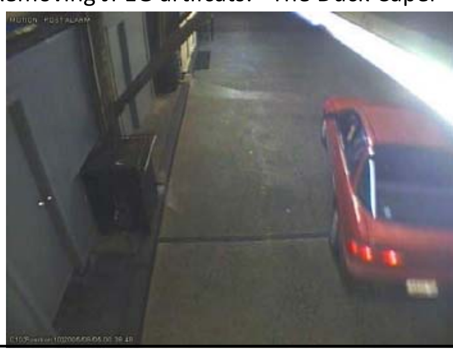
Math review

Lecture 3

Overview

Projects

- Removing JPEG artifcats: “The Duck Caper”



Projects

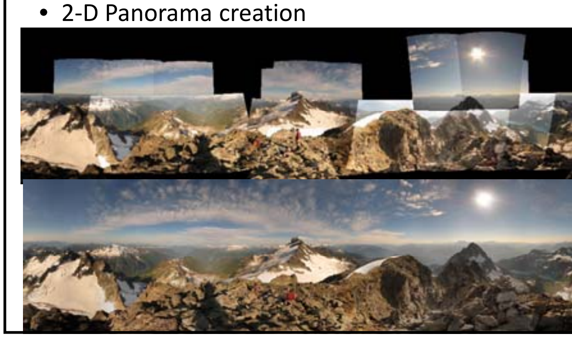
- Hot or Not

Projects

- David Hockney’s Joiners

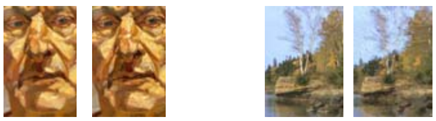

Projects

- 2-D Panorama creation



Projects

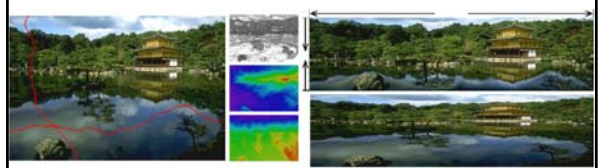
- Image Analogies



A A' B B'

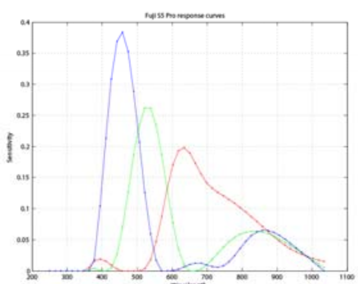
Projects

- Seam carving



Projects

- Spectral response curve estimtation



Overview of today

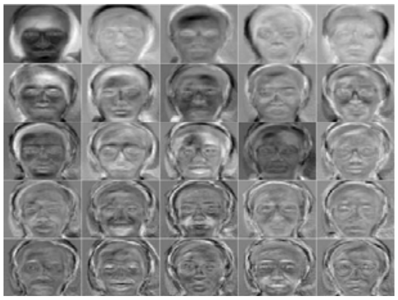
- Linear Algebra review
- Least Squares Optimization
- Linear Systems
- Fourier domain

Whiteboard

PCA - Faces

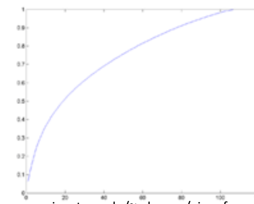
http://www.cs.princeton.edu/~cdecoro/eigenfaces/

PCA - Faces



http://www.cs.princeton.edu/~cdecoro/eigenfaces/

PCA - Faces



http://www.cs.princeton.edu/~cdecoro/eigenfaces/

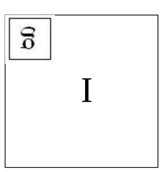
Overview of today

- Linear Algebra review
- Least Squares Optimization
- Linear Systems
- Fourier domain

Whiteboard

Convolution

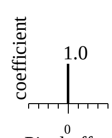
$$f[m,n] = I \otimes g = \sum_{k,l} I[m-k, n-l]\, g[k,l]$$



Slide credit: Bill Freeman

Linear filtering (warm-up slide)

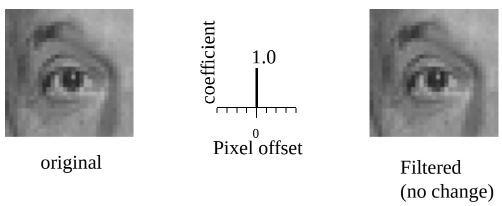
original



?

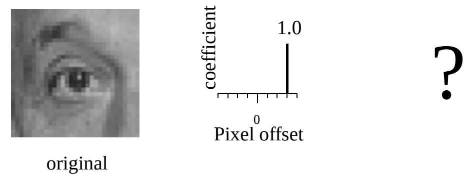
Slide credit: Bill Freeman

Linear filtering (warm-up slide)



