Thinking in Frequency

Computer Vision

Jia-Bin Huang, Virginia Tech

Dali: “Gala Contemplating the Mediterranean Sea” (1976)
Administrative stuffs


• Office hours - Jia-Bin (440 Whittemore Hall )
  • Monday at 1:00 PM – 2:00 PM (final project) – sign up here
  • Friday at 3:00 PM – 4:00 PM (lectures, HW discussions)

• MATLAB tutorial session by Akrit
  • Friday 3-4 PM, Whittemore Hall 340A
  • Bring your laptop with MATLAB installed

• HW 1 will be posted tomorrow (Sept 2). Due date: Sept 19.
Previous class: Image Filtering

• Linear filtering is sum of dot product at each position
  • Can smooth, sharpen, translate (among many other uses)

• Gaussian filters
  • Low pass filters, separability, variance

• Attend to details:
  • filter size, extrapolation, cropping

• Noise models and nonlinear image filters
Today’s class

• Review of image filtering in spatial domain
  • Application: representing textures
  • Noise models and nonlinear image filters

• Fourier transform and frequency domain

• Frequency view of filtering

• Image downsizing and interpolation
Demo

Review: questions

Fill in the blanks:

a) _ = D * B
b) A = _ * _
c) F = D * _
d) _ = D * D
Application: Representing Texture
Texture and Material

http://www-cvr.ai.uiuc.edu/ponce_grp/data/texture_database/samples/
Texture and Orientation

http://www-cvr.ai.uiuc.edu/ponce_grp/data/texture_database/samples/
Texture and Scale

http://www-cvr.ai.uiuc.edu/ponce_grp/data(texture_database/samples/)
What is texture?

Regular or stochastic patterns caused by bumps, grooves, and/or markings
How can we represent texture?

• Compute responses of blobs and edges at various orientations and scales
Overcomplete representation: filter banks

Code for filter banks: www.robots.ox.ac.uk/~vgg/research/texclass/filters.html
Filter banks

• Process image with each filter and keep responses (or squared/abs responses)
How can we represent texture?

• Measure responses of blobs and edges at various orientations and scales

• Idea 1: Record simple statistics (e.g., mean, std.) of absolute filter responses
Can you match the texture to the response?

Filters

1

2

3

Mean abs responses

A

B

C
Representing texture by mean abs response
Representing texture

• Idea 2: take vectors of filter responses at each pixel and cluster them, then take histograms (more on this in coming weeks)
Denoising and Nonlinear Image Filtering

- **Salt and pepper noise**: contains random occurrences of black and white pixels

- **Impulse noise**: contains random occurrences of white pixels

- **Gaussian noise**: variations in intensity drawn from a Gaussian normal distribution

Source: S. Seitz
Reducing salt-and-pepper noise

- What’s wrong with Gaussian filtering?
Alternative idea: Median filtering

- A **median filter** operates over a window by selecting the median intensity in the window.

![Median filter diagram](image)

- Is median filtering linear?
Median filter

• Is median filtering linear?
• Let’s try filtering

\[
\begin{array}{ccc}
A & & B \\
1 & 1 & 1 & 0 & 0 & 0 \\
1 & 1 & 2 & + & 0 & 1 & 0 \\
2 & 2 & 2 & & 0 & 0 & 0 \\
\end{array}
\]

Median(A) = 1 \quad Median(B) = 0 \quad Median(A+B) = 2

Violate linearity

\[\text{filter}(f_1 + f_2) \neq \text{filter}(f_1) + \text{filter}(f_2)\]
Median filter

- What advantage does median filtering have over Gaussian filtering?
  - Robustness to outliers

<table>
<thead>
<tr>
<th>Input</th>
<th>Median</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>⋯⋯⋯⋯⋯⋯</td>
<td>⋯⋯⋯⋯⋯ ⋯⋯⋯⋯⋯</td>
<td>⋯⋯⋯⋯⋯⋯</td>
</tr>
</tbody>
</table>

Source: K. Grauman
Median filter

Salt-and-pepper noise  Median filtered

• MATLAB: `medfilt2(image, [h w])`

Source: M. Hebert
Gaussian vs. median filtering

3x3  5x5  7x7

Gaussian

Median
Other non-linear filters

• Weighted median (pixels further from center count less)

• Clipped mean (average, ignoring few brightest and darkest pixels)

• Bilateral filtering (weight by spatial distance and intensity difference)
Bilateral Filters

• Edge preserving: weights similar pixels more

\[
I_p^b = \frac{1}{W_p^b} \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(\|I_p - I_q\|) I_q
\]

\[
W_p^b = \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(\|I_p - I_q\|)
\]

Guided Image Filters

\[ B = \text{imguidedfilter}(A, G); \]

Kaiming He, Jian Sun, Xiaou Tang, Guided Image Filtering. PAMI 2013
Why does the Gaussian give a nice smooth image, but the box filter give edgy artifacts?
Hybrid Images

• A. Oliva, A. Torralba, P.G. Schyns, “Hybrid Images,” SIGGRAPH 2006
Why do we get different, distance-dependent interpretations of hybrid images?
Why does a lower resolution image still make sense to us? What do we lose?
Thinking in terms of frequency
Jean Baptiste Joseph Fourier (1768-1830)

had crazy 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
  • there are some subtle restrictions

...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.
How would math have changed if the Slanket or Snuggie had been invented?

