Announcements / Questions

• ???
Image Frequencies

- Low frequency components = slow changes in pixel intensity
  - broad areas of uniform intensity / color

- High frequency components = rapid changes in pixel intensity
  - sharp or abrupt changes in intensity / color
  - i.e., edges!

- Note: we will not use frequency-domain representations of images to any major degree
Example: Low Frequency Content
Example:
High Frequency Content
Convolution Kernels

• An output, or new, pixel value is computed from the input value of neighboring pixels
  – ex.: a weighted sum of a neighborhood of values

• Convolution of an N x N matrix (kernel) with the image
  – “weights” = coefficients of the kernel
  – coefficients of the kernel determine its function

• Kernel size choice
  – smaller kernel $\rightarrow$ less computation
  – larger kernel $\rightarrow$ higher quality results
Convolution Illustration

FIGURE 3.1 How a convolution is performed.
Convolution Example

A small portion of an image
(pixel values shown are greyscale)

A 3x3 convolution kernel

Some of the output pixel values produced by convolving the image with the kernel
Convolution (cont’d)

- How deal with edge pixels (boundaries, again!)?
  - zero fill (add black border around image), or
  - duplicate the edge pixels, or
  - don’t process the edges!

- What does convolution of \textit{color} images mean?
  1. filter the luminance channel only; do not change the chroma information
  2. convolve each of R, G, and B independently
Smoothing (Blurring) Filters

- **Smoothing** = eliminate details (high frequencies)
  - eliminates pixelation effects, other noise
  - low-pass filters

- **Convolution kernels for smoothing**
  - Usually, sum of kernel coefficients = 1
  - i.e., preserves average image intensity

- **Examples of smoothing**
  - simple averaging (all pixels weighted equally)
  - gaussian blurring (coefficient values approximate the normal distribution)
Example: Smoothing (Simple Averaging)
Example: Gaussian Smoothing
Example: Smoothing of “Noise”
Sharpening (Edge Detection) Filters

• Edge-detection, enhancement, or sharpening
  – preserve the abrupt change (edges)
  – same as remove the areas of constant / similar color
  – high-pass filter

• Convolution kernels for sharpening filters
  – generally, the sum of kernel coefficients = 0
    • average image intensity almost 0 (black)
  – both positive and negative coefficients
    • differences in signs emphasizes differences (first-order derivative) in pixel values

• Any “negative” values that result saturate at 0 (black)
Examples: Edge Detection
Edge Detection Kernels

- Some first order (gradient) kernels
  - Prewitt row
    \[
    \begin{pmatrix}
    1 & 0 & -1 \\
    1 & 0 & -1 \\
    1 & 0 & -1 \\
    \end{pmatrix}
    \]
  - Sobel row
    \[
    \begin{pmatrix}
    1 & 0 & -1 \\
    2 & 0 & -2 \\
    1 & 0 & -1 \\
    \end{pmatrix}
    \]

- Combined row and column operators for arbitrary direction
Prewitt Result
Sobel Result
Edge Direction

- Assymmetric kernels detect edges from specific directions
- Example (Prewitt):

NorthWest:

```
-1  -1  1
-1  -2  1
 1  1  1
```

North:

```
-1  -1  -1
 1  -2  1
 1  1  1
```

NorthEast:

```
1   -1  -1
1   -2  -1
 1  1  1
```

West:

```
-1  1  1
-1  -2  1
-1  1  1
```

East:

```
 1  1  -1
 1  -2  -1
 1  1  -1
```
Directional Kernel Results
Second Order Derivative Kernels

• Second order kernels
  – Non-directional
  – Results in closed curves (contours)
  – Example: Laplacian

\[
\begin{pmatrix}
  0 & -1 & 0 \\
  -1 & 4 & -1 \\
  -1 & 8 & -1 \\
\end{pmatrix} \quad \begin{pmatrix}
  0 & -1 & 0 \\
  -1 & 4 & -1 \\
  -1 & 8 & -1 \\
\end{pmatrix}
\]

• Replace output pixel values with sign changes (zero crossings)
LAPLACIAN Example

A B C D E F G

-1 -1 -1
-1 8 -1
-1 -1 -1
Edge “Sharpening”

- Mix edges detected with some amount of original image
Discrete Cosine Transform (DCT)

- A frequency transform
  - uses a different set of basis functions than the Fourier transform
- Fundamental part of JPEG (image) and MPEG (video) compression
- More when we talk about JPEG next time...
“Statistical” Filters

- *Not* based on the convolution operation

- **Example:** median filter is used for smoothing
  - preserves edges better than blurring

- **Implementing median filter**
  1. sort pixel values in a region or *neighborhood*
  2. find the median value
  3. use this as the value of the pixel in the middle of the neighborhood
  - neighborhood size and shape?
Median Filter Example

A small portion of an image
(pixel values shown are greyscale)

Select median value of 3x3 neighborhood

Some of the output pixel values produced by median filtering over a 3x3 neighborhood

<table>
<thead>
<tr>
<th>50</th>
<th>80</th>
<th>110</th>
<th>150</th>
</tr>
</thead>
<tbody>
<tr>
<td>70</td>
<td>100</td>
<td>110</td>
<td>180</td>
</tr>
<tr>
<td>100</td>
<td>120</td>
<td>130</td>
<td>200</td>
</tr>
<tr>
<td>200</td>
<td>220</td>
<td>240</td>
<td>250</td>
</tr>
</tbody>
</table>
Median Filter Example

Before  After
Median Filter of Noise
Other “Statistical” Filters

1. Minimum filter
   - Replace pixel value with the minimum (darkest) value of its neighborhood
   - result: “thinning” of the bright areas, “growing” of dark areas

2. Maximum filter
   - replace pixel value with the maximum (brightest) value in its neighborhood
   - result: growing the bright areas, thinning the dark areas

3. “Pixellate” functions
   - Related to the median filter
Max / Min
Examples

Original

After applying “Maximum filter”

After applying “minimum filter”
Pixellate Examples
Image Transforms

\[ H(u, v) = \frac{1}{MN} \left[ \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} h(x, y) \cos \left( \frac{2\pi ux}{M} + \frac{2\pi vy}{N} \right) - \right. \]
\[ \left. j \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} h(x, y) \sin \left( \frac{2\pi ux}{M} + \frac{2\pi vy}{N} \right) \right] \]

- \( H(u, v) \) is component with frequency \( u < M/2, \ v < N/2 \)
  - \( H(0,0) \) mapped to center of the image

- Computing the transform: correlate with the 2-D sine and cosine wave functions, as before

- \( M, N \) are size of image

- \( h(x,y) \) is pixel intensity at position \( x,y \)
Transform Example

FIGURE 7.5  Fourier transform of a spot: (a) original image; (b) Fourier transform.
Filtering In The Frequency Domain

• ...for another day...
Sources Of Info

- **Recommended**
  - [Crane97] *A Simplified Approach to Image Processing*
    - Chapter 3

- **Optional**
  - [Smith97] *The Scientist and Engineer’s Guide to Digital Signal Processing*
    - Chapters 23-25 (selected parts discussed in lecture)
  - [Watkins97] *Modern Image Processing*