EEE-6512: Image Processing and Computer Vision

October 9, 2017

Lecture #7: Spatial Domain Filtering

Damon L. Woodard, Ph.D.
Dept. of Electrical and Computer Engineering
dwoodard@ece.ufl.edu

Chapter Outline

- Convolution
- Smoothing by Convolving with a Gaussian
- Computing the First Derivative
- Computing the Second Derivative
- Nonlinear Filters
- Grayscale Morphological Operators

Preliminaries

Limitations of Point Operations

- They don’t know where they are in the image
- They don’t know anything about their neighbors
- Most image features (edges, textures, etc.) involve the use of a spatial neighborhood of pixels
- If we want to manipulate or enhance these features, we need to go beyond point operations.
- Things that point operators cannot do:
  - **Blurring or Smoothing**
  - **Sharpening**

Spatial Filtering

- The word “filtering” has been borrowed from the frequency domain.
- Filters are classified as:
  - **Low-pass** (i.e., preserve low frequencies)
  - **High-pass** (i.e., preserve high frequencies)
  - **Band-pass** (i.e., preserve frequencies within a band)
  - **Band-reject** (i.e., reject frequencies within a band

Preliminaries

- A **spatial filter** is an image operation where each pixel value $I(u,v)$ is changed by a function of the intensities of pixels in a neighborhood of (u,v).

Spatial Filtering Methods

- Two types of Spatial Filtering (**Linear and Non-Linear**)
- Two types of Linear Spatial Filtering Method (**Correlation and Convolution**)

Key Properties of Linear Filters

Linearity:

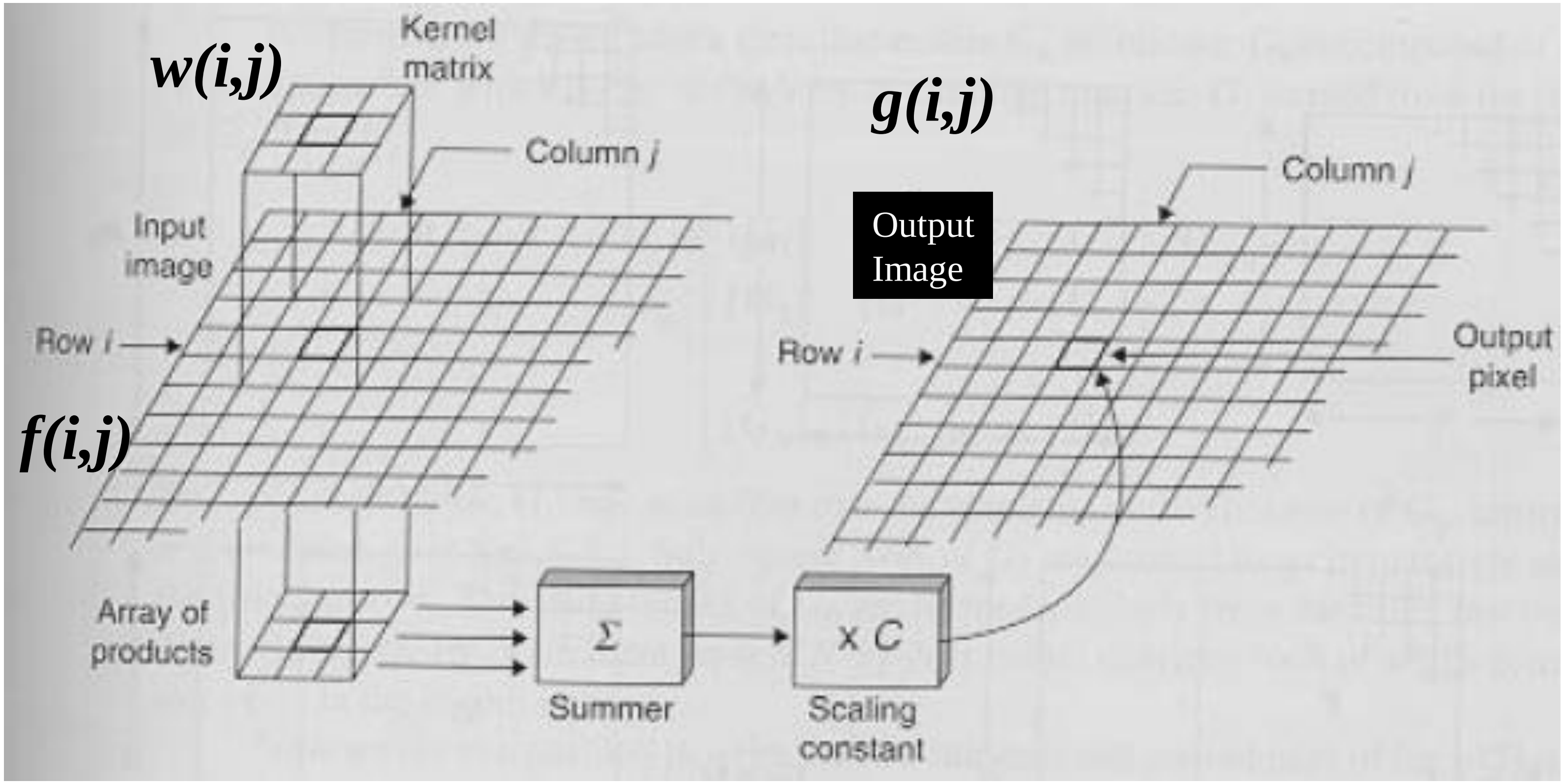
$\texttt{filter}(f_1 + f_2) = \texttt{filter}(f_1) + \texttt{filter}(f_2)$

