Modifying Images and Sounds using the Mathematics of Fourier Transforms

Colin Macdonald

www.math.sfu.ca/∼cbm

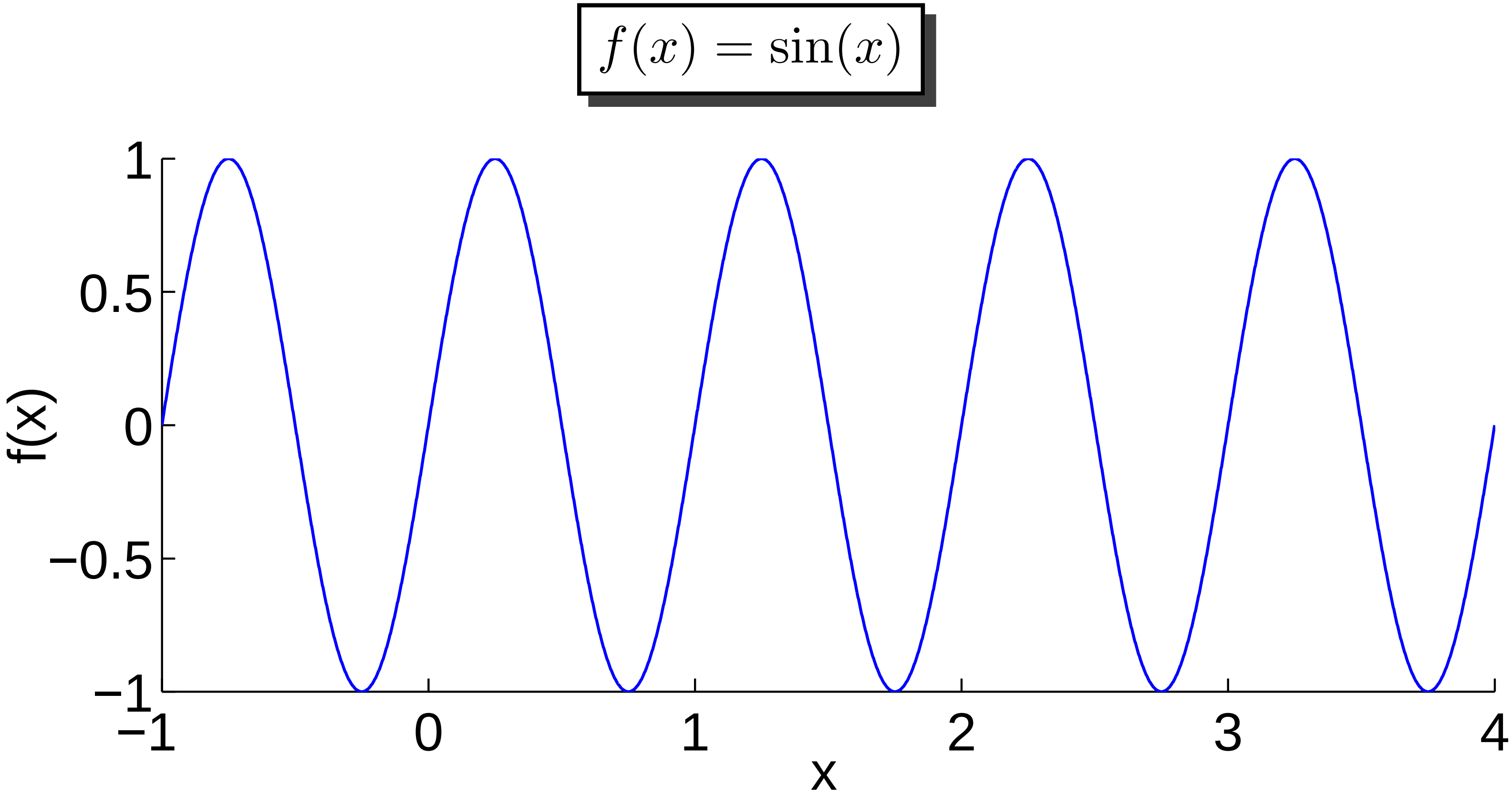
Mathematics Department

Simon Fraser University

Burnaby, BC

Image processing demo code is based on code by Dr. Dave Muraki

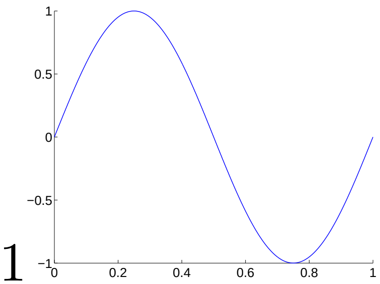
Sine functions

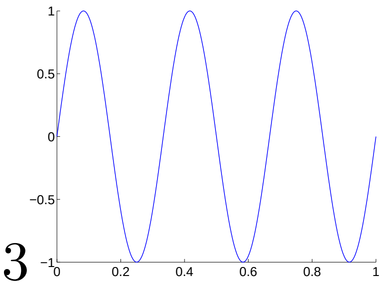


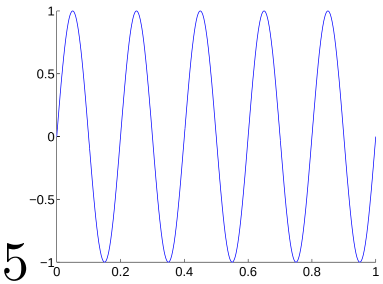
Sine functions

$$\boxed{f(x) = a\sin(k \cdot 2\pi x)}$$

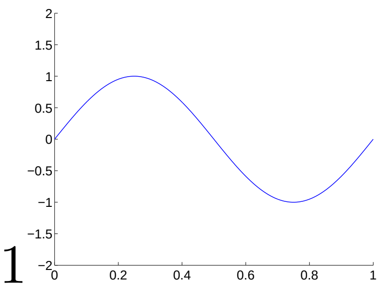
- Increasing the "wavenumber" k:

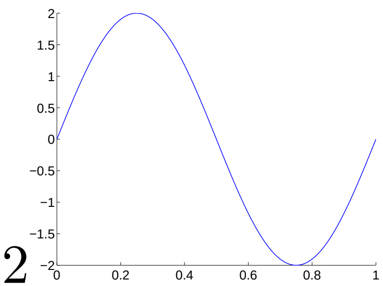
$k = 1$ 

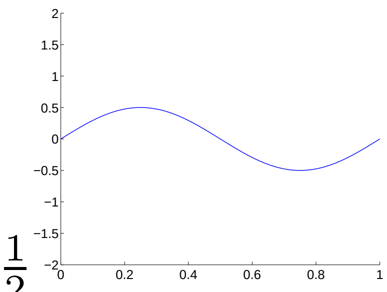
$k = 3$ 

$k = 5$ 

- Changing the amplitude a:

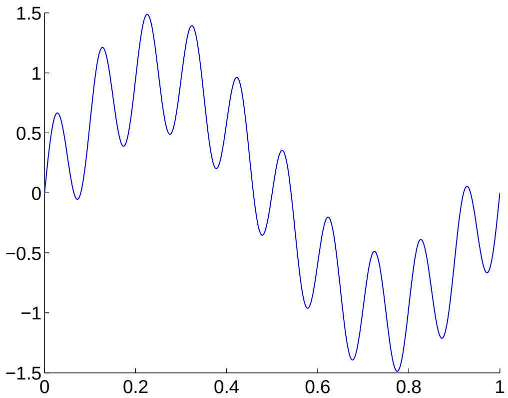
$a = 1$ 

$a = 2$ 

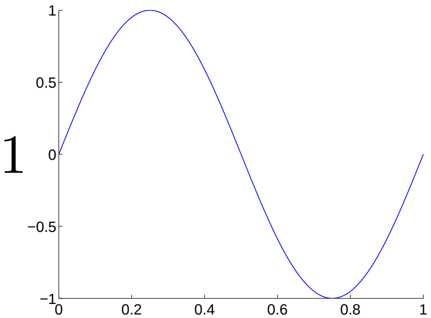
$a = \frac{1}{2}$ 

Decomposing a function

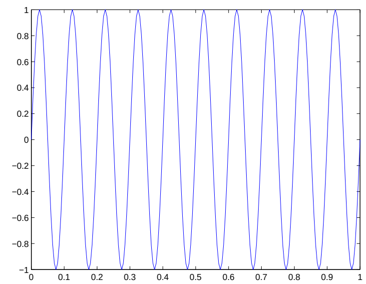
- Can express many functions as a sum of various sine functions:



$= 1$



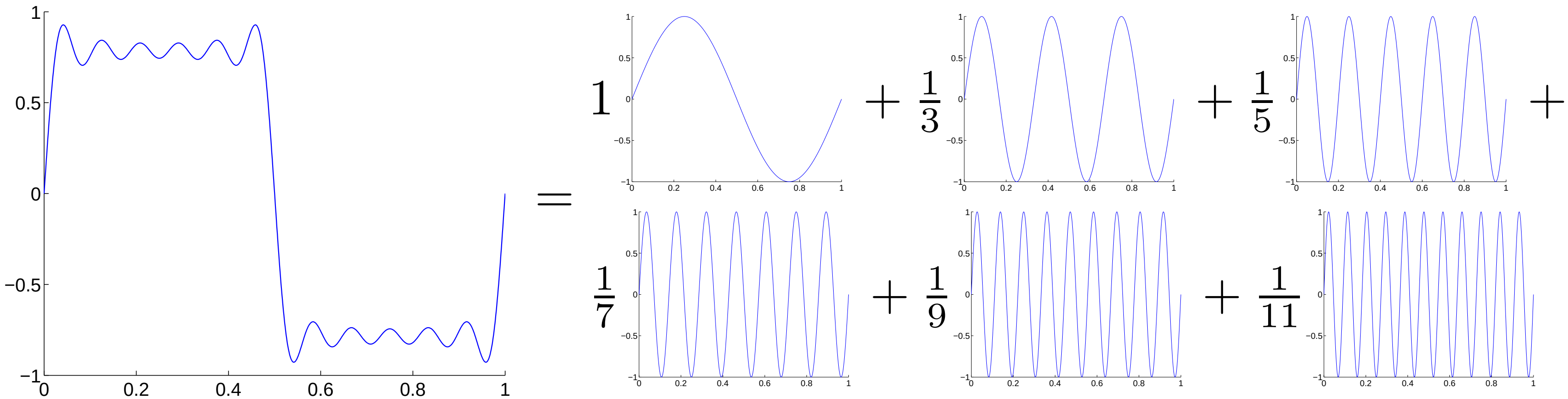
$+ \frac{1}{2}$



- $(k = 1, a = 1)$ and $(k = 10, a = \frac{1}{2})$.

Decomposing a function

- Another example:



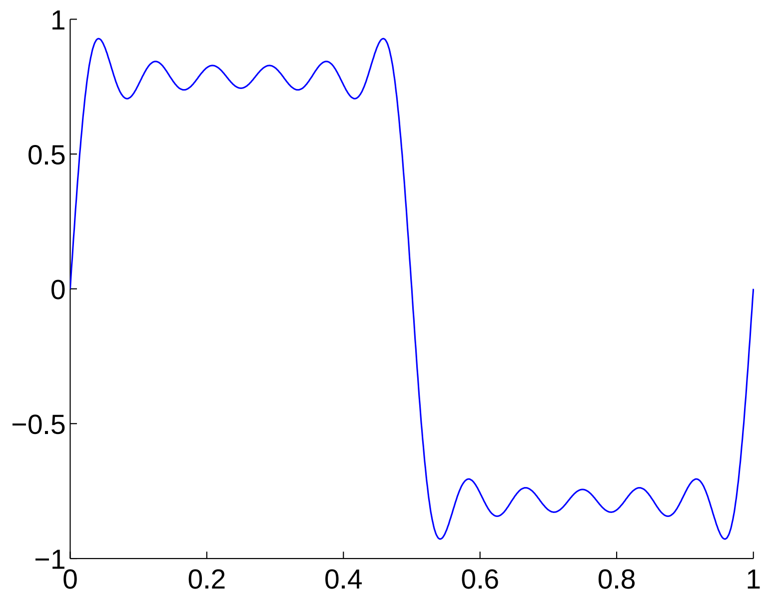
- $(k = 1, a = 1)$,
  $(k = 3, a = \frac{1}{3})$,
  $(k = 5, a = \frac{1}{5})$,
  $(k = 7, a = \frac{1}{7})$,
  $(k = 9, a = \frac{1}{9})$,
  $(k = 11, a = \frac{1}{11})$.

