Practical Algorithms for Image Analysis*, 2000, p. 249, 262.

*2D FFT can be computed as two discrete Fourier transforms in 1 dimension*

# 2D Discrete Fourier Transform

Edge represents highest frequency,
smallest resolvable length (2 pixels)

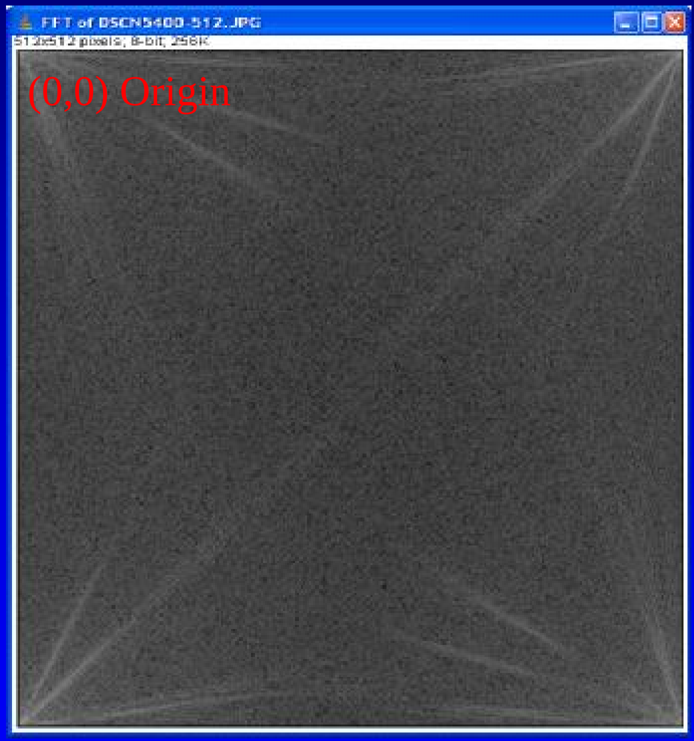



Center represents lowest frequency,
which represents average pixel value

# 2D FFT Example

FFTs Using ImageJ

ImageJ Steps: (1) File | Open, (2) Process | FFT | FFT


Spatial Domain

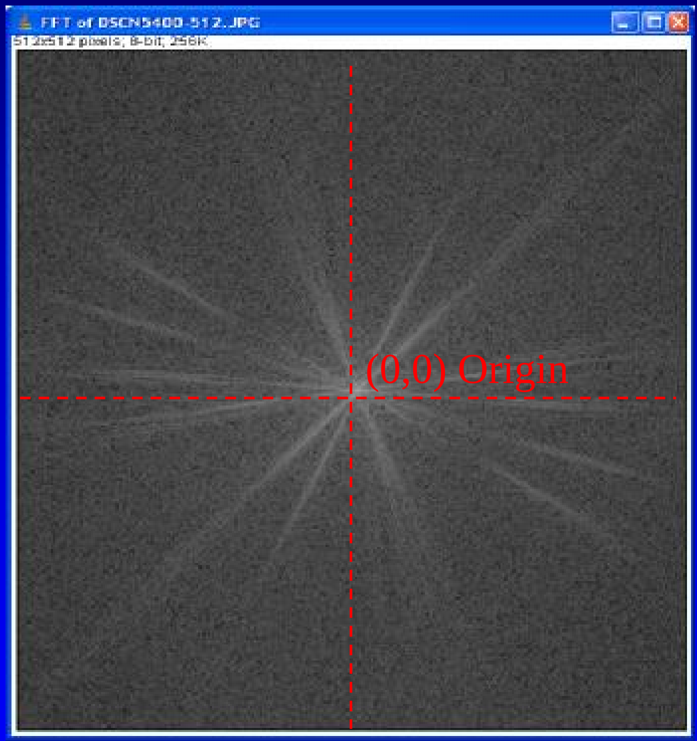
Frequency Domain

# 2D FFT Example

FFTs Using ImageJ

ImageJ Steps: Process | FFT | Swap Quadrants


Spatial Domain




Default display is to swap quadrants

Frequency Domain



*Regularity in image manifests itself in the degree of order or randomness in FFT pattern.*