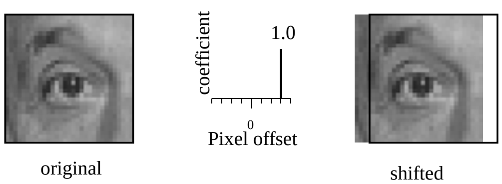
Slide credit: Bill Freeman

Linear filtering



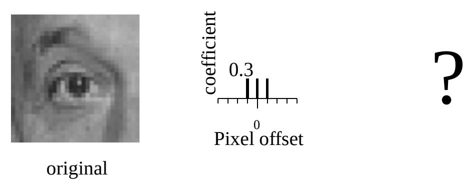
Slide credit: Bill Freeman

shift



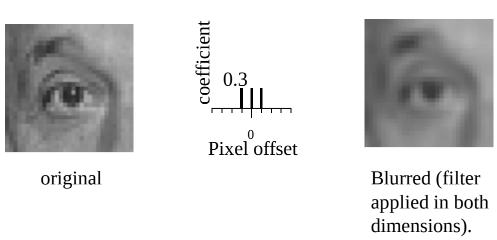
Slide credit: Bill Freeman

Linear filtering



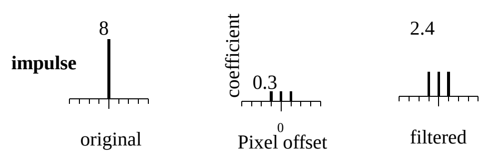
Slide credit: Bill Freeman

Blurring

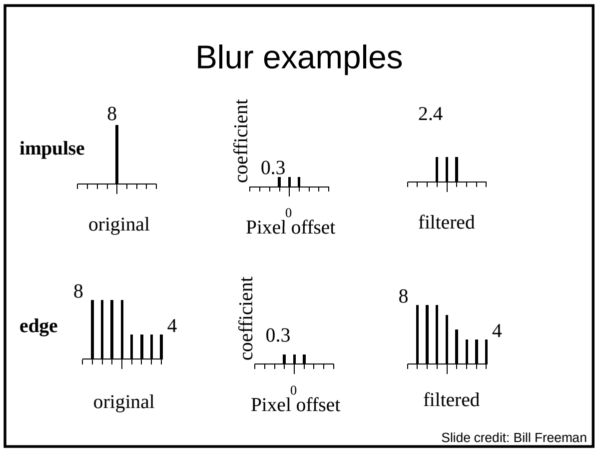


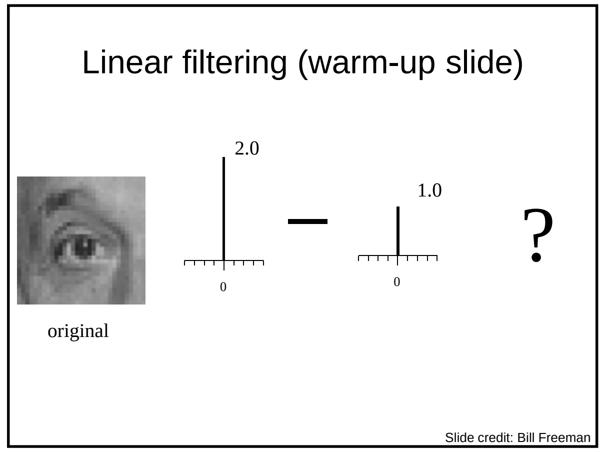
Slide credit: Bill Freeman

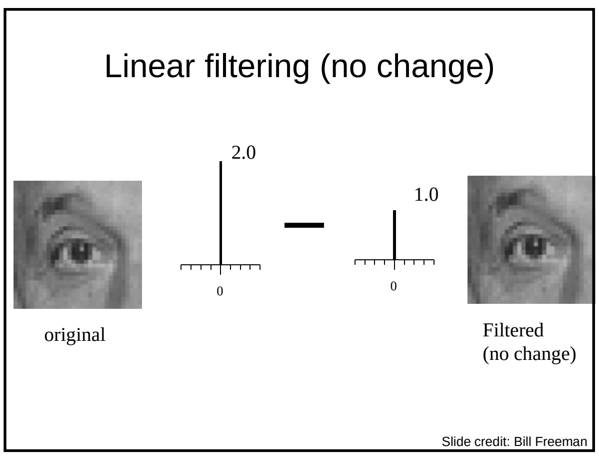
Blur examples

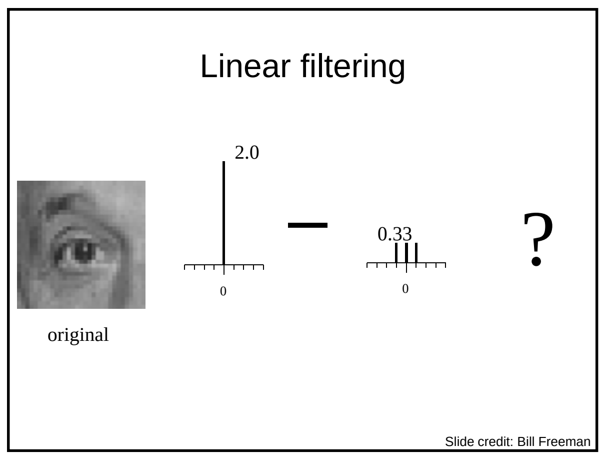


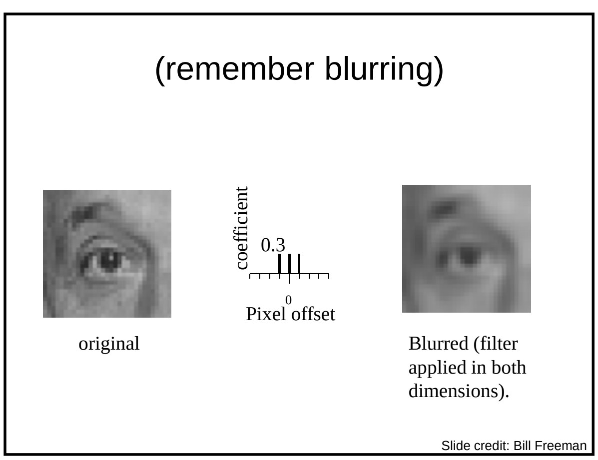
Slide credit: Bill Freeman

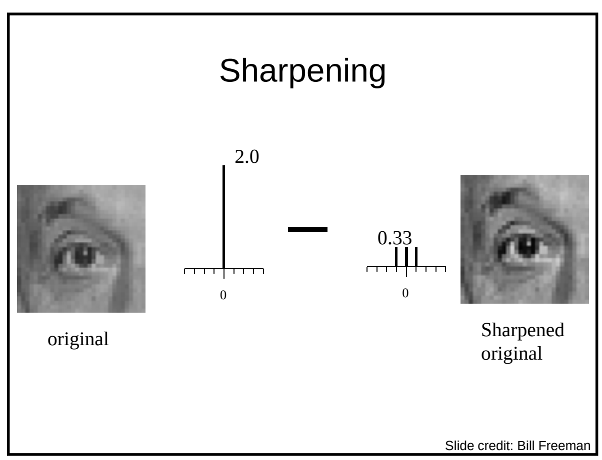

Blur examples
impulse
8
original
coefficient
0.3
0
Pixel offset
2.4
filtered
edge
8
4
original
coefficient
0.3
0
Pixel offset
8
4
filtered
Slide credit: Bill Freeman


