2005

Medical image enhancement

Alina Monica Trifas
Louisiana State University and Agricultural and Mechanical College, mtrifas@hotmail.com

Follow this and additional works at: https://digitalcommons.lsu.edu/gradschool_dissertations

Part of the Computer Sciences Commons

Recommended Citation
https://digitalcommons.lsu.edu/gradschool_dissertations/589

This Dissertation is brought to you for free and open access by the Graduate School at LSU Digital Commons. It has been accepted for inclusion in LSU Doctoral Dissertations by an authorized graduate school editor of LSU Digital Commons. For more information, please contact gradetd@lsu.edu.
MEDICAL IMAGE ENHANCEMENT

A Dissertation

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
in
The Department of Computer Science

by
Alina Monica Trifas
B.S., University of Bucharest, 1996
M.S., University of Bucharest, 1999
December, 2005
In loving memory of

Elena Brinzaru (My Mother)
Acknowledgements

I would like to thank my advisor, Dr. John Tyler, for his support, encouragement, and patience during my years of graduate study. His experience and knowledge made this research possible. I would like to thank Dr. Oleg Pianykh for his constant guidance in my research work.

I would also like to thank Dr. Doris Carver, Dr. Jerry Trahan, Dr. Donald Kraft, Dr. Bijaya Karki and Dr. David Bertolini for taking the time to be on my committee.
Table of Contents

Acknowledgements........................................................................................................... iii

List of Tables...................................................................................................................... ix

List of Figures..................................................................................................................... x

Abstract..........................................................................................................................xxi

Chapter 1  Introduction..................................................................................................... 1
1.1 Introduction to Computer Vision................................................................. 1
1.2 What Is Digital Image Processing........................................................... 2
1.3 The Origins of Digital Image Processing............................................. 5
1.4 Examples of Fields that Use Digital Image Processing..................... 8
  1.4.1 Gamma-Ray Imaging............................................................... 9
  1.4.2 X-ray Imaging................................................................. 11
  1.4.3 Imaging in the Ultraviolet Band........................................ 13
  1.4.4 Imaging in the Visible and Infrared Bands.......................... 15
  1.4.5 Imaging in the Microwave Band...................................... 20
  1.4.6 Imaging in the Radio Band.............................................. 21
  1.4.7 Examples of Using Other Imaging Modalities................. 22
1.5 Fundamentals Steps in Digital Image Processing......................... 28
1.6 Components of an Image Processing System................................ 33

Chapter 2  Image Enhancement in the Spatial Domain................................. 38
2.1 Background............................................................................................................. 39
2.2 Gray Level Transformations........................................................................... 41
  2.2.1 Negatives.................................................................................. 42
  2.2.2 Log Transformations............................................................... 42
  2.2.3 Power-Law Transformations.................................................. 43
  2.2.4 Piecewise-Linear Transformation Function......................... 49
    2.2.4.1 Contrast Stretching.......................................................... 51
    2.2.4.2 Gray-Level Slicing.......................................................... 52
    2.2.4.3 Bit-Plane Slicing................................................................. 54
  2.3 Histogram Processing......................................................................................... 57
    2.3.1 Histogram Equalization.......................................................... 61
    2.3.2 Histogram Matching................................................................. 65
  2.4 Enhancement Using Arithmetic/Logic Operations............................ 71
    2.4.1 Image Subtraction.................................................................. 73
    2.4.2 Image Averaging................................................................... 78
  2.5 Spatial Filtering................................................................................................. 81
  2.6 Smoothing Spatial Filters................................................................................. 84
    2.6.1 Linear Smoothing Filters....................................................... 85
    2.6.2 Order-Statistics Filters........................................................... 87
  2.7 Sharpening Spatial Filters............................................................................. 94
    2.7.1 Fundamental Properties of Derivatives......................... 94
    2.7.2 The Second Derivative Used in Enhancement.................. 96
      2.7.2.1 Method Description......................................................... 96
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3.3</td>
<td>Alternative Computational Theories</td>
<td>213</td>
</tr>
<tr>
<td>4.3.3.1</td>
<td>Texture Analysis</td>
<td>213</td>
</tr>
<tr>
<td>4.3.3.2</td>
<td>Structure from Shading</td>
<td>215</td>
</tr>
<tr>
<td>4.3.4</td>
<td>A Theoretical Synthesis</td>
<td>218</td>
</tr>
<tr>
<td>4.3.4.1</td>
<td>Local Spatial Frequency Filters</td>
<td>218</td>
</tr>
<tr>
<td>4.3.4.2</td>
<td>Exploiting the Structure of Natural Images</td>
<td>221</td>
</tr>
</tbody>
</table>

Chapter 5  Multiscale Contrast Enhancement | 228 |
| 5.1     | Image Pyramids | 228  |
| 5.1.1   | Method Description | 230  |
| 5.1.2   | The Gaussian Pyramid | 231  |
| 5.1.2.1 | Gaussian Pyramid Generation | 231  |
| 5.1.2.2 | The Generating Kernel | 232  |
| 5.1.2.3 | Weighting Functions | 233  |
| 5.1.2.4 | Gaussian Pyramid Interpolation | 234  |
| 5.1.3   | The Laplacian Pyramid | 234  |
| 5.1.3.1 | Laplacian Pyramid Generation | 234  |
| 5.1.3.2 | Decoding | 235  |
| 5.1.4   | Using Laplacian Pyramid for Contrast Enhancement | 236  |