# 2D FFT Example

## FFTs Using ImageJ

ImageJ Steps:  (1) File | Open, (2) Process | FFT | FFT


Overland Park Arboretum and Botanical Gardens, June 2006

Spatial Domain

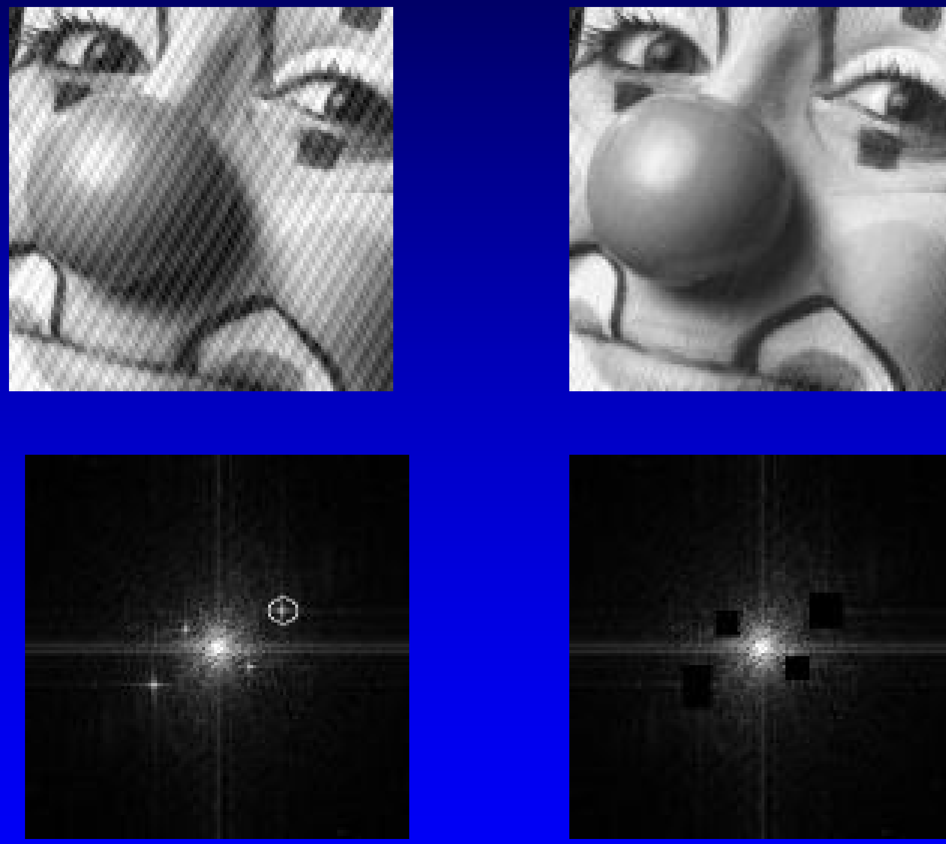

Frequency Domain



*Regularity in image manifests itself in the degree of order or randomness in FFT pattern.*

