Image Processing

Intensity Transformations
Chapter 3

Prof. Vidya Manian
Dept. of Electrical and Computer Engineering
Overview

• Background
• Basic intensity transformation functions
• Histogram processing
3x3 neighborhood of a point \((x,y)\)
Intensity or gray-level or mapping transformation function

• $S = T(r)$

• Resulting image has higher contrast by darkening the intensity levels below $k$ and brightening the levels above $k$

• Contrast stretching-values of $r$ lower than $k$ are compressed by the transformation function into a narrow range of $s$, toward black, vice versa for values of $r$ higher than $k$
Contrast stretching and thresholding functions

\[ s = T(r) \]

\[ s_0 = T(r_0) \]

Dark \rightarrow Light

\[ s = T(r) \]

Dark \rightarrow Light

\[ k \]

\[ r_0 \]
Thresholding function

• T(r) produces a two-level (binary) image – limiting case
• Intensity transformations are used for image enhancement
• Also for image segmentation
• Two types: point processing and neighborhood processing
Basic intensity transformation functions

- $S = T(r)$
- Mapping from $r$ to $s$ are implemented via a lookup table
- For 8-bits, lookup table has 256 entries.
- Types of functions: linear (negative and identity transformations)
- Power-law (nth power and nth root transformations)
Basic intensity transformation functions
Image negatives

• Negative transformation $s = L - 1 - r$
• Log transformation: $s = c \log (1 + r)$
• $c$ is a constant, $r \geq 0$
• Maps range of low intensity values in the input to wider range of output levels
• Expands dark pixels in an image and compresses the higher-level values (white pixels)
• Useful for Fourier spectrum: spectra values range from 0 to $10^6$ or higher.
• Detail lost in the display of Fourier spectrum
Original image and negative image obtained using the negative transformation
Fourier spectrum and log transformation
• Fourier spectrum with values in range of 0 to 1.5 x 10^6 (Fig. 3.5 a) only few pixels are white.
• First use log transformation with c=1, range of values become 0 to 6.2
• Fig 3.5b has better scaling and shows more detail.
Power law (Gamma) transformations

• $s = cr^\gamma$
  Where $c$ and $\gamma$ are positive constants.
• $s = c(r+\varepsilon)^\gamma$ to account for an offset due to display calibration
• Power-law curves with fractional values of $\gamma$ map a narrow range of dark input values to a wider range of output values, and opposite for higher values of input levels
• $\gamma > 1$ has opposite effect of $\gamma < 1$. Also called Gamma correction
• Device dependent, CRT displays, image capture, printing
**FIGURE 3.6** Plots of the equation $s = cr^\gamma$ for various values of $\gamma$ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.
• Preprocess image before input to monitor by 
  \( s = r^{(1/2.5)} = r^{(0.4)} \)
• Gamma corrected output has appearance close to original
• Applicable to scanners, printers
• Use of digital images for commercial purpose over Internet has increased
• Solution – display image after gamma correction to value that represents “average” of the types of monitors and computer systems to be used to display the image.
FIGURE 3.7
(a) Intensity ramp image. (b) Image as viewed on a simulated monitor with a gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).
FIGURE 3.9
(a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0, \text{ and } 5.0$, respectively. (Original image for this example courtesy of NASA.)
Contrast stretching

![Graph showing contrast stretching](image)

**FIGURE 3.10**
Contrast stretching. 
(a) Form of transformation function. (b) A low-contrast image. 
(c) Result of contrast stretching. 
(d) Result of thresholding. 
(Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)
**FIGURE 3.11** (a) This transformation highlights intensity range \([A, B]\) and reduces all other intensities to a lower level. (b) This transformation highlights range \([A, B]\) and preserves all other intensity levels.
FIGURE 3.12  (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)
Bit plane representation of an 8-bit image
FIGURE 3.14  (a) An 8-bit gray-scale image of size $500 \times 1192$ pixels. (b) through (i) Bit planes 1 through 8, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image.
FIGURE 3.15 Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).
Four basic image types
Histogram equalization
Transformation functions for Histogram equalization in Fig. 3. 20

**FIGURE 3.21**
Transformation functions for histogram equalization. Transformations (1) through (4) were obtained from the histograms of the images (from top to bottom) in the left column of Fig. 3.20 using Eq. (3.3-8).
Example of histogram equalization

**FIGURE 3.19** Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.
Histogram

**FIGURE 3.23**
(a) Image of the Mars moon Phobos taken by NASA’s *Mars Global Surveyor.*
(b) Histogram. (Original image courtesy of NASA.)
Histogram equalized image

