Review – Frequency Domain Techniques

- Thinking images in terms of frequency.

- Treat images as infinite-size, continuous periodic functions.
Review - Fourier Transform

We want to understand the frequency $w$ of our signal. So, let’s reparametrize the signal by $w$ instead of $x$:

$$f(x) \rightarrow \text{Fourier Transform} \rightarrow F(w)$$

For every $w$ from 0 to inf, $F(w)$ holds the amplitude $A$ and phase $f$ of the corresponding sine $A \sin(\omega x + \phi)$

- How can $F$ hold both? Complex number trick!

$$F(\omega) = R(\omega) + iI(\omega)$$

$$A = \pm \sqrt{R(\omega)^2 + I(\omega)^2} \quad \phi = \tan^{-1} \frac{I(\omega)}{R(\omega)}$$

We can always go back:

$$F(w) \rightarrow \text{Inverse Fourier Transform} \rightarrow f(x)$$

Slide credit: A. Efros
**Review - Fourier Transform**

- Fourier transform stores the magnitude and phase at each frequency
  - Magnitude encodes how much signal there is at a particular frequency
  - Phase encodes spatial information (indirectly)
  - For mathematical convenience, this is often notated in terms of real and complex numbers

\[
\text{Amplitude: } A = \pm \sqrt{R(\omega)^2 + I(\omega)^2} \quad \text{Phase: } \phi = \tan^{-1} \frac{I(\omega)}{R(\omega)}
\]
Review - Discrete Fourier transform

- **Forward transform**

\[ F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)} \]

for \( u = 0, 1, 2, \ldots, M - 1, v = 0, 1, 2, \ldots, N - 1 \)

- **Inverse transform**

\[ f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M + vy/N)} \]

for \( x = 0, 1, 2, \ldots, M - 1, y = 0, 1, 2, \ldots, N - 1 \)

\( u, v \) : the transform or frequency variables
\( x, y \) : the spatial or image variables

Euler’s definition of \( e^{j\theta} \)
Review - The Fourier Transform

- Represent function on a new basis
  - Think of functions as vectors, with many components
  - We now apply a linear transformation to transform the basis
    - dot product with each basis element
- In the expression, u and v select the basis element, so a function of x and y becomes a function of u and v
- Basis elements have the form $e^{-i2\pi(ux + vy)}$
Review - The Fourier Transform

How to interpret a 2-d Fourier Spectrum

Horizontal orientation

Vertical orientation

45 deg.

fx in cycles/image

Low spatial frequencies

Log power spectrum

High spatial frequencies

Slide credit: B. Freeman and A. Torralba
Review - The Convolution Theorem

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

\[ F[g * h] = F[g]F[h] \]

• The inverse Fourier transform of the product of two Fourier transforms is the convolution of the two inverse Fourier transforms

\[ F^{-1}[gh] = F^{-1}[g] * F^{-1}[h] \]

• Convolution in spatial domain is equivalent to multiplication in frequency domain!
Review - Filtering in frequency domain
Today

- Sampling
- Gabor wavelets, Steerable filters
Today

• Sampling

• Gabor wavelets, Steerable filters
Sampling

Why does a lower resolution image still make sense to us? What do we lose?

Image: http://www.flickr.com/photos/igoms/136916757/

Slide credit: D. Hoiem
Sampled representations

- How to store and compute with continuous functions?
- Common scheme for representation: samples
  - write down the function’s values at many points
Reconstruction

- Making samples back into a continuous function
  - for output (need realizable method)
  - for analysis or processing (need mathematical method)
  - amounts to “guessing” what the function did in between

Slide credit: S. Marschner
Sampling in digital audio

- Recording: sound to analog to samples to disc
- Playback: disc to samples to analog to sound again
  - how can we be sure we are filling in the gaps correctly?