# Application of FFT in Image Processing

## Noise Removal



FFT

Noise Pattern
Stands Out as
Four Spikes

Edit FFT

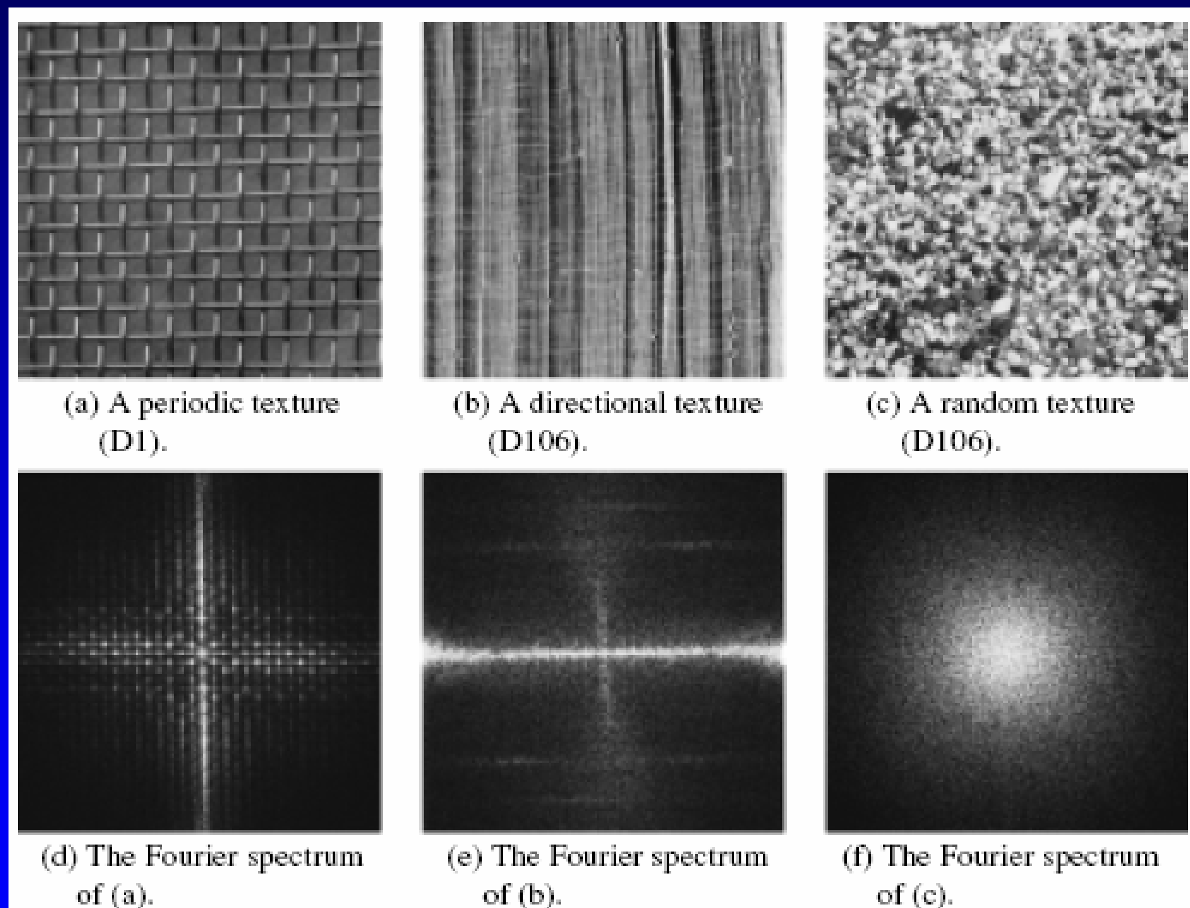
Four Noise
Spikes Removed

Inverse
FFT

Source: www.mediacy.com/apps/fft.htm, Image Pro Plus FFT Example. Last seen online in 2004.

# Application of FFT

## Pattern/Texture Recognition



(a) A periodic texture (D1).

(b) A directional texture (D106).

(c) A random texture (D106).

(d) The Fourier spectrum of (a).

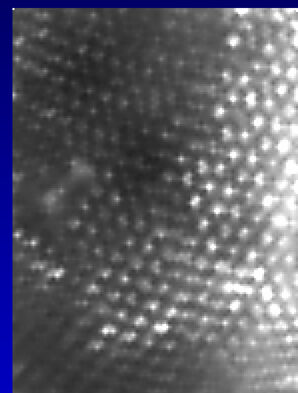
(e) The Fourier spectrum of (b).

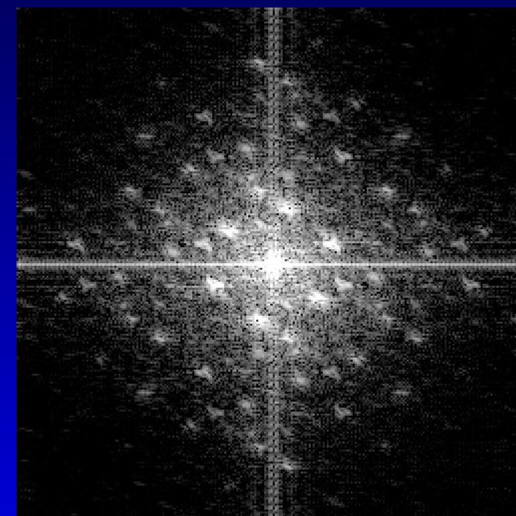
(f) The Fourier spectrum of (c).

