IMAGE FORMATION

1

SAMPLING & CONVOLUTION

- 2D signals in time and frequency domain
- Spatial sampling – resizing an image
- Intensity quantization
  - how many bits used to store a pixel in an image
  - impact on the quality of the image
- Correlation and convolution
- Location of specific patterns using a template (cross correlation)
- Filtering techniques

2

IMAGE FORMATION

- Sampling
- Quantization
- Resize an image with sampling
- Quantize colors

3

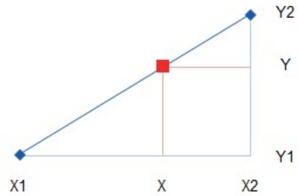
SAMPLING

- Selection/rejection of image pixels
- Spatial operation
- Increase or reduce the size of an image (up sampling, down sampling)
- Upsampling
  - New larger images will have some pixels that have no corresponding pixels in the original smaller image
  - Guess those pixel values
    - An aggregate - the mean value of its nearest known one or more pixel-neighbor values
    - An interpolated value using pixel-neighbors with bilinear or cubic interpolation
  - Nearest neighbor-based up-sampling may result in a poor quality output image

4

UPSAMPLING & INTERPOLATION

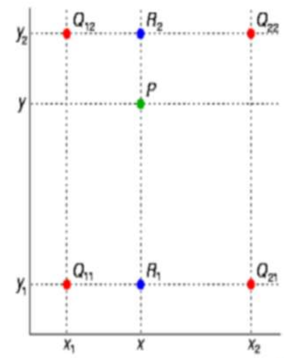
- Bi-linear interpolation
  - 2D analogue of linear interpolation



$$\frac{(X - X1)}{(X2 - X1)} = \frac{(Y - Y1)}{(Y2 - Y1)}$$

$$Y = Y1 + (X - X1)\frac{(Y2 - Y1)}{(X2 - X1)}$$

$$\begin{aligned} f(x,y) \approx & \frac{f(Q_{11})}{(x_2-x_1)(y_2-y_1)}(x_2-x)(y_2-y) \\ & + \frac{f(Q_{21})}{(x_2-x_1)(y_2-y_1)}(x-x_1)(y_2-y) \\ & + \frac{f(Q_{12})}{(x_2-x_1)(y_2-y_1)}(x_2-x)(y-y_1) \\ & + \frac{f(Q_{22})}{(x_2-x_1)(y_2-y_1)}(x-x_1)(y-y_1). \end{aligned}$$

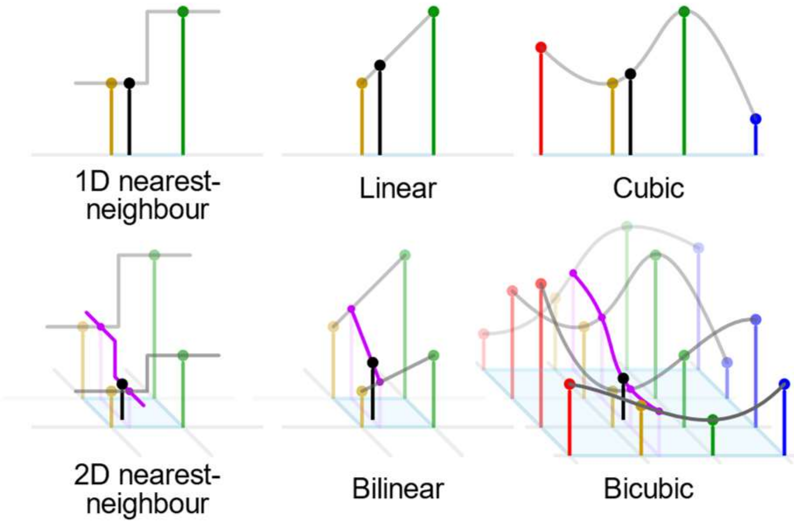


5

UPSAMPLING & INTERPOLATION

- Bi-cubic interpolation
  - Extension of cubic interpolation
  - Interpolated surface is smoother than corresponding surfaces obtained by bi-linear or nearest neighbor interpolation
  - Accomplished using Lagrange polynomials, cubic splines, or cubic convolution algorithm
  - Cubic spline interpolation in a 4 x 4 environment