Shift invariance: same behavior regardless of pixel location

`filter(shift(f)) = shift(filter(f))`

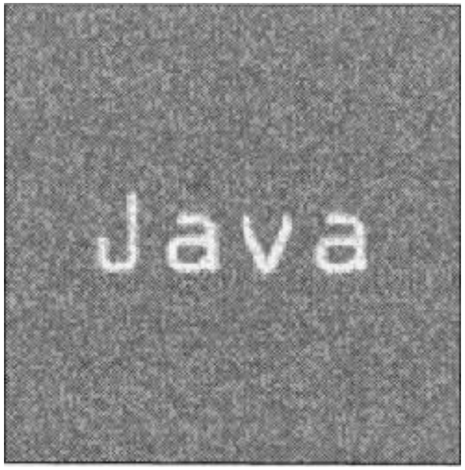
Any linear, shift-invariant operator can be represented as a convolution

Correlation



$$g(i,j) = w(i,j) \bullet f(i,j) = \sum_{s=-K/2}^{K/2} \sum_{t=-K/2}^{K/2} w(s,t) f(i+s, j+t)$$

Correlation (cont.)



Often used in applications where we need to measure the similarity between images or parts of images (e.g., template matching).

Convolution

- Similar to correlation except that the mask is first **flipped** both horizontally and vertically.

$$g(i,j) = w(i,j) * f(i,j) = \sum_{s=-K/2}^{K/2} \sum_{t=-K/2}^{K/2} w(s,t) f(i-s, j-t)$$

Note: if w(i, j) is symmetric, that is w(i, j)=w(-i,-j), then convolution is equivalent to correlation!

Convolution

1D Convolution

- The discrete convolution of a 1D signal *f* with a kernel *g* is defined as:

$$f'(x) = f(x) \circledast g(x) \equiv \sum_{i=-\infty}^{\infty} f(x-i)g(i)$$
$$= \sum_{i=-\tilde{w}}^{w-\tilde{w}-1} f(x-i)g(i)$$

- The **origin** $\tilde{w}$ of the kernel indicates the location where the result is stored, which is usually defined to be the index nearest the center.

$$\tilde{w} \equiv \left\lfloor \tfrac{1}{2}(w-1) \right\rfloor$$

1D Convolution (cont'd)

- Convolution is closely related to **cross-correlation**, which is defined as:

$$f'_{corr}(x) = f(x) \check{\circledast} g(x) \equiv \sum_{i=-\infty}^{\infty} f^*(x+i)g(i) = \sum_{i=-\bar{w}}^{w-\bar{w}-1} f^*(x+i)g(i)$$