Source: Lee and Chen, A New Method for Coarse Classification of Textures and Class Weight Estimation for Texture Retrieval, *Pattern Recognition and Image Analysis*, Vol. 12, No. 4, 2002, pp. 400–410.

# Application of FFT

Pattern/Texture Recognition



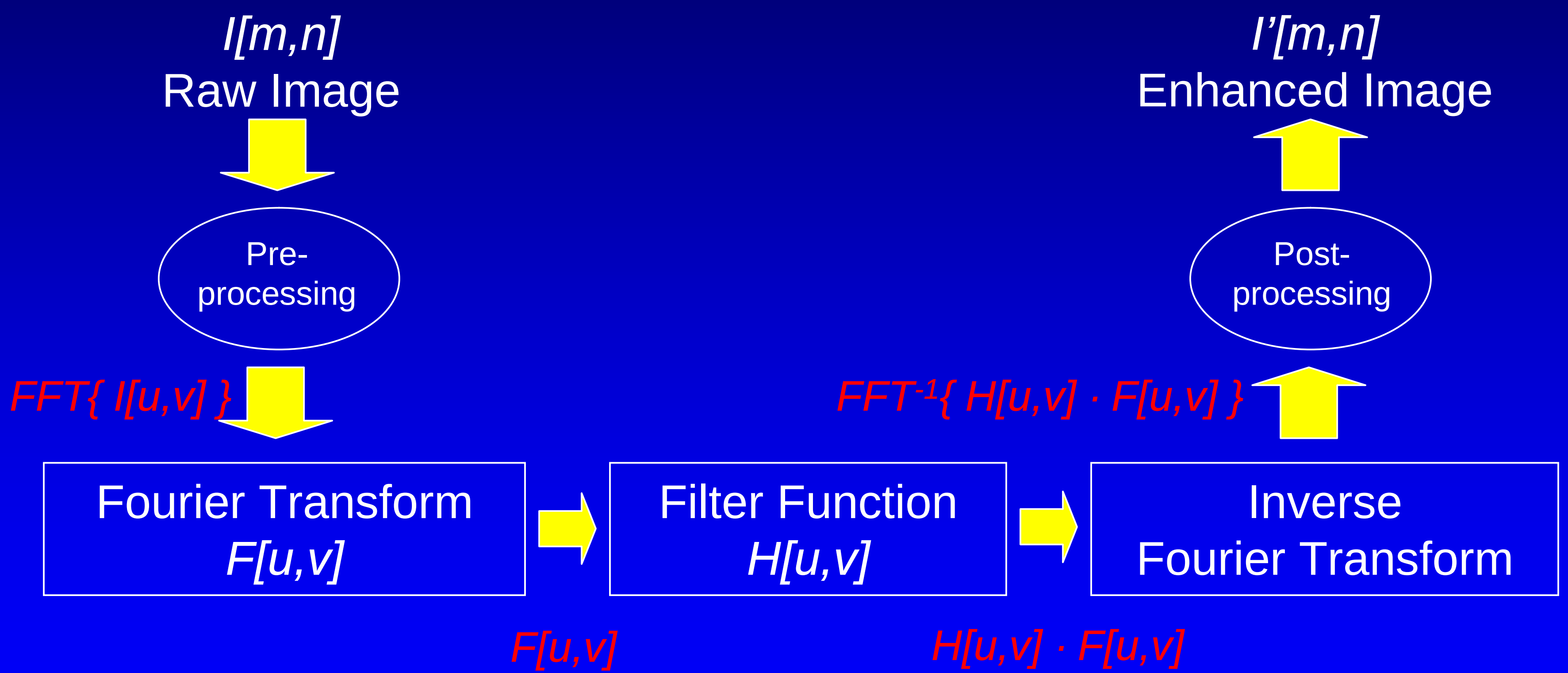
The Drosophila eye is a great example a cellular crystal with its hexagonally closed-packed structure. The absolute value of the Fourier transform (right) shows its hexagonal structure.

