Capture Frequency - Rolling `Shutter’

Rolling Shutter: 果冻效应

当使用电子快门来拍摄高速移动的物件时，

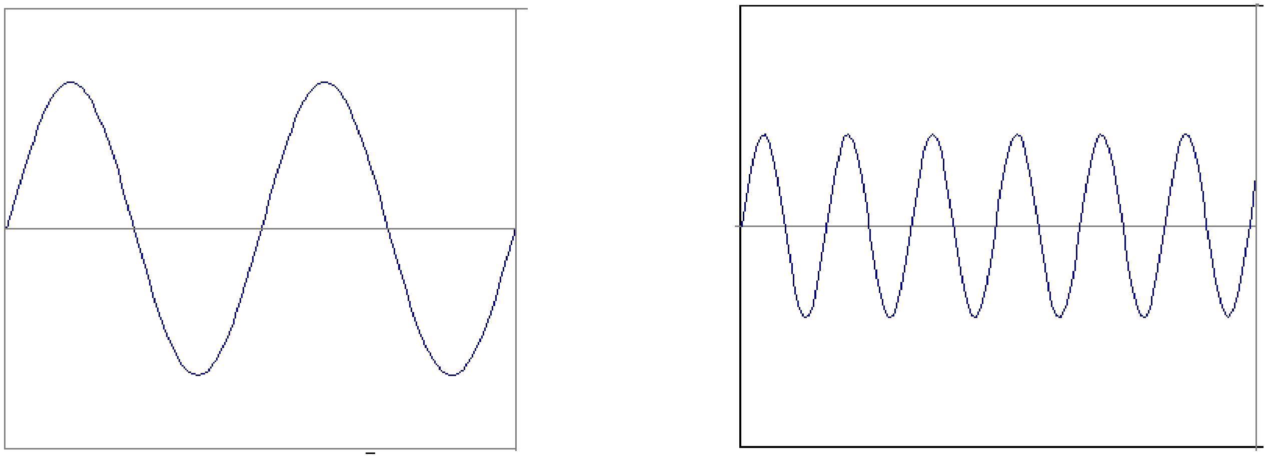
原本垂直的物件拍摄出的画面却为倾斜甚至变形。

未来媒体研究中心
CENTER FOR FUTURE MEDIA

电子科技大学
University of Electronic Science and Technology of China

James Hays

I understand frequency as in waves…



…but how does this relate to the complex signals we see in natural images?

…to image frequency?

Another way of thinking about frequency

FOURIER SERIES & FOURIER TRANSFORMS 傅里叶级数与傅里叶变换

Fourier series

Jean Baptiste Joseph Fourier (1768-1830)

让·巴普蒂斯·约瑟夫·傅里叶

A bold idea (1807):

> ***Any*** *univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies.*

Fourier, Joseph (1768-1830)

French mathematician who discovered that any periodic motion can be written as a superposition of sinusoidal and cosinusoidal vibrations. He developed a mathematical theory of heat in *Théorie Analytique de la Chaleur (Analytic Theory of Heat),* (1822), discussing it in terms of differential equations.

Fourier was a friend and advisor of Napoleon. Fourier believed that his health would be improved by wrapping himself up in blankets, and in this state he tripped down the stairs in his house and killed himself. The paper of Galois which he had taken home to read shortly before his death was never recovered.

SEE ALSO: Galois

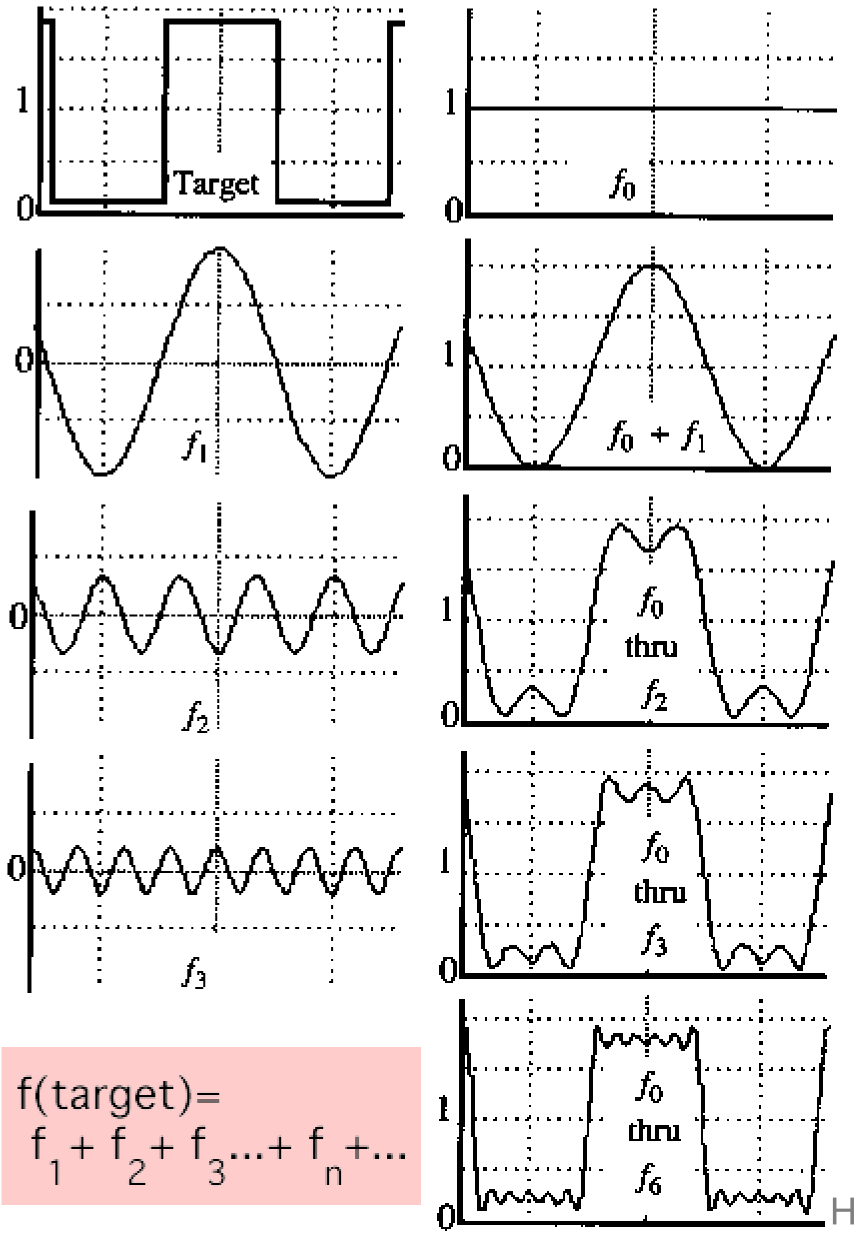
Additional biographies: MacTutor (St. Andrews), Bonn

由于傅里叶极度痴迷热学，他认为热能包治百病，于是在一个夏天，他关上了家中的门窗，穿上厚厚的衣服，坐在火炉边，于是他被活活热死了，1830年5月16日卒于法国巴黎。[百度百科]

Fourier series

A bold idea (1807):

***Any** univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies.*



Our building block:

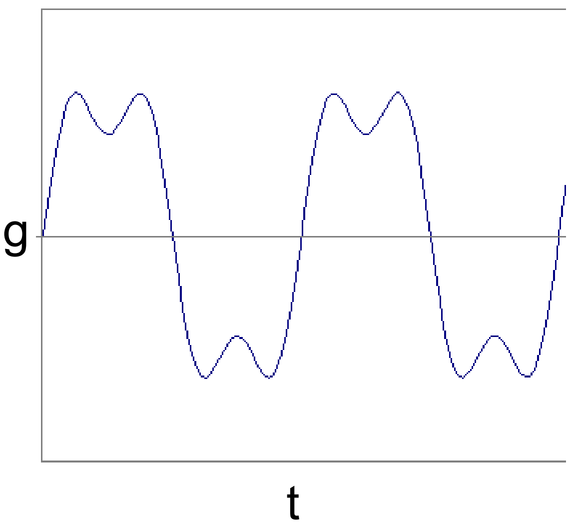
$$A\sin(\omega t) + B\cos(\omega t)$$

Add enough of them to get any signal *g(t)* you want!

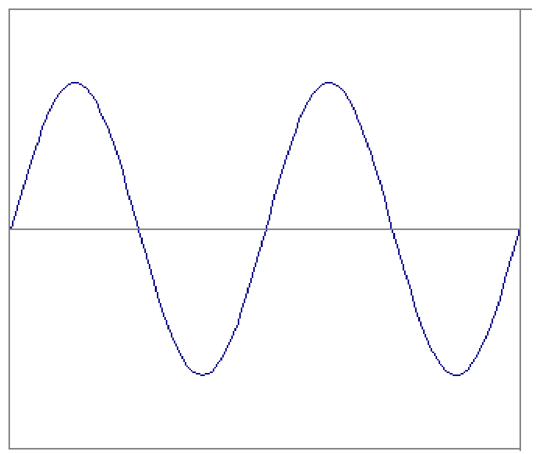
f(target)=
$f_1 + f_2 + f_3 ... + f_n + ...$

$$t = [0,2], f = 1$$

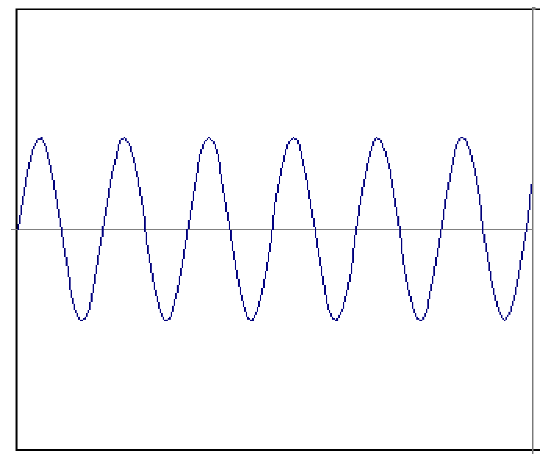
$$g(t) = \sin(2\pi f\, t) + (1/3)\sin(2\pi(3f)\, t)$$