Slide credit: James Hays

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

© 1996-2007 Eric W. Weisstein

I’m wearing it as a joke!
A sum of sines

Our building block:

$$A \sin(\omega x + \phi)$$

Add enough of them to get any signal $f(x)$ you want!
Frequency Spectra

- Example: \( g(t) = \sin(2\pi f t) + (1/3)\sin(2\pi(3f) t) \)
Frequency Spectra
Frequency Spectra
Frequency Spectra
Frequency Spectra

\[\text{Frequency Spectra} = \text{Line} + \text{Wave} = \text{Line} + \text{Wave} = \text{Line} + \text{Wave}\]
Frequency Spectra

\[ \text{Frequency Spectra} \]

\[ \text{Diagram showing frequency spectra.} \]
Frequency Spectra

= 

+ 

= 

=
Frequency Spectra

\[
\sum_{k=1}^{\infty} \frac{1}{k} \sin(2\pi kt)
\]
Example: Music

• We think of music in terms of frequencies at different magnitudes
Other signals

• We can also think of all kinds of other signals the same way

Cats(?)
Fourier analysis in images

Intensity Image

Fourier Image

http://sharp.bu.edu/~slehar/fourier/fourier.html#filtering
Signals can be composed

http://sharp.bu.edu/~slehar/fourier/fourier.html#filtering
More: http://www.cs.unm.edu/~brayer/vision/fourier.html
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

Amplitude: \[ A = \pm \sqrt{R(\omega)^2 + I(\omega)^2} \]

Phase: \[ \phi = \tan^{-1} \frac{I(\omega)}{R(\omega)} \]

Euler’s formula: \[ e^{inx} = \cos(nx) + i \sin(nx) \]
Salvador Dali invented Hybrid Images?

Salvador Dali

“Gala Contemplating the Mediterranean Sea, which at 30 meters becomes the portrait of Abraham Lincoln”, 1976
Log Magnitude

Strong Vertical Frequency (Sharp Horizontal Edge)

Strong Horz. Frequency (Sharp Vert. Edge)

Diagonal Frequencies

Low Frequencies

Log Magnitude
Man-made Scene
Can change spectrum, then reconstruct
Low and High Pass filtering
Computing the Fourier Transform

\[ H(\omega) = \mathcal{F} \{ h(x) \} = Ae^{j\phi} \]

Continuous

\[ H(\omega) = \int_{-\infty}^{\infty} h(x)e^{-j\omega x} \, dx \]

Discrete

\[ H(k) = \frac{1}{N} \sum_{x=0}^{N-1} h(x)e^{-j \frac{2\pi k x}{N}} \quad k = -N/2..N/2 \]

Fast Fourier Transform (FFT): NlogN
The Convolution Theorem

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

\[ F[g \ast h] = F[g] F[h] \]

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

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

• Convolution in spatial domain is equivalent to multiplication in frequency domain!
Properties of Fourier Transforms

• Linearity

\[ F(ax(t) + by(t)) = aF[x(t)] + bF[y(t)] \]

• Fourier transform of a real signal is symmetric about the origin

• The energy of the signal is the same as the energy of its Fourier transform

See Szeliski Book (3.4)
Filtering in spatial domain

\[
\begin{bmatrix}
1 & 0 & -1 \\
2 & 0 & -2 \\
1 & 0 & -1 \\
\end{bmatrix}
\]
Filtering in frequency domain

FFT

Inverse FFT
FFT in Matlab

• Filtering with fft

im = ... % “im” should be a gray-scale floating point image
[imh, imw] = size(im);
fftsize = 1024; % should be order of 2 (for speed) and include padding
im_fft = fft2(im, fftsize, fftsize); % 1) fft im with padding
hs = 50; % filter half-size
fil = fspecial('gaussian', hs\*2+1, 10);
fil_fft = fft2(fil, fftsize, fftsize); % 2) fft fil, pad to same size as image
im_fil_fft = im_fft .* fil_fft; % 3) multiply fft images
im_fil = ifft2(im_fil_fft); % 4) inverse fft2
im_fil = im_fil(1+hs:size(im,1)+hs, 1+hs:size(im, 2)+hs); % 5) remove padding

• Displaying with fft

figure(1), imagesc(log(abs(fftshift(im_fft)))), axis image, colormap jet
Questions

Which has more information, the phase or the magnitude?