5.2     | Contrast Enhancement by Wavelet Analysis | 238  |
| 5.2.1   | Preview | 238  |
| 5.2.2   | Wavelet Applications | 240  |
| 5.2.2.1 | Scale Aspects | 240  |
| 5.2.2.2 | Time Aspects | 240  |
| 5.2.2.3 | Wavelet Decomposition as a Whole | 241  |
| 5.2.3   | Fourier Analysis versus Wavelet Analysis | 242  |
| 5.2.4   | What is Wavelet Analysis? | 245  |
| 5.2.5   | The Continuous Wavelet Transform | 246  |
| 5.2.6   | The Discrete Wavelet Transform | 248  |
| 5.2.7   | Wavelet Reconstruction | 250  |
| 5.2.8   | Two-Dimensional Wavelet Transform | 252  |
| 5.2.9   | Using Wavelet Analysis for Contrast Enhancement | 255  |
| 5.2.10  | Quantitative Measures of Error | 257  |

Chapter 6  Summary | 263  |
| 6.1     | Conclusion | 263  |
| 6.2     | Implementation and Future Work | 266  |

References | 270  |

Appendix  Theoretical Approaches to Vision | 282  |
<p>| A.1     | Introduction to Visual Perception | 282  |
| A.2     | Visual Systems | 285  |
| A.2.1   | The Human Eye | 285  |
| A.2.1.1 | Eye and Brain | 286  |
| A.2.1.2 | Anatomy of the Eye | 287  |
| A.2.1.3 | Physiological Optics | 287  |
| A.2.2   | The Retina | 288  |
| A.2.2.1 | Neurons | 288  |
| A.2.2.2 | Photoreceptors | 290  |</p>
<table>
<thead>
<tr>
<th>A.6.5 The Category-Based Stage</th>
<th>339</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vita</td>
<td>341</td>
</tr>
</tbody>
</table>
List of Tables

Table 1.1 Thematic bands in NASA’s LANDSAT satellite...........................................16

Table 3.1 Values of percent of the image power included in circles
        with different radii..............................................................143

Table 4.1 Functional differences between Magno and Parvo LGN cells.......................179

Table 5.1 Error measurements for Figure 5.25 (constant=3)....................................260

Table 5.2 Error measurements for Figure 5.26 (constant=10).................................260

Table 5.3 Error measurements for Figure 5.27 (constant=20).................................261

Table A.1 Three psychological theories of visual perception....................................303
List of Figures

Figure 1.1 The first picture of moon by a U.S. spacecraft ................................. 6
Figure 1.2 The electromagnetic spectrum arranged according to energy per photon .... 9
Figure 1.3 (a) Bone scan. (b) PET image. (c) Cygnus Loop (d) Gamma Radiation .... 10
Figure 1.4 Examples of X-ray imaging. (a) Chest X-ray ...................................... 11
Figure 1.4 (b) Aortic angiogram ........................................................................... 12
Figure 1.4 (c) Brain CT ......................................................................................... 13
Figure 1.5 Examples of ultraviolet imaging (a) Normal corn. (b) Corn with smut .... 14
Figure 1.6 Examples of light microscopy images .................................................. 15
Figure 1.7 LANDSAT satellite images of the Washington, D.C. area ................. 17
Figure 1.8 Multispectral image of Hurricane Andrew taken by NOAA GEOS sensors........................................................................................................... 18
Figure 1.9 Some examples of manufactured goods checked using digital image processing ........................................................................................................... 19
Figure 1.10 More examples of imaging in the visual spectrum............................ 20
Figure 1.11 Spaceborne radar image of mountains in southeast Tibet ............... 21
Figure 1.12 MRI image of a human bone .............................................................. 22
Figure 1.13 Images of the Crab Pulsar (in the center of these images) covering the electromagnetic spectrum ................................................................. 22
Figure 1.14 Cross-sectional image of a seismic model ....................................... 23
Figure 1.15 Example of ultrasound imaging (baby) ........................................... 25
Figure 1.16 (a) Magnified 250 times SEM image of a tungsten filament following thermal failure. (b) Magnified 2500 times SEM image of damaged integrated circuit ............................................................................................... 26
Figure 1.17 (a) and (b) Fractal images. (c) and (d) Images generated from 3-D computer models of the objects shown ......................................................... 27
Figure 1.18 Fundamentals steps in digital image processing .............................. 29
Figure 1.19 Components of a general-purpose image processing system .......... 34
Figure 2.1 A 3x3 neighborhood about a point (x, y) in an image..............................39
Figure 2.2 Gray-level transformation functions for contrast enhancement......................41
Figure 2.3 Some basic gray-level transformation functions
used for image enhancement..................................................42
Figure 2.4 The negative image was obtained.................................................................43
Figure 2.5 Fourier spectrum.......................................................................................44
Figure 2.6 Result of applying the log transformation to the Fourier spectrum.................44
Figure 2.7 Plots of the equation $s = cr^\gamma$ for various values of $\gamma$ ($c = 1$ in all cases)......45
Figure 2.8 Magnetic resonance (MR) image of a head..................................................46
Figure 2.9 Result of applying the power-law transformation
with $c = 1$ and $\gamma = 0.6$ ..................................................................................47
Figure 2.10 Result of applying the power-law transformation
with $c = 1$ and $\gamma = 0.4$ ..................................................................................47
Figure 2.11 Result of applying the power-law transformation
with $c = 1$ and $\gamma = 0.3$ ..................................................................................48
Figure 2.12 Magnetic resonance (MR) image of human brain.......................................49
Figure 2.13 Result of applying the power-law transformation
with $c = 1$ and $\gamma = 3.0$ ..................................................................................50
Figure 2.14 Result of applying the power-law transformation
with $c = 1$ and $\gamma = 4.0$ ..................................................................................50
Figure 2.15 Result of applying the power-law transformation
with $c = 1$ and $\gamma = 5.0$ ..................................................................................51
Figure 2.16 Contrast stretching. Form of transformation function.................................52
Figure 2.17 A low-contrast image (human abdomen)......................................................53
Figure 2.18 Result of contrast stretching applied to the image in Figure 2.17....................53
Figure 2.19 Result of thresholding applied to the image in Figure 2.17.........................54
Figure 2.20 Gray-level slicing transformations...............................................................55
Figure 2.21 Bit-plane representation of an 8-bit image..................................................56
Figure 2.22 A 8-bit fractal image........................................................................56
Figure 2.23 The eight bit planes of the image in Figure 2.22.........................57
Figure 2.24 A dark image and its histogram.......................................................59
Figure 2.25 A bright image and its histogram......................................................59
Figure 2.26 A low-contrast image and its histogram..........................................60
Figure 2.27 A high-contrast image and its histogram.........................................60
Figure 2.28 A gray-level transformation function that is both single
dvalued and monotonically increasing.........................................................62
Figure 2.29 (a) The dark image from Figure 2.24.
(b) Result of histogram equalization. (c) The equalized histogram..............65
Figure 2.30 (a) The bright image from Figure 2.25.
(b) Result of histogram equalization. (c) The equalized histogram..............66
Figure 2.31 (a) The low-contrast image from Figure 2.26.
(b) Result of histogram equalization. (c) The equalized histogram..............66
Figure 2.32 (a) The high-contrast image from Figure 2.27.
(b) Result of histogram equalization. (c) The equalized histogram..............67
Figure 2.33 (a) Image of the Mars moon Phobos taken by NASA’s Mars Global Surveyor (b) Its histogram.............................................70
Figure 2.34 (a) Histogram equalization transformation of Figure 2.33 (a).
(b) Histogram equalized image for Figure 2.33. (c) Histogram of (b)..........70
Figure 2.35 Enhanced image using histogram matching.................................72
Figure 2.36 A magnetic resonance (MR) image of a bone.................................74
Figure 2.37 A blurred version of the image in Figure 2.36.................................74
Figure 2.38 Difference between images in Figures 2.36 and 2.37.....................75
Figure 2.39 Histogram-equalized difference image........................................75
Figure 2.40 Enhancement by image subtraction...............................................76
Figure 2.41 (a) Image of Galaxy Pair NGC 3314.
(b) Image corrupted by additive Gaussian noise with zero mean and standard deviation of 64 gray levels.
(c)-(f) Results of averaging $K = 8, 16, 64, 128$ noisy images.................80
Figure 2.42 The spatial filtering process. The magnified drawing shows a $3 \times 3$ mask and the image section directly below it. .......................... 82