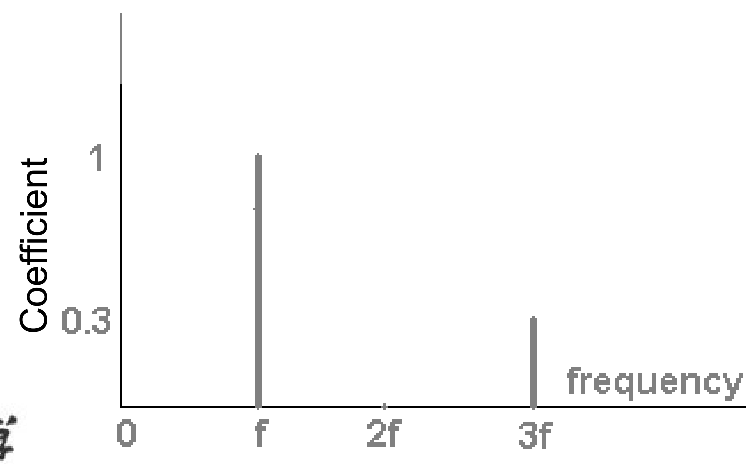


=

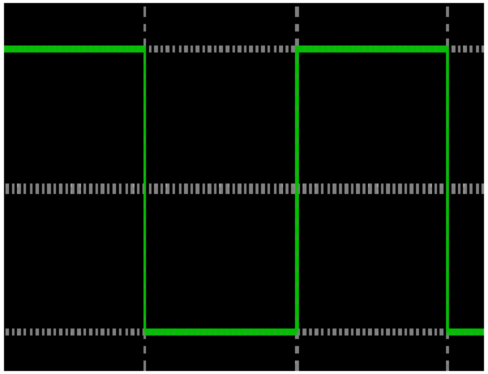


+

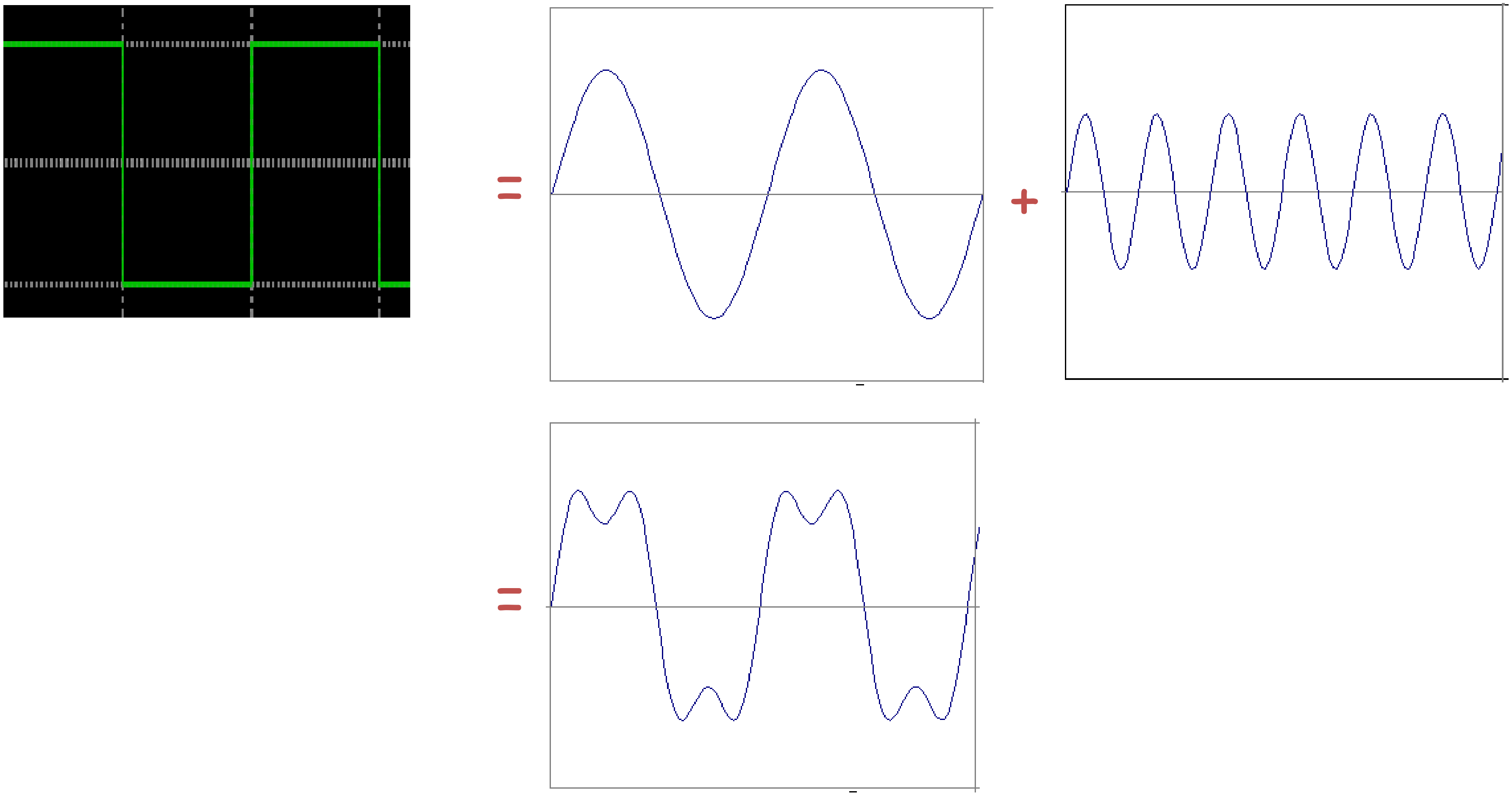




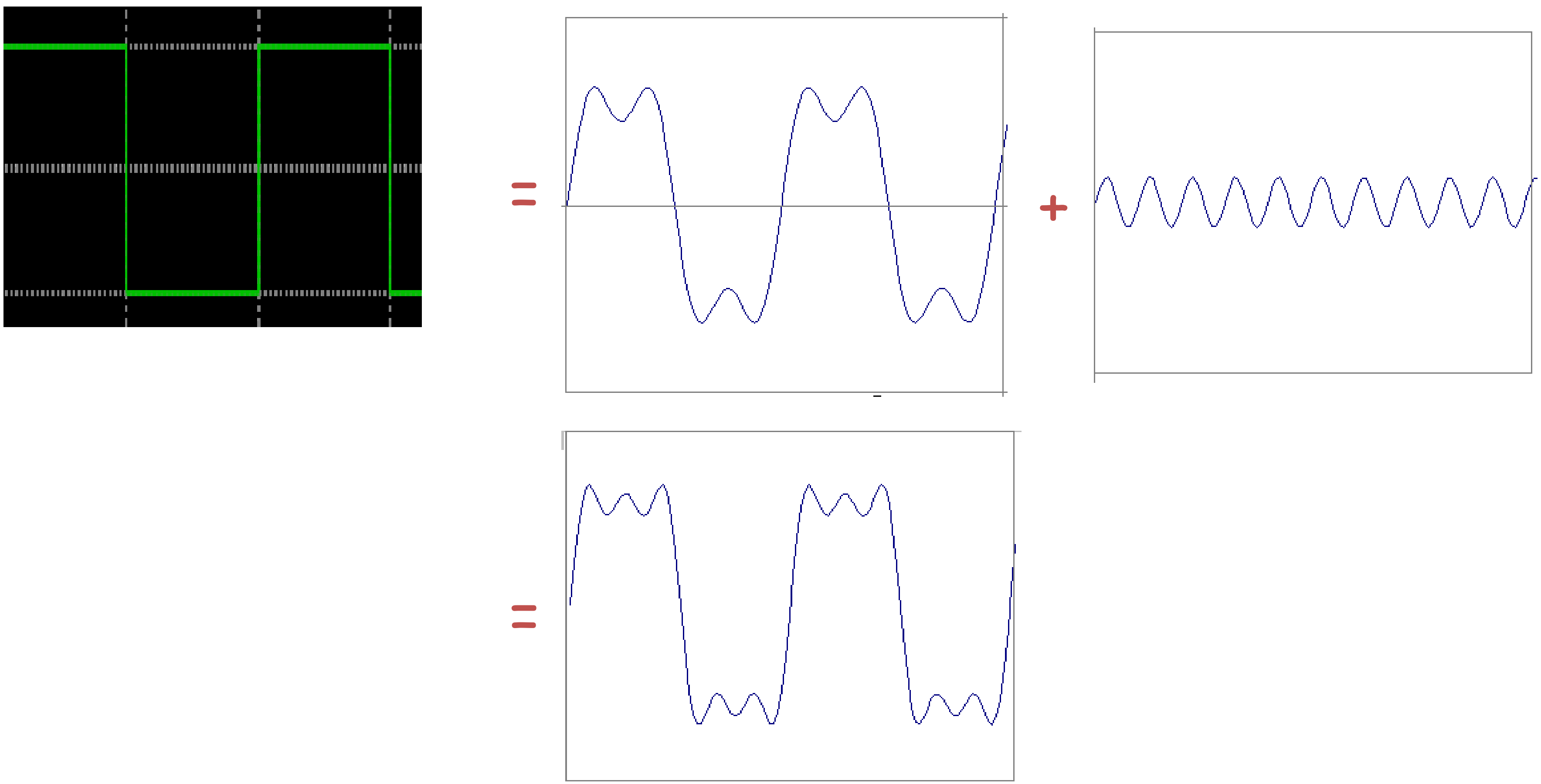
Square wave spectra



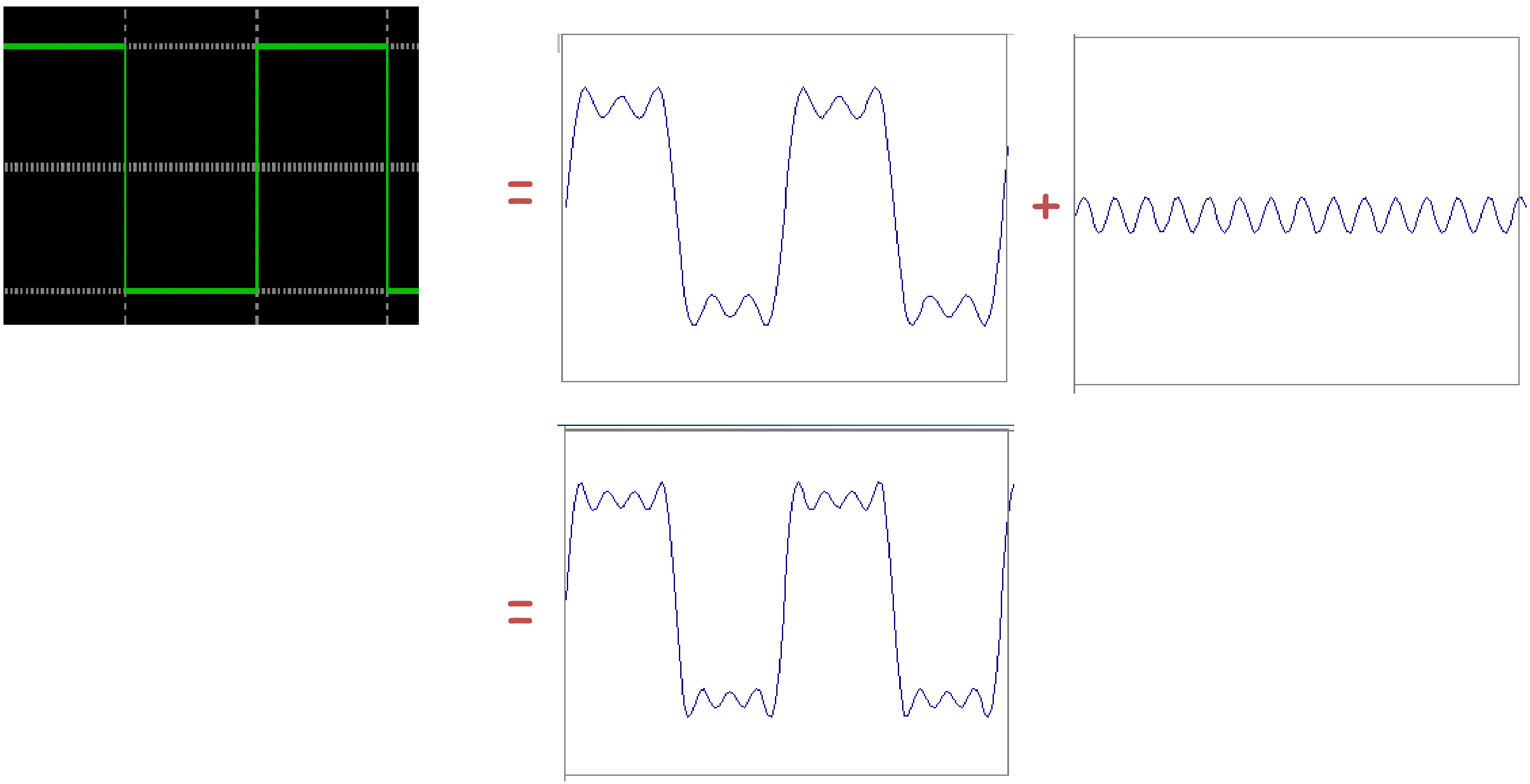
Square wave spectra



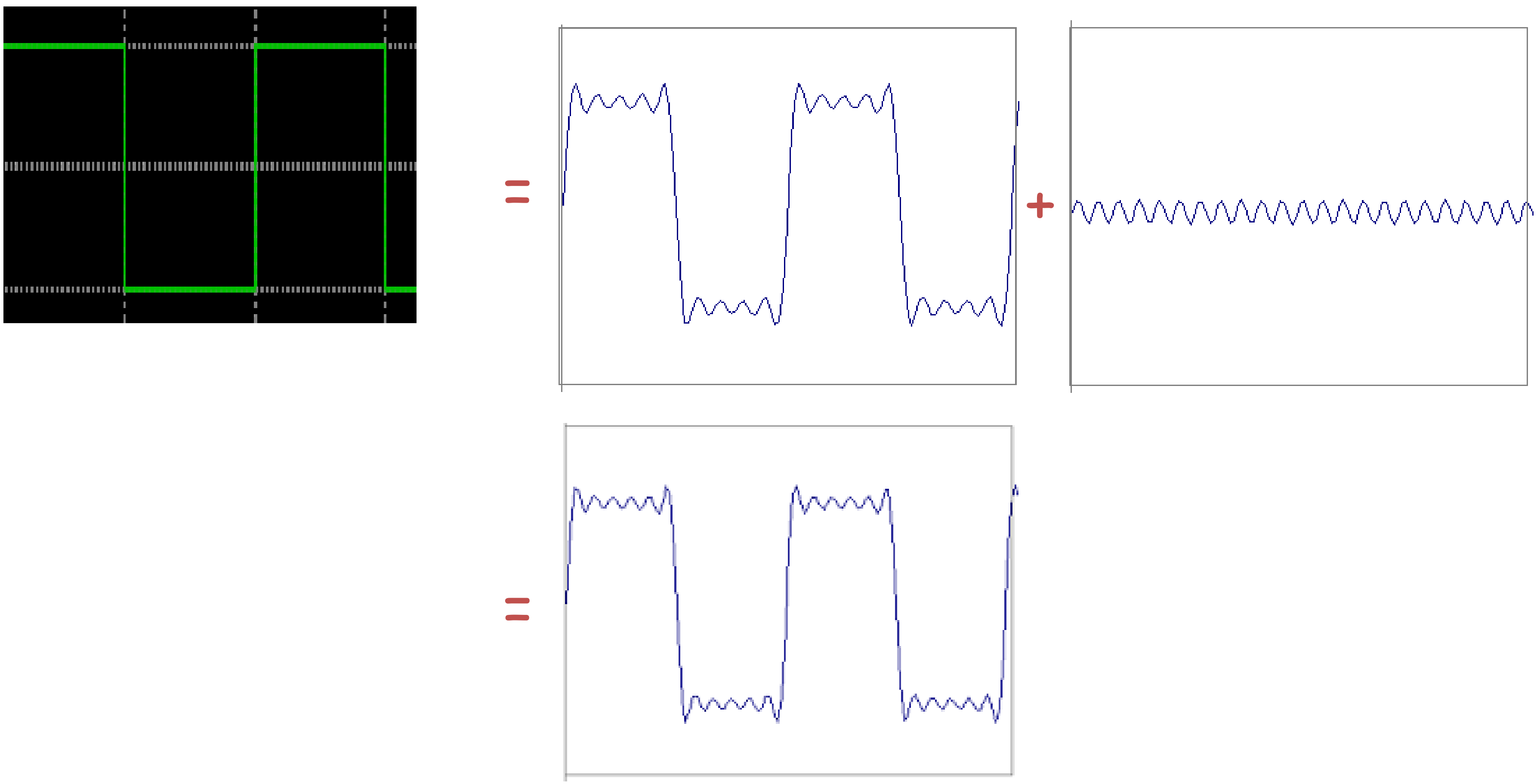
Square wave spectra



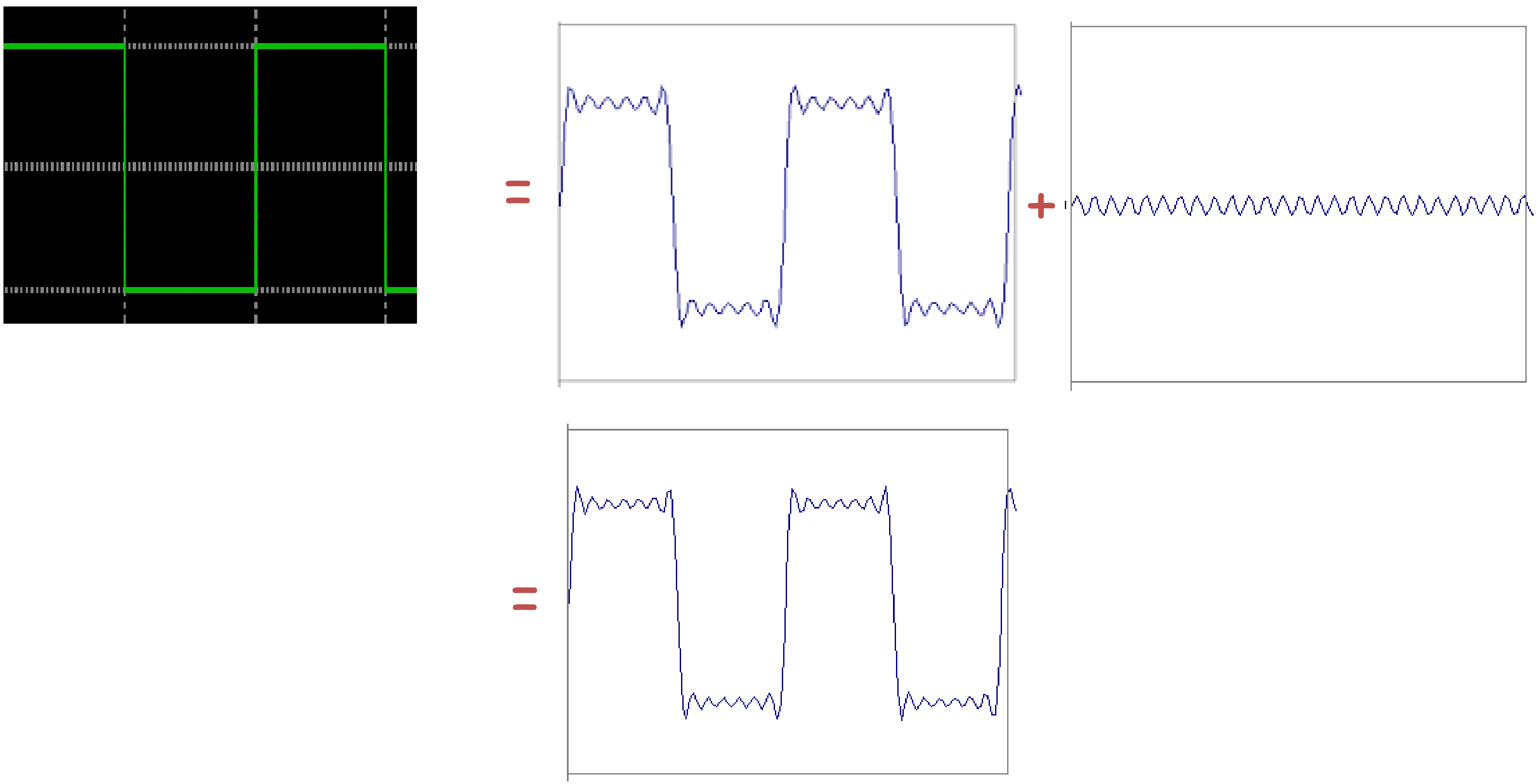
Square wave spectra



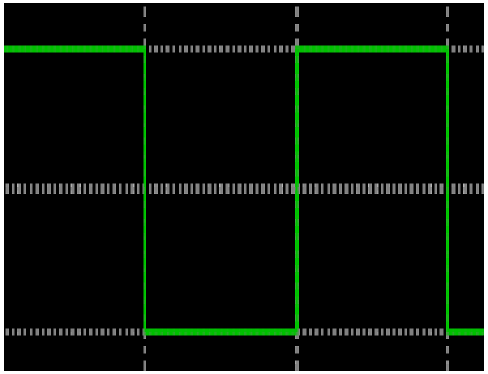
Square wave spectra



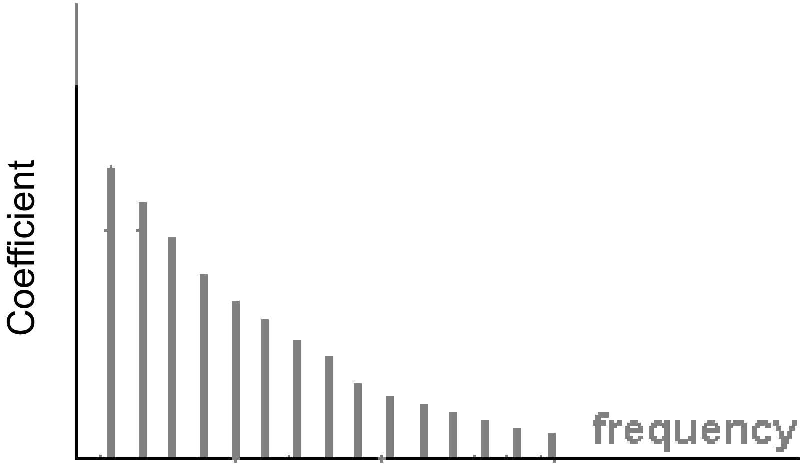
Square wave spectra

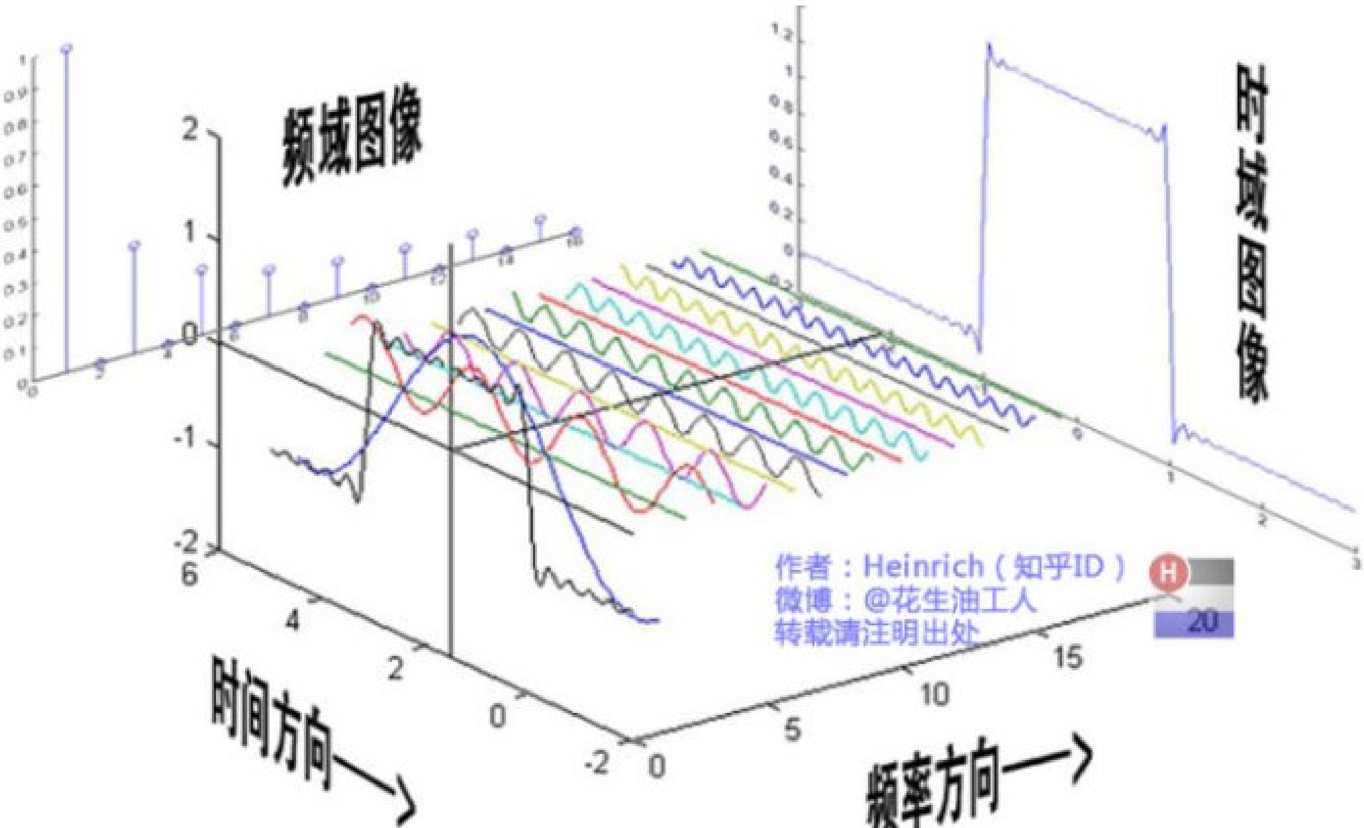


Square wave spectra



$$= \quad A\sum_{f=1}^{\infty}\frac{1}{f}\sin(2\pi f t)$$





频域图像
时域图像
时间方向
频率方向
作者：Heinrich（知乎ID）
微博：@花生油工人
转载请注明出处

Jean Baptiste Joseph Fourier (1768-1830)