What happens if you take the phase from one image and combine it with the magnitude from another image?
Filtering

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

Match the spatial domain image to the Fourier magnitude image

1  2  3  4  5
A  B  C  D  E
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

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

Why does this look so crufty?
Aliasing! What do we do?
Image sub-sampling
Even worse for synthetic images

Source: L. Zhang
Aliasing problem

• 1D example (sinewave):

Source: S. Marschner
Aliasing problem

• 1D example (sinewave):
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”
Aliasing

- Occurs when your sampling rate is not high enough to capture the amount of detail in your image
- Can give you the wrong signal/image—an alias

- To do sampling right, need to understand the structure of your signal/image
- To avoid aliasing:
  - sampling rate $\geq 2 \times$ max frequency in the image
    - said another way: $\geq$ two samples per cycle
  - This minimum sampling rate is called the Nyquist rate

Source: L. Zhang
Wagon-wheel effect

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)

(See http://www.michaelbach.de/ot/mot_wagonWheel/index.html)
Wagon-wheel effect

https://www.youtube.com/watch?v=Q0wzkND_ooU
Sampling an image

Examples of GOOD sampling
Undersampling

Examples of BAD sampling -> Aliasing
Anti-aliasing

Forsyth and Ponce 2002
Gaussian (low-pass) pre-filtering

Solution: filter the image, \textit{then} subsample
Subsampling with Gaussian pre-filtering

• Solution: filter the image, then subsample

Source: S. Seitz
Compare with...

1/2

1/4  (2x zoom)

1/8  (4x zoom)

Source: S. Seitz
Why does a lower resolution image still make sense to us? What do we lose?

Why do we get different, distance-dependent interpretations of hybrid images?
Clues from Human Perception

- Early processing in humans filters for various orientations and scales of frequency
- Perceptual cues in the mid-high frequencies dominate perception
- When we see an image from far away, we are effectively subsampling it

Early Visual Processing: Multi-scale edge and blob filters
Hybrid Image in FFT

Hybrid Image

Low-passed Image + High-passed Image
Upsampling

• This image is too small for this screen:
• How can we make it 10 times as big?
• Simplest approach:
  repeat each row
  and column 10 times
• ("Nearest neighbor interpolation")
Recall how a digital image is formed

\[ F[x, y] = \text{quantize}\{f(xd, yd)\} \]

- It is a discrete point-sampling of a continuous function
- If we could somehow reconstruct the original function, any new image could be generated, at any resolution and scale

Adapted from: S. Seitz
Recall how a digital image is formed

\[ F[x, y] = \text{quantize}\{f(xd, yd)\} \]

- It is a discrete point-sampling of a continuous function
- If we could somehow reconstruct the original function, any new image could be generated, at any resolution and scale

Adapted from: S. Seitz
**Image interpolation**

- What if we don’t know $f$?
  - Guess an approximation: $\tilde{f}$
  - Can be done in a principled way: filtering
  - Convert $F$ to a continuous function:
    \[ f_F(x) = F\left(\frac{x}{d}\right) \text{ when } \frac{x}{d} \text{ is an integer, } 0 \text{ otherwise} \]
  - Reconstruct by convolution with a reconstruction filter, $h$
    \[ \tilde{f} = h \ast f_F \]

Adapted from: S. Seitz
Image interpolation

-sinc(x) - “Ideal” reconstruction

-\[ II(x) \] - Nearest-neighbor interpolation

-\[ \Lambda(x) \] - Linear interpolation

-gauss(x) - Gaussian reconstruction

Source: B. Curless
Reconstruction filters

• What does the 2D version of this hat function look like?

\[ h(x) \]

performs linear interpolation

\[ h(x, y) \]

(tent function) performs \textbf{bilinear interpolation}

Often implemented without cross-correlation

• E.g., \url{http://en.wikipedia.org/wiki/Bilinear_interpolation}

Better filters give better resampled images

• \textbf{Bicubic} is common choice
Image interpolation

Original image:  x 10

Nearest-neighbor interpolation  Bilinear interpolation  Bicubic interpolation
Image interpolation

Also used for *resampling*
Things to Remember

• Sometimes it makes sense to think of images and filtering in the frequency domain
  • Fourier analysis

• Can be faster to filter using FFT for large images (N logN vs. N^2 for auto-correlation)

• Images are mostly smooth
  • Basis for compression

• Remember to low-pass before sampling
Thank you

- Enjoy your long weekend!

- Next class:
  - Pyramid, template matching