Reconstruction

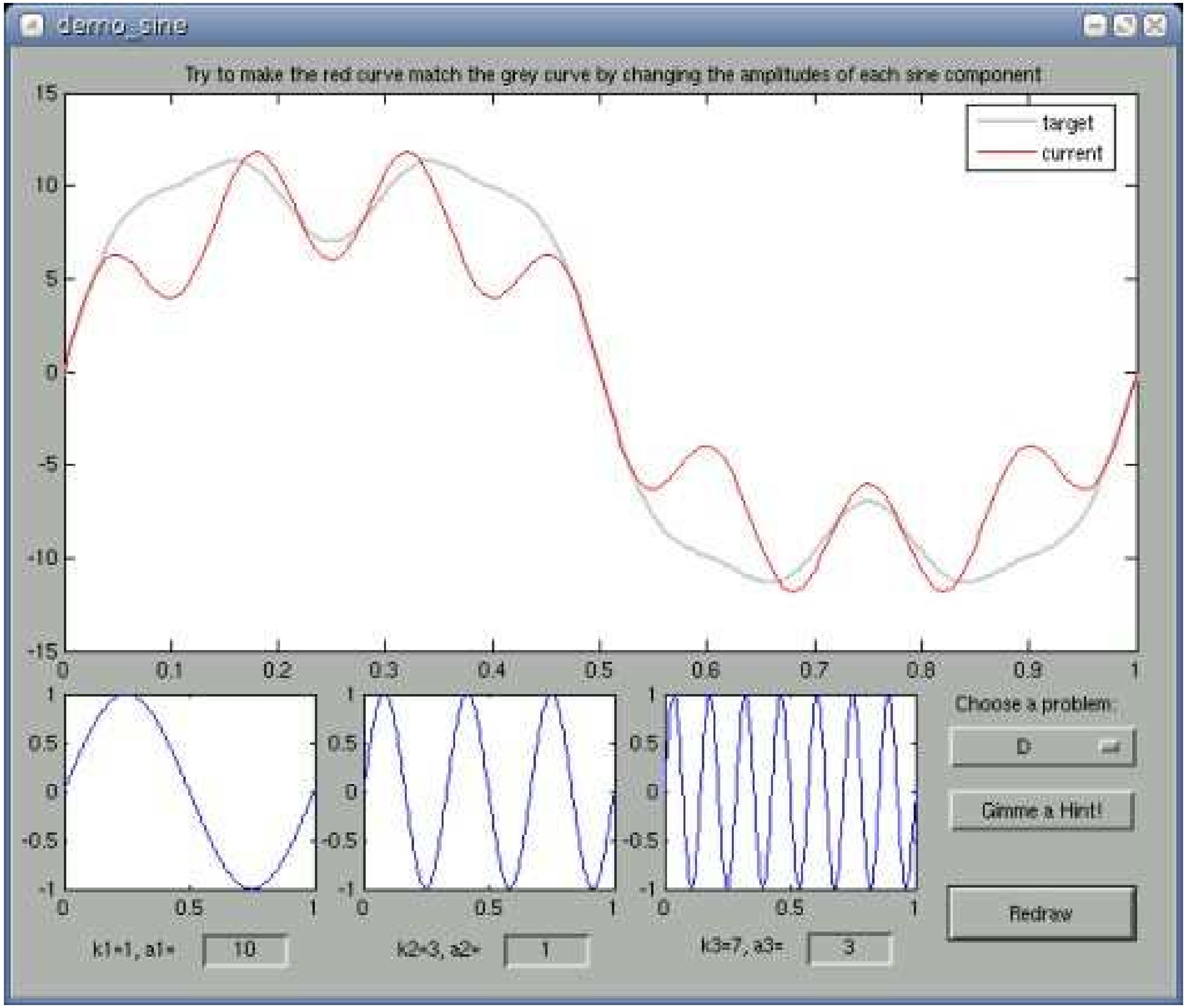
- Give amplitude and frequency information: "$(k = 1, a = 1)$, $(k = 3, a = \frac{1}{3})$, ..."
- Reconstruction:

$$f(x) = (1)\sin(1 \cdot 2\pi x) + \left(\frac{1}{3}\right)\sin(3 \cdot 2\pi x) + \left(\frac{1}{5}\right)\sin(5 \cdot 2\pi x) + \ldots$$

$$\begin{aligned}&(k = 1, a = 1),\\ &(k = 3, a = \tfrac{1}{3}),\\ &(k = 5, a = \tfrac{1}{5}),\\ &(k = 7, a = \tfrac{1}{7}),\\ &(k = 9, a = \tfrac{1}{9}),\\ &(k = 11, a = \tfrac{1}{11})\end{aligned} \quad \Longrightarrow$$