A bold idea (1807):

> ***Any*** *univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies.*

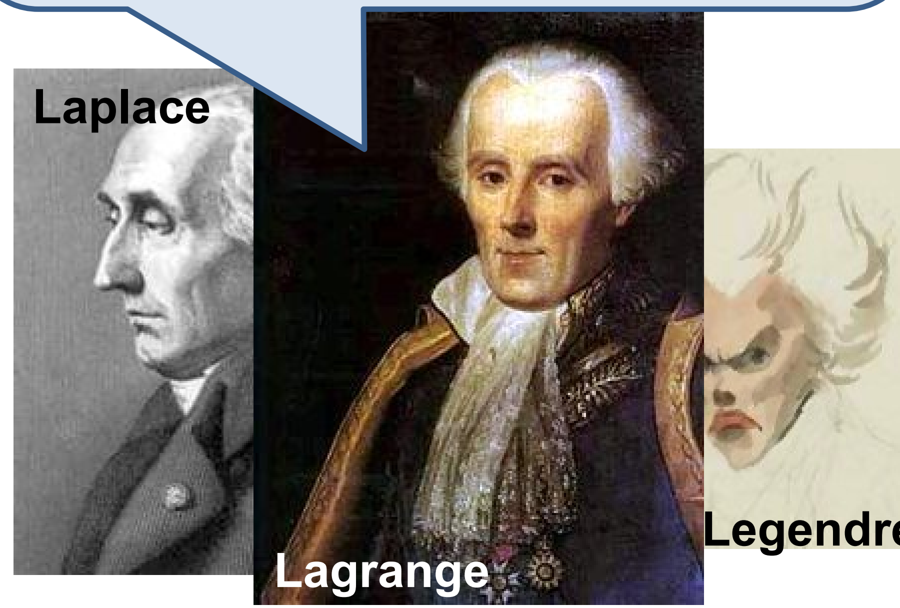
Don’t believe it?

- Neither did Lagrange, Laplace, Poisson and other big wigs
- Not translated into English until 1878!

But it’s (mostly) true!

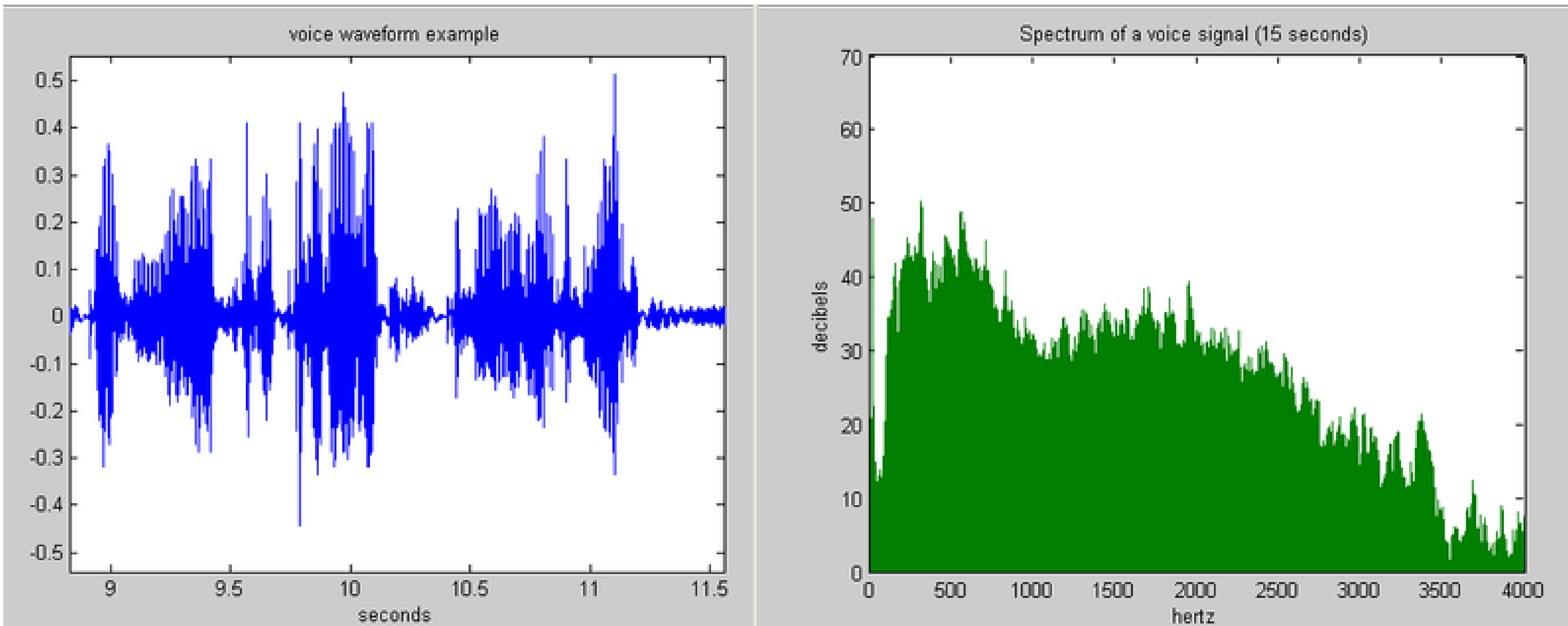
- Called Fourier Series

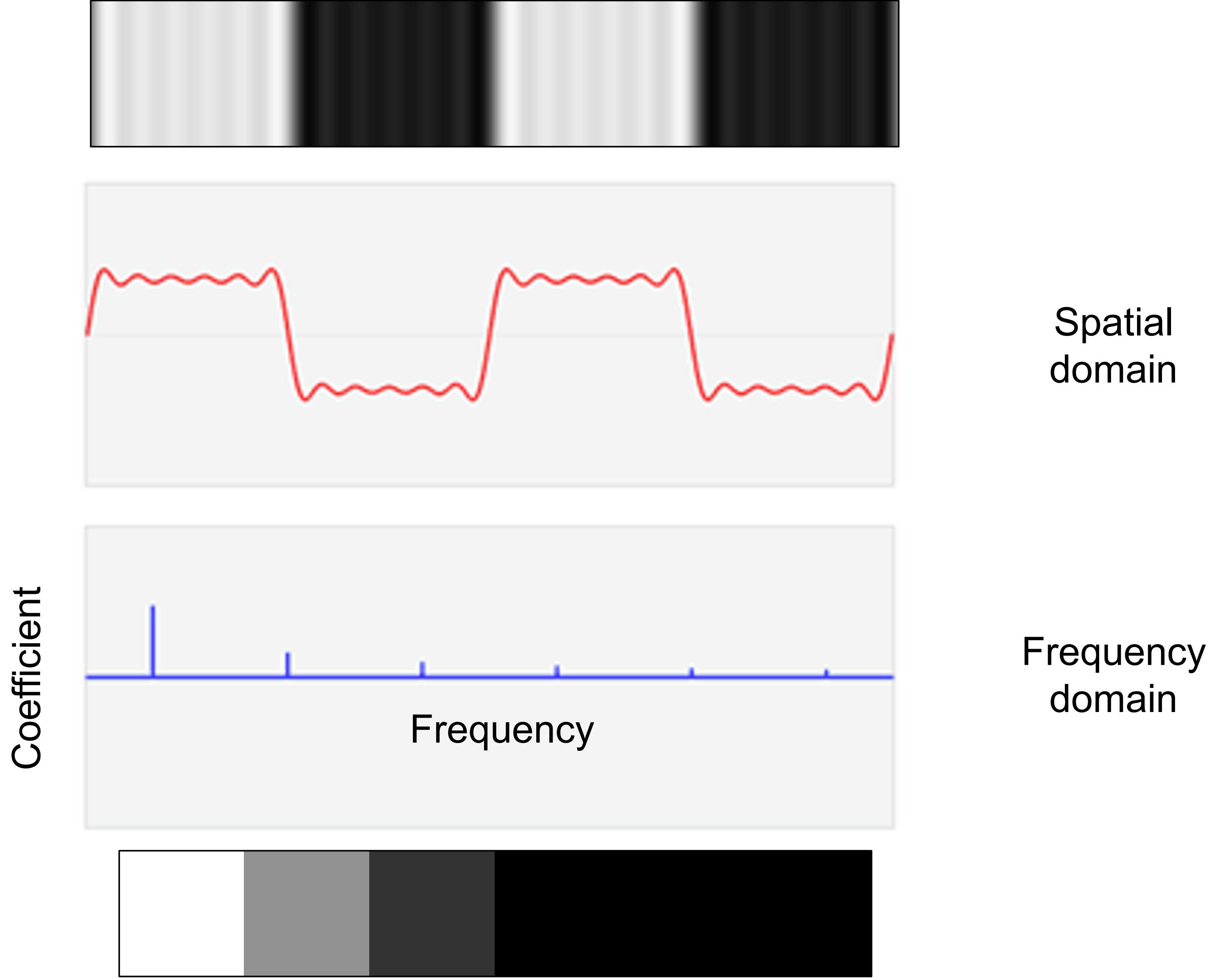
...the manner in which the author arrives at these equations is not exempt of difficulties and...his analysis to integrate them still leaves something to be desired on the score of generality and even rigour.



Example: Music

- We think of music in terms of frequencies at different magnitudes



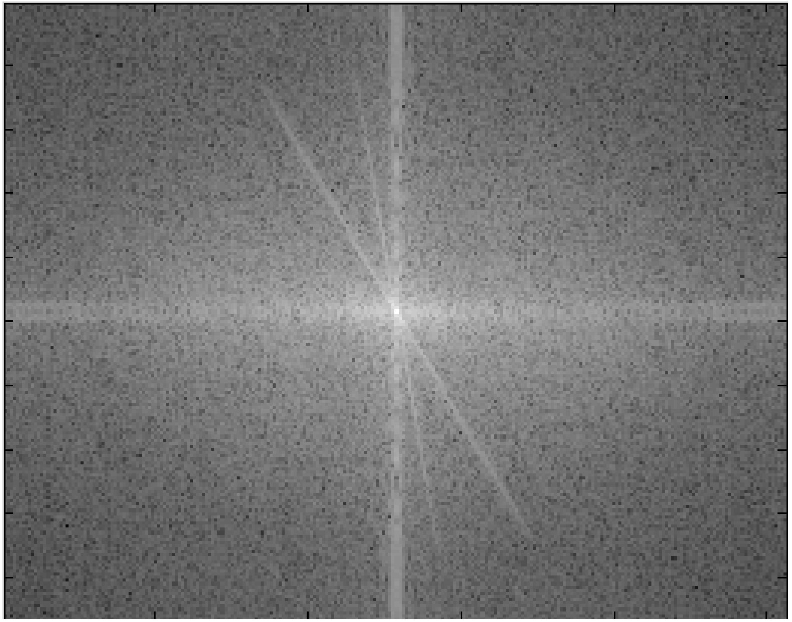


Wikipedia – Fourier transform

Natural image

Natural image

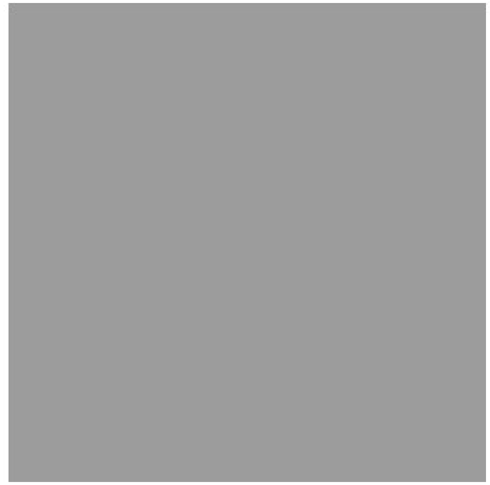
Fourier decomposition
Frequency coefficients (amplitude)



What does it mean to be at pixel x,y?
What does it mean to be more or less bright in the Fourier decomposition image?

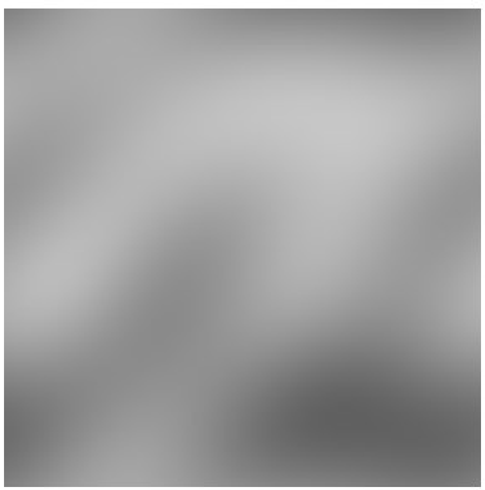
Basis reconstruction

Full image



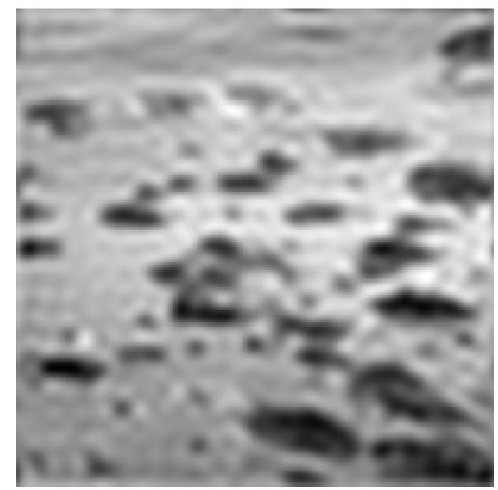
First 1 basis fn

First 4 basis fns



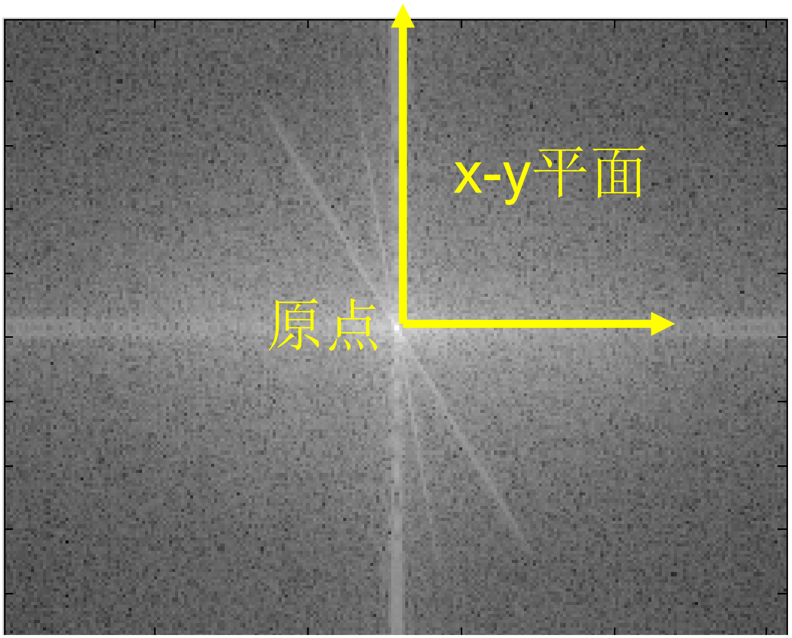
First 9 basis fns

First 16 basis fns



First 400 basis fns

Danny Alexander



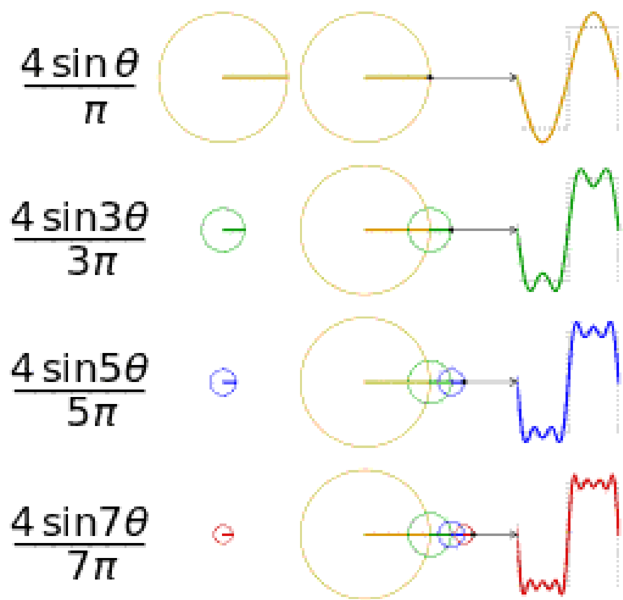
- 距离原点越远=频率越高=原图中灰度值的变化越频繁。
- 灰度值越大=幅值越大=原图中灰度值变化的范围越大。