Linear filtering (warm-up slide)
2.0
0
1.0
0
?
original
Slide credit: Bill Freeman

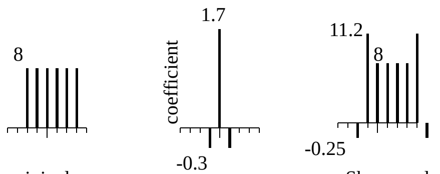

Linear filtering (no change)
2.0
0
1.0
0
original
Filtered
(no change)
Slide credit: Bill Freeman


Linear filtering
2.0
0
0.33
0
?
original
Slide credit: Bill Freeman


(remember blurring)
coefficient
0.3
0
Pixel offset
original
Blurred (filter
applied in both
dimensions).
Slide credit: Bill Freeman


Sharpening
2.0
0
0.33
0
original
Sharpened
original
Slide credit: Bill Freeman

Sharpening example

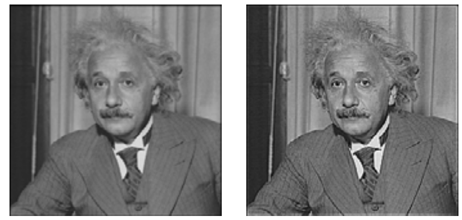


original

Sharpened
(differences are
accentuated; constant
areas are left untouched).

Slide credit: Bill Freeman

Sharpening



before **after**

Slide credit: Bill Freeman

Spatial resolution and color

original

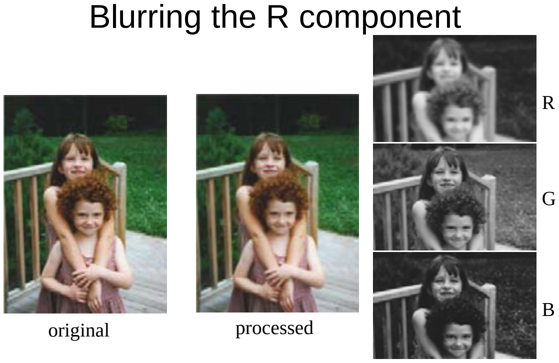
Slide credit: Bill Freeman

Blurring the G component

original processed

Slide credit: Bill Freeman

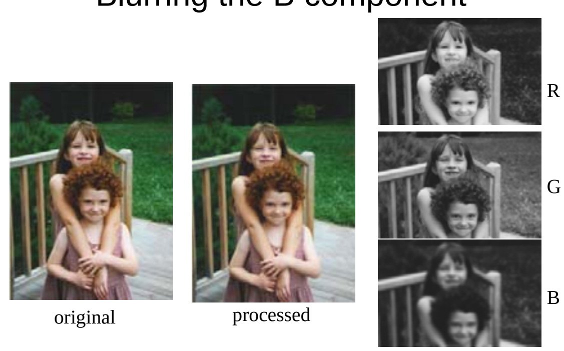
Blurring the R component



original processed

Slide credit: Bill Freeman

Blurring the B component



original processed

Slide credit: Bill Freeman

From W. E. Glenn, in Digital Images and Human Vision, MIT Press, edited by Watson, 1993

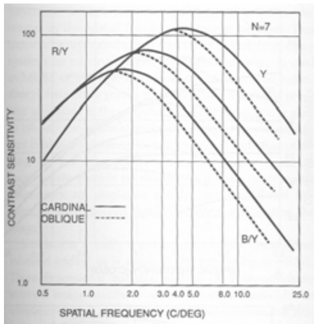
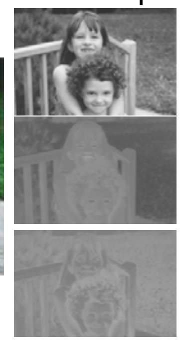


Figure 6.1
Contrast sensitivity threshold functions for static luminance gratings (Y) and isoluminance chromaticity gratings (R/Y,B/Y) averaged over seven observers.

Slide credit: Bill Freeman

Lab color components

L

a

b

A rotation of the color coordinates into directions that are more perceptually meaningful:
L: luminance,
a: red-green,
b: blue-yellow

Slide credit: Bill Freeman

Blurring the L Lab component

original

processed

L

a

b

Slide credit: Bill Freeman

Blurring the a Lab component

original

processed

L

a

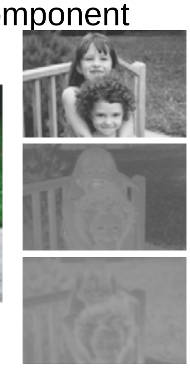
b

Slide credit: Bill Freeman

Blurring the b Lab component

original

processed



L

a

b

Slide credit: Bill Freeman

Overview of today

- Linear Algebra review
- Least Squares Optimization
- Linear Systems
- Fourier domain

Linear image transformations

- In analyzing images, it's often useful to make a change of basis.

transformed image

$$\vec{F} = U\vec{f}$$

Vectorized image

Fourier transform, or
Wavelet transform, or
Steerable pyramid transform

Slide credit: Bill Freeman

Self-inverting transforms

Same basis functions are used for the inverse transform

$$\vec{f} = U^{-1}\vec{F}$$
$$= U^{+}\vec{F}$$

U transpose and complex conjugate

Slide credit: Bill Freeman

An example of such a transform: the Fourier transform

discrete domain

Forward transform

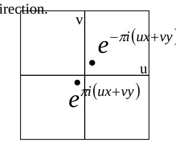
$$F[m,n] = \sum_{k=0}^{M-1}\sum_{l=0}^{N-1} f[k,l]\, e^{-\pi i\left(\frac{km}{M}+\frac{\ln}{N}\right)}$$