FIGURE 3.24
(a) Transformation function for histogram equalization.
(b) Histogram-equalized image (note the washed-out appearance).
(c) Histogram of (b).
Specified Histogram

**FIGURE 3.25**
(a) Specified histogram.
(b) Transformations.
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).
Steps for Histogram Matching

• Obtain Pr(r) from input image and obtain values of s
• Use specified pdf to obtain the transformation function G(z)
• Obtain the inverse transformation z=G⁻¹(s); mapping from s to z
• Obtain output image by first equalizing input image. For each pixel with value s, perform inverse mapping z=G⁻¹(s) to obtain corresponding output pixel
• When all pixels are processed the pdf of output image is equal to specified pdf
Local Histogram equalization

**FIGURE 3.26** (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization applied to (a), using a neighborhood of size $3 \times 3$. 
Global and local histogram equalization

**FIGURE 3.27** (a) SEM image of a tungsten filament magnified approximately 130×. (b) Result of global histogram equalization. (c) Image enhanced using local histogram statistics. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)
Fundamentals of spatial filtering

• Filter: accepting (passing) or rejecting certain frequency components

• Spatial filter consists of (1) a neighborhood (a small rectangle) and (2) predefined operation performed on the image pixels encompassed by the neighborhood

• Linear and non-linear filters
Basics of Spatial Filtering

• Neighborhood subimage-filter, mask, kernel, template, or window
• Values in mask – coefficients
• Response $R$ of linear filtering with the filter mask at a point $(x,y)$ in the image is:
  $$R = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \ldots$$
  $$+ w(0,0)f(x,y) + \ldots + w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1)$$
Mechanics of spatial filtering

• $W(0,0)$ center coefficient of the filter
• Mask of size $mxm$, $m=2a+1$ and $n=2b+1$, where $a$ and $b$ are positive integers
• Smallest size $3x3$
The mechanics of spatial filtering. The magnified drawing shows a $3 \times 3$ mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.
Spatial correlation and convolution

• Correlation is the process of moving a filter mask over the image and computing the sum of products at each location.
• Convolution is the same except that the filter is rotated by 180 degrees.
• Rotation is equivalent to horizontal flipping
• In 2D rotation is equivalent to flipping the mask along one axis and then the other
• Fig. 3.29 shows 1-D convolution and correlation
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.
Correlation and convolution

(a) Padded $f$

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(b) $g(x, y) w(x, y)$

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(c) Initial position for $w$

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(d) Full correlation result

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(e) Cropped correlation result

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(f) Rotated $w$

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(g) Full convolution result

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(h) Cropped convolution result

<table>
<thead>
<tr>
<th>$(x, y)$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Smoothing spatial filters

Two $3 \times 3$ smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to the sum of the values of its coefficients, as is required to compute an average.
Smoothing linear filters

• An \(mxn\) mask would have a normalizing constant equal to \(1/mn\).

• A spatial averaging filter in which all coefficients are equal is called a box filter.

• The diagonal terms are further away from center and hence are weighted less than the immediate neighbors.

• Sum of coefficients =16 is useful for easy computer implementation (a power of 2).
Effect of smoothing as a function of filter size

- Average filters of sizes $m = 3, 5, 9, 15$ and 35 pixels
- Larger size windows for blurring removes smaller objects from image
- A gross representation of objects of interest is obtained
- Intensity of smaller objects blends with background and larger objects become bloblike
(a) Original image, of size $500 \times 500$ pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes $n = 3, 5, 9, 15$, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their gray levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size $30 \times 120$ pixels.
Order-statistics (nonlinear) filters

- Response of nonlinear filters based on ordering (ranking) the pixels contained in the image area encompassed by the filter
- Median filter - to remove impulse noise (salt and pepper noise)
- Sort the pixel values in the window (say 3x3) and assign the median value to the center pixel
- Forces pixels with distinct intensity levels to be more like its neighbors
- Max, min filters
Image filtered with a 15x15 averaging mask

**FIGURE 3.34** (a) Image of size 528 $\times$ 485 pixels from the Hubble Space Telescope. (b) Image filtered with a 15 $\times$ 15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)
**FIGURE 3.35** (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a $3 \times 3$ averaging mask. (c) Noise reduction with a $3 \times 3$ median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)
Sharpening spatial filters