《数字图像处理与机器视觉:
Visual C++与Matlab实现》
人民邮电出版社

Fourier Series

$$f(x)=\frac{a_0}{2}+\sum_{n=1}^{+\infty} a_n * cos(x\omega n)+b_n * sin(x\omega n)$$

对于任意一个周期为T的函数f(x)，它可以被表示成一系列正弦波和余弦波的叠加



2D Fourier Series

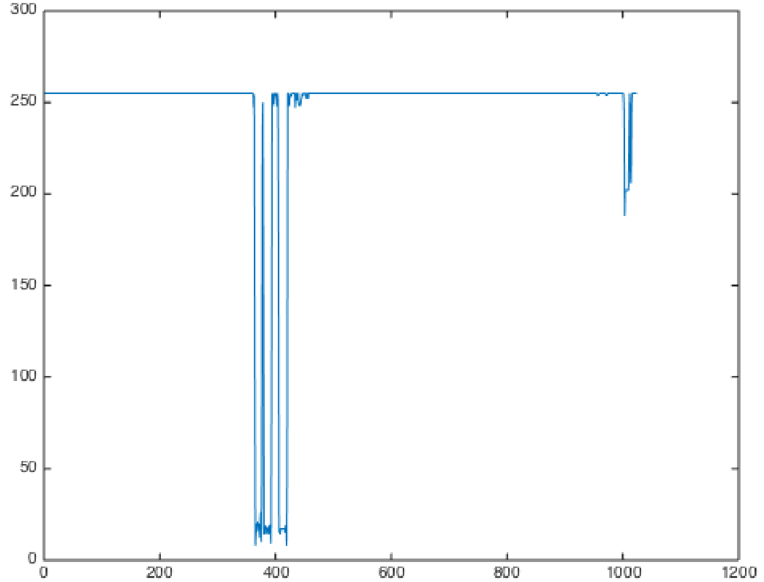
$$F(\omega)=\mathcal{F}(f(t))=\int_{-\infty}^{+\infty} f(t)e^{-i\omega t}dt$$

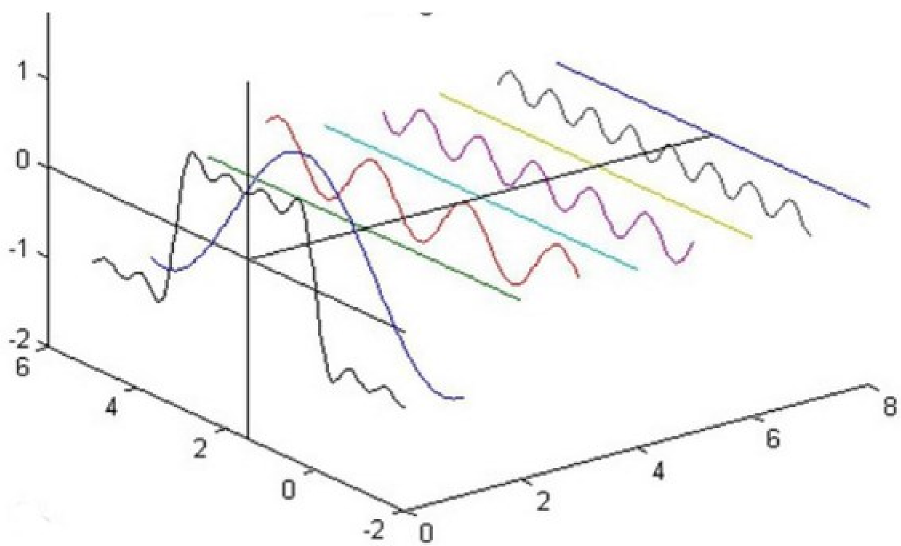
现实中的一个任意数据f(t)，是由很多频率的正弦，余弦波叠加组成。

- 对于图像信号
  - 图片是二维，公式是一维
  - 图片数据有限且离散，公式无限且连续
  - 二维矩阵中每一列（或者每一行）数据都可以单独看作一个波
  - 对一个MxN的矩阵，可以看作N个Mx1的波

2D Fourier Series Decomposition

- 二维的分解
  - 在二维傅立叶变换里，先分别对每一列（行）做傅立叶变换，会得到同样大小的傅立叶系数向量
  - 再在另一个维度（上面是行这里就是列，上面是列这里就是行）对这些系数做傅立叶变换。





2D Fourier Series Decomposition

- 图片中的像素值

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{P}[a_n * cos(n\omega x) + b_n * sin(n\omega x)]$$

  - 其中ω是所有频率波中的单位频率，它决定了波叠加的效果
  - 如果可以求出a_n和b_n的值，就完成了离散傅立叶变换
  - **图像中一列数据组成的波，可以看作是P个频率的正弦和余弦波叠加而成，这些波具有不同的振幅，我们的目的就是求出这些振幅的具体数值。**

2D Fourier Series Decomposition

- 图片中的像素值

[redacted]

– 如何求ω?
（决定数据伸缩程度）

  - 一列图像的像素值，如果它的大小是N x 1，也就是说有N个f(x)的值，而未知数共有2P+1个

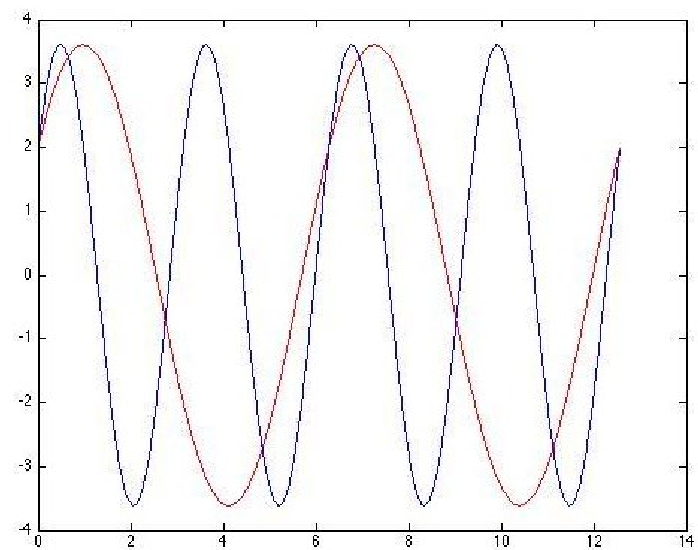
令 $a_0 = 0$, $P = 1$, $a_1 = 2$, $b_1 = 3$, 随意取俩个频率 $\omega_1 = 1$, $\omega_2 = 2$,

  - **如何求ω?**

(决定数据伸缩程度）

$$f(x) = a_1 * cos(\omega_1 x) + b_1 * sin(\omega_1 x) = 2cos(x) + 3sin(x)$$

$$f(x) = a_1 * cos(\omega_2 x) + b_1 * sin(\omega_2 x) = 2cos(2x) + 3sin(2x)$$



未来媒体研究中心
CENTER FOR FUTURE MEDIA
电子科技大学
University of Electronic Science and Technology of China

2D Fourier Series Decomposition

- 图片中的像素值

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{P}[a_n * cos(n\omega x) + b_n * sin(n\omega x)]$$

  - 对一个图片，其中一列数据(N x 1的向量)做傅立叶变换，把这列数据看作是周期为**T = N**的周期数据
  - **在一张图片的上下俩侧平铺无数张同样的图片，那我们取出的这一列数据就变成一个T = N的周期函数中的一段**
  - **图像上的一列像素是由P个频率的波叠加而成的，这些波的单位频率$\omega = \frac{2\pi}{N}$，上述公式未知数可解**

2D Fourier Series Decomposition

- 图片中的像素值

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{P}[a_n * cos(n\omega x) + b_n * sin(n\omega x)]$$

$$cos(\theta) = \frac{1}{2}(e^{i\theta} + e^{-i\theta}) \qquad sin(\theta) = -\frac{i}{2}(e^{i\theta} - e^{-i\theta})$$

$$\Rightarrow \quad f(x) = \frac{a_0}{2} + \sum_{n=1}^{P}(\frac{a_n - ib_n}{2}e^{in\omega x} + \frac{a_n + ib_n}{2}e^{-in\omega x})$$

$\Rightarrow$ 当n=1,2,3,…,P时

$$\boxed{c_n = \frac{a_n - ib_n}{2}, \quad c_{2P+1-n} = \frac{a_n + ib_n}{2}, \quad c_0 = \frac{a_0}{2},}$$

2D Fourier Series Decomposition

$$cos(\theta) = \frac{1}{2}(e^{i\theta} + e^{-i\theta}) \qquad sin(\theta) = -\frac{i}{2}(e^{i\theta} - e^{-i\theta})$$

$$\Rightarrow \quad f(x) = \frac{a_0}{2} + \sum_{n=1}^{P}\left(\frac{a_n - ib_n}{2}e^{in\omega x} + \frac{a_n + ib_n}{2}e^{-in\omega x}\right)$$

$\Rightarrow$ 当n=1,2,3,…,P时

$$\boxed{\begin{aligned} c_n &= \frac{a_n - ib_n}{2}, \\ c_{2P+1-n} &= \frac{a_n + ib_n}{2}, \\ c_0 &= \frac{a_0}{2}, \end{aligned}}$$

由于$\omega = \frac{2\pi}{N}$ $\quad e^{i\omega(N-n)x} = e^{-i\omega nx}, N = 2P + 1, n = 1, 2, 3...$

$$\Rightarrow \quad f(x) = \sum_{m=0}^{N-1} \boxed{c_m} * e^{i*\omega*m*x} \qquad N = 2P + 1$$

2D Fourier Series Decomposition

$$f(x) = \sum_{m=0}^{N-1} \boxed{c_m} * e^{i*\omega*m*x} \qquad N = 2P + 1$$

一列N x 1的图像数据提供N个f(x)，而右式有N个未知数c_m

⟹ **傅里叶变换就是在计算 c_m的值！**

$$f(0) = c_0 * e^0 + c_1 * e^{i0\omega*1} + c_2 * e^{i0\omega*2} + \ldots + c_{N-1} * e^{i0\omega*(N-1)}$$

$$f(1) = c_0 * e^0 + c_1 * e^{i\omega*1} + c_2 * e^{i\omega*2} + \ldots + c_{N-1} * e^{i\omega*(N-1)}$$

$$f(2) = c_0 * e^{i2\omega*0} + c_1 * e^{i2\omega*1} + c_2 * e^{i2\omega*2} + \ldots + c_{N-1} * e^{i2\omega*(N-1)}$$

$$f(N-1) = c_0 * e^{i(N-1)\omega*0} + c_1 * e^{i(N-1)\omega*1} + c_2 * e^{i(N-1)\omega*2} + \ldots + c_{N-1} * e^{i(N-1)\omega*(N-1)}$$

2D Fourier Series Decomposition

$$f(x) = \sum_{m=0}^{N-1} \boxed{c_m} * e^{i*\omega*m*x} \qquad N = 2P + 1$$

$$T = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ e^{i\omega*0} & e^{i\omega*1} & e^{i\omega*2} & \cdots & e^{i\omega*(N-1)} \\ e^{i2\omega*0} & e^{i2\omega*1} & e^{i2\omega*2} & \cdots & e^{i2\omega*(N-1)} \\ \cdot & & & & \\ \cdot & & & & \\ \cdot & & & & \\ e^{i(N-1)\omega*0} & e^{i(N-1)\omega*1} & e^{i(N-1)\omega*2} & \cdots & e^{i(N-1)\omega*(N-1)} \end{bmatrix} \in R^{N\times N}$$

c_m和像素f(n)的关系： $$T * \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \cdots \\ c_{N-1} \end{bmatrix} = \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ \cdots \\ f(N-1) \end{bmatrix} \Rightarrow C = T^{-1} * f$$

2D Fourier Series Decomposition

$$f(x) = \sum_{m=0}^{N-1} \boxed{c_m} * e^{i*\omega*m*x}$$

$$N = 2P + 1$$

$$\omega = \frac{2\pi}{T}$$

$$T^{-1} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ e^{-i\omega*0} & e^{-i\omega*1} & e^{-i\omega*2} & \dots & e^{-i\omega*(N-1)} \\ e^{-i2\omega*0} & e^{-i2\omega*1} & e^{-i2\omega*2} & \dots & e^{-i2\omega*(N-1)} \\ \vdots & & & & \\ e^{-i(N-1)\omega*0} & e^{-i(N-1)\omega*1} & e^{-i(N-1)\omega*2} & \dots & e^{-i(N-1)\omega*(N-1)} \end{bmatrix} \in R^{N \times N}$$

$$C = T^{-1} * f$$

$$\boxed{F(x) = c_n = \frac{1}{N} \sum_{n=0}^{N-1} f(n) e^{-in\frac{2\pi}{N}x}}$$

2D Fourier Series Decomposition

1D

$$\boxed{F(x) = c_n = \frac{1}{N}\sum_{n=0}^{N-1} f(n)e^{-in\frac{2\pi}{N}x}}$$

2D

$$\boxed{F(x,y) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)e^{-in\frac{2\pi}{N}x}e^{-im\frac{2\pi}{M}y}}$$

2D Fourier Series Decomposition

- 频谱图的理解
  - 频谱图里每一个像素上的值就是c_n的值

$$\boxed{F(x) = c_n = \frac{1}{N}\sum_{n=0}^{N-1} f(n)e^{-in\frac{2\pi}{N}x}}$$

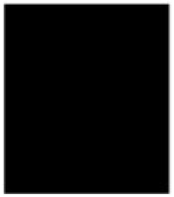
$$c_n = \frac{a_n - ib_n}{2} \qquad ||c_n|| = \sqrt{a_n^2 + b_n^2}$$

  - 用来代表这个频率波的振幅大小

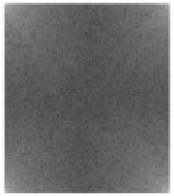
2D Fourier Series Decomposition

- 频谱图的理解
  - 当n=1,2,...N时，c_n和$c_{\{N-n\}}$共轭对称
  - 对于像素f(p,q), $\omega_1 = \frac{2\pi * p}{N}$，$\omega_2 = \frac{2\pi * q}{M}$
  - 越靠近坐标角落，频率越低
  - 每个点的值也可以反映某个频率的波的振幅
  - 便于观察，互换角落和图片中心位置的数值

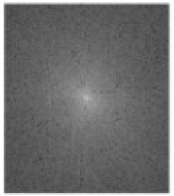
原图



DFT变换后的值



DFT映射到可显示空间的值



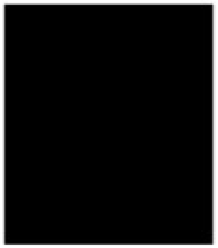
把四个角上的值转移到图片中心

2D Fourier Series Decomposition

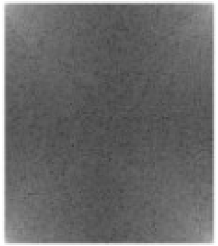
$$c_n = \frac{a_n - ib_n}{2} \qquad ||c_n|| = \sqrt{a_n^2 + b_n^2}$$

- 频谱图的理解
  - 用来代表这个频率波的振幅大小
  - 一个N x 1的信号，其由$\frac{N-1}{2}$个频率的波叠加，计算所得数据左右上下对称
  - 当n=1,2,...N时，c_n和$c_{\{N-n\}}$共轭对称