Demo – decomposing into sine waves

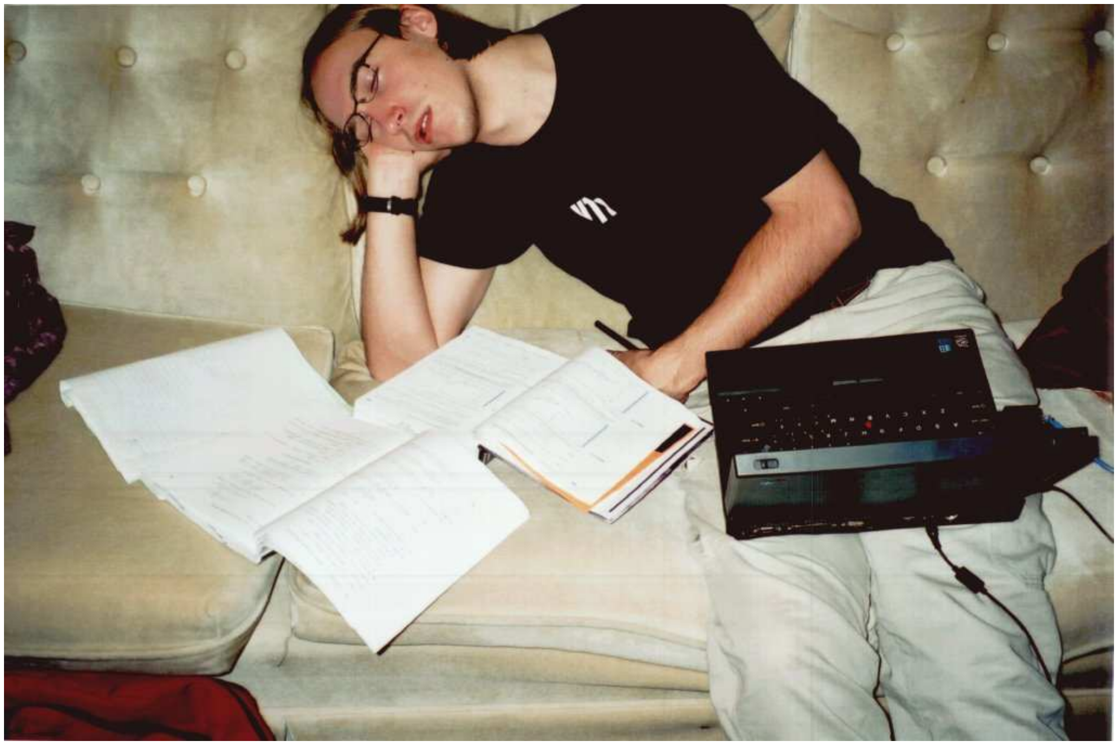


Decomposition/Reconstruction seems *hard*. . .

- Difficult with three wavenumbers. But 1 second of CD audio has 44000 possible wavenumbers!
- Use student labour?

Decomposition/Reconstruction seems *hard*. . .

- Difficult with three wavenumbers. But 1 second of CD audio has 44000 possible wavenumbers!
- Use student labour? Not reliable enough:



- Calculus can provide a solution.

The Fourier transform

- Fourier Transform decomposes $f(x)$ into amplitude and frequency (wavenumber) information.
- Transforms $f(x)$ into $a(k)$.
- Transforms functions of space x into functions of wavenumber k.

The Fourier transform

- Fourier Transform decomposes $f(x)$ into amplitude and frequency (wavenumber) information.
- Transforms $f(x)$ into $a(k)$.
- Transforms functions of space x into functions of wavenumber k.
- Warning! more equations:

$$a(k) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-i2\pi kx} dx$$

- discrete case:

$$a_k = \frac{1}{N} \sum_{n=0}^{N-1} f_n e^{-i2\pi kn/N}$$

- Understanding is more important than equations!

The inverse Fourier transform

- Inverse Fourier Transform recovers $f(x)$ from amplitude and frequency (wavenumber) information.
- Transforms $a(k)$ into $f(x)$.
- Transforms functions of wavenumber k into functions of space x.

The inverse Fourier transform

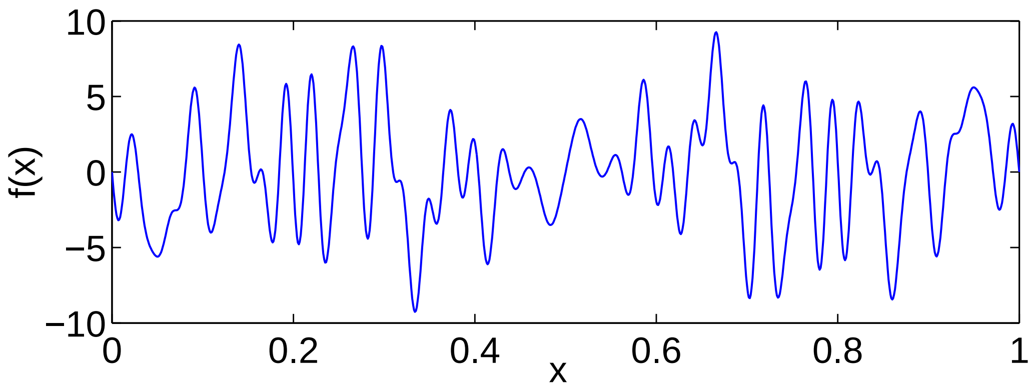
- Inverse Fourier Transform recovers $f(x)$ from amplitude and frequency (wavenumber) information.
- Transforms $a(k)$ into $f(x)$.
- Transforms functions of wavenumber k into functions of space x.
- The gory details:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} a(k) e^{i2\pi kx} dk$$

- discrete case:

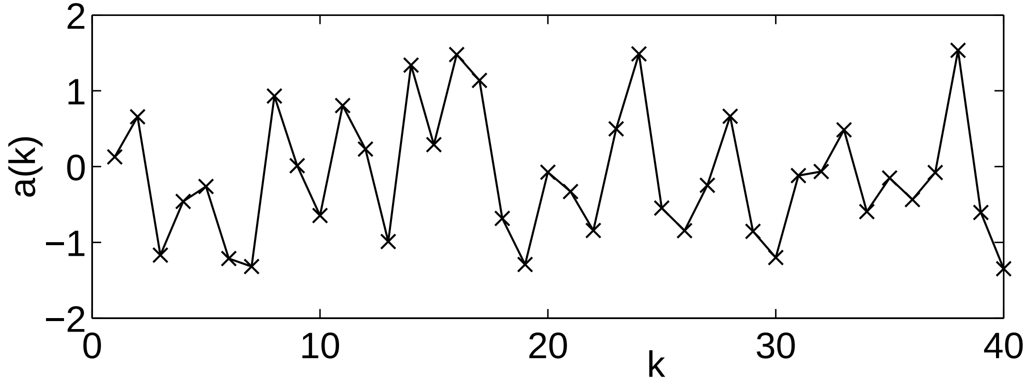
$$f_n = \sum_{n=0}^{N-1} a_k e^{i2\pi kn/N}$$