6



INTERPOLATION

7

DOWN-SAMPLING

- For each pixel in a new smaller image – multiple pixels in the original larger image
- Dropping some pixels from the larger image in a systematic way
  - For example, dropping every other row and column if we want an image a fourth of the size of the original image
- Computing the new pixel value as an aggregate value of the corresponding multiple pixels in the original image
- not very good for shrinking images
  - creates an aliasing effect
  - patchy and bad output
- Anti-aliasing
  - single pixel in the output image corresponds to 25 pixels in the input image, but sampling the value of a single pixel instead
  - averaging over a small area in the input image, ANTIALIAS – high-quality down-sampling filter
  - Smoothing an image – LPF such as Gaussian filter before down-sampling

8

QUANTIZATION

- Related to the intensity of an image
- Number of bits used per pixel
- Typically quantized to 256 gray levels
- As number of bits for pixel storage decreases, the quantization error increases
- Artificial boundaries or contours and pixelating – poor quality
- Color quantization – convert () with the P mode, color argument as the maximum number of possible colors
- Signal to Noise ratio (SNR) – mean divided by the standard deviation
- Image quality decreases as the number of bits to store a pixel reduces
  - Reduces image size (number of bits/pixel reduced) – poor quality
- Higher the SNR – better the quality

9

CONVOLUTION

- Operates on two images – input image, mask or kernel or filter or window
- Modify spatial frequency characteristics
- Determine the value of a central pixel – adding the weighted values of all of its neighbors
- Traversing the kernel through the image
- Applies a general purpose filter effect on the input image
- Achieve various effects with appropriate kernels
  - Smoothing
  - Sharpening
  - Embossing
  - Edge detection

10

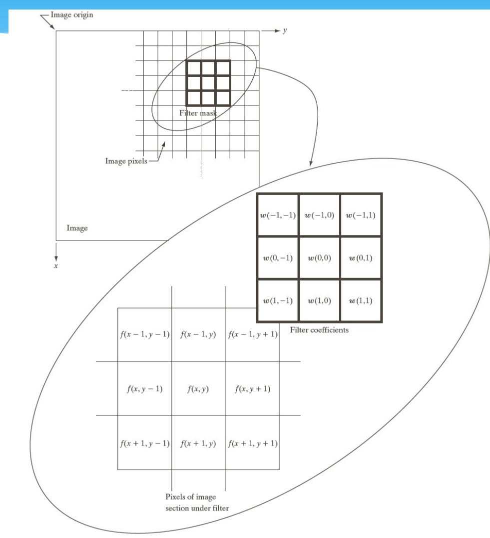
Mechanics of Spatial Filtering

- g(m,n) = T[f(m,n)], g(m,n) = h(m,n)*f(m,n)
  - if the operator T is linear & shift invariant
  - h(m,n) is of finite extent, G(u,v) = H(u,v)F(u,v)
  - can be interpreted in the frequency domain as a filtering operation.
    - It has the effect of filtering frequency components (passing certain frequency components and stopping others).
  - If h(m, n) is a 3x3 mask, output g(m, n) is computed by sliding the mask over each pixel of the image f(m, n).
- Special care is required for the pixels at the border of image f(m, n). This depends on the so-called boundary condition. Common choices are:
  - The mask is truncated at the border (free boundary)
  - The image is extended by appending extra rows/columns at the boundaries. The extension is done by repeating the first/last row/column or by setting them to some constant (fixed boundary).
  - The boundaries wrap around (periodic boundary).

Spatial Filtering

- Procedure
  - Move the filter mask from point to point
  - Response of the filter at that point is calculated using a predefined relationship
    - Response is given by a sum of products of the filter coefficients and the corresponding image pixels in the area spanned by the filter mask
    - For a 3 x 3 mask
      - R = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + … + w(0,0)f(x,y) + … + w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1) or R = $\text{sum}_{i=1}^{9} w_i z_i$
      - w(0,0) coincides with image value f(x,y)
        - Mask is centered at (x,y)