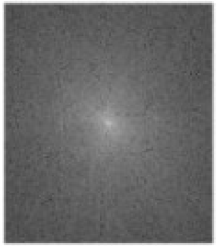
原图



DFT变换后的值

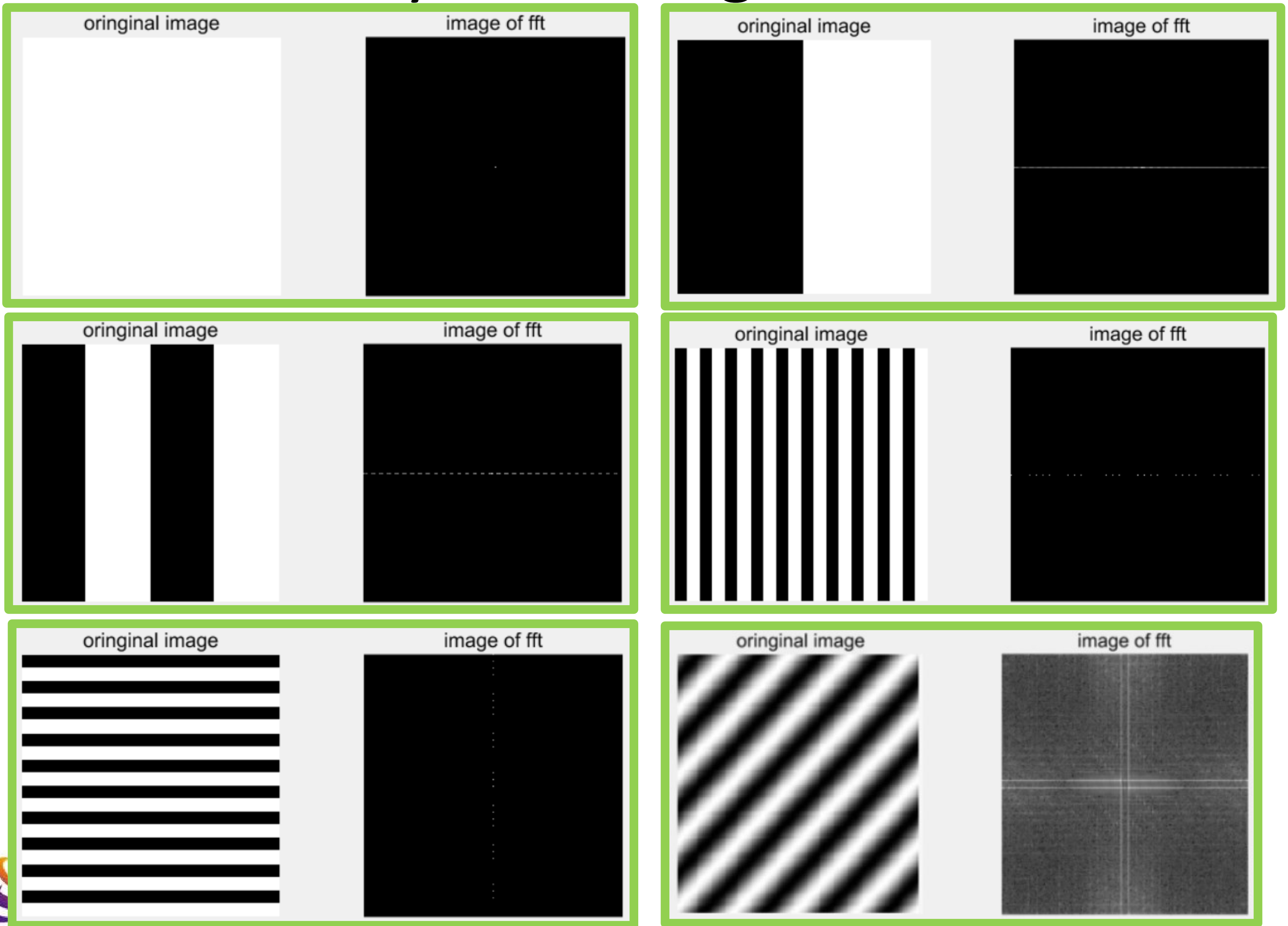


DFT映射到可显示空间的值



把四个角上的值转移到图片中心

Fourier analysis in images

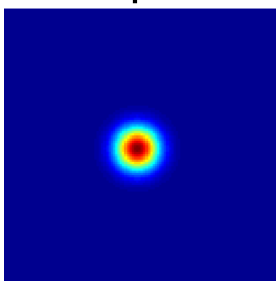


Think-Pair-Share

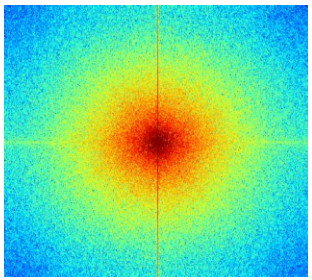
1 - D
2 – B
3 – A
4 - E
5 - C

Match the spatial domain image to the Fourier magnitude image

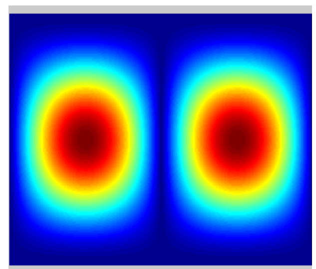
1



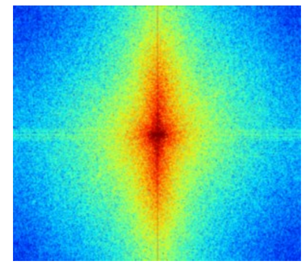
2



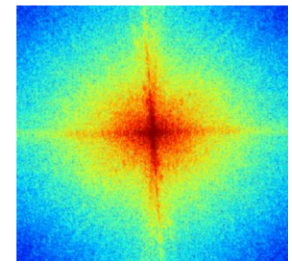
3



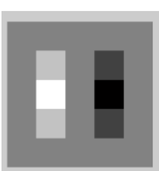
4



5

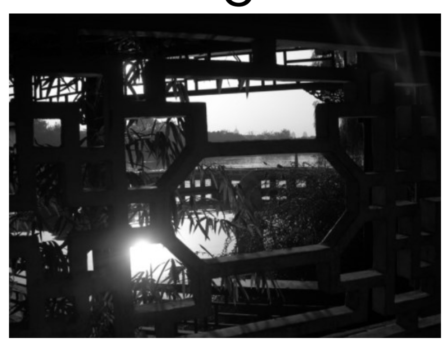


A

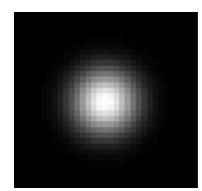


B

C



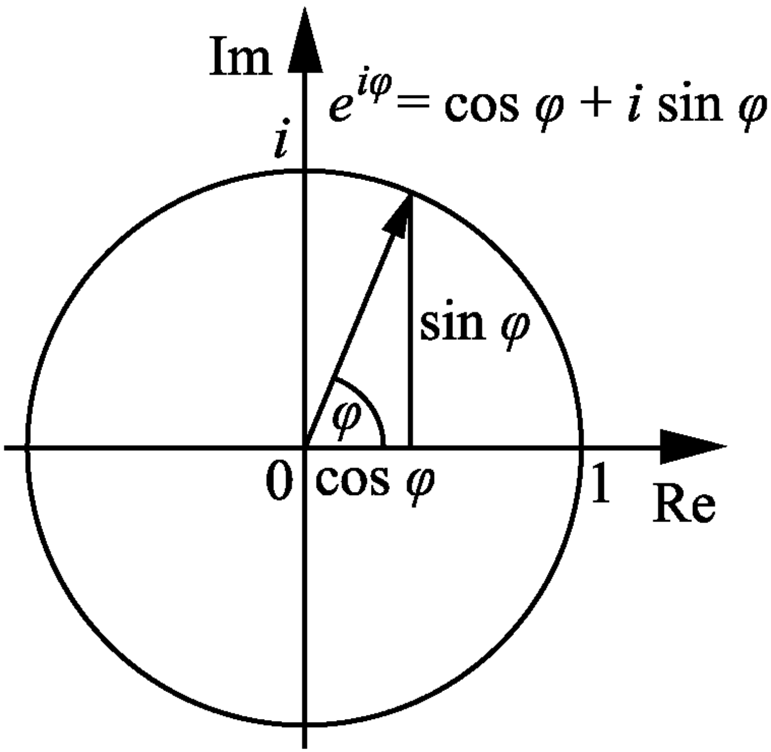
D



E

Fourier Transform

- Stores the amplitude and phase at each frequency:
  - For mathematical convenience, this is often notated in terms of real and complex numbers
  - Related by Euler's formula



Amplitude encodes how much signal there is at a particular frequency:

$$A = \pm\sqrt{\mathrm{Re}(\varphi)^2 + \mathrm{Im}(\varphi)^2}$$

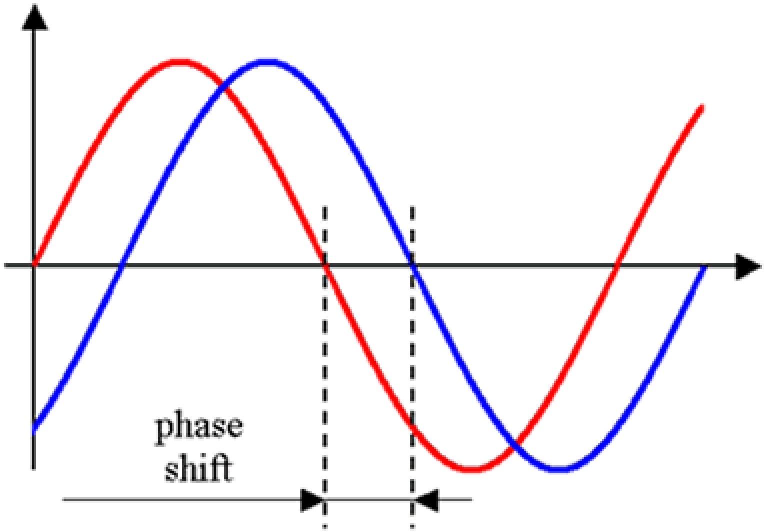
Phase encodes spatial information (indirectly):

$$\phi = \tan^{-1}\frac{\mathrm{Im}(\varphi)}{\mathrm{Re}(\varphi)}$$

未来媒体研究中心 CENTER FOR FUTURE MEDIA | 电子科技大学 University of Electronic Science and Technology of China

Hays

Amplitude / Phase



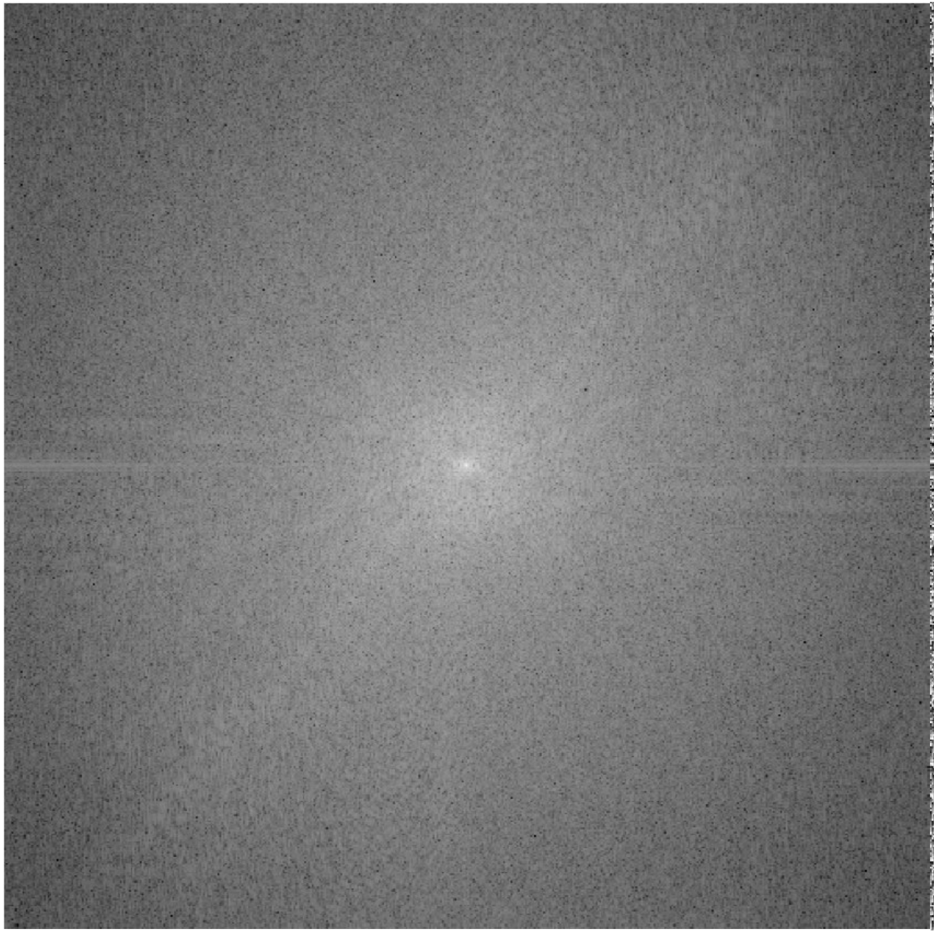
- Amplitude tells you “how much”
- Phase tells you “where”

- Translate the image?
  - Amplitude unchanged
  - Adds a constant to the phase.

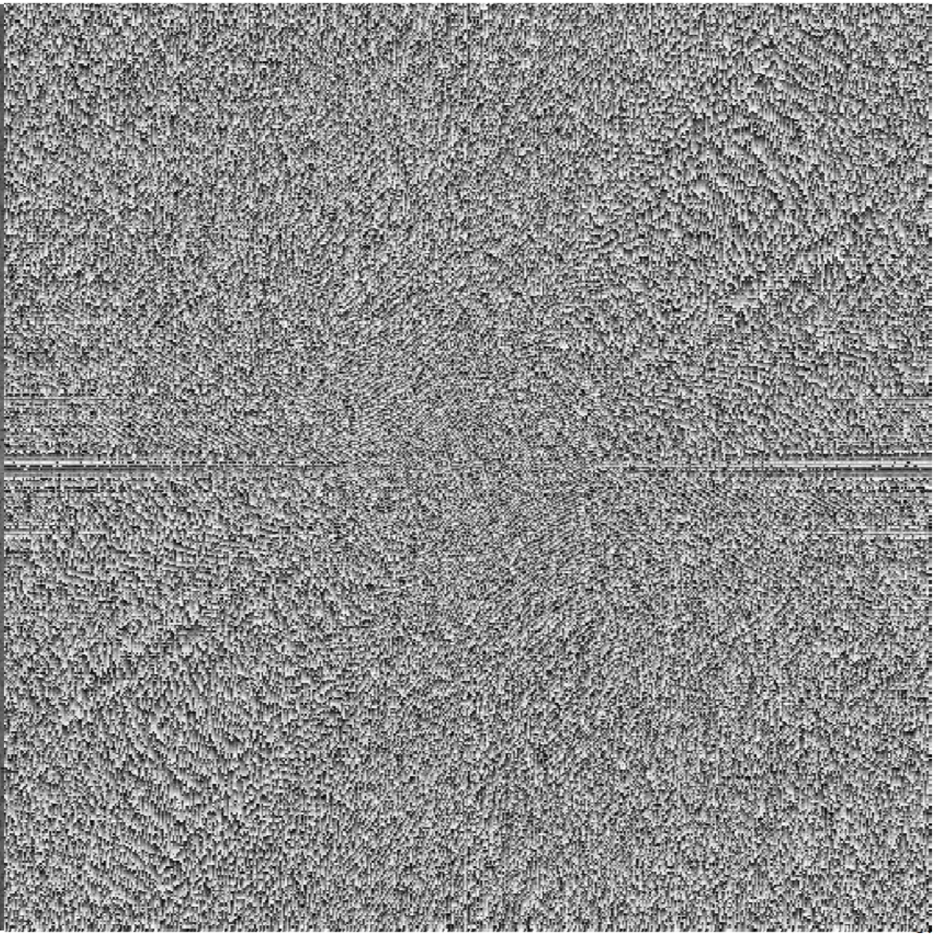
What about phase?

What about phase?