Figure 2.43 A general representation of a general $3 \times 3$ spatial filter mask.......................... 83

Figure 2.44 Two $3 \times 3$ smoothing (averaging) filter masks. ............................................ 86

Figure 2.45 A magnetic resonance (MR) image of a bone.................................................. 88

Figure 2.46 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 3$ .......................... 88

Figure 2.47 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 5$ ..................... 89

Figure 2.48 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 9$ ..................... 89

Figure 2.49 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 15$ .................... 90

Figure 2.50 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 35$ .................... 90

Figure 2.51 Lena image corrupted by salt-and-pepper noise.............................................. 92

Figure 2.52 Noise reduction with a $3 \times 3$ averaging mask.............................................. 93

Figure 2.53 Noise reduction with a $3 \times 3$ median filter.................................................. 93

Figure 2.54 Filter masks used to implement the Laplacian............................................... 97

Figure 2.55 A blurred image of a bone.............................................................................. 99

Figure 2.56 Laplacian filtered image.............................................................................. 99

Figure 2.57 Laplacian image scaled for display purposes................................................. 100

Figure 2.58 Image enhanced using the Laplacian............................................................... 100

Figure 2.59 Composite Laplacian masks........................................................................... 101

Figure 2.60 A blurred image of a bone.............................................................................. 102

Figure 2.61 Result of filtering with the mask in Figure 2.59 (a).......................................... 103

Figure 2.62 Result of filtering with the mask in Figure 2.59 (b).......................................... 103

Figure 2.63 Masks for the high-boost filtering ................................................................. 104
Figure 2.64 A dark magnetic resonance image of a human head.........................105

Figure 2.65 Laplacian of image in Figure 2.64, computed
with the mask in Figure 2.63(b) using $A = 0$.........................106

Figure 2.66 Laplacian enhanced image using
the mask in Figure 2.63 (b) with $A = 1$.........................106

Figure 2.67 Laplacian enhanced image using
the mask in Figure 2.63 (b) with $A = 1.7$.........................107

Figure 2.68 Robert cross-gradient operators.................................109

Figure 2.69 Sobel operators...........................................110

Figure 2.70 A medical image of a knee..................................110

Figure 2.71 Sobel gradient of the medical image in Figure 2.70........111

Figure 2.72 (a) Optical image of contact lens
(with defects on the boundary at 4 and 5 o’clock). (b) Sobel gradient........112

Figure 3.1 (a) A discrete function of M points, and (b) its Fourier spectrum.
(c) A discrete function with twice the number of nonzero points and
(d) its Fourier spectrum......................................................117