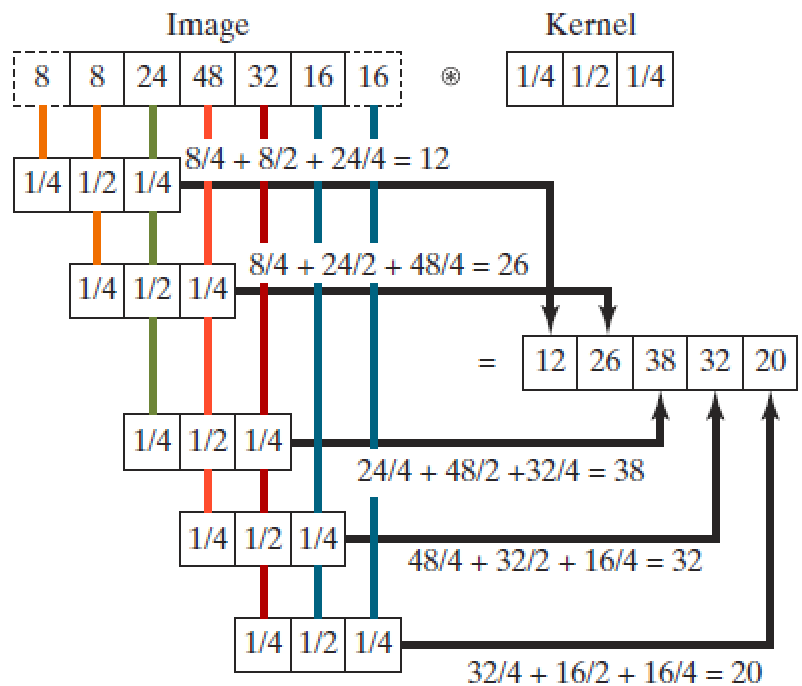
- (*) indicates the complex conjugate.

1D Convolution (cont'd)

- **Smoothing kernels:** perform an averaging of the values in a local neighborhood and therefore reduce the effects of noise.
  - Such kernels are often used as the first stage of preprocessing an image that has been corrupted by noise, in order to restore the original image.
- **Differentiating kernels:** accentuate the places where the signal is changing rapidly in value.
  - They are therefore used to extract useful information from images, such as the boundaries of objects, for purposes such as object detection.

1D Convolution (cont'd)

Figure 5.1 An example of 1D convolution.



Convolution as Matrix Multiplication

- Sometimes it is convenient to view discrete convolution as the multiplication of a matrix by a vector to produce another vector:
  - the input vector is formed from the original signal,
  - the matrix is formed from the convolution kernel,
  - the output vector is the result of the convolution.
- Example:

$$\underbrace{\begin{bmatrix} 12 \\ 26 \\ 38 \\ 32 \\ 20 \end{bmatrix}}_{f'} = \underbrace{\frac{1}{4}\begin{bmatrix} 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 & 1 \end{bmatrix}}_{\substack{\textit{convolution matrix} \\ \mathbf{G}}} \underbrace{\begin{bmatrix} 8 \\ 8 \\ 24 \\ 48 \\ 32 \\ 16 \\ 16 \end{bmatrix}}_{f}$$

Convolution as Fourier Multiplication

- Convolution in the spatial domain is equivalent to multiplication in the frequency domain.

$$f'(x) = f(x) \circledast g(x) = \mathcal{F}^{-1}\{\mathcal{F}\{f(x)\} \cdot \mathcal{F}\{g(x)\}\}$$

Linear Versus Nonlinear Systems

- A system is said to be **linear** if both the *scaling* and *additivity* properties hold for all possible inputs:

$$\mathcal{L}(\alpha f) = \alpha \mathcal{L}(f) \qquad \text{(scaling)}$$

$$\mathcal{L}(f_1 + f_2) = \mathcal{L}(f_1) + \mathcal{L}(f_2) \quad \text{(additivity)}$$

- Together, these properties are referred to as *superposition:*

$$\mathcal{L}(\alpha_1 f_1 + \alpha_2 f_2) = \alpha_1 \mathcal{L}(f_1) + \alpha_2 \mathcal{L}(f_2) \quad \text{(superposition)}$$

- If a system is not linear, then it is said to be **nonlinear**.

Linear Versus Nonlinear Systems (cont'd)

- A system is called **shift-invariant** if a shift in the input causes a shift in the output by the same amount.

$$f'(x - x_0) = \mathcal{L}(f(x - x_0))$$

- **Linear shift-invariant systems:** systems that are particularly important due to their convenient mathematical properties.

$$f'(x) = f(x) \circledast g(x) = \sum_{i=-\infty}^{\infty} f(x - i)g(i)$$

2D Convolution

- **2D Convolution:** used to perform filtering on a 2D image.

$$I'(x,y) = I(x,y) \circledast G(x,y) = \sum_{i=0}^{w-1}\sum_{j=0}^{h-1} I(x+\tilde{w}-i, y+\tilde{h}-j)G(i,j)$$

where *w* and *h* are the width and height of the kernel, respectively.

Questions?