Slide credit: S. Marschner
Sampling Theorem

Continuous signal: (Real world signal)

\[ f(x) \]

Shah function (Impulse train): (What the image measures)

\[ s(x) = \sum_{n=-\infty}^{\infty} \delta(x - nx_0) \]

Sampled function:

\[ f_s(x) = f(x)s(x) = f(x) \sum_{n=-\infty}^{\infty} \delta(x - nx_0) \]

Slide credit: S. Narasimhan
Sampling Theorem

Sampled function:

\[ f_s(x) = f(x)s(x) = f(x) \sum_{n=-\infty}^{\infty} \delta(x-nx_0) \]

\[ F_S(u) = F(u) * S(u) = F(u) * \frac{1}{x_0} \sum_{n=-\infty}^{\infty} \delta\left(u - \frac{n}{x_0}\right) \]

Slide credit: S. Narasimhan
Fourier Transform Pairs

Note that these are derived using angular frequency \( e^{-i\omega x} \)
Sampling Theorem

Sampled function:

\[ f_s(x) = f(x)s(x) = f(x) \sum_{n=\infty}^{\infty} \delta(x - nx_0) \]

\[ F_s(u) = F(u) \ast S(u) = F(u) \ast \sum_{n=-\infty}^{\infty} \frac{1}{x_0} \delta\left(u - \frac{n}{x_0}\right) \]

Slide credit: S. Narasimhan
**Sampling Theorem**

Sampled function:

\[ f_s(x) = f(x) s(x) = f(x) \sum_{n=-\infty}^{\infty} \delta(x - nx_0) \]

\[ F_S(u) = F(u) * S(u) = F(u) * \frac{1}{x_0} \sum_{n=-\infty}^{\infty} \delta\left(u - \frac{n}{x_0}\right) \]

Slide credit: S. Narasimhan
Subsampling by a factor of 2

Throw away every other row and column to create a $1/2$ size image

Slide credit: D. Hoiem
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
  - also was always indistinguishable from higher frequencies
  - aliasing: signals “traveling in disguise” as other frequencies

Slide credit: S. Marschner
Aliasing problem

• Sub-sampling may be dangerous….

• Characteristic errors may appear:
  – “Wagon wheels rolling the wrong way in movies”
  – “Checkerboards disintegrate in ray tracing”
  – “Striped shirts look funny on color television”
Moire patterns in real-world images. Here are comparison images by Dave Etchells of Imaging Resource using the Canon D60 (with an antialias filter) and the Sigma SD-9 (which has no antialias filter). The bands below the fur in the image at right are the kinds of artifacts that appear in images when no antialias filter is used. Sigma chose to eliminate the filter to get more sharpness, but the resulting apparent detail may or may not reflect features in the image.
More examples

Check out Moire patterns on the web.

Slide credit: A. Farhadi
Aliasing in video

Imagine a spoked wheel moving to the right (rotating clockwise). Mark wheel with dot so we can see what’s happening.

If camera shutter is only open for a fraction of a frame time (frame time = 1/30 sec. for video, 1/24 sec. for film):

Without dot, wheel appears to be rotating slowly backwards! (counterclockwise)
Aliasing in graphics

Disintegrating textures
Sampling and aliasing

256x256  128x128  64x64  32x32  16x16
Sampling Theorem

Sampled function:

\[ f_s(x) = f(x)s(x) = f(x) \sum_{n=-\infty}^{\infty} \delta(x - nx_0) \]

\[ F_s(u) = F(u) * S(u) = F(u) * \frac{1}{x_0} \sum_{n=-\infty}^{\infty} \delta\left(u - \frac{n}{x_0}\right) \]

Slide credit: S. Narasimhan
Nyquist Frequency

If \( u_{\text{max}} > \frac{1}{2x_0} \)

When can we recover \( F(u) \) from \( F_S(u) \)?

