Image Convolution

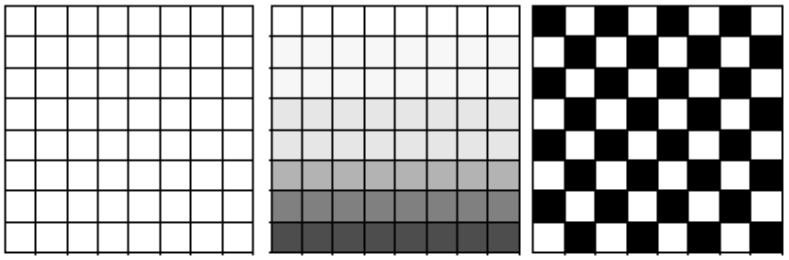
Jamie Ludwig
Satellite Digital Image Analysis, 581
Portland State University

Key words

- Filtering
- Convolution
- Matrix
- Color values
- kernel

Spatial frequencies

- Convolution filtering is used to modify the spatial frequency characteristics of an image.



What is convolution?

- Convolution is a general purpose filter effect for images.
- Is a matrix applied to an image and a mathematical operation comprised of integers
- It works by determining the value of a central pixel by adding the weighted values of all its neighbors together
- The output is a new modified filtered image

The process of image convolution

- A convolution is done by multiplying a pixel's and its neighboring pixels color value by a matrix
- **Kernel:** A kernel is a (usually) small matrix of numbers that is used in image convolutions.
  - Differently sized kernels containing different patterns of numbers produce different results under convolution.
  - The size of a kernel is arbitrary but 3x3 is often used

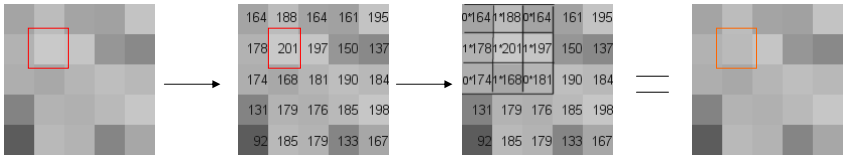
Example kernel:

0	1	0
1	1	1
0	1	0

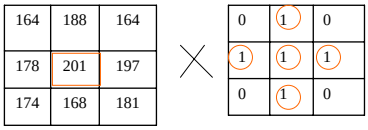
Why convolve an image?

- Smooth
- Sharpen
- Intensify
- Enhance

Example



Original image | Image with color values placed over it | Image with 3x3 kernel placed over it | Output image



Color values | Kernel

Divided by the sum of the kernel

932\5 = new pixel color

Convolution Formula

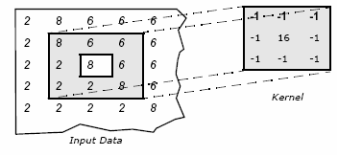
$$V = \left| \frac{\sum_{i=1}^{q} \left(\sum_{j=1}^{q} f_{ij} d_{ij} \right)}{F} \right|$$

Where:

- f_{ij} = the coefficient of a convolution kernel at position i,j (in the kernel)
- d_{ij} = the data value of the pixel that corresponds to f_{ij}
- q = the dimension of the kernel, assuming a square kernel (if q = 3, the kernel is 3 × 3)
- F = either the sum of the coefficients of the kernel, or 1 if the sum of coefficients is 0
- V = the output pixel value

In cases where V is less than 0, V is clipped to 0.

More examples



integer [(-1 × 8) + (-1 × 6) + (-1 × 6) + (-1 × 2) + (16 × 8) + (-1 × 6) +
(-1 × 2) + (-1 × 2) + (-1 × 8) ÷ (-1 + -1 + -1 + -1 + 16 + -1 + -1 + -1 + -1)]

= int [(128-40) / (16-8)]

= int (88 / 8) = int (11) = 11

What do we do with edge pixels?

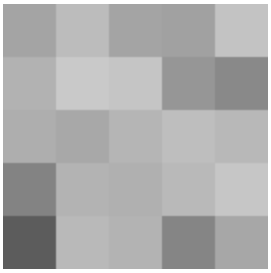
- Wrap the image
- Ignore edge pixels and only compute for those pixels with all neighbors
- Duplicate edge pixels so the pixel at (2,n) (where n would be non-positive will have a value of (2,1)

-1 = no data

	-1	-1	-1	-1	-1	
-1	164	188	164	161	195	-1
-1	178	201	197	150	137	-1
-1	174	168	181	190	184	-1
-1	131	179	176	185	198	-1
-1	92	185	179	133	167	-1
	-1	-1	-1	-1	-1	

	1	1	1	1	1	
1	164	188	164	161	195	1
1	178	201	197	150	137	1
1	174	168	181	190	184	1
1	131	179	176	185	198	1
1	92	185	179	133	167	1
	1	1	1	1	1	

- Original Image



- Smoothed modified image

Some other kernel examples

1	1	1
1	1	1
1	1	1

Unweighted 3x3 smoothing kernel

0	1	0
1	4	1
0	1	0

Weighted 3x3 smoothing kernel with Gaussian blur

0	-1	0
-1	5	-1
0	-1	0

Kernel to make image sharper

-1	-1	-1
-1	9	-1
-1	-1	-1

Intensified sharper image

Gaussian Blur

Sharpened image

Example of smoothing kernel

- A larger kernel area when using a smoothing kernel increases smoothing area

5x5 smoothing kernel

0	1	2	1	0
1	4	8	4	1
2	8	16	8	2
1	4	8	4	1
0	1	2	1	0

Main points

- Start out with an image
- The choice of kernel affects the output image
- Base your choice of kernel on the desired results for the image (smooth, blur, enhance, sharpen)
- Low Pass and high pass filters will be discussed later in the class

Pre-what?

Sources

- http://www.dspguide.com/ch24/1.htm
- http://wally.cs.iupui.edu/n351/raster/filterDemo.html
- **wally.cs.iupui.edu/n351/raster/filterDemo.html**
- http://www.websupergoo.com
- Mather, P. M. 2004. Computer Processing of Remotely Sensed Images, An Introduction. West Sussex. John Wiley & Sons Ltd.