Fourier transform summary



Fourier
Transform
"Decompose" $\Downarrow$

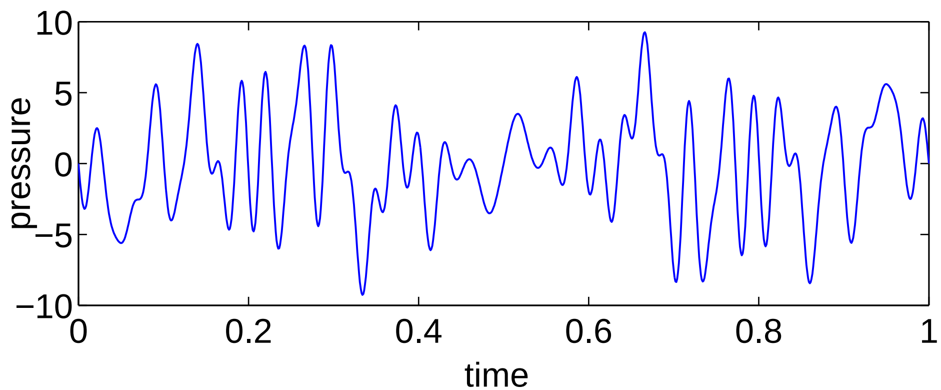
$\Uparrow$ Inverse
Fourier
Transform
"Reconstruct"



- ". . . as if functions circulated at ground level and their transforms in the underworld" [Bracewell, 1965]

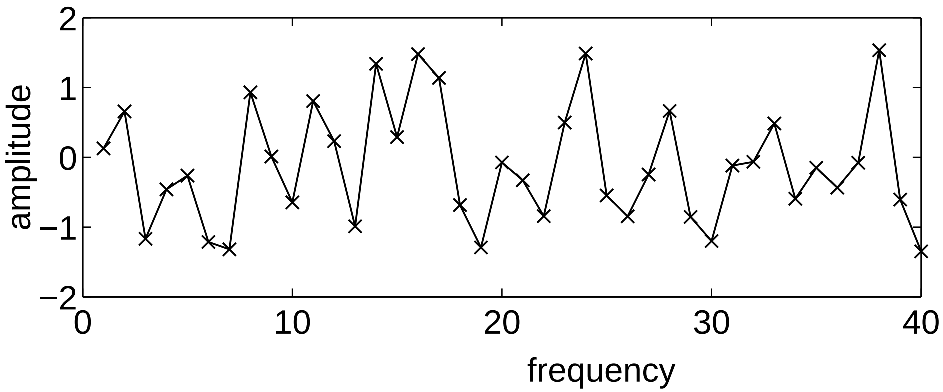
Audio signals and Fourier transforms

- Sound is pressure against your ears changing in time:



Fourier Transform $\Downarrow$

$\Uparrow$ Inverse Fourier Transform



- Time domain $\Longleftrightarrow$ frequency domain.

Fourier transforms for music

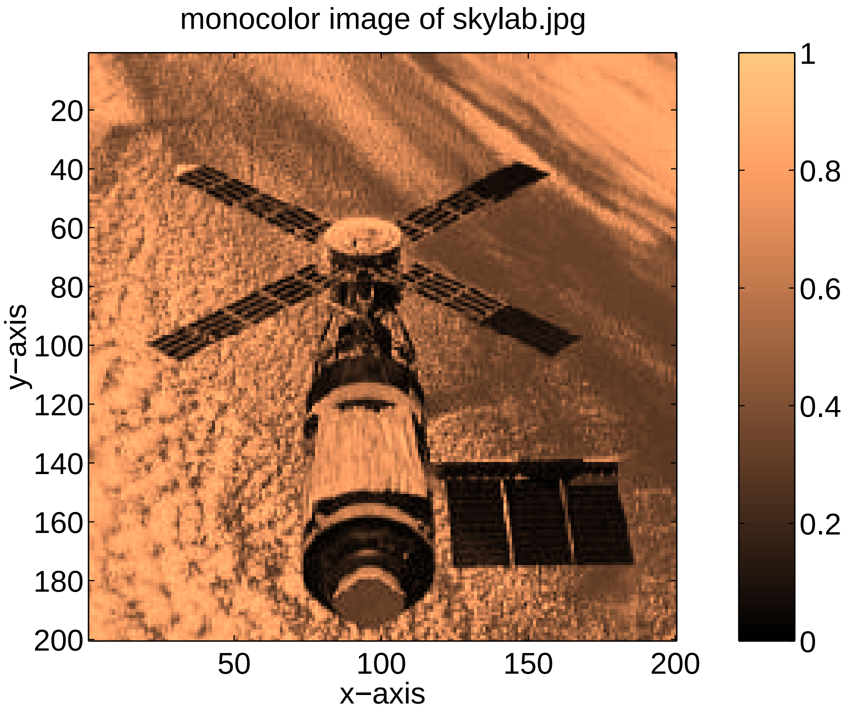
- Visualization:

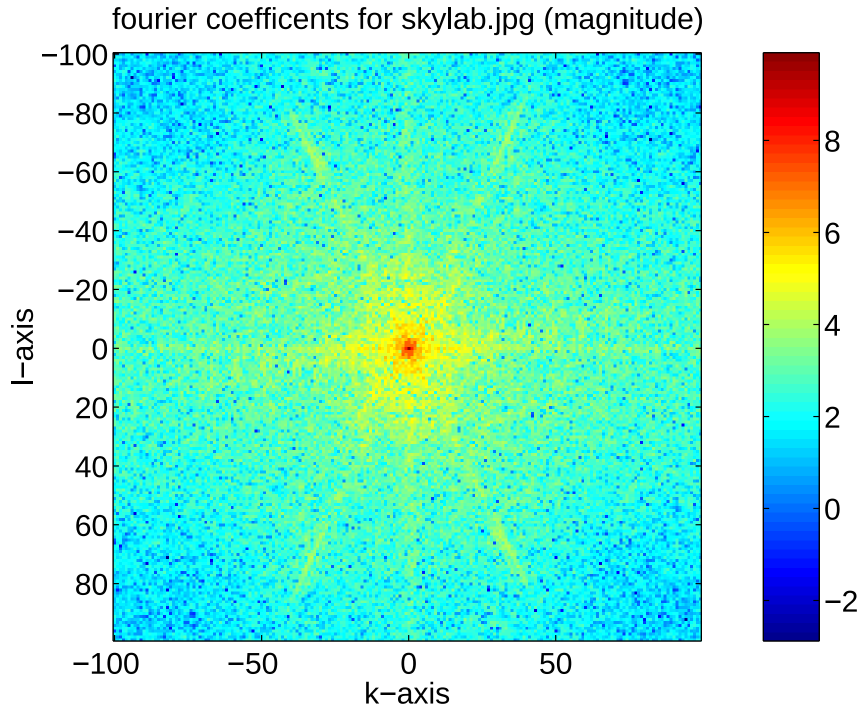
- “Bass Boost”:

$\boxed{\text{music}} \longrightarrow \boxed{\text{F.T.}} \longrightarrow \boxed{\text{Increase “small } k\text{” freq.}} \longrightarrow \boxed{\text{Inv. F.T.}} \longrightarrow \boxed{\text{music}}$

Fourier transform for images

- Images are 2-D

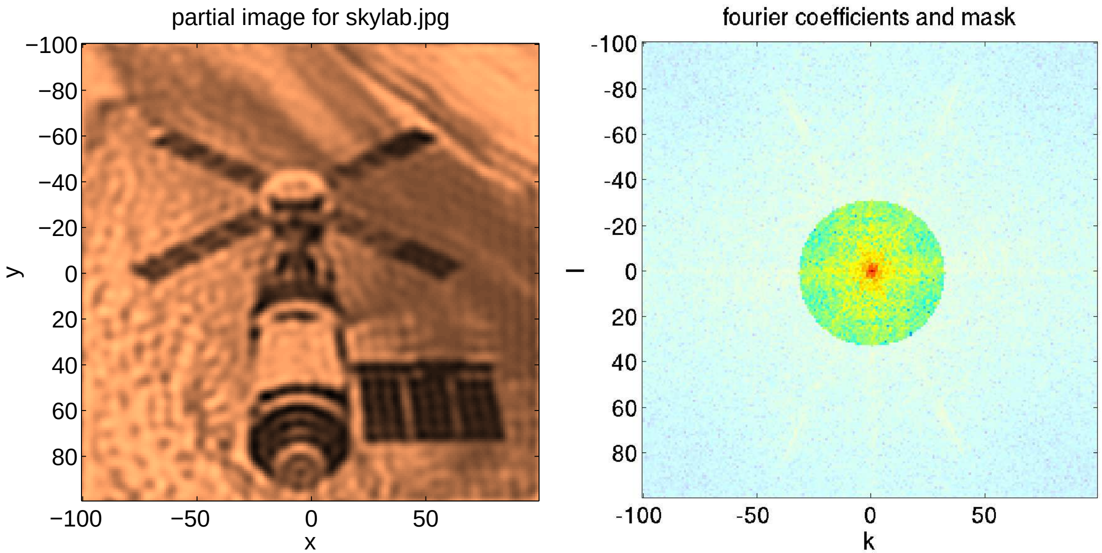




- Transform $f(x, y)$ into $a(k, l)$. Horizontal wavenumber k and vertical wavenumber l.

Fourier transform for images

Broad features:



Detail:

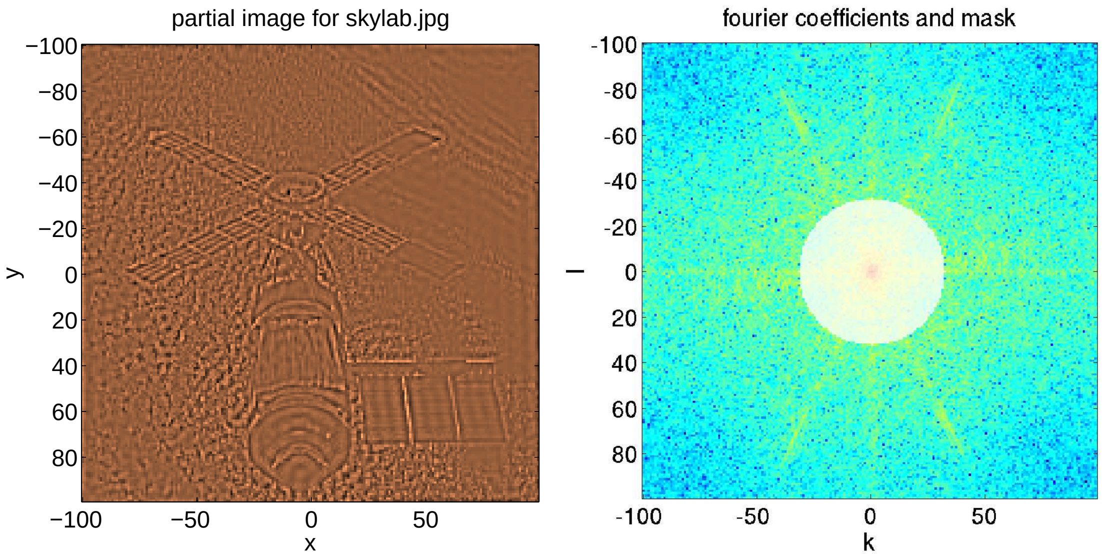
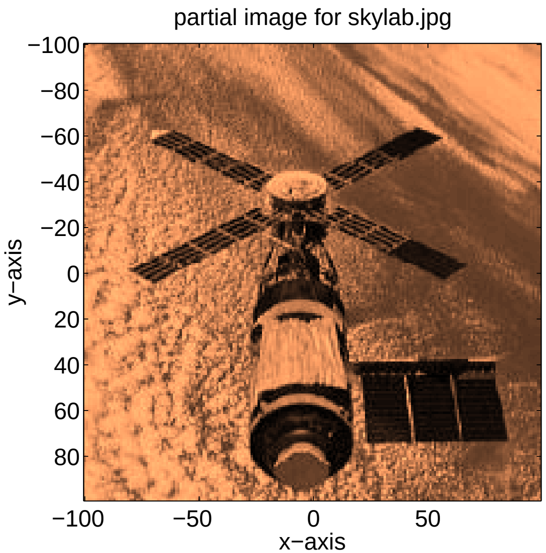
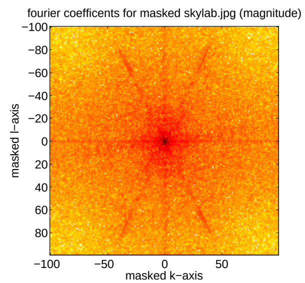
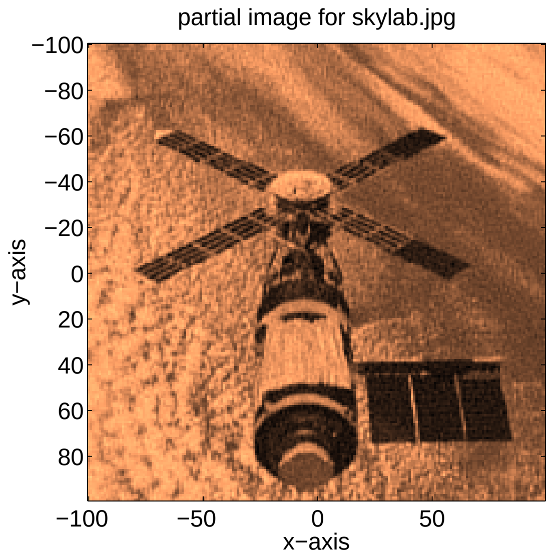