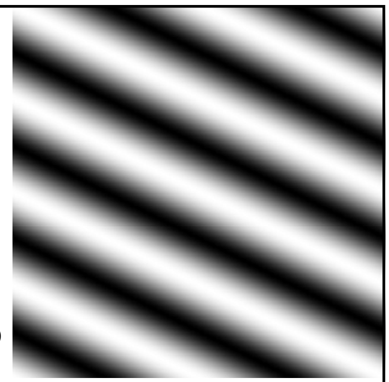
Inverse transform

$$f[k,l] = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1} F[m,n]\, e^{+\pi i\left(\frac{km}{M}+\frac{\ln}{N}\right)}$$

Slide credit: Bill Freeman

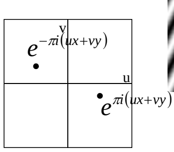
To get some sense of what basis elements look like, we plot a basis element --- or rather, its real part --- as a function of x,y for some fixed u, v. We get a function that is constant when (ux+vy) is constant. The magnitude of the vector (u, v) gives a frequency, and its direction gives an orientation. The function is a sinusoid with this frequency along the direction, and constant perpendicular to the direction.

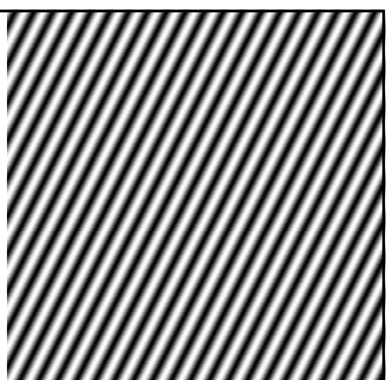




Slide credit: Bill Freeman

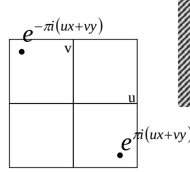
Here u and v are larger than in the previous slide.

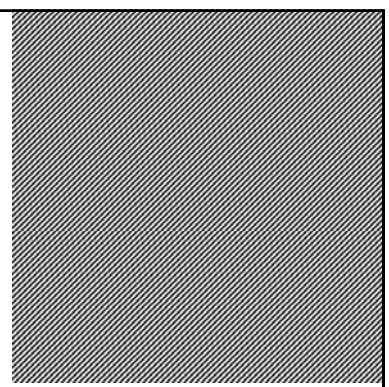




Slide credit: Bill Freeman

And larger still...





Slide credit: Bill Freeman

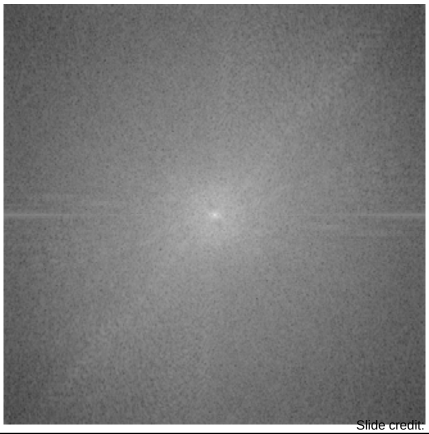
Phase and Magnitude

- Fourier transform of a real function is complex
  - difficult to plot, visualize
  - instead, we can think of the phase and magnitude of the transform
- Phase is the phase of the complex transform
- Magnitude is the magnitude of the complex transform
- Curious fact
  - all natural images have about the same magnitude transform
  - hence, phase seems to matter, but magnitude largely doesn't
- Demonstration
  - Take two pictures, swap the phase transforms, compute the inverse - what does the result look like?

Slide credit: Bill Freeman

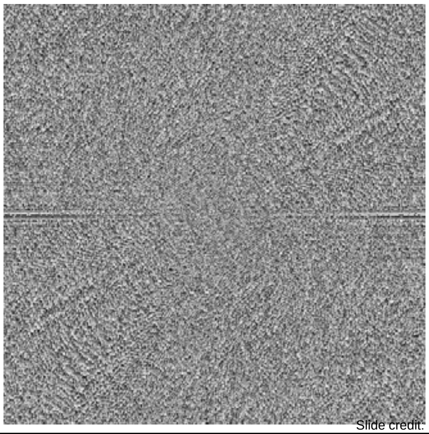
Slide credit: Bill Freeman

This is the magnitude transform of the cheetah pic



Slide credit: Bill Freeman

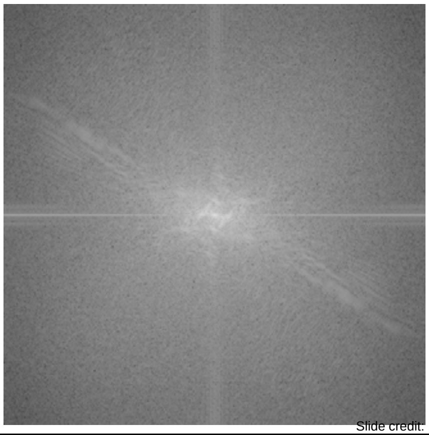
This is the phase transform of the cheetah pic