**Only if** \( u_{\text{max}} \leq \frac{1}{2x_0} \) (Nyquist Frequency)

We can use

\[
C(u) = \begin{cases} 
  x_0 & |u| < \frac{1}{2x_0} \\
  0 & \text{otherwise} 
\end{cases}
\]

Then \( F(u) = F_S(u)C(u) \) and \( f(x) = \text{IFT}[F(u)] \)

Sampling frequency must be greater than \( 2u_{\text{max}} \)

Slide credit: S. Narasimhan
Nyquist-Shannon Sampling Theorem

- When sampling a signal at discrete intervals, the sampling frequency must be $\geq 2 \times f_{\text{max}}$
- $f_{\text{max}} = \text{max frequency of the input signal}$
- This will allow to reconstruct the original perfectly from the sampled version

Slide credit: D. Hoiem
2D example

Good sampling

Bad sampling

Slide credit: N. Kumar
Anti-aliasing

Solutions:

• Sample more often

• Get rid of all frequencies that are greater than half the new sampling frequency
  – Will lose information
  – But it’s better than aliasing
  – Apply a smoothing filter
Preventing aliasing

• Introduce lowpass filters:
  – remove high frequencies leaving only safe, low frequencies
  – choose lowest frequency in reconstruction (disambiguate)
Impulse Train

• Define a \textit{comb} function (impulse train) in 1D as follows

\[ \text{comb}_M[x] = \sum_{k=-\infty}^{\infty} \delta[x - kM] \]

where \( M \) is an integer
Impulse Train in 2D (bed of nails)

\[ \text{comb}_{M,N}(x, y) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(x - kM, y - lN) \]

- Fourier Transform of an impulse train is also an impulse train:

\[ \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(x - kM, y - lN) \iff \frac{1}{MN} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta\left( u - \frac{k}{M}, v - \frac{l}{N} \right) \]

\[ \text{comb}_{M,N}(x, y) \]  
\[ \text{comb}_{\frac{1}{M}, \frac{1}{N}}(u, v) \]

As the comb samples get further apart, the spectrum samples get closer together!

Slide credit: B. K. Gunturk
Impulse Train in 1D

\[ \text{comb}_2(x) \]

\[ \frac{1}{2} \text{comb}_1(u) \]

- Remember:

Scaling: \[ f(ax) \]

\[ \frac{1}{|a|} F \left( \frac{u}{a} \right) \]

Slide credit: B. K. Gunturk
Sampling low frequency signal

\[ f(x) \]

\[ F(u) \]

\[ \text{comb}_M (x) \]

\[ \text{comb}_{\frac{1}{M}} (u) \]

Multiply:

\[ f(x) \text{comb}_M (x) \]

Convolve:

\[ F(u) \ast \text{comb}_{\frac{1}{M}} (u) \]

Slide credit: B. K. Gunturk
Sampling low frequency signal

\[ f(x) \overset{\sim}{\rightarrow} F(u) \]

\[ f(x) \text{comb}_M(x) \overset{\sim}{\rightarrow} F(u) \ast \text{comb}_{\frac{1}{M}}(u) \]

“No problem” if \( \frac{1}{M} > 2W \)

Slide credit: B. K. Gunturk
Sampling low frequency signal

If there is no overlap, the original signal can be recovered from its samples by low-pass filtering.

Slide credit: B. K. Gunturk
Overlap: The high frequency energy is folded over into low frequency. It is “aliasing” as lower frequency energy. And you cannot fix it once it has happened.
Sampling high frequency signal

\[ f(x) \]

\[ f(x) \ast h(x) \]

\[ \left[ f(x) \ast h(x) \right] \text{comb}_M(x) \]

\[ F(u) \]

Anti-aliasing filter

\[ \frac{1}{2M} \]

\[ \frac{1}{M} \]

Slide credit: B. K. Gunturk
Sampling high frequency signal

• Without anti-aliasing filter:

\[ f(x) \text{comb}_M(x) \]

• With anti-aliasing filter:

\[ [f(x) * h(x)] \text{comb}_M(x) \]

Slide credit: B. K. Gunturk
Sampling high frequency signal

- Without anti-aliasing filter:
  \[ f(x) \cdot \text{comb}_M(x) \]

- With anti-aliasing filter:
  \[ \left[ f(x) \ast h(x) \right] \cdot \text{comb}_M(x) \]
Algorithm for downsampling by factor of 2

1. Start with image(h, w)
2. Apply low-pass filter
   \[ \text{im\_blur} = \text{imfilter(image, fspecial('gaussian', 7, 1))} \]
3. Sample every other pixel
   \[ \text{im\_small} = \text{im\_blur(1:2:end, 1:2:end);} \]
Anti-aliasing