Image compression

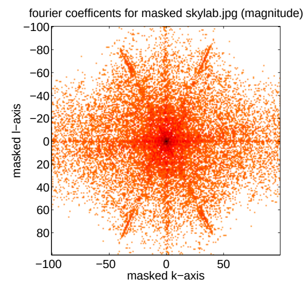


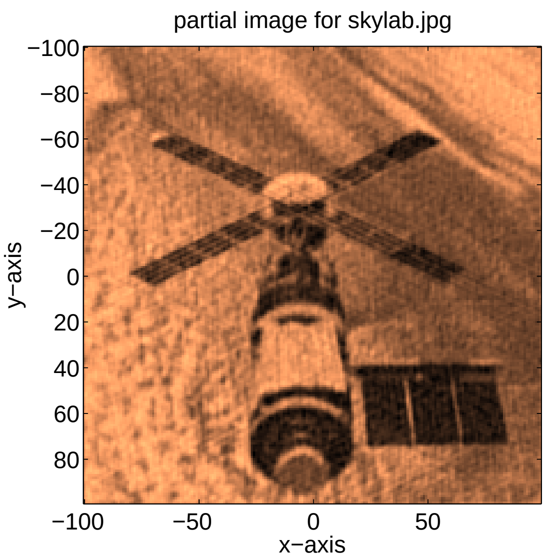
100% Fourier coefficients



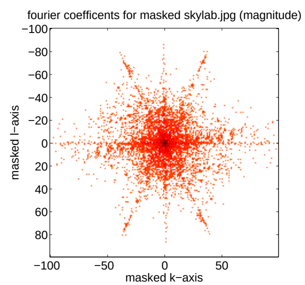


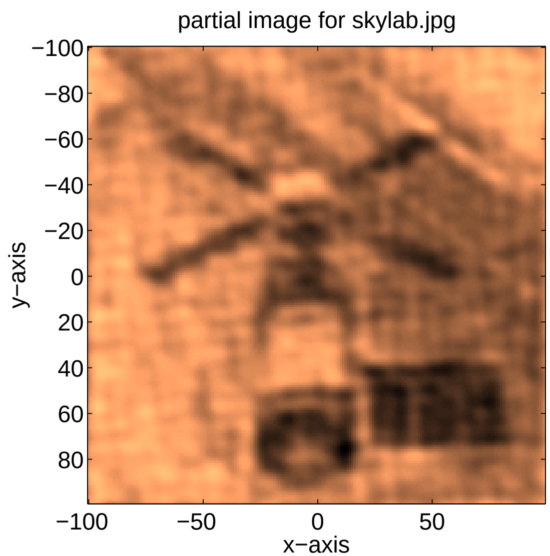
33% Fourier coefficients



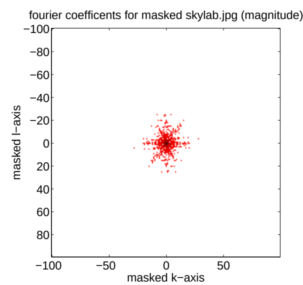


10% Fourier coefficients





1% Fourier coefficients



Demo

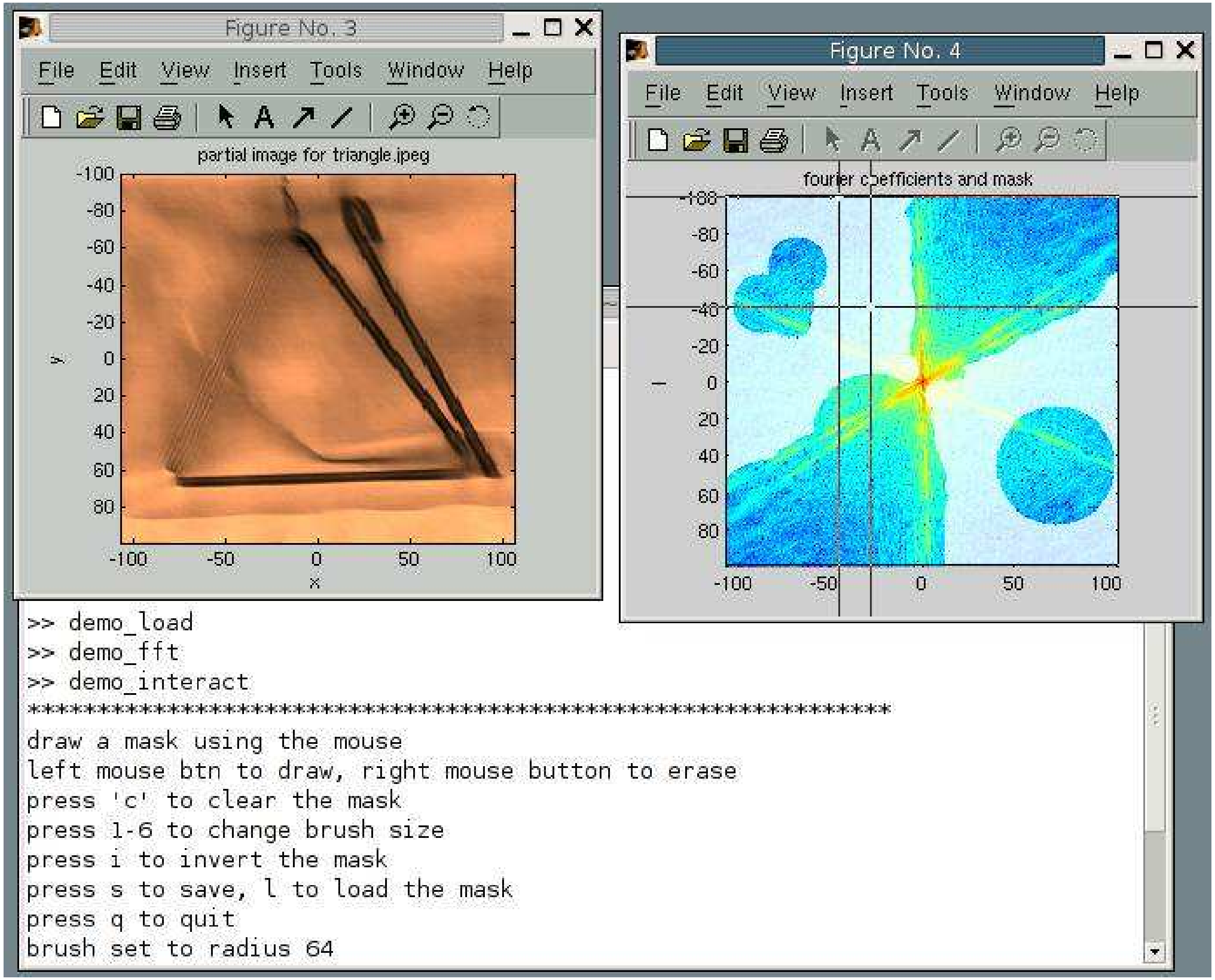
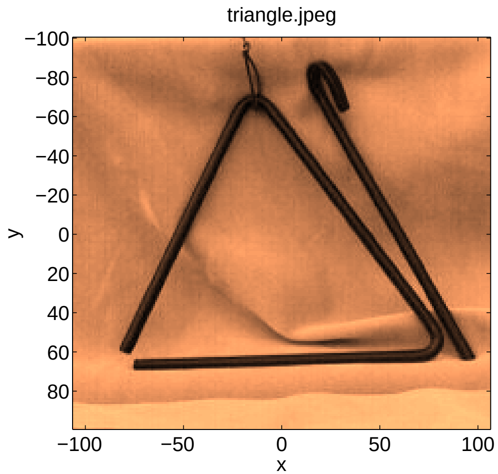
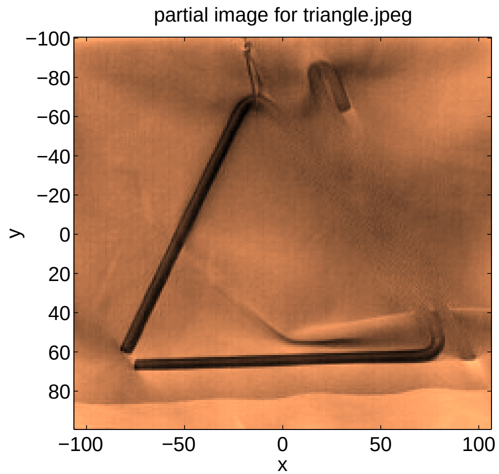


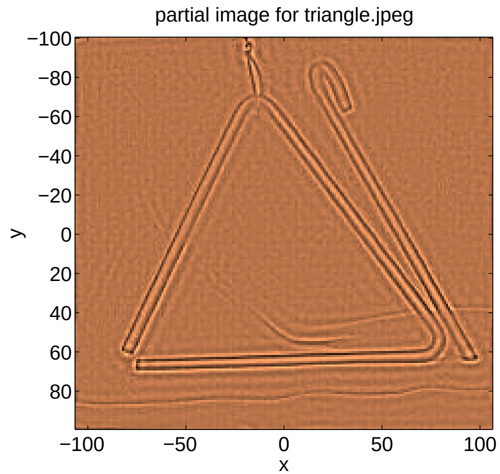
Figure No. 3
File Edit View Insert Tools Window Help
partial image for triangle.jpeg