Slide credit: Bill Freeman

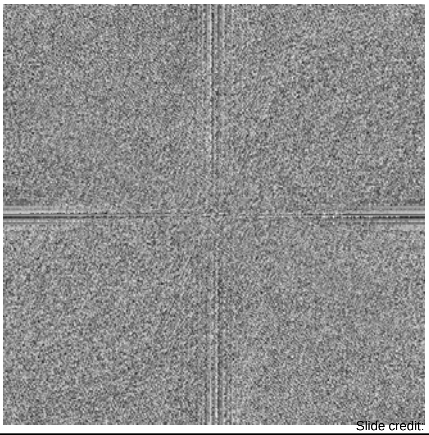
Slide credit: Bill Freeman

This is the magnitude transform of the zebra pic



Slide credit: Bill Freeman

This is the phase transform of the zebra pic

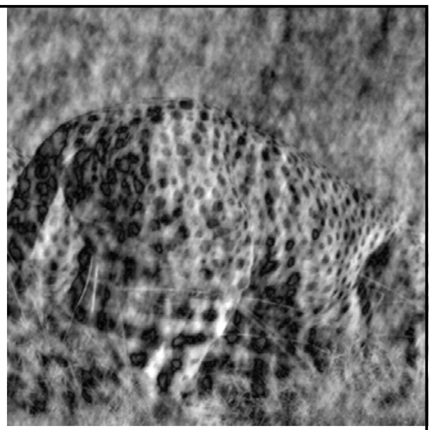


Slide credit: Bill Freeman

Reconstruction with zebra phase, cheetah magnitude

Slide credit: Bill Freeman

Reconstruction with cheetah phase, zebra magnitude



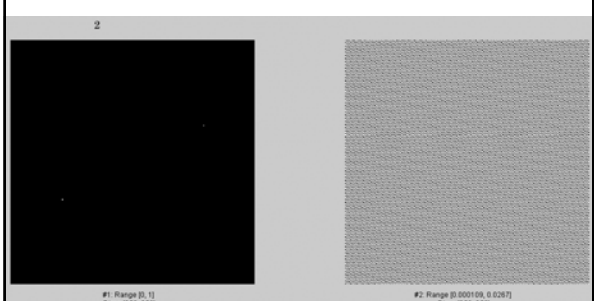
Slide credit: Bill Freeman

Example image synthesis with Fourier basis.

- Following are 16 images showing the reconstruction of an image from a random selection of Fourier basis functions.
- Note, the selection of basis functions to include was not made according to basis magnitude. Doing that would have given us an approximate version of the image much sooner.