Slide credit: Forsyth and Ponce
Subsampling without pre-filtering

1/2

1/4  (2x zoom)

1/8  (4x zoom)

Slide credit: S. Seitz
Subsampling with Gaussian pre-filtering

Gaussian 1/2

G 1/4

G 1/8

Slide credit: S. Seitz
by dropping pixels

250 pixel width

 gaussian filter

Slide credit: S. Marschner
Up-sampling

How do we compute the values of pixels at fractional positions?
Up-sampling

How do we compute the values of pixels at fractional positions?

Bilinear sampling:
\[ f(x + a, y + b) = (1 - a)(1 - b) f(x, y) + a(1 - b) f(x + 1, y) + (1 - a)b f(x, y + 1) + ab f(x + 1, y + 1) \]

Bicubic sampling fits a higher order function using a larger area of support.

Slide credit: A. Farhadi
Up-sampling Methods
Up-sampling

Nearest neighbor

Bilinear

Bicubic

Slide credit: A. Farhadi
Up-sampling

Nearest neighbor  Bilinear  Bicubic

Slide credit: A. Farhadi
Today

- Sampling
- Gabor wavelets, Steerable filters
Fourier Filtering

Multiply by a filter in the frequency domain => convolve with the filter in spatial domain.

Images from Steve Lehar [http://cns-alumni.bu.edu/~slehar](http://cns-alumni.bu.edu/~slehar) An Intuitive Explanation of Fourier Theory
Phase Caries More Information

Raw Images:

Magnitude and Phase:

Reconstruct (inverse FFT) mixing the magnitude and phase images

Phase “Wins”

Slide credit: S. Thrun
What is a good representation for image analysis?

• Fourier transform domain tells you “what” (textural properties), but not “where”.

• Pixel domain representation tells you “where” (pixel location), but not “what”.

• Want an image representation that gives you a local description of image events—what is happening where.

Slide credit: B. Freeman and A. Torralba
Analyzing local image structures

Too much

Too little

Slide credit: B. Freeman and A. Torralba
The image through the Gaussian window

$$h(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

Too much

Too little

Probably still too little...

...but hard enough for now

Slide credit: B. Freeman and A. Torralba
Analysis of local frequency

Fourier basis:

\[ e^{j2\pi u_0 x} \]

Gabor wavelet:

\[ \psi(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}} e^{j2\pi u_0 x} \]

We can look at the real and imaginary parts:

\[ \psi_c(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}} \cos(2\pi u_0 x) \]

\[ \psi_s(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}} \sin(2\pi u_0 x) \]

Slide credit: B. Freeman and A. Torralba
Gabor wavelets

\[ \psi_c(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \cos(2\pi u_0 x) \]

\[ \psi_s(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \sin(2\pi u_0 x) \]

Slide credit: B. Freeman and A. Torralba
Gabor filters

Gabor filters at different scales and spatial frequencies

Top row shows anti-symmetric (or odd) filters; these are good for detecting odd-phase structures like edges. Bottom row shows the symmetric (or even) filters, good for detecting line phase contours.

Slide credit: B. Freeman and A. Torralba
Fig. 5. Top row: illustrations of empirical 2-D receptive field profiles measured by J. P. Jones and L. A. Palmer (personal communication) in simple cells of the cat visual cortex. Middle row: best-fitting 2-D Gabor elementary function for each neuron, described by (10). Bottom row: residual error of the fit, indistinguishable from random error in the Chi-squared sense for 97 percent of the cells studied.
Quadrature filter pairs

- A quadrature filter is a complex filter whose real part is related to its imaginary part via a Hilbert transform along a particular axis through the origin.

\[ \psi(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} e^{j2\mu_0 x} \]

Slide credit: B. Freeman & A. Torralba
Quadrature filter pairs

Contrast invariance! (same energy response for white dot on black background as for a black dot on a white background).