Amplitude



Phase

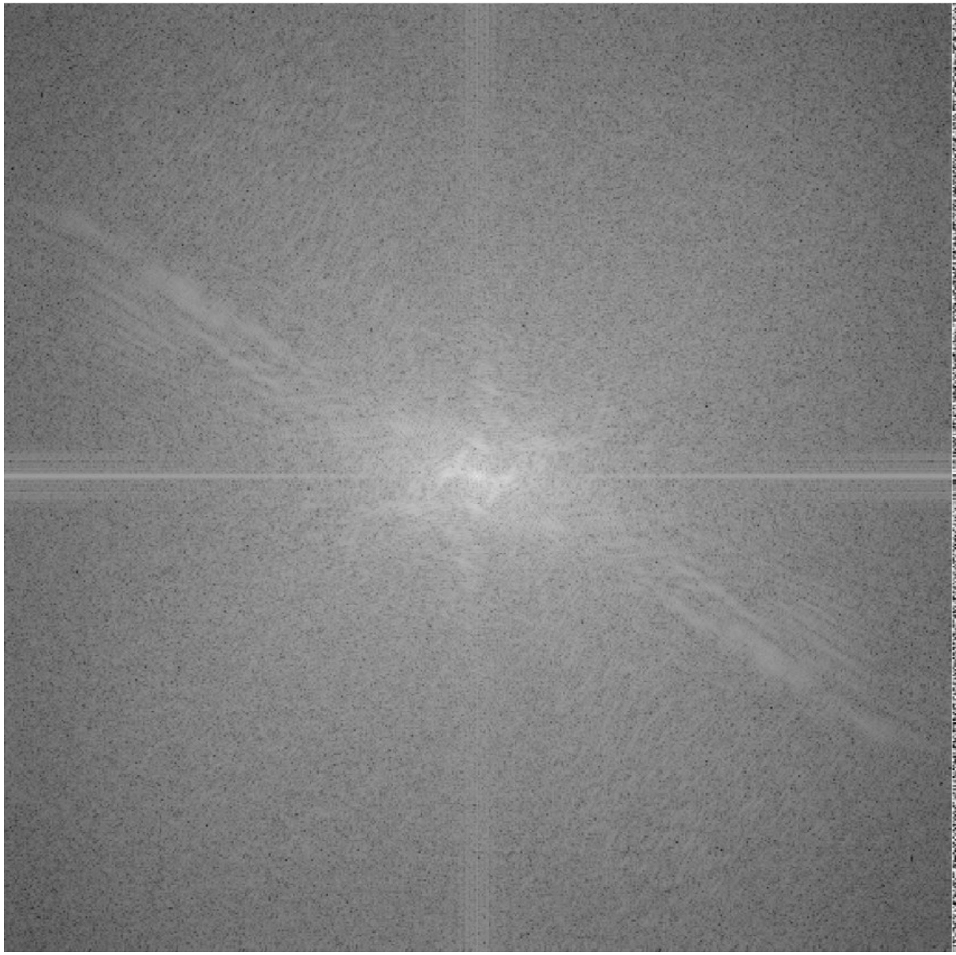


What about phase?

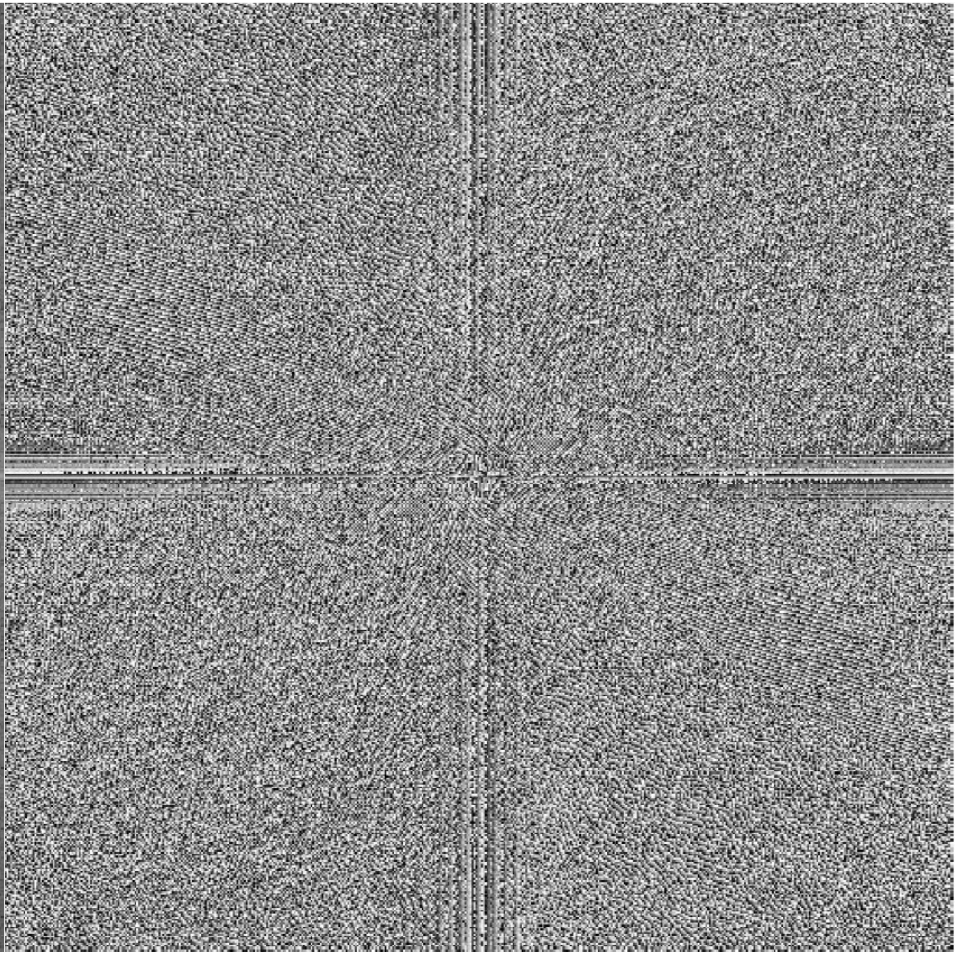
Efros

What about phase?

Amplitude



Phase



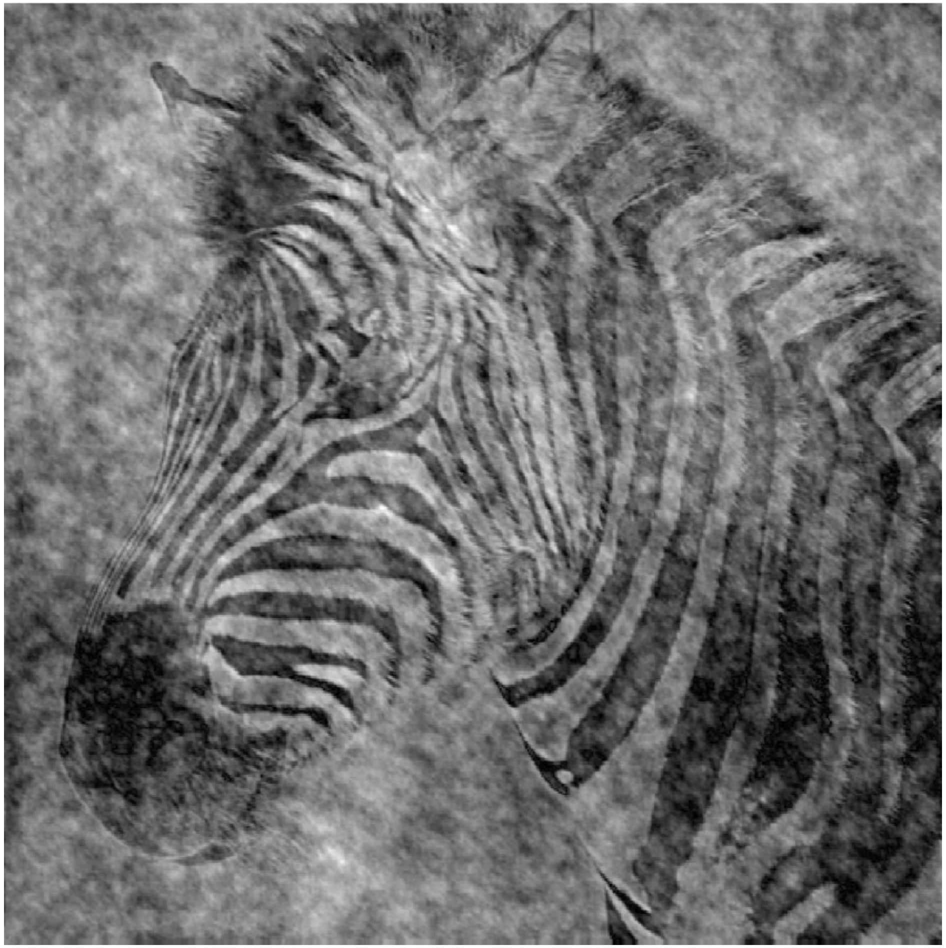
Efros

Think-Pair-Share

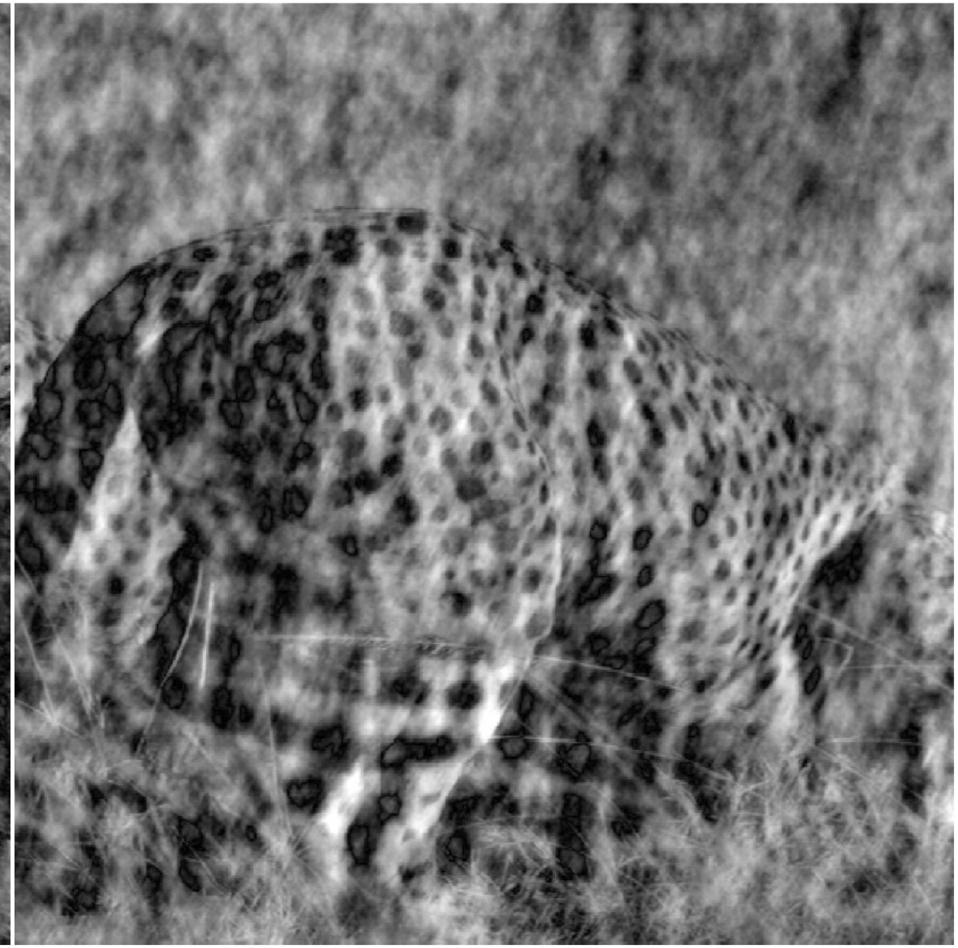
- In Fourier space, where is more of the information that we see in the visual world?
  - Amplitude
  - Phase

Cheebra

Zebra phase, cheetah amplitude



Cheetah phase, zebra amplitude



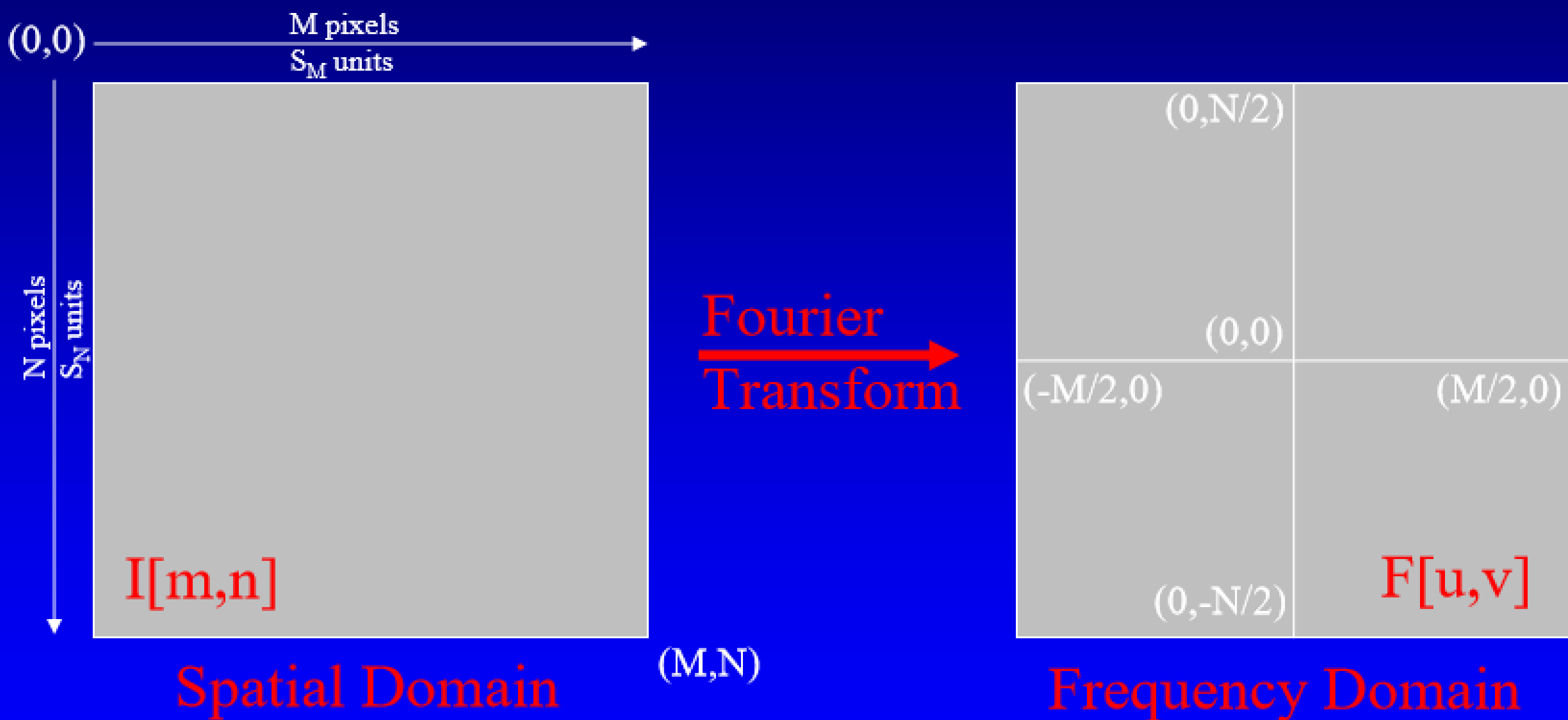
Efros

- The frequency amplitude of natural images are quite similar
  - Heavy in low frequencies, falling off in high frequencies
  - Will *any* image be like that, or is it a property of the world we live in?
- Most information in the image is carried in the phase, not the amplitude
  - Not quite clear why

Efros

2D Discrete Fourier Transform

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

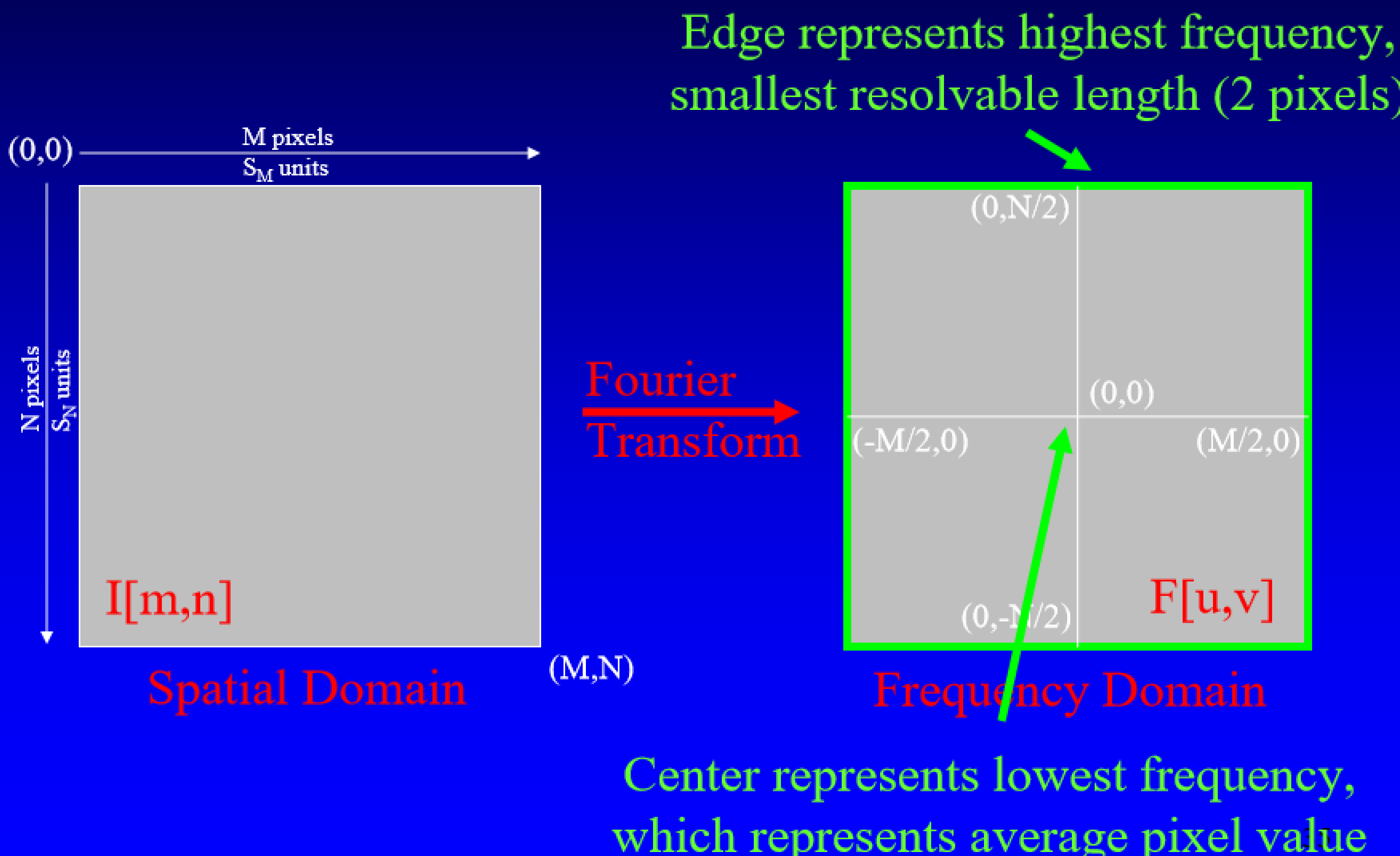


Source: Seul et al, *Practical Algorithms for Image Analysis*, 2000, p. 249, 262.