Figure 3.2 A 256x256 pixel medical image (a bone) on a black background........120

Fig. 3.3 Centered Fourier spectrum shown after log transformation ...............121

Figure 3.4 Scanning electron microscope image of a damaged integrated circuit.....122

Figure 3.5 Fourier spectrum of the image in Figure 3.4..................................123

Figure 3.6 Block image..........................................................124

Figure 3.7 The Fourier spectrum of the image in Figure 3.6..............................124

Figure 3.8 Basics steps for filtering in the frequency domain..........................126

Figure 3.9 Lena image..................................................................127

Figure 3.10 Result of filtering Lena image with a notch filter.........................128

Figure 3.11 A magnetic resonance brain image..........................................129

Figure 3.12 A two-dimensional lowpass filter function..................................130

Figure 3.13 Result of lowpass filtering the original image..............................130
Figure 3.35 (a) Original image. (b)-(d) Results of ideal highpass filtering with 
\[ D_0 = 15, \ 30, \ 50 \] respectively..........................................................154

Figure 3.36 Perspective plot of a Butterworth highpass filter transfer function........155

Figure 3.37 Butterworth highpass filter displayed as a projection onto the \((u,v)\) plane.....156

Figure 3.38 Butterworth lowpass filter radial cross section of order 2....................156

Figure 3.39 (a) Original image. (b)-(d) Results of highpass filtering the image in (a) 
using a BHPF of order 2 with \[ D_0 = 15, \ 30, \ 50 \] respectively...............157

Figure 3.40 Perspective plot of a GHPF transfer function......................................158

Figure 3.41 Gaussian highpass filter displayed as a projection onto the \((u,v)\) plane....158

Figure 3.42 Gaussian highpass filter radial cross section for \[ D_0 = 2 \] ...............159

Figure 3.43 (a) Original image. (b)-(d) Results of highpass filtering the image in (a) 
using a GHPF with \[ D_0 = 15, \ 30, \ 50 \] respectively.................................160

Figure 3.44 3-D plot of the Laplacian in the frequency domain..............................162

Figure 3.45 The Laplacian in the frequency domain projected onto the \((u,v)\) plane.....162

Figure 3.46 A blurred image of a bone.................................................................163

Figure 3.47 Laplacian filtered image.................................................................164

Figure 3.48 Laplacian image scaled...............................................................164

Figure 3.49 Image enhanced using (3.4-12)......................................................165

Figure 3.50 A dark magnetic resonance image of a human head.........................167

Figure 3.51 Laplacian of the image in figure 3.50..............................................168

Figure 3.52 Image obtained using (3.4-17) with \[ A = 2 \] .................................168

Figure 3.53 Image obtained using (3.4-17) with \[ A = 2.7 \] ...............................169

Figure 3.54 A blurred version of a chest X-ray image........................................170

Figure 3.55 Result of Butterworth highpass filtering.........................................171

Figure 3.56 Result of high-frequency emphasis filtering.....................................171

Figure 3.57 Result of performing histogram equalization on the image in Figure 3.56....172
Figure 4.1 Response of on-center, off-surround ganglion cells.................................175

Figure 4.2  Receptive field structure of ganglion cells.............................................176

Figure 4.3 Neural inputs to an on-center, off-surround bipolar cell............................177

Figure 4.4 Cross section through the lateral geniculate nucleus (LGN) of a macaque monkey.................................................................178

Figure 4.5 Mapping from eyes to layers of the LGN.................................................180

Figure 4.6 Receptive fields and response profiles of vertically oriented simple cells in area V1.................................................................181

Figure 4.7 Possible neural wiring of a complex cell from simple cells.........................184

Figure 4.8 Sinusoidal gratings..................................................................................186

Figure 4.9 Constructing a square wave by adding sinusoidal components...............189

Figure 4.10 Constructing a plaid grating by adding square wave gratings at different orientations.................................................................190

Figure 4.11 Spatial frequency content of a complex image....................................190

Figure 4.12 The magnitude of the Fourier transform for a clear medical image........192

Figure 4.13 The magnitude of the Fourier transform for the blurred version of the same medical image.........................................................193

Figure 4.14 The magnitude of the Fourier transform the sharpened version of the same medical image as before.........................................................193

Figure 4.15 Average values of the amplitudes of spatial frequency, computed over high-frequency components of the DCT coefficient matrix for a normal image (blue), its blurred (red) and sharpened (green) versions..............194

Figure 4.16 Contrast sensitivity functions...............................................................196

Figure 4.17 Demonstration of the shape of the contrast sensitivity function for luminance gratings.................................................................197

Figure 4.18 Contrast sensitivity before and after adaptation to a sinusoidal grating.......198

Figure 4.19 The multiple spatial frequency channels hypothesis..............................199

Figure 4.20 Gabor functions....................................................................................201

Figure 4.21 Simple edge operators..........................................................................205
Figure 4.22 First and second derivatives of luminance edges.................................208

Figure 4.23 Discrete (A) versus continuous (B) versions of a second-order omnidirectional edge operator...............................................................209

Figure 4.24 Finding edges by detecting zero-crossings.......................................210

Figure 4.25 Edge detection at different spatial scales........................................211

Figure 4.26 A neural model for the Marr-Hildreth zero-crossing algorithm.............212

Figure 4.27 Edge operators for Canny’s algorithm..............................................214

Figure 4.28 An application of Canny’s edge algorithm........................................214

Figure 4.29 Demonstration of texture analysis....................................................215