Slide credit: B. Freeman and A. Torralba
Quadrature filter pairs

edge

energy response to an edge

Slide credit: B. Freeman and A. Torralba
Quadrature filter pairs

energy response to a line

Slide credit: B. Freeman and A. Torralba
How quadrature pair filters work

Figure 3-5: Frequency content of two bandpass filters in quadrature. (a) even phase filter, called $G$ in text, and (b) odd phase filter, $H$. Plus and minus sign illustrate relative sign of regions in the frequency domain. See Fig. 3-6 for calculation of the frequency content of the energy measure derived from these two filters.

Slide credit: B. Freeman and A. Torralba
How quadrature pair filters work

(a) Fourier transform of $G^*G$

(b) Fourier transform of $H^*H$

(c) Fourier transform of $G^*G + H^*H$

Figure 3-6: Derivation of energy measure frequency content for the filters of Fig. 3-5. (a) Fourier transform of $G^*G$. (b) Fourier transform of $H^*H$. Each squared response has 3 lobes in the frequency domain, arising from convolution of the frequency domain responses. The center lobe is modulated down in frequency while the two outer lobes are modulated up. (There are two sign changes which combine to give the signs shown in (b).) To convolve $H$ with itself, we flip it in $f_x$ and $f_y$, which interchanges the $+$ and $-$ lobes of Fig. 3-5 (b). Then we slide it over an unflipped version of itself, and integrate the product of the two. That operation will give positive outer lobes, and a negative inner lobe. However, $H$ has an imaginary frequency response, so multiplying it by itself gives an extra factor of $-1$, which yields the signs shown in (b)). (c) Fourier transform of the energy measure, $G^*G + H^*H$. The high frequency lobes cancel, leaving only the baseband spectrum, which has been demodulated in frequency from the original bandpass response. This spectrum is proportional to the sum of the auto-correlation functions of either lobe of Fig. 3-5 (a) and either lobe of Fig. 3-5 (b).

Slide credit: B. Freeman and A. Torralba
Oriented Filters

- Gabor wavelet: \( \psi(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}} e^{j2\pi u_0 x} \)

- Tuning filter orientation:
  \[
  x' = \cos(\alpha)x + \sin(\alpha)y \quad \text{and} \quad y' = -\sin(\alpha)x + \cos(\alpha)y
  \]

Slide credit: B. Freeman and A. Torralba
Simple example

“Steerability” -- the ability to synthesize a filter of any orientation from a linear combination of filters at fixed orientations.

\[ G_\theta = \cos(\theta)G_0 + \sin(\theta)G_{90} \]

Filter Set:

0°  
90°  
Synthesized 30°

Response:

Raw Image


Slide credit: B. Freeman and A. Torralba
**Steerable filters**

Derivatives of a Gaussian:

\[
\begin{align*}
  h_x(x,y) &= \frac{\partial h(x,y)}{\partial x} = \frac{-x}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \\
  h_y(x,y) &= \frac{\partial h(x,y)}{\partial y} = \frac{-y}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}}
\end{align*}
\]

An arbitrary orientation can be computed as a linear combination of those two basis functions:

\[
h_\alpha(x,y) = \cos(\alpha) h_x(x,y) + \sin(\alpha) h_y(x,y)
\]

The representation is “shiftable” on orientation: We can interpolate any other orientation from a finite set of basis functions.

\[
\cos(\theta) + \sin(\theta) = \text{result}
\]

Freeman & Adelson, 1992

Slide credit: B. Freeman and A. Torralba
Steerable filters

Fig. 3. Steerable filter system block diagram. A bank of dedicated filters process the image. Their outputs are multiplied by a set of gain maps that adaptively control the orientation of the synthesized filter.
Local image representations

A pixel

[r, g, b]

An image patch

Gabor filter pair in quadrature

Gabor jet

V1 sketch: hypercolumns


Slide credit: B. Freeman and A. Torralba
Gabor Filter Bank

or = [4 4 4 4];
or = [12 6 3 2];
Not for image reconstruction. It does NOT cover the entire space!

Slide credit: B. Freeman and A. Torralba
Summary

- Sampling
- Gabor wavelets, Steerable filters
Next week

• Image pyramids