-100 -80 -60 -40 -20 0 20 40 60 80
-100 -50 0 50 100
x
y
Figure No. 4
File Edit View Insert Tools Window Help
fourier coefficients and mask
-80 -60 -40 -20 0 20 40 60 80
-100 -50 0 50 100
>> demo_load
>> demo_fft
>> demo_interact
****
draw a mask using the mouse
left mouse btn to draw, right mouse button to erase
press 'c' to clear the mask
press 1-6 to change brush size
press i to invert the mask
press s to save, l to load the mask
press q to quit
brush set to radius 64

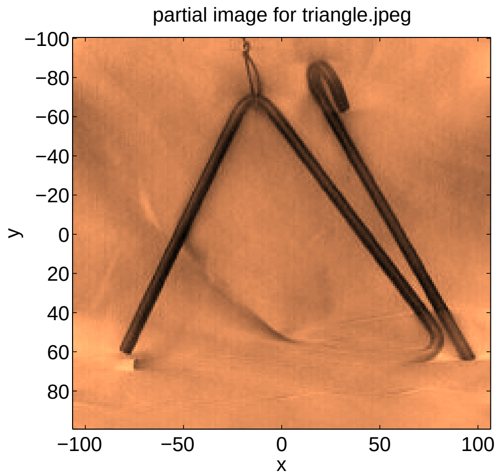
Exercises

- Can you draw a Fourier coefficient mask to produce the following?









Solutions to exercises