Figure 4.30 Demonstration of shading analysis....................................................216

Figure 4.31 The architecture of a three-layer feedforward network used to learn shading analysis.................................................................217

Figure 4.32 A theoretical hypothesis about the architecture of image processing......219

Figure 4.33 Natural versus unnatural images......................................................223

Figure 4.34 Compact versus sparse coding........................................................224

Figure 4.35 Receptive fields learned from viewing natural images via principle component analysis.................................................................225

Figure 4.36 Receptive fields learned from exposure to natural images using sparse coding constraints.................................................................226

Figure 4.37 Representing image structure via a pyramid of local oriented filters at different spatial frequencies.........................................................226

Figure 5.1 Examples of Gaussian Pyramid (left) and Laplacian Pyramid (right)......229

Figure 5.2 The first levels of a Gaussian Pyramid..............................................231

Figure 5.3 depicts the images contained in the Gaussian pyramid generated for $a = 0.4$ .........................................................................................233

Figure 5.4 We can expand the subsampled image $n$ times to recover what the image is supposed to look like in the original size...............................235

Figure 5.5 Image features such as edges and bars appear enhanced in the Laplacian Pyramid.................................................................236
Figure 5.6 Improving a blurred medical image using the multiscale decomposition……..238

Figure 5.7 Fourier analysis produces a representation of the signal in the frequency domain……………………………………..242

Figure 5.8 Short-Time Fourier Transform (STFT) provides a representation of the signal as a two-dimensional function of time and frequency…………………..243

Figure 5.9 Wavelet analysis is a windowing technique with variable-size regions…………243

Figure 5.10 Comparison between the time-based, frequency-based, STFT and wavelet views of the same signal……………………………………..244

Figure 5.11 A sinusoidal signal with a small discontinuity…………………………244

Figure 5.12 A plot of the Fourier coefficients and wavelet coefficients for the previous signal……………………………………..245

Figure 5.13 Sine waves and wavelets………………………………………………..246

Figure 5.14 The effect of scale factor on sinusoids ..........................................247

Figure 5.15 The effect of scale factor on wavelets……………………………………247

Figure 5.16 Shifting a wavelet……………………………………………………….248

Figure 5.17 The first stage of DWT…………………………………………………249

Figure 5.18 Performing DWT at level $j$ …………………………………………..250

Figure 5.19 Wavelet decomposition tree for $J = 3$ ………………………………….251

Figure 5.20 IDWT performs wavelet reconstruction…………………………………252

Figure 5.21 Image of a woman (Barb image)………………………………………….254

Figure 5.22 One-level transform of an image of a woman …………………………254

Figure 5.23 Two-dimensional DWT on level $j$ ………………………………………256

Figure 5.24 Two-dimensional IDWT on level $j$ ………………………………………257

Figure 5.25 Improving a blurred medical image with multiplier=3……………….258

Figure 5.26 Improving a blurred medical image with multiplier=10……………….258

Figure 5.27 Improving a blurred medical image with multiplier=20………………259

Figure 5.28 Improving a blurred medical image using multiple-level wavelet analysis…………………………………………………..259
Figure A.1 A real-world scene on Berkeley campus.............................................282
Figure A.2 The human visual system.................................................................286
Figure A.3 A cross section of the human eye......................................................288
Figure A.4 A typical neuron.............................................................................289
Figure A.5 The human retina...........................................................................291
Figure A.6 Scanning electron micrograph of rods and cones...........................292
Figure A.7 Distribution of rods (solid curve) and cones (shaded region) in the human retina.............................................................292
Figure A.8 Visual areas of the human cortex.....................................................295
Figure A.9 Two visual pathways in the monkey cortex......................................296
Figure A.10 The location of primary visual cortex in macaque monkeys...........297
Figure A.11 Interconnections between cortical layers.......................................299
Figure A.12 Diagram of the visual pathways hypothesis....................................300
Figure A.13 Emergent properties of a configuration..........................................305
Figure A.14 Roberts’s “blocks world” computer vision program.......................316
Figure A.15 A three-layer feedforward connectionist network used in backpropagation learning.............................................................318
Figure A.16 Receptive fields in cortical cells.....................................................321
Figure A.17 A black box diagram.....................................................................326
Figure A.18 Representation as a homomorphic mapping..................................329
Figure A.19 Bottom-up versus top-down processes.........................................333
Figure A.20 Four stages of visual processing....................................................334
Abstract

Each image acquired from a medical imaging system is often part of a two-dimensional (2-D) image set whose total presents a three-dimensional (3-D) object for diagnosis. Unfortunately, sometimes these images are of poor quality. These distortions cause an inadequate object-of-interest presentation, which can result in inaccurate image analysis. Blurring is considered a serious problem. Therefore, “deblurring” an image to obtain better quality is an important issue in medical image processing.

In our research, the image is initially decomposed. Contrast improvement is achieved by modifying the coefficients obtained from the decomposed image. Small coefficient values represent subtle details and are amplified to improve the visibility of the corresponding details. The stronger image density variations make a major contribution to the overall dynamic range, and have large coefficient values. These values can be reduced without much information loss.
Chapter 1

Introduction

There have been many advances in the presentation of data, but few can rival those in image processing. The miniaturization, increased speed and increased memory sizes of computers have led to exploitation of a wide variety of image processing applications. Today, there is almost no technical endeavor that is not impacted by digital image processing. There are imaging systems to reconstruct 3D images of human body internal organs from computer tomography (CT) scans; 3D images of aircraft approaching an airport displayed inside a crystal for air traffic control; photographs acquired digitally and enhanced before printing; robots that navigate and perform complex object manipulation using processed stereo imagery; etc. Yet, we have not taken full advantage of the existing-developed imaging systems.