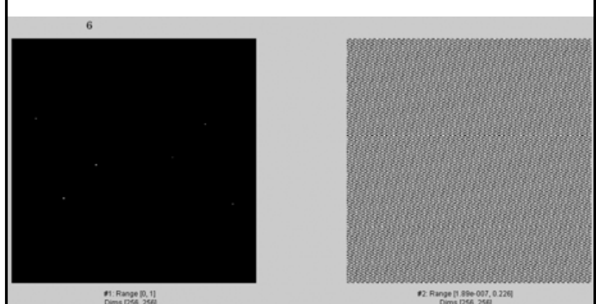
Slide credit: Bill Freeman

2



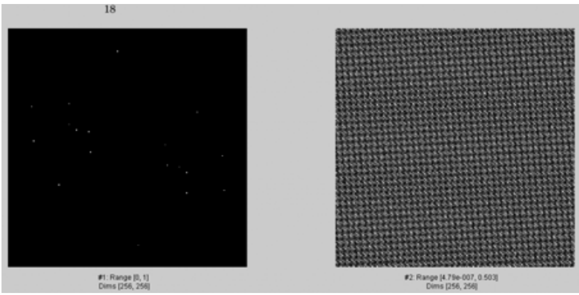
Slide credit: Bill Freeman

6



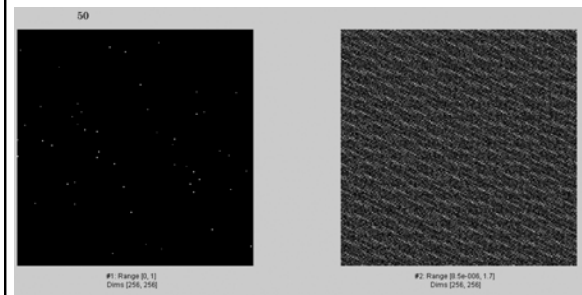
Slide credit: Bill Freeman

18



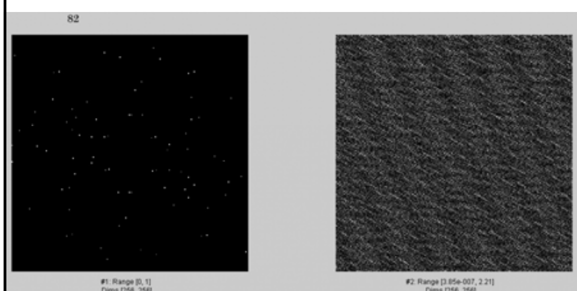
Slide credit: Bill Freeman

50



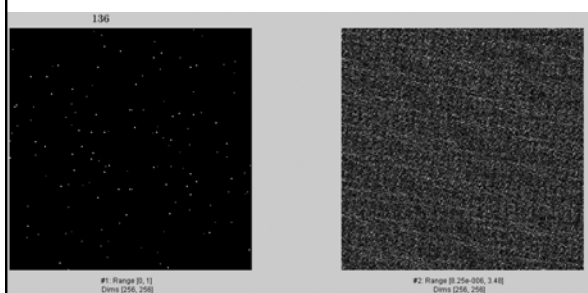
Slide credit: Bill Freeman

82



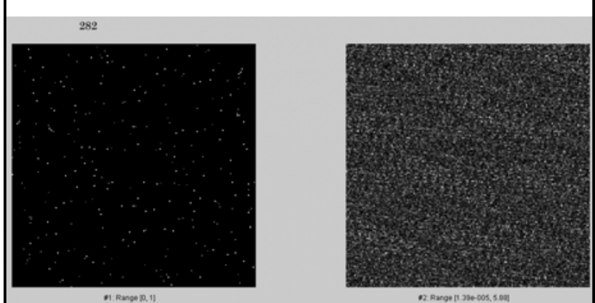
Slide credit: Bill Freeman

136



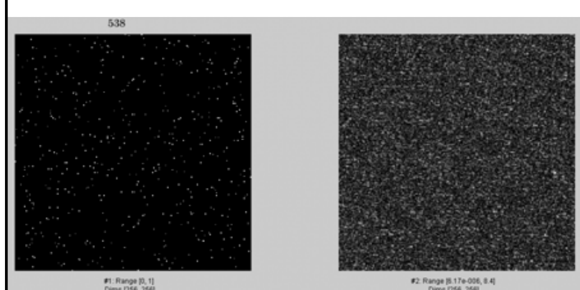
Slide credit: Bill Freeman

282



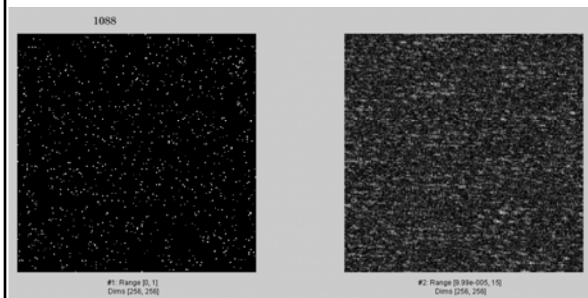
Slide credit: Bill Freeman

538



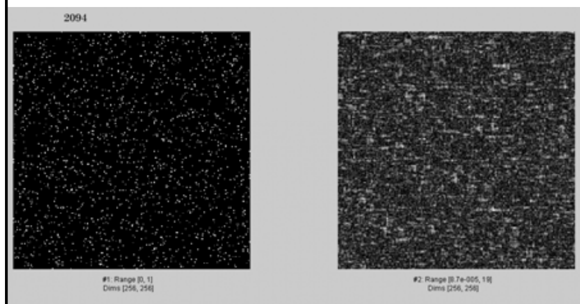
Slide credit: Bill Freeman

1088



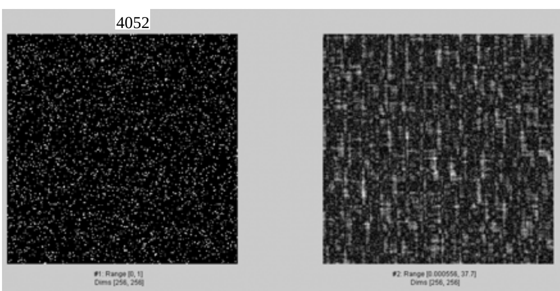
Slide credit: Bill Freeman

2094



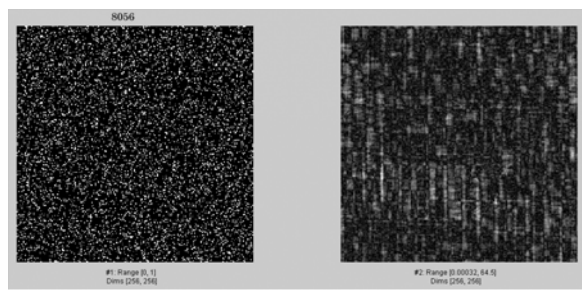
Slide credit: Bill Freeman

4052.



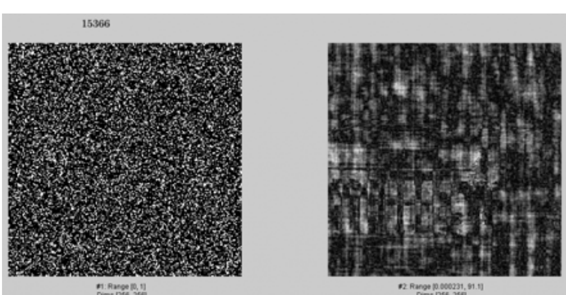
Slide credit: Bill Freeman

8056.



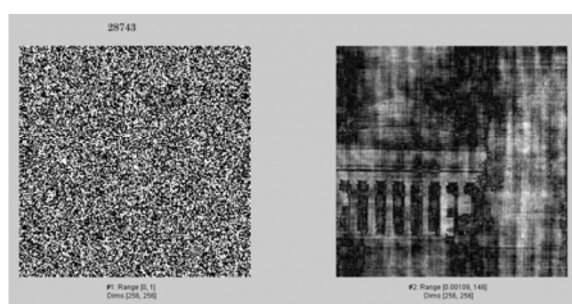
Slide credit: Bill Freeman

15366



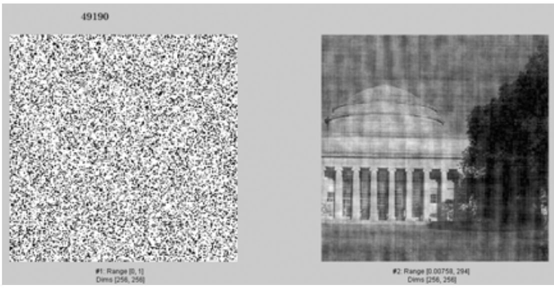
Slide credit: Bill Freeman

28743



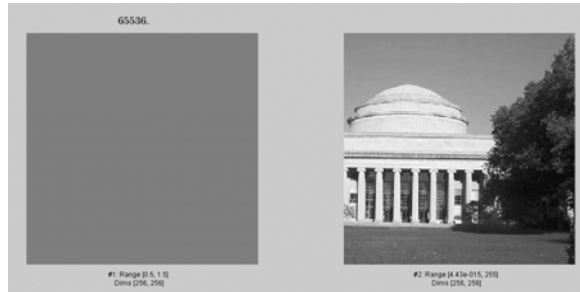
Slide credit: Bill Freeman

49190.



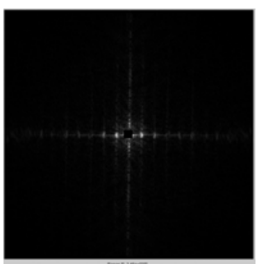
Slide credit: Bill Freeman

65536.



