Frequency analysis, pyramids, texture analysis, applications (face detection, category recognition)
Outline

– Measuring frequencies in images: Definitions, properties
– Sampling issues
– Relation with Gaussian smoothing
– Image pyramids
– Gabor filters and wavelets
– Examples: Texture classification and category recognition
A representation for image changes

- We need a change of basis to move from pixel intensities (space domain) to frequency domain.

“A measure of image content at this frequency and orientation

“dot”
Fourier basis

...etc.
• More formally, content at frequency \( u \) is obtained by taking the magnitude of the “dot product” with the function \( \cos 2\pi u \)

• Same with \( \sin 2\pi u \)
Combining cos and sin: Fourier Transform

1-D case:

\[ F(u) = \int f(x) e^{-i2\pi ux} \, dx \]

\[ f(x) = \int F(u) e^{i2\pi ux} \, du \]

2-D case:

\[ F(u, v) = \iint f(x, y) e^{-i2\pi (ux + vy)} \, dx \, dy \]

\[ f(x, y) = \iint F(u, v) e^{i2\pi (ux + vy)} \, du \, dv \]
Examples (1D)
Example from Ponce & Forsyth
Example from Ponce & Forsyth
Questions

• How do discrete images differ from continuous images?
• How do we avoid aliasing while sampling?
Sampling an image

Examples of GOOD sampling
Undersampling

Examples of BAD sampling -> Aliasing
Constructing a pyramid by taking every second pixel leads to layers that badly misrepresent the top layer.
Low-pass filtering before sampling

- The minimum frequency at which we must sample a signal in order to be able to fully reconstruct it called the **Nyquist frequency**.
- Nyquist frequency = 2 times the maximum frequency contained in the waveform.
- The message of the FT is that high frequencies lead to trouble with sampling.

- Solution: suppress high frequencies before sampling
  - multiply the FT of the signal with something that suppresses high frequencies
  - or convolve with a low-pass filter
- Common solution: use a Gaussian
  - multiplying FT by Gaussian is equivalent to convolving image with Gaussian.
Questions

• How can we represent images at multiple scales?
Sampling without smoothing. Top row shows the images, sampled at every second pixel to get the next; bottom row shows the magnitude spectrum of these images.
Sampling with smoothing. Top row shows the images. We get the next image by smoothing the image with a Gaussian with sigma 1 pixel, then sampling at every second pixel to get the next; bottom row shows the magnitude spectrum of these images.
Representation of scale
Representation of scale

The Gaussian pyramid

- Smooth with gaussians, because
  - a gaussian \times gaussian = another gaussian
- Synthesis
  - smooth and subsample
- Analysis
  - Start with the top image (coarse) and move to lower (fine) image layers

Applications

- Search for correspondence
  - look at coarse scales, then refine with finer scales
  - Lowers computational cost
- Edge tracking
  - a “good” edge at a fine scale has parents at a coarser scale
Laplacian Pyramids

- Given input $I$
- Construct Gaussian pyramid $I^G_1, \ldots, I^G_n$
- Take the difference between consecutive levels:
  - $I^L_k = I^G_k - I^G_{k-1}$
- Image $I^L_k$ is an approximation of the Laplacian at scale number $k$
  - Laplacian is a band-pass filter: Both high frequencies (edges and noise) and low frequencies (slow variations of intensity across the image)
Laplacian Pyramids
Frequency/Scale Tradeoff: Gabor filters and wavelets
Low frequency content in all directions

High frequency at 45° orientation

Fourier Transform = content of entire image at all frequencies and orientations
Frequency content in local neighborhood at every point in image

\[ I' = I \ast \text{STFT} \]
Frequency content in local neighborhood at every point in image

\[ I' = I \ast \text{Gabor} \]
Gabor Filter Example

Odd (antisymmetric)

\[ e^{-\frac{x^2}{2\sigma^2}} \sin (2\pi \omega x) \]

Even (Symmetric)

\[ e^{-\frac{x^2}{2\sigma^2}} \cos (2\pi \omega x) \]

\[ e^{-\frac{x^2}{2\sigma^2}} + 2\pi i \omega x \]
High frequency along axis

Lower frequency along diagonal

Even lower frequency

\[ e^{-\frac{x^2+y^2}{2\sigma^2}} \cos (2\pi (k_xx + k_yy)) \]
Odd Gabor filter

First Derivative
Even Gabor filter

Laplacian
If scale small compared to inverse frequency, the Gabor filters become derivative operators

\[ \sigma = 2 \quad f = 1/6 \]

\[ \approx G_x^\sigma \]

\[ \approx G_{xx}^\sigma \]
Can we select the frequency and the scale $\sigma$ arbitrarily?
Frequency/scale mismatch: $1/\text{frequency} \gg \text{scale}$

Gabor filter
High frequency

Frequency/scale mismatch:
$1/\text{frequency} \ll \text{scale}$

Large scale

Gabor filter
Wavelets: Scale and frequency are made consistent by scaling one basis filter. The scale of the filter is changed and, at the same time, the frequency is also changed.
Example
Example: Face Detection