Interest in digital image processing is oriented toward two principal areas: improvement of pictorial information for human interpretation; and processing of image data for storage, transmission, and representation in autonomous machines.

1.1 Introduction to Computer Vision

Computer vision is concerned with processing towards a goal of having a symbolic representation of the image (Fischler and Firschein, [1987]). The symbolic representation should correspond to the representation that humans obtain when they process an image. The objects in the image should be correctly labeled. Computer vision is concerned with scene interpretation. An interpretation is basically a mapping between a symbolic description of the scene and the structure of the image.

Another term is computational vision. Computational vision examines issues involved in extracting information from images (Wechsler, [1990]). One often is involved
with changing the form of the representation. The transformation from one representation of the image to another should make the solution to the analysis problem easier. Computational vision is concerned with issues of discrete data, resolution, representation, numerical methods, and computational complexity.

**Image processing** is a term with some correspondence to computational vision. Image processing usually refers to methods of transforming an image into another image with desirable properties. These properties might be that certain features in the image are enhanced or that the output image is more suitable for subsequent processing.

**Image understanding** is another term often used. This term has overlap with computer vision. It basically refers to processing methods that build symbolic descriptions of the image which are consistent with human interpretation. The symbolic description should accurately reflect the objects and their relationship to the scene.

### 1.2 What Is Digital Image Processing?

An image may be defined as a two-dimensional function, \( f(x, y) \), where \( x \) and \( y \) are spatial coordinates, and the value of \( f \) at any pair of coordinates \( (x, y) \) is called the intensity or gray level of the image at that point. When \( x, y \) and the amplitude values of \( f \) are all finite, discrete quantities, we call the image a digital image. The field of digital image processing includes the processing digital images using a digital computer. A digital image contains a finite number of elements, each having a particular location and value. These elements are referred to as picture elements, image elements, and pixels. Pixel is the term most widely used to denote the elements of a digital image.

Vision is the most advanced of our senses, so images play the most important role in human perception. Humans are limited to the visual band of the electromagnetic (EM) spectrum. Imaging machines cover almost the entire EM spectrum, ranging from gamma to radio waves. They can operate on images generated by sources that humans are not
accustomed to associating with images. These include ultrasound, electron microscopy, and computer-generated images. Thus, digital image processing encompasses a wide and varied field of applications.

There is no general agreement among authors regarding where image processing stops and other related areas, such as image analysis and computer vision, start. Some authors define image processing as a discipline in which both the input and output of a process are images. This could be a limiting and artificial boundary. For example, with this definition, the task of computing the average intensity of an image (which produces one number) would not be considered an image processing operation. On the other hand, computer vision may have, among its goals, to use computers to solve a task similar to that processed by human vision, including learning and being able to make inferences and take actions based on visual inputs. This area is a branch of artificial intelligence (AI) that has, among its objectives, the simulation of human intelligence. The area of image analysis (also called image understanding) is in between image processing and computer vision.

There are no clear-cut boundaries in the continuum from image processing at one end to computer vision at the other. However, one useful paradigm is to consider three types of computerized processes: low-level, intermediate (mid)-level and high-level.

Low-level vision is related to local properties concerned with continuity or discontinuity of intensity, texture, or color (Fischler and Firschein, [1987]). Noise reduction, smoothing, contrast enhancement, and edge detection methods fall into the category of low-level vision. These image processing methods are often general and can be applied to a variety of different image classes. Low-level vision is sometimes called early vision (Forsyth and Ponce, [2002]). Some other terminology related to low-level vision are the 2.5 dimensional sketch or the primal sketch (Winston, [1992]). This processing would determine features at each point such the edge properties, texture properties, or surface normals for 3-D
imagery. The grouping of points into regions corresponding to physical properties is often called forming the intrinsic image (Ballard and Brown [1982]). The intrinsic properties of an object are properties such as shape, color, reflectance, 3-D surface, and texture. The intrinsic image would have values for these quantities at each point. This processing derives meaningful quantities at a level above the raw pixel values. A low-level process is characterized by the fact that both its inputs and outputs are images.

Another level of vision processing is called intermediate-level vision (Fischler and Firschein, [1987]). Intermediate level vision is concerned with integrating local or point features into global constructs or regions. Examples would be edge point aggregation into lines and point aggregation into regions. This relates to image partitioning or segmentation into regions which perceptually relate to objects in the image. Typical models are lines, parallel lines, perpendicular lines or polygons which represent simple geometric objects such as roads, doors, and buildings that commonly occur in scenes. A mid-level process is characterized by the fact that its inputs generally are images, but its outputs are attributes extracted from those images.

High-level vision utilizes higher levels of processing and more complex object relationships in the modeling of objects (Fischler and Firschein [1987]). For example, processing methods might involve formal logic, constraint programming, rules, frames and other artificial intelligence programming methodologies. There is a more complex knowledge representation. Inferencing is involved in scene interpretation. The vision system has more semantic content and therefore the software is more specifically tailored to the specific image class. This corresponds to the more cognitive levels of image interpretation by humans. This methodology is sometimes called late vision.

These different levels of vision may overlap in any practical vision system. The concepts of the levels are useful mainly in characterizing the complexity of the vision system.
It is also useful in relating to the computational processing in the human vision system.