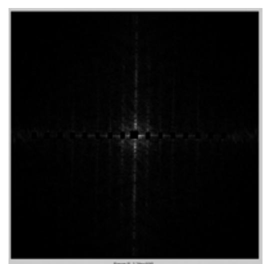
Slide credit: Bill Freeman

Fourier transform magnitude

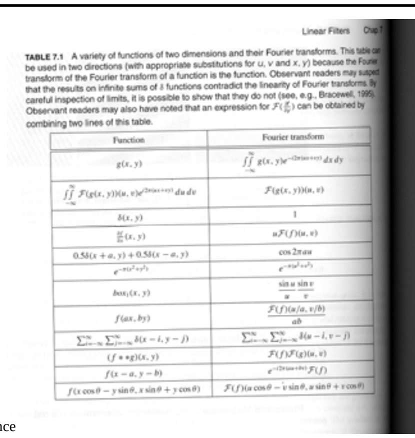


Slide credit: Bill Freeman

Masking out the fundamental and harmonics from periodic pillars



Slide credit: Bill Freeman



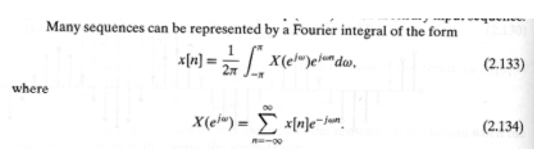
Linear Filters Chap. 7

TABLE 7.1 A variety of functions of two dimensions and their Fourier transforms. This table can be used in two directions (with appropriate substitutions for u, v and x, y) because the Fourier transform of the Fourier transform of a function is the function. Observant readers may suspect that the results on infinite sums of δ functions contradict the linearity of Fourier transforms. By careful inspection of limits, it is possible to show that they do not (see, e.g., Bracewell, 1995). Observant readers may also have noted that an expression for $\mathcal{F}(\frac{\partial f}{\partial y})$ can be obtained by combining two lines of this table.

Function	Fourier transform
$g(x, y)$	$\iint_{-\infty}^{\infty} g(x, y)e^{-i2\pi(ux+vy)}\,dx\,dy$
$\iint_{-\infty}^{\infty} \mathcal{F}(g(x, y))(u, v)e^{i2\pi(ux+vy)}\,du\,dv$	$\mathcal{F}(g(x, y))(u, v)$
$\delta(x, y)$	1
$\frac{\partial f}{\partial x}(x, y)$	$u\mathcal{F}(f)(u, v)$
$0.5\delta(x + a, y) + 0.5\delta(x - a, y)$	$\cos 2\pi au$
$e^{-\pi(x^2+y^2)}$	$e^{-\pi(u^2+v^2)}$
$box_1(x, y)$	$\frac{\sin u}{u}\frac{\sin v}{v}$
$f(ax, by)$	$\frac{\mathcal{F}(f)(u/a, v/b)}{ab}$
$\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}\delta(x - i, y - j)$	$\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}\delta(u - i, v - j)$
$(f **g)(x, y)$	$\mathcal{F}(f)\mathcal{F}(g)(u, v)$
$f(x - a, y - b)$	$e^{-i2\pi(au+bv)}\mathcal{F}(f)$
$f(x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta)$	$\mathcal{F}(f)(u\cos\theta - v\sin\theta, u\sin\theta + v\cos\theta)$

Forsyth&Ponce

Discrete-time, continuous frequency Fourier transform



Many sequences can be represented by a Fourier integral of the form

$$x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega, \qquad (2.133)$$

where

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}. \qquad (2.134)$$

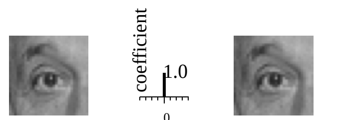
Oppenheim, Schafer and Buck, Discrete-time signal processing, Prentice Hall, 1999

Why is the Fourier domain particularly useful?

- It tells us the effect of linear convolutions.
- There is a fast algorithm for performing the DFT, allowing for efficient signal filtering.
- The Fourier domain offers an alternative domain for understanding and manipulating the image.

Slide credit: Bill Freeman

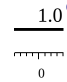
Analysis of our simple filters



original

Filtered (no change)

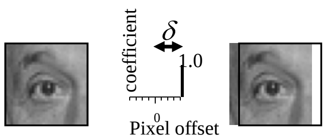
$$F[m] = \sum_{k=0}^{M-1} f[k] e^{-\pi i \left(\frac{km}{M}\right)}$$
$$= 1$$



constant

Slide credit: Bill Freeman

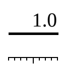
Analysis of our simple filters



original

shifted

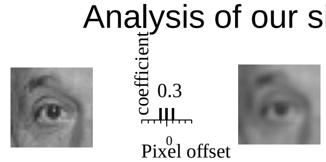
$$F[m] = \sum_{k=0}^{M-1} f[k] e^{-\pi i \left(\frac{km}{M}\right)}$$
$$= e^{-\pi i \frac{\delta m}{M}}$$



Constant magnitude, linearly shifted phase

Slide credit: Bill Freeman

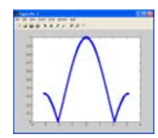
Analysis of our simple filters



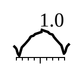
original

blurred

$$F[m] = \sum_{k=0}^{M-1} f[k] e^{-\pi i \left(\frac{km}{M}\right)}$$
$$= \frac{1}{3}\left(1 + 2\cos\left(\frac{\pi m}{M}\right)\right)$$