• Highlights transition in intensities
• Blurring by pixel averaging
• Sharpening by spatial differentiation
• Enhances edges and other discontinuities such as noise
• Based on first and second order derivatives
First and second derivatives of a 1D signal
Sharpening spatial filters

\[
\frac{\partial f}{\partial x} = f(x + 1) - f(x)
\]

\[
\frac{\partial^2 f}{\partial x^2} = f(x + 1) + f(x - 1) - 2f(x)
\]

Change between adjacent pixels
First and second derivative
(1) Must be zero in flat areas;
(2) Non zero at the onset of gray-level step or ramp
(3) First derivative-non zero along ramps; second derivative-
    Zero along ramps of constant slope
The Laplacian for enhancement (the second derivative)

\[ \nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \]

In the x direction
\[ \frac{\partial^2 f}{\partial x^2} = f(x + 1, y) + f(x - 1, y) - 2f(x, y) \]

In the y direction
\[ \frac{\partial^2 f}{\partial x^2} = f(x, y + 1) + f(x, y - 1) - 2f(x, y) \]

Sum min g
\[ \nabla^2 f = [f(x, y + 1) + f(x, y - 1) + f(x + 1, y) + f(x - 1, y)] - 4f(x, y) \]
(a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4).
(b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.
Second derivative-the Laplacian

**FIGURE 3.38**
(a) Blurred image of the North Pole of the moon.
(b) Laplacian without scaling.
(c) Laplacian with scaling.
(d) Image sharpened using the mask in Fig. 3.37(a).
(e) Result of using the mask in Fig. 3.37(b).
(Original image courtesy of NASA.)
Blurring using Gaussian filter

FIGURE 3.40
(a) Original image.
(b) Result of blurring with a Gaussian filter.
(c) Unsharp mask. (d) Result of using unsharp masking.
(e) Result of using highboost filtering.
(a) Image of the North Pole of the moon.
(b) Laplacian-filtered image.
(c) Laplacian image scaled for display purposes.
(d) Image enhanced by using Eq. (3.7-5).
(Original image courtesy of NASA.)
The Gradient for enhancement (the first derivative)

\[ \nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \]

\[ \nabla f = \text{mag}(\nabla f) \]
\[ = [G_x^2 + G_y^2]^{1/2} \]
\[ = \left[ \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right]^{1/2} \]

\[ \nabla f \approx \left| (z_7 + 2z_8 + z_9) - (z_1 + 2z_1 + z_3) \right| + \left| (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \right| \]
A $3 \times 3$ region of an image (the $z$'s are gray-level values) and various masks used to compute the gradient at point labeled $z_5$. 

**Roberts**

<table>
<thead>
<tr>
<th>-1</th>
<th>-1</th>
<th>-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>-1</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**Prewitt**

<table>
<thead>
<tr>
<th>-1</th>
<th>-2</th>
<th>-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>-1</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Composite Laplacian mask

(a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (c) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)
Sobel gradient
FIGURE 3.43
(a) Image of whole body bone scan.
(b) Laplacian of (a).
(c) Sharpened image obtained by adding (a) and (b).
(d) Sobel gradient of (a).
Optical image of contact lens (note defects on the boundary at 4 and 5 o’clock).
(b) Sobel gradient.
(Original image courtesy of Mr. Pete Sites, Perceptrics Corporation.)
Exercise in group

• Propose a set of intensity-slicing transformations capable of producing all the individual bit planes of an 8bit monochrome image. $T(r)=0$ in range $[0,127]$ and $T(r)=255$ for $r$ in range $[128,255]$ produces an image of the 8th bit plane in an 8-bit image.

• Given the Gaussian pdf what is the transformation function you would use for histogram equalization.
Exercises on spatial filtering

• 3.1, 3.2, 3.4, 3.5, 3.6, 3.7, 3.8 to 3.11, 3.13
• 3.15, 3.17, 3.19,

• 3.21, 3.23, 3.24, 3.25, 3.27, 3.28, 3.29
• 3.1) Give a single intensity transformation function for spreading the intensities of an image so the lowest intensity is 0 and highest is L-1.

• 3.5) What effect would setting to zero the lower-order bit planes have on the histogram of an image in general.
• 3.6) Explain why discrete histogram equalization does not in general yield a flat histogram.

• Answer: To get a flat histogram, pixel intensities should be redistributed so that there are L groups of n/L pixels with same intensity and n=MN.
3.7) Suppose that a digital image is subjected to histogram equalization. Show that a second pass of histogram equalization (on the histogram equalized image) will produce exactly the same result as the first pass.