1.3 The Origins of Digital Image Processing

One of the first applications of digital images was in the newspaper industry, when pictures were first sent by submarine cable between London and New York. By introducing the Bartlane cable picture transmission system in the early 1920s, the time required to transport a picture across the Atlantic was reduced from more than one week to less than three hours. Specialized printing equipment coded pictures for cable transmission and then reconstructed them at the receiving end.

The early Batlane systems could code images in five distinct levels of gray. The number of levels was increased to 15 in 1929. The reproduction system was improved by using a system for developing a film plate via light beams, modulated by the coded picture tape.

We gave examples of early usage of digital images. However, these are not considered digital image processing results in the context of the previous definition because computers were not involved in their creation. Thus, the history of digital image processing is related to the development of the digital computer. Digital images require substantial amount of storage and computational power. Progress in the field of digital image processing has been dependent upon the development of digital computers and supporting technologies that include data storage, display, and transmission.

The first computers powerful enough to perform meaningful image processing tasks appeared in the early 1960s. The availability of those machines and the onset of the space program during that time determined the birth of what we call digital image processing today. The combination of those two developments brought into focus the potential of digital image processing concepts. Work on improving images from a space probe started at the Jet
Propulsion Laboratory (Pasadena, California) in 1964 when pictures of the moon transmitted by Ranger 7 were processed by a computer to correct various types of image distortion inherent in the on-board television camera. Figure 1.1 (Gonzales and Woods [2001], p.5) shows the first image of the moon taken by Ranger 7 on July 31, 1964 at 9:09 a.m. Eastern Daylight Time (EDT), about 17 minutes before impacting the lunar surface (the markers, called reseau marks are used for geometric corrections). This is the first image of the moon taken by a U.S. spacecraft. The experience gained with Ranger 7 helped to improve methods used to restore images from the Surveyor missions to the moon, the Mariner series of flyby missions to Mars, the Apollo manned flights to the moon, and others.

Figure 1.1 The first picture of moon by a U.S. spacecraft

In parallel with space applications, digital image processing techniques began in the late 1960s and early 1970s in medical imaging and remote Earth resources astronomy. An invention in the early 1970s of computerized axial tomography (CAT), also called computerized tomography (CT), is one of the most important events in the application of image processing in medical diagnosis. In computerized axial tomography, a ring of detectors encircles an object (or patient) and an X-ray source, concentric with the detector ring, rotates about the object. The X-rays pass through the object and are collected at the opposite end by the corresponding detectors in the ring. The source rotates and this procedure is repeated.
The algorithms used in tomography construct an image representing a “slice” through the object. Motion of the object in a direction perpendicular to the ring of detectors produces a set of such slices, which form a three-dimensional (3-D) representation of the inside of the object. Tomography was invented independently by Sir Godfrey N. Hounsfield and Professor Allan M. Cormack. They shared the Nobel Prize in Medicine for their invention in 1979. X-rays were discovered in 1895 by Wilhelm Conrad Roentgen. He received the 1901 Nobel Prize in Physics. Those two inventions made possible some of the most active application areas of image processing today.

From the 1960s until the present, the field of image processing has developed strongly. Digital image processing techniques are used in a broad range of applications. Computer-based techniques enhance the contrast or code the intensity levels into color for easier interpretation of X-rays and other images used in industry, medicine, and the biological sciences. Geographers use similar procedures to study pollution patterns from aerial and satellite imagery. Image enhancement and restoration procedures are used to process degraded images of objects that cannot be recovered or images of experimental results too expensive to be done again. In archeology, image processing methods have restored blurred pictures of rare artifacts lost or damaged after being photographed. In physics and related fields, computer techniques are used to enhance images of experiments in high-energy plasmas and electron microscopy. Similarly successful applications of image processing concepts can be found in astronomy, biology, nuclear medicine, law enforcement, defense, and industrial applications.

These examples enumerate processing results intended for human interpretation. The digital image processing techniques are also applied in solving problems related to machine perception. In this case, interest focuses on extracting information from an image in a form suitable for computer processing. Often, this information presents little resemblance to visual
features that humans use when they interpret the content of an image. Statistical moments, Fourier transform coefficients and multidimensional distance measures are examples of the type of information used in machine perception. Automatic character recognition, industrial machine vision for product assembly and inspection, military recognizance, automatic processing of fingerprints, screening of X-rays and blood samples, and machine processing of aerial and satellite imagery for weather prediction and environmental assessment utilize image processing techniques. The continuing decline in the ratio of computer price to performance and the expansion of networking and communication bandwidth via the World Wide Web and the Internet have created new opportunities for development of digital image processing.

1.4 Examples of Fields that Use Digital Image Processing

Today, there is almost no area of technical endeavor not impacted in some way by digital image processing. We can cover only a few of these applications in the context of this discussion.

The areas of application of digital image processing are so varied that some form of organization is necessary in the attempt to describe the breadth of this field. One way to develop an understanding of the extent of image processing applications is to classify images according to their source (e.g., visual, X-ray, and so on). The principal energy source for images in use today is the electromagnetic energy spectrum. Other important sources of energy include acoustic, ultrasonic, and electronic (in the form of electron beams used in electron microscopy). Synthetic images, used for modeling and visualization, are generated by computer.