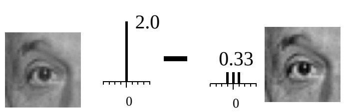


Low-pass filter



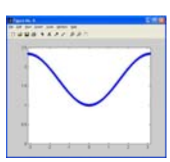
Slide credit: Bill Freeman

Analysis of our simple filters



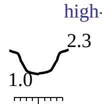
original

sharpened



$$F[m] = \sum_{k=0}^{M-1} f[k] e^{-\pi i \left(\frac{km}{M}\right)}$$
$$= 2 - \frac{1}{3}\left(1 + 2\cos\left(\frac{\pi m}{M}\right)\right)$$

high-pass filter



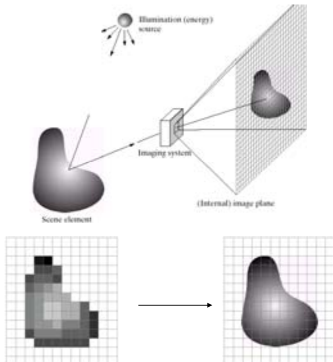
Slide credit: Bill Freeman

Convolution versus FFT

- 1-d FFT: O(NlogN) computation time, where N is number of samples.
- 2-d FFT: 2N(NlogN), where N is number of pixels on a side
- Convolution: K N^2, where K is number of samples in kernel
- Say $N=2^{10}$, K=100. 2-d FFT: 20 2^{20}, while convolution gives 100 2^{20}

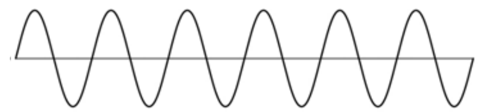
Slide credit: Bill Freeman

Sampling and Reconstruction



Sampling and Reconstruction

- Simple example: a sign wave

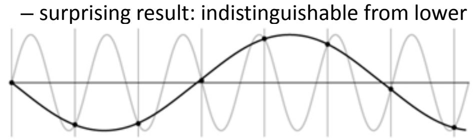


Undersampling

- What if we "missed" things between the samples?
- Simple example: undersampling a sine wave
  - unsurprising result: information is lost

Undersampling

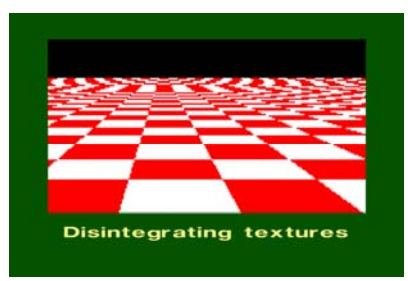
- What if we "missed" things between the samples?
- Simple example: undersampling a sine wave
  - unsurprising result: information is lost
  - surprising result: indistinguishable from lower



Undersampling

- What if we "missed" things between the samples?
- Simple example: undersampling a sine wave
  - unsurprising result: information is lost
  - surprising result: indistinguishable from lower frequency

Aliasing in images



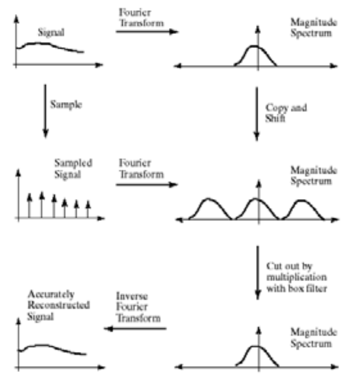
The Fourier transform of a sampled signal

$$F\left(\text{Sample}_{2D}(f(x,y))\right) = F\left(f(x,y) \sum_{i=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \delta(x-i, y-j) \right)$$

$$= F(f(x,y)) ** F\left(\sum_{i=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \delta(x-i, y-j) \right)$$

$$= \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} F(u-i, v-j)$$

Slide credit: Bill Freeman



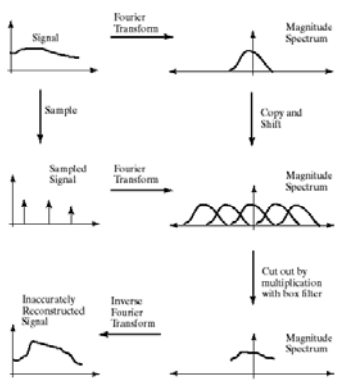


Image half-sizing

This image is too big to fit on the screen. How can we reduce it?

How to generate a half-sized version?

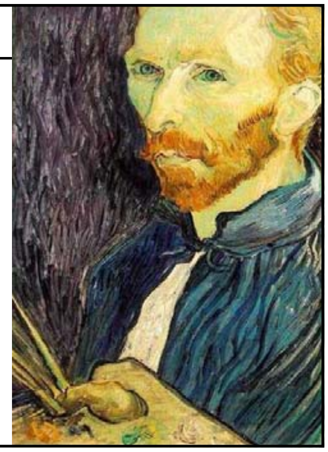
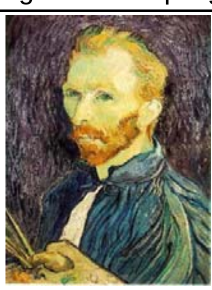


Image sub-sampling



1/4

1/8

Throw away every other row and column to create a 1/2 size image
- called *image sub-sampling*

Slide by Steve Seitz

Image sub-sampling

1/2 1/4 (2x zoom) 1/8 (4x zoom)

Aliasing! What do we do?

Slide by Steve Seitz

Gaussian (lowpass) pre-filtering

G 1/8

G 1/4

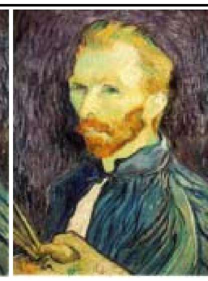
Gaussian 1/2

Solution: filter the image, *then* subsample

- Filter size should double for each ½ size reduction. Why?

Slide by Steve Seitz

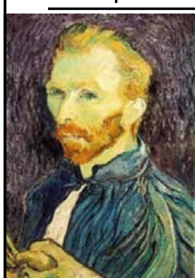
Subsampling with Gaussian pre-filtering



Gaussian 1/2 G 1/4 G 1/8

Slide by Steve Seitz

Compare with...



1/2 1/4 (2x zoom) 1/8 (4x zoom)

Slide by Steve Seitz