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

Earl F. Glynn

2D Discrete Fourier Transform



Earl F. Glynn

The Convolution Theorem

- The Fourier transform of the convolution of two functions is the product of their Fourier transforms

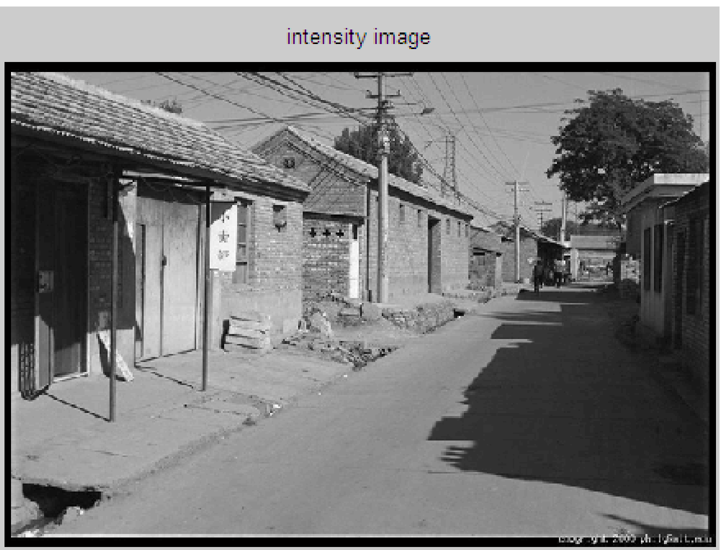
$$\mathrm{F}[g * h] = \mathrm{F}[g]\,\mathrm{F}[h]$$

- **Convolution** in spatial domain is equivalent to **multiplication** in frequency domain!

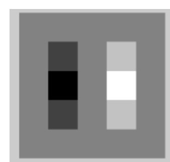
$$g * h = \mathrm{F}^{-1}[\mathrm{F}[g]\,\mathrm{F}[h]]$$