Spatial Filtering



Dept. of E & C, MSRIT

13

Example

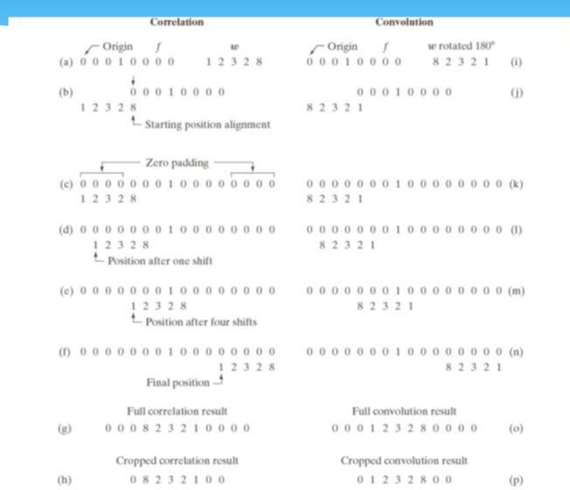
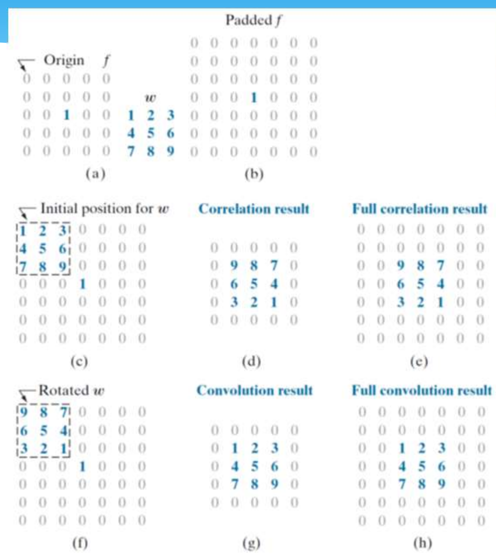


FIGURE 3.29 Illustration of 1-D correlation and convolution of a filter with a discrete unit impulse. Note that correlation and convolution are functions of *displacement*.

Dept. of E & C, MSRIT

14

Example



Smoothing Filters

* designed to smooth or flatten the image by reducing the rapid pixel-to-pixel variation in gray values.
  * Blurring: preprocessing step for removing small (unwanted) details before extracting the relevant (large) object, bridging gaps in lines/curves
  * Noise Reduction: Mitigate the effect of noise by linear or nonlinear operations.
* Image smoothing by averaging (low pass filtering)
  * An averaging mask is a mask with positive weights, which sum to 1. It computes a weighted average of the pixel values in a neighborhood. This operation is sometimes called neighborhood averaging

Smoothing Linear Filters

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

$\frac{1}{16} \times$

1	2	1
2	4	2
1	2	1

Smoothing Kernels

* Box kernel
* Gaussian kernel

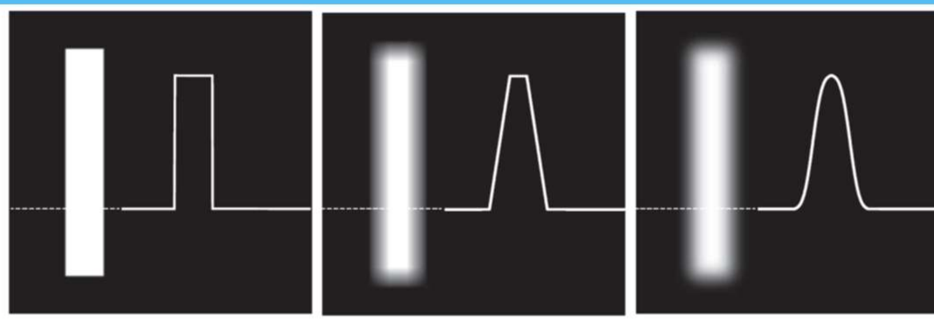
$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

$\frac{1}{4.8976} \times$

0.3679	0.6065	0.3679
0.6065	1.0000	0.6065
0.3679	0.6065	0.3679

Smoothing Spatial Filters



a b c

FIGURE 3.38 (a) Image of a white rectangle on a black background, and a horizontal intensity profile along the scan line shown dotted. (b) Result of smoothing this image with a box kernel of size 71×71, and corresponding intensity profile. (c) Result of smoothing the image using a Gaussian kernel of size 151×151, with $K = 1$ and $\sigma = 25$. Note the smoothness of the profile in (c) compared to (b). The image and rectangle are of sizes 1024×1024 and 768×128 pixels, respectively.