Source: http://www.rpgroup.caltech.edu/courses/PBL/size.htm

*Could FFT of Drosophila eye be used to identify/quantify subtle phenotypes?*

# Application of FFT

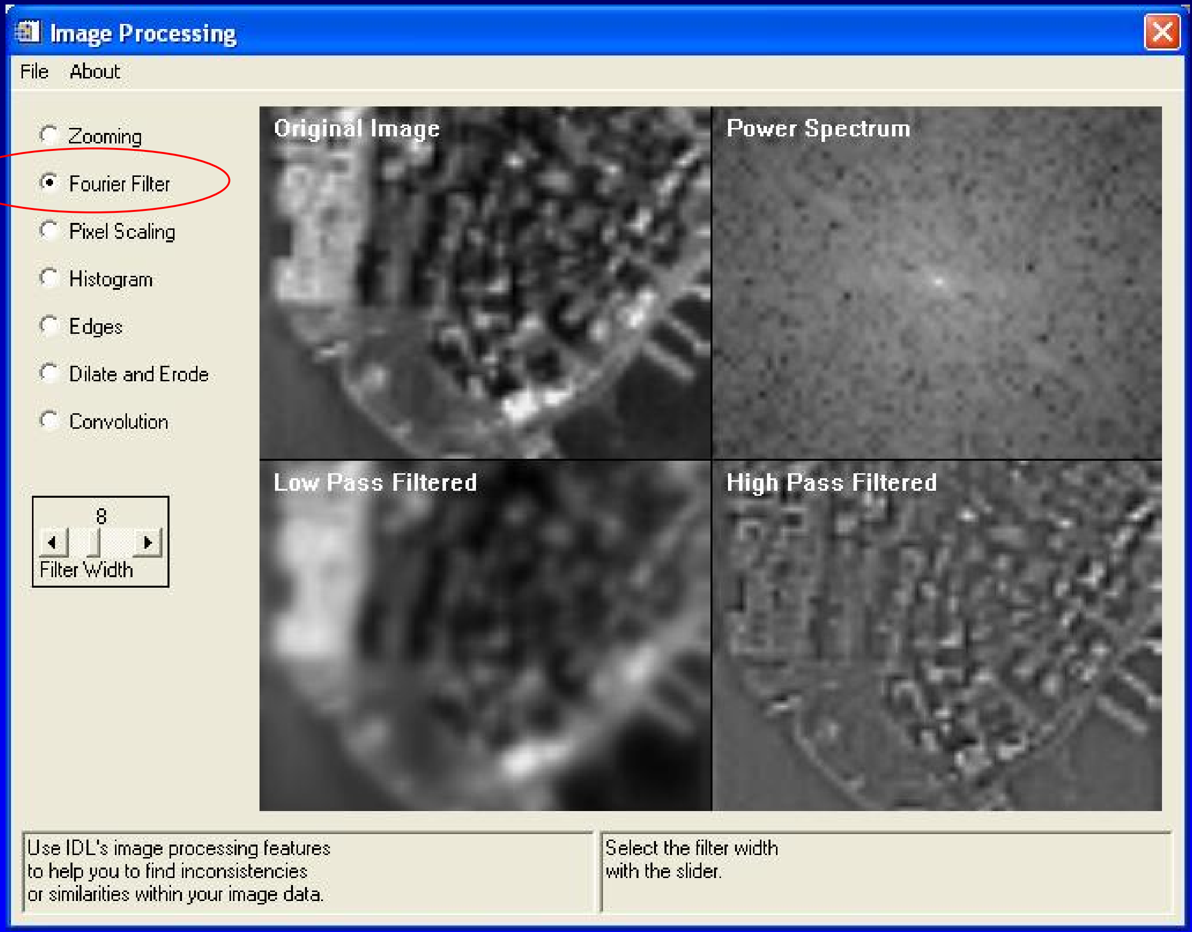
## Filtering in the Frequency Domain: Convolution