Filtering in spatial domain

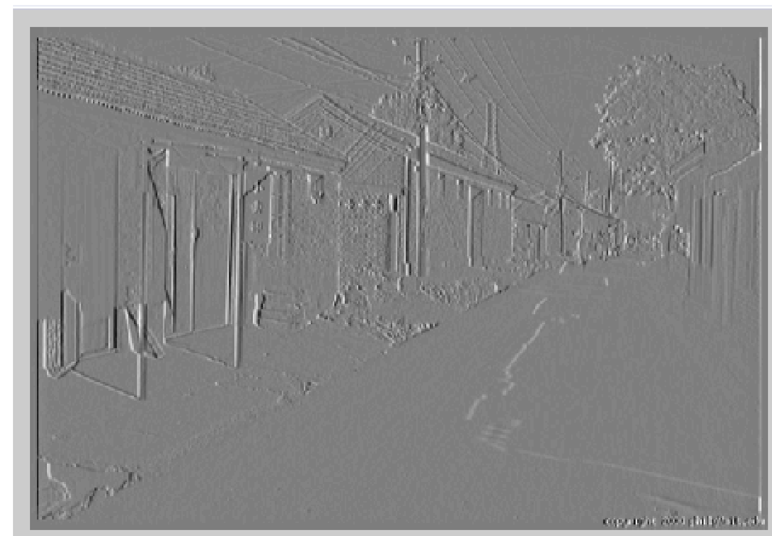
1	0	-1
2	0	-2
1	0	-1



*

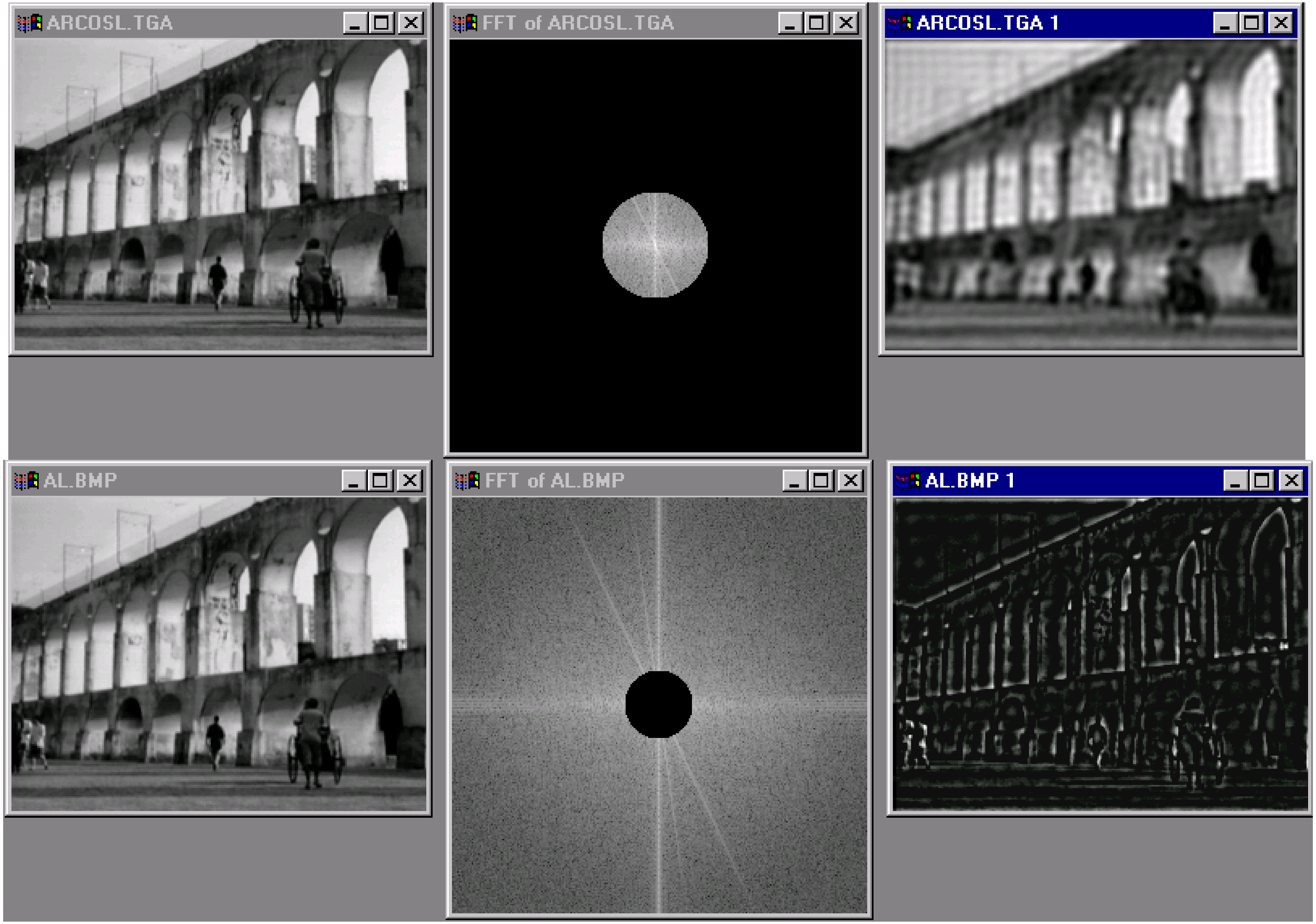


=

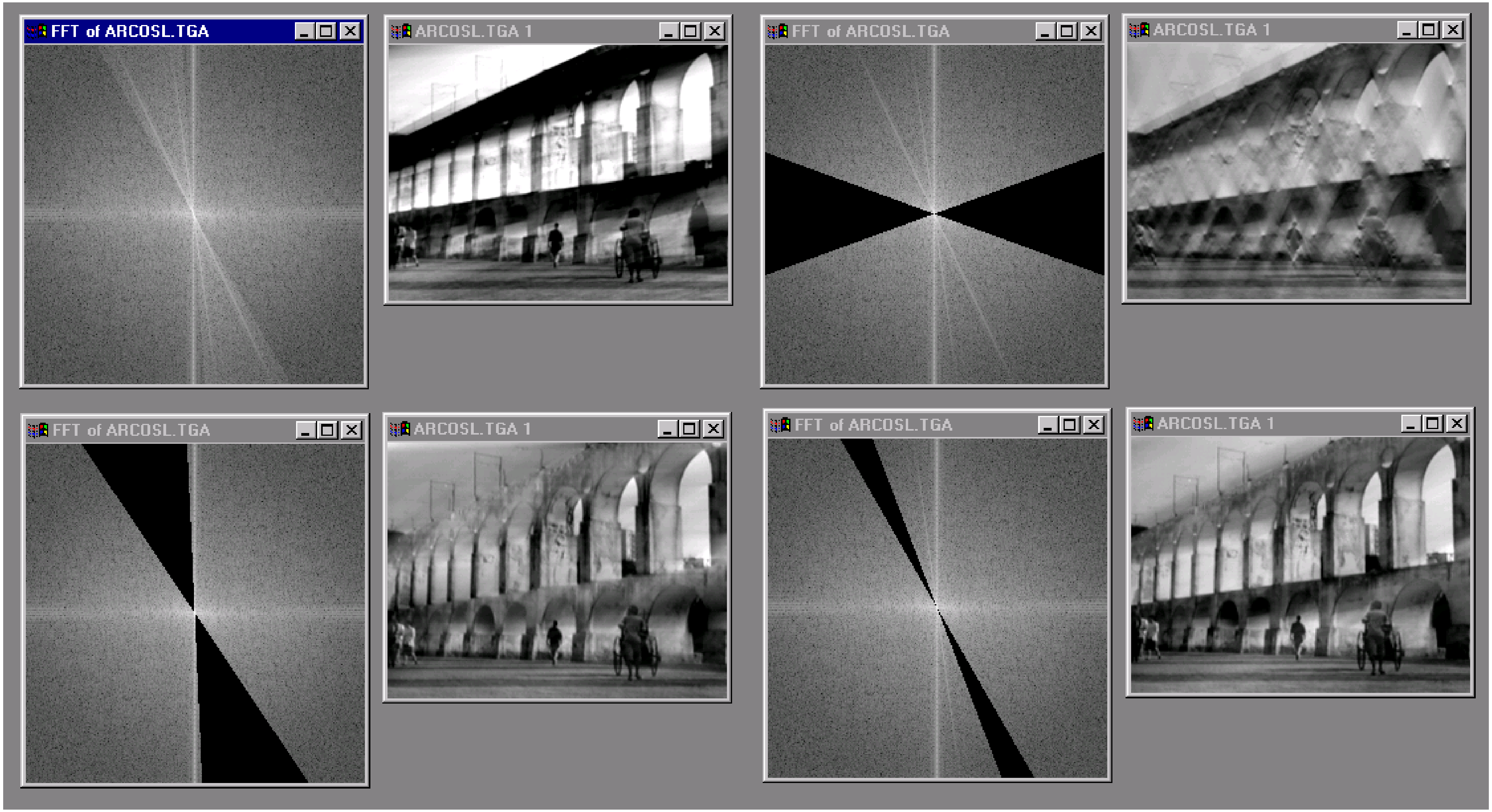


Hays

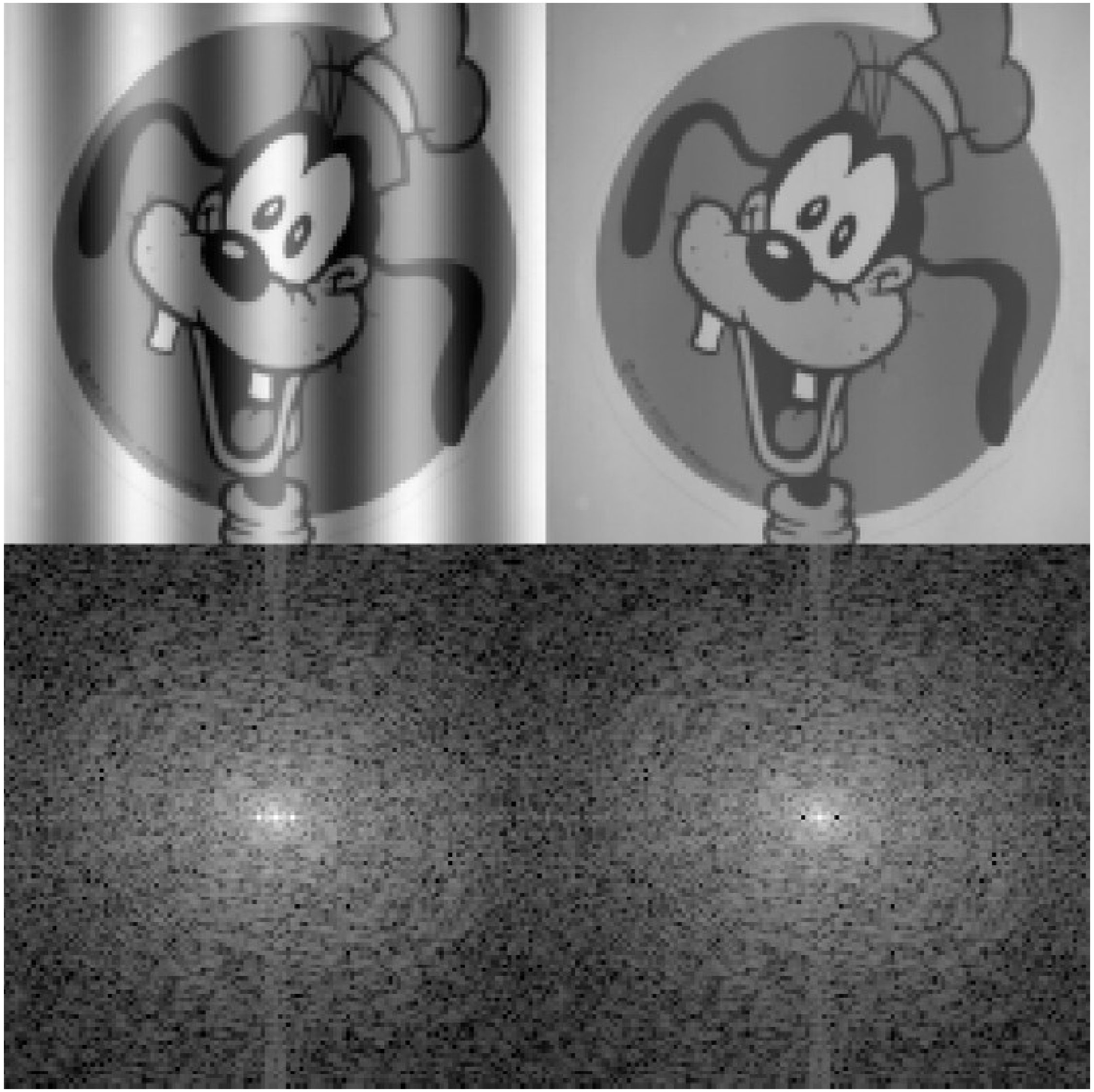
Low and High Pass filtering



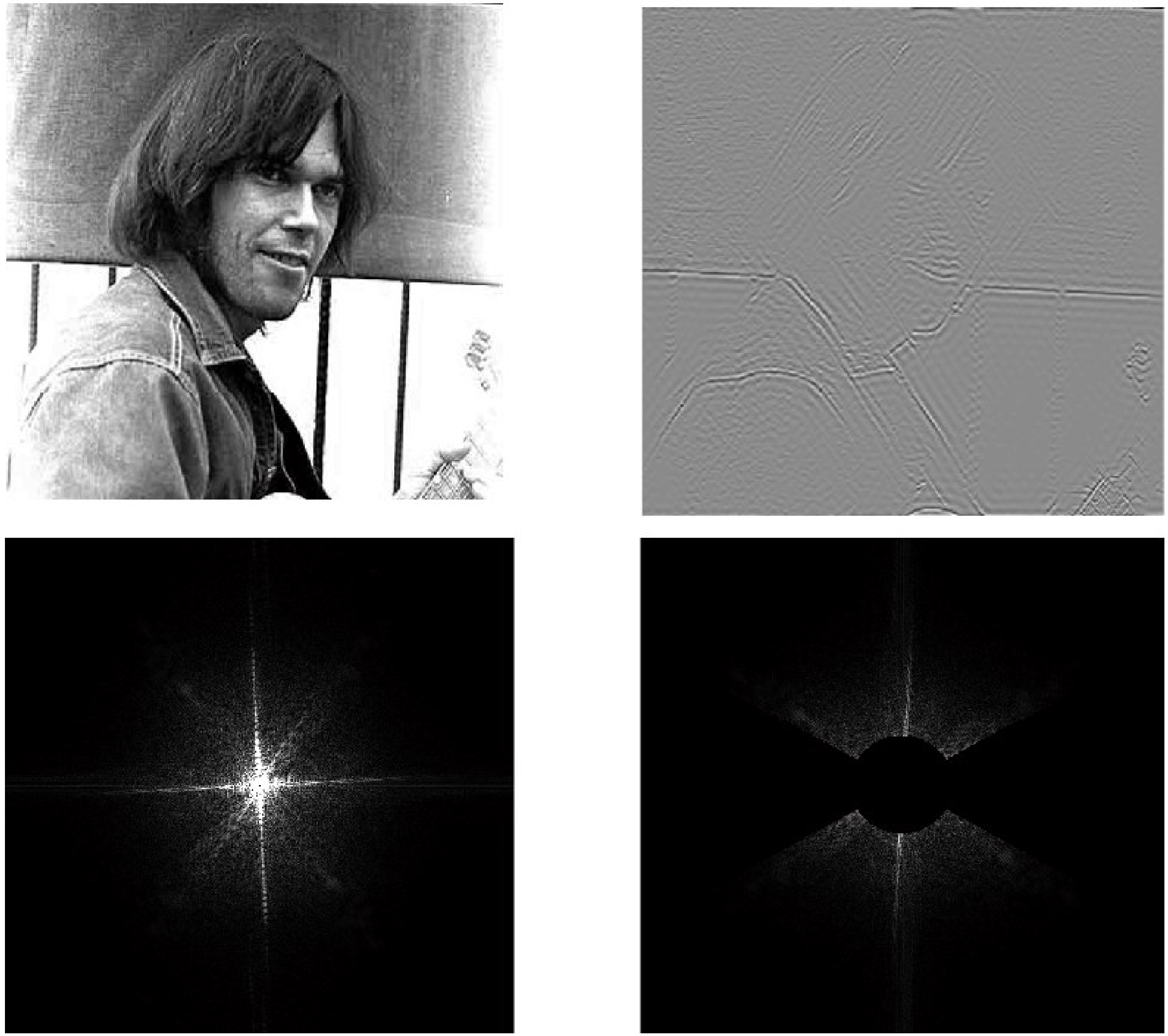
Now we can edit frequencies!



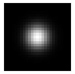
Removing frequency bands

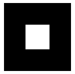


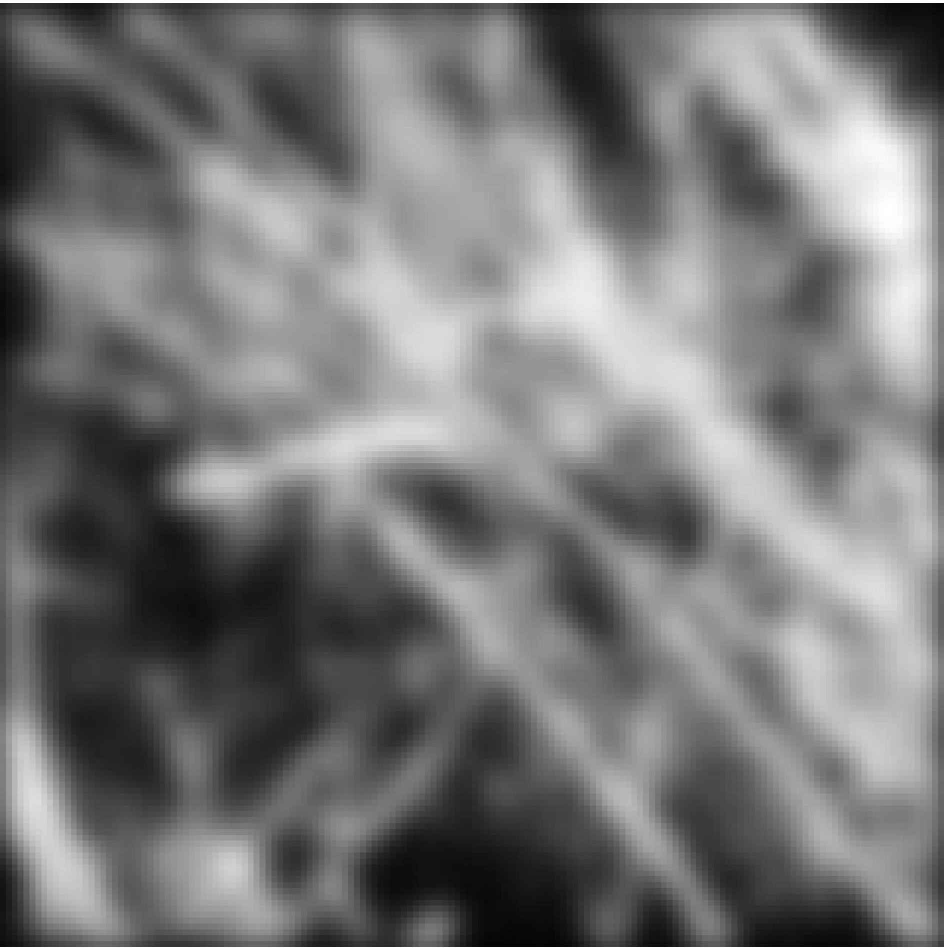
High pass filtering + orientation

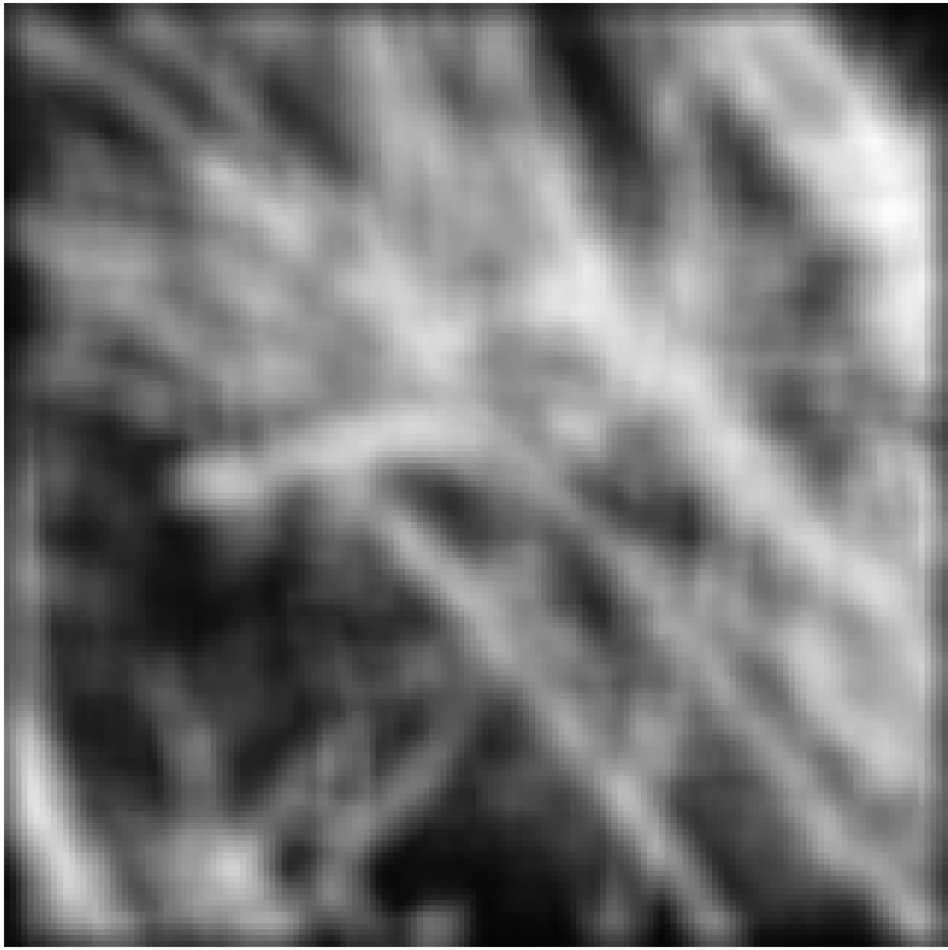


Why does the Gaussian filter give a nice smooth image, but the square filter give edgy artifacts?

Gaussian 

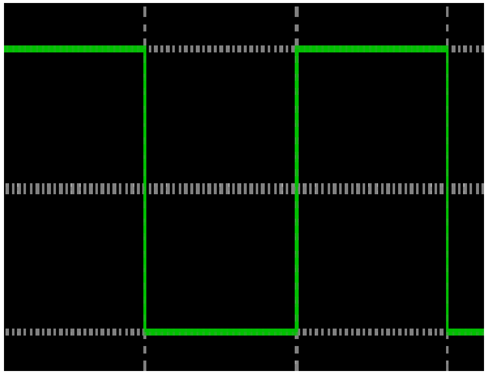
Box filter 



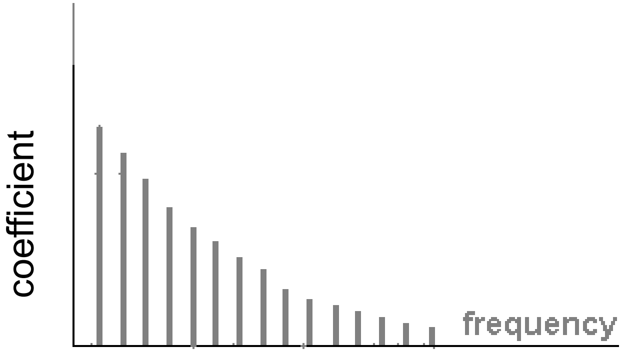


Why do we have those lines in the image?

- Sharp edges in the image need _all_ frequencies to represent them.



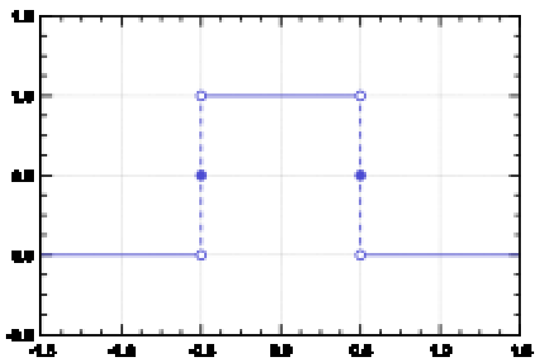
$$= A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)$$



Box filter / sinc （辛克） filter duality

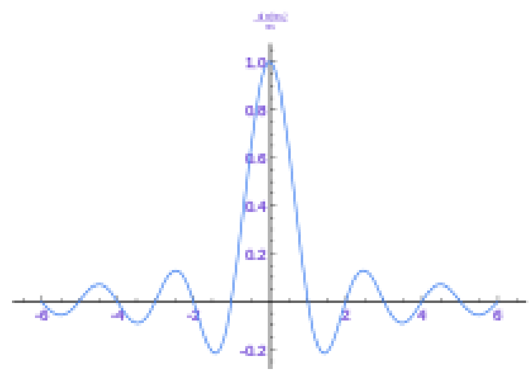
- What is the spatial representation of the hard cutoff (box) in the frequency domain?
- http://madebyevan.com/dft/

Box filter



Sinc filter

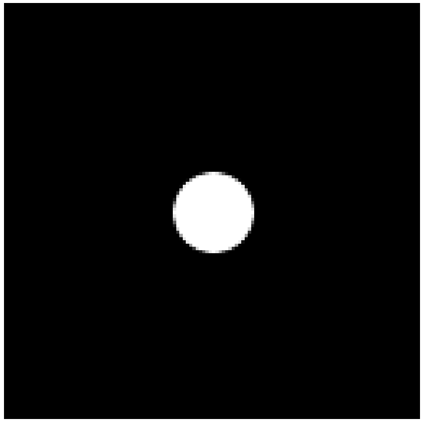
sinc(x) = sin(x) / x



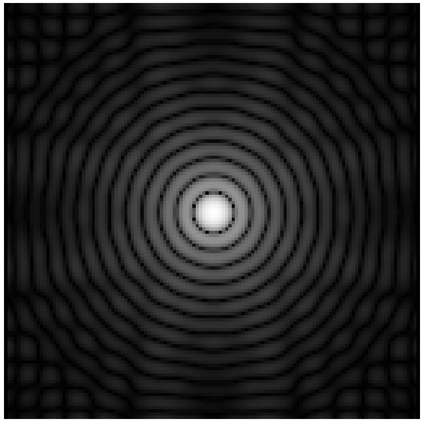
Spatial Domain ⟺ Frequency Domain

Frequency Domain ⟺ Spatial Domain

Box filter (spatial)



Frequency domain magnitude

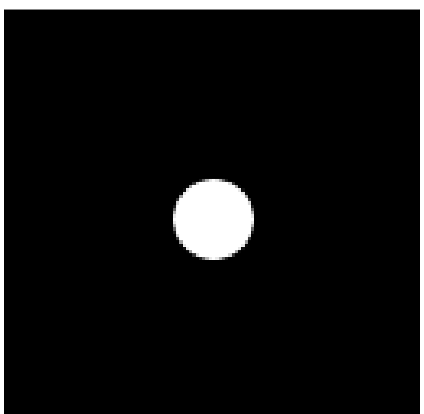


Gaussian filter duality

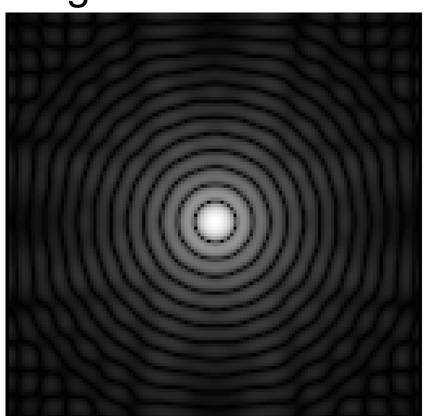
- Fourier transform of one Gaussian…

  …is *another Gaussian (with inverse variance).*
- Why is this useful?
  - Smooth degradation in frequency components
  - No sharp cut-off
  - No negative values
  - Never zero (infinite extent)

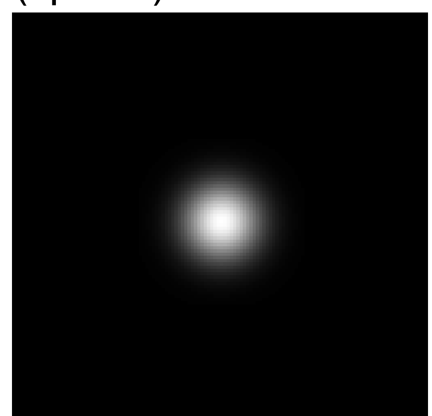
Box filter (spatial)



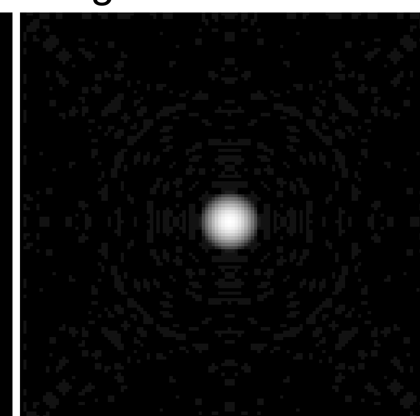
Frequency domain magnitude



Gaussian filter (spatial)



Frequency domain magnitude



Deconvolution is hard.

- Active research area.
- Even if you know the filter (non-blind deconvolution), it is still hard and requires strong *regularization* to counteract noise.
- If you don’t know the filter (blind deconvolution), then it is harder still.

A few questions…

If we have infinite frequencies, why does the image end?

- Sampling theory. Frequencies higher than Nyquist frequency end up falling on an existing sample.
  - i.e., they are 'aliases' for existing samples!
- Nyquist frequency is half the sampling frequency.

A few questions

Why is frequency decomposition centered in middle, and duplicated and rotated?

- From Euler:
  - $\cos(x) + i\sin(x) = e^{ix}$
  - $\cos(\omega t) = \frac{1}{2}\left(e^{-i\omega t} + e^{+i\omega t}\right)$
- Coefficients for negative frequencies (i.e., 'backwards traveling' waves)

- FFT of a real signal is conjugate symmetric
  - i.e., f(-x) = f*(x)

A few questions

How is the Fourier decomposition computed?

Intuitively, by correlating the signal with a set of waves of increasing signal!

Notes in hidden slides.

Plus: http://research.stowers-institute.org/efg/Report/FourierAnalysis.pdf