19

EXAMPLES

- Blur an image – Box kernel
- Detect edges from a gray scale image – Laplace kernel
- convolve2d () – mode, boundary, fillvalue
  - mode = 'full' – This is the default mode, in which the output is the full discrete linear convolution of the input
  - mode = 'valid' – This ignores edge pixels and only computes for those pixels with all neighbors (pixels that do not need zero-padding). The output image size is less than the input image size for all kernels (except 1 x 1)
  - mode = 'same' – The output image has the same size as the input image; it is centered with regards to the 'full' output
  - RGB image – apply convolution separately for each channel
  - ndimage.convolve () – convolve RGB image directly

20

CORRELATION VS CONVOLUTION

- convolution flips the kernel twice (horizontal and vertical axis) before computing the weighted combination
- Example:
  - Cross correlation with an eye template image
  - Location of the eye in the raccoon-face image
  - Location – best match with the template (largest cross correlation value)

21

SHARPENING SPATIAL FILTERS

- Highlight fine detail in an image
- Enhance detail that has been blurred (either in error or as a natural effect of image acquisition)
- Blurring vs Sharpening
  - Blurring/smooth is done in spatial domain by pixel averaging of neighbors
  - process of integration
  - Sharpening is an inverse process, to find the difference in a neighborhood
  - done by spatial differentiation

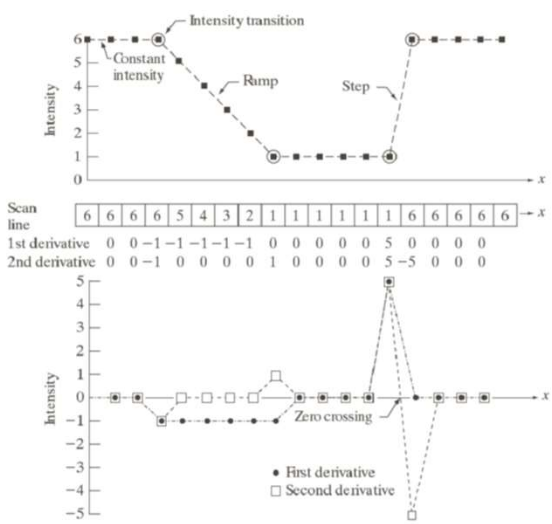
22

DERIVATIVE OPERATOR

- Strength of the response of a derivative operator is proportional to the degree of discontinuity of the image at the point at which the operator is applied
- Image differentiation
  - enhances edges and other discontinuities (noise)
  - deemphasizes area with slowly varying gray-level values
- First order derivative $\frac{\partial f}{\partial x} = f(x+1) - f(x)$
- Second order derivative $\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$

23

EXAMPLE



24

2D DERIVATIVES

- Partial derivatives along the two spatial axes

Gradient operator
(linear operator)

$$\nabla \text{f} = \frac{\partial f(x,y)}{\partial x \partial y} = \frac{\partial f(x,y)}{\partial x} + \frac{\partial f(x,y)}{\partial y}$$

Laplacian operator
(non-linear)

$$\nabla^2 f = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

25

DISCRETE FORM OF LAPLACIAN

$$\frac{\partial^2 f}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2f(x,y)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2f(x,y)$$

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

$$\nabla^2 f = [f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)]$$

26

OTHER IMPLEMENTATIONS

- Give the same result
- When combining (add/subtract) a Laplacian filtered image with another image
- Highlights gray level discontinuities
- Deemphasizes regions with slowly varying gray levels
- Grayish edge lines and other discontinuities, superimposed on a dark featureless background

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

27

SOLUTION

- Add original and Laplacian image (careful with Laplacian filter used)

$$g(x,y)=\begin{cases} f(x,y)-\nabla^2 f(x,y) \\ f(x,y)+\nabla^2 f(x,y) \end{cases}$$

if the center coefficient of the Laplacian mask is negative

if the center coefficient of the Laplacian mask is positive

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

−1	−1	−1
−1	9	−1
−1	−1	−1

28

IMAGE SHARPENING MASK

$$g(x,y)=\begin{cases} f(x,y)-\nabla^2 f(x,y) \\ f(x,y)+\nabla^2 f(x,y) \end{cases}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 9 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ -1 & 8 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

29

HIGH BOOST MASK

0	−1	0
−1	$A+4$	−1
0	−1	0

−1	−1	−1
−1	$A+8$	−1
−1	−1	−1

30

IMAGE EMBOSSING

- Computer graphics technique
  - Each pixel of an image is replaced either by a highlight or a shadow
  - depending on light/dark boundaries on the original image
  - Low contrast areas are replaced by a gray background
- filtered image will represent the rate of color change at each location of the original image
- results in an image resembling a paper or metal embossing of the original image
- Directional difference filter

31

EMBOSS FILTER MASKS

$$\begin{pmatrix} 0 & +1 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} +1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ +1 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & +1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & +1 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ +1 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} -2 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} -2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

32

SUMMARY

- 2D DFT, applications in image processing (filtering)
- Sampling and Quantization
- FFT algorithms (image denoising and restoration)
- Correlation and Convolution
- Application of correlation in template matching

33