Input

Wavelet decomposition

From Henry Schneiderman
Example: Texture Classification

- Profound observation: Cows and buildings don’t look the same!
- Basic idea: Model the distribution of “texture” over the image (or over a region) and classify in different classes based on the texture models learned from training examples.
The Concept of “Texton”

Multiple training images of the same texture

Filter responses over a bank of filters

Clustering

Texton Dictionary
Example of Filter Banks

Isotropic Gabor

Gaussian derivatives at different scales and orientations

'S'

'LM'

'MR8'
Example Textons (LM)
Example: Visual words in photographs

Images

Word maps

Visual dictionary
Modeling Texton Distributions

Training image

Filter Responses

Histogram of textons in the image

Model = Histogram of textons in the image
Analogy with Text Analysis

Political observers say that the government of Zorgia does not control the political situation. The government will not hold elections …
The ZH-20 unit is a 200Gigahertz processor with 2Gigabyte memory. Its strength is its bus and high-speed memory......
Classification

Input Image (or Region of an Input Image)

Model

Compare with Stored Models from Training Images

Models of Plastic

Models of Grass
Example Classification

Input Region

Filter responses

Texton Map

Histogram

Model

Textons

Tree

Cow

Building

Car
Examples
## Example Performance (Confusion Matrix)

### Inferred label

<table>
<thead>
<tr>
<th>True label</th>
<th>Building</th>
<th>Grass</th>
<th>Tree</th>
<th>Cow</th>
<th>Sky</th>
<th>Aeroplane</th>
<th>Face</th>
</tr>
</thead>
<tbody>
<tr>
<td>Building</td>
<td>22</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Grass</td>
<td>62</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tree</td>
<td></td>
<td></td>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cow</td>
<td></td>
<td></td>
<td></td>
<td>22</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Sky</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>44</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aeroplane</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>Face</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>
• Sources:

• Current research topics:
  – How many textons/words?
  – What filters?
  – How to construct clusters?
  – How to compare histogram distributions?
  – How to exploit the spatial distribution of textons (these examples completely ignore the relative positions of textons in the image)?
\[ G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \]

**Gaussian**

Separable, low-pass filter

**Derivatives of Gaussian**

\[ \frac{\partial G_\sigma(x, y)}{\partial x} \propto x e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \frac{\partial G_\sigma(x, y)}{\partial y} \propto y e^{-\frac{x^2+y^2}{2\sigma^2}} \]

\[ \nabla G_\sigma = \left[ \frac{\partial G_\sigma}{\partial x} \quad \frac{\partial G_\sigma}{\partial y} \right] \]

**Laplacian**

Not-separable, approximated by A difference of Gaussians. Output of convolution is Laplacian of image: Zero-crossings correspond to edges

\[ \nabla^2 G_\sigma(x, y) = \frac{\partial^2 G_\sigma(x, y)}{\partial x^2} + \frac{\partial^2 G_\sigma(x, y)}{\partial y^2} \]

**Fourier Transform**

\[ F(g)(u,v) = \int g(x, y)e^{-2\pi i(ux+vy)} \, dudv \]

\[ F(f * g) = F(f) \cdot F(g) \quad F(\frac{\partial g}{\partial x}) = uF(g) \]

**Directional Derivatives**

\[ \cos \theta \frac{\partial G_\sigma}{\partial x} + \sin \theta \frac{\partial G_\sigma}{\partial y} \]

Output of convolution is magnitude of derivative in direction \( \theta \). Filter is linear combination of derivatives in \( x \) and \( y \).
Directional smoothing

\[ e = \frac{e_{1x+b_1y} + \sigma_1^2}{\sigma_2^2} - \frac{e_{2x+b_2y} + \sigma_1^2}{\sigma_2^2} \]

Steerability

Generalization of property of derivatives: \( \Phi \) is steerable if the rotated filter can be expressed as a linear combination of basis filters.

\[ \Phi_\theta = \sum_i a_i \Theta_i \Phi_i \]

Gabor Filters

Even: \( \cos(2\pi(k_x x + k_y y)e^{-\frac{x^2+y^2}{2\sigma^2}}) \)

Odd: \( \sin(2\pi(k_x x + k_y y)e^{-\frac{x^2+y^2}{2\sigma^2}}) \)

Compute the local contribution of frequency \( f = \sqrt{k_x^2 + k_y^2} \) in the direction \((k_x, k_y)\) at scale \( \sigma \).

Even filters approximate 2nd derivative, odd filters approximate 1st derivative.

Wavelets

Decompose the image locally using

Replication of a basis function over scale

\[ \psi_s(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t}{s}\right) \]

\[ \psi_k(t) = 2^{-\frac{k}{2}} \psi(2^{-k} t) \]

Gaussian Pyramids

Gaussian smooth image and subsample at each stage

\[ I_{k+1} = S \downarrow G_\sigma \ast I_k \]

Laplacian Pyramids

Compute Laplacian by difference of Gaussian at each stage

\[ L_k = I_k - S \uparrow G_\sigma \ast I_{k+1} \]