Source: Gonzalez and Woods, *Digital Image Processing* (2nd ed), 2002, p. 159

# Application of FFT

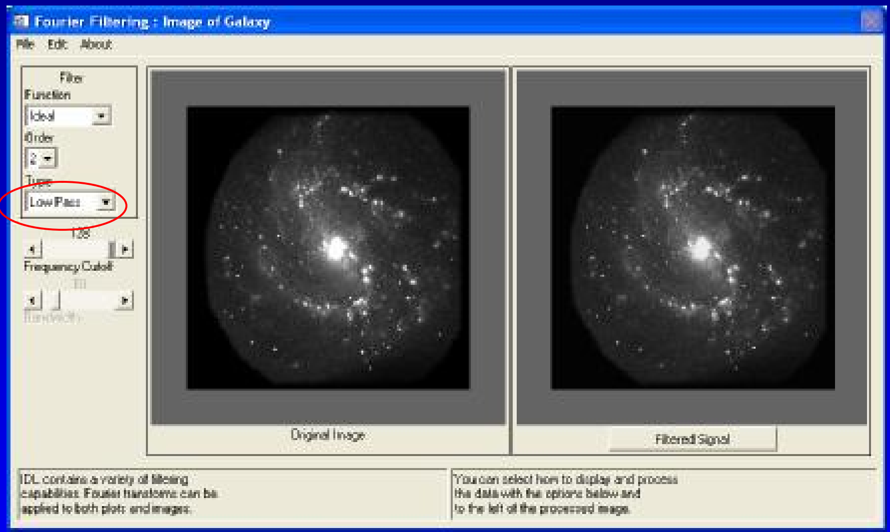
## Filtering: IDL Fourier Filter Demo

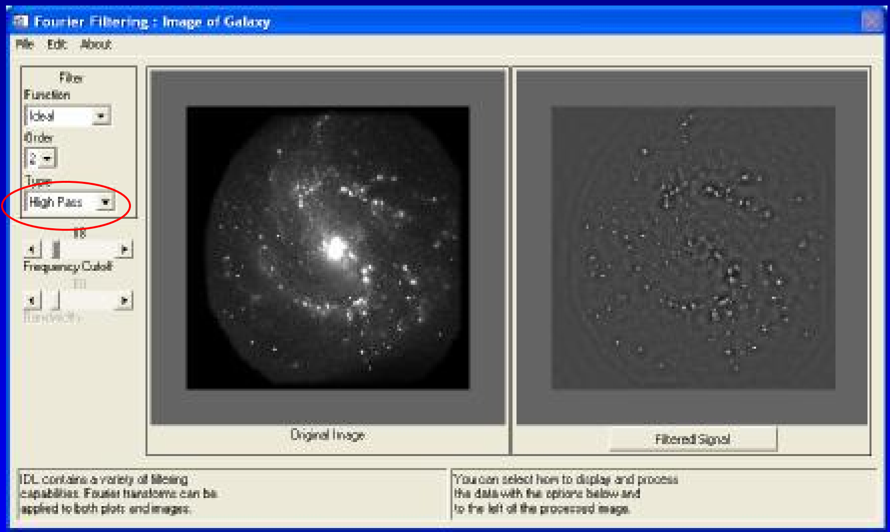



*IDL Run Demo, Data Analysis, Image Processing, Image Processing Demo*

# Application of FFT

Filtering:  IDL Fourier Filtering Demo



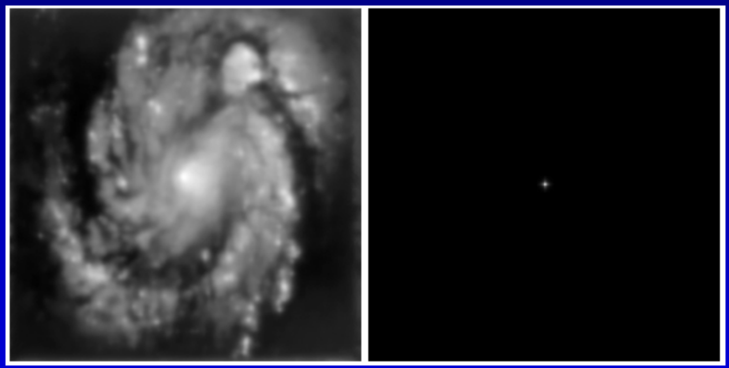




*IDL Run Demo, Data Visualization, Images, Fourier Filtering*

# Application of FFT

## Deblurring: Deconvolution

The Point Spread Function (PSF) is the Fourier transform of a filter.
(the PSP says how much blurring there will be in trying to image a point).



Hubble image and measured PSF

*Dividing the Fourier transform of the PSF into the transform of the blurred image, and performing an inverse FFT, recovers the unblurred image.*

FFT(Unblurred Image) * FFT(Point Spread Function) = FFT(Blurred Image)

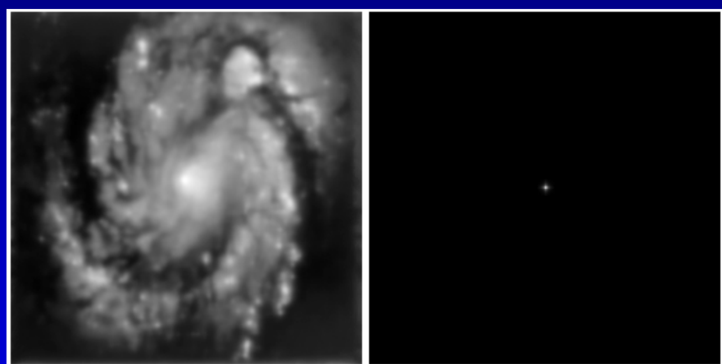
Unblurred Image = $FFT^{-1}$[ FFT(Blurred Image) / FFT(Point Spread Function) ]

Source: http://www.reindeergraphics.com/index.php?option=com_content&task=view&id=179&Itemid=127

# Application of FFT

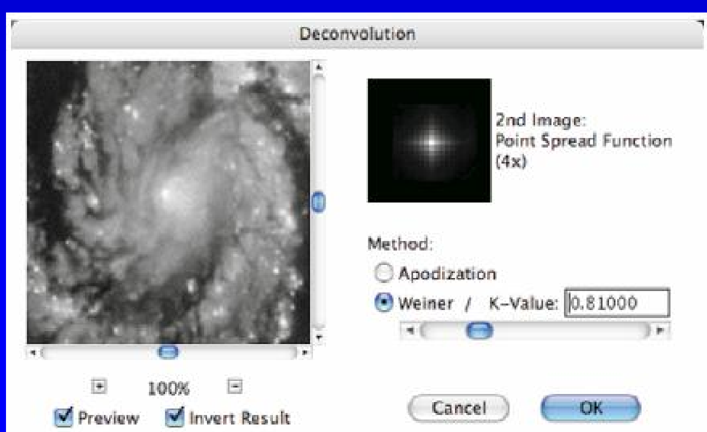
## Deblurring: Deconvolution

The Point Spread Function (PSF) is the Fourier transform of a filter.
(the PSP says how much blurring there will be in trying to image a point).



Hubble image and measured PSF

*Dividing the Fourier transform of the PSF into the transform of the blurred image, and performing an inverse FFT, recovers the unblurred image.*




Deblurred image

Source: http://www.reindeergraphics.com/index.php?option=com_content&task=view&id=179&Itemid=127

# Summary

• Fourier Analysis is a powerful tool even when periodicity is not directly a part of the problem being solved.

• Discrete Fourier Transforms (DFT) are well-suited for computation by computer, especially when using Fast Fourier Transform (FFT) algorithms.

• Fourier Analysis can be used to remove noise from a signal or image.

• Interpretation of the complex Fourier Transform is not always straightforward.

• Convolution and Deconvolution are "simple" in Fourier transform space to restore or enhance images.

• There are many other image processing uses of Fourier Analysis, such as image compression [JPGs use the Discrete Cosine Transform (DCT), which is a special kind of DFT]