Images based on radiation from the EM spectrum are the most familiar, especially images in the X-ray and visual bands of the spectrum. Electromagnetic waves can be described as propagating sinusoidal waves of varying wavelengths, or as a stream of massless
particles, each traveling in a wavelike pattern and moving at the speed of light. Each massless particle contains a certain amount of energy. Each bundle of energy is called a photon. The spectrum shown in Figure 1.2 (Gonzales and Woods [2001], p.7), ranging from gamma rays (highest energy) at one end to radio waves (lowest energy) at the other is obtained by grouping of the spectral bands according to energy per photon. The bands of the EM spectrum are not distinct but rather transition smoothly from one to the other.

![Energy Spectrum Diagram](image)

**Figure 1.2** The electromagnetic spectrum arranged according to energy per photon.

### 1.4.1 Gamma-Ray Imaging

The most important uses of imaging based on gamma rays include nuclear medicine and astronomical observations. In nuclear medicine, the patient is injected with a radioactive isotope that emits gamma rays as it decays. Images are produced from the emissions collected by gamma ray detectors. Figure 1.3 (a) (Gonzales and Woods [2001], p.8) shows an image of a complete bone scan obtained by using gamma-ray imaging. Images of this type are used to locate sites of bone pathology, such as infections or tumors. Figure 1.3 (b) (Gonzales and Woods [2001], p.8) shows another major modality of nuclear imaging called positron emission tomography (PET). The principle is the same as with X-ray tomography, mentioned briefly in section 1.3. Instead of using an external source of X-ray energy, the patient takes a radioactive isotope that emits positrons as it decays. When a positron meets an electron, both are annihilated and two gamma rays are given off. These are detected and a tomographic image is created. The image shown in Figure 1.3 (b) is one sample of a sequence that constitutes a 3-D rendition of the patient. This image shows a tumor in the brain and one in the lung, easily visible as small white masses.
A star in the constellation of Cygnus exploded about 15,000 years ago; it generated a superheated stationary gas cloud (known as Cygnus Loop) that produces an array of colors. Figure 1.3 (c) (Gonzales and Woods [2001], p.8) shows the Cygnus Loop imaged in the gamma-ray band. This image was produced by using the natural radiation of the object imaged. Figure 1.3 (d) (Gonzales and Woods [2001], p.8) shows an image of gamma radiation from a valve in a nuclear reactor.

Figure 1.3 (a) Bone scan. (b) PET image. (c) Cygnus Loop (d) Gamma Radiation
1.4.2 X-ray Imaging

X-rays are among the oldest sources of EM radiation used for imaging. The best known use of X-rays is medical diagnosis, but they are also used in industry and other areas, such as astronomy. X-rays for medical and industrial imaging are generated using an X-ray tube, a vacuum tube with a cathode and anode. The cathode is heated; this causes free electrons to be released. These electrons flow at high speed to the positively charged anode. When the electrons strike a nucleus, energy is released in the form of X-ray radiation. The energy of the X-rays is controlled by a voltage applied across the anode. The number of X-rays is controlled by a current applied to the filament in the cathode. Figure 1.4 (a) (image provided by LSU Health Sciences Center) shows a chest X-ray generated by placing the patient between an X-ray source and a film sensitive to X-ray energy. The intensity of the X-rays is modified by absorption as they pass through the patient, and the resulting energy that falls on the film develops it. Two methods are used in digital radiography to produce digital images:

![Figure 1.4 Examples of X-ray imaging. (a) Chest X-ray.](image)
(1) digitizing X-ray films; or (2) having the X-rays that pass through the patient fall directly onto devices that convert X-rays to light. The light signal is captured by a light-sensitive digitizing system.

Angiography is another important application in an area called contrast-enhancement radiography. This procedure produces images (called angiograms) of blood vessels. A catheter is inserted into an artery or vein in the groin. The catheter is threaded into the blood vessel and guided to the area to be studied. When the catheter reaches the part under investigation, an X-ray contrast medium is injected through the catheter. This enhances the contrast of the blood vessels, so that the radiologist is able to see irregularities or blockages. Figure 1.4 (b) (Gonzales and Woods [2001], p.10) shows an example of an aortic angiogram.

![Aortic angiogram](image)

The catheter can be seen as inserted into the large blood vessel on the lower left of the picture. Angiography is a major area of digital image processing, where image subtraction is used to enhance further the blood vessels under observation.

One of the best known uses of X-rays in medical imaging is computerized axial tomography. Due to their resolution and 3-D capabilities, CAT scans changed drastically
medicine from the moment they became available in the early 1970s. Each CAT image is a “slice” taken perpendicularly through the patient. Numerous slices are generated as the patient is moved in a longitudinal direction. The group of such images forms a 3-D rendition of the inside of the patient, with the longitudinal resolution proportional to the number of slice images taken. Figure 1.4 (c) (image provided by LSU Health Sciences Center) shows a typical brain CAT slice image.

Figure 1.4 (c) Brain CT

1.4.3. Imaging in the Ultraviolet Band

Applications of ultraviolet “light” are diverse. They include lithography, industrial inspection, microscopy, lasers, biological imaging, and astronomical observations. We will use examples from microscopy.
Ultraviolet light is used in fluorescence microscopy, one of the fastest growing areas of microscopy. Fluorescence was discovered in the middle of nineteenth century, when it was first observed that the mineral fluorspar fluoresces when ultraviolet light is directed upon it. The ultraviolet light itself is not visible. When a photon of ultraviolet radiation collides with an electron in an atom of a fluorescent material, the electron is lifted to a higher energy level. That’s why the excited electron relaxes to a lower level and emits light in the form of a lower-energy photon in the visible (red) light region. The fluorescence microscope uses an excitation light to irradiate a prepared specimen and then separates the weaker radiating fluorescent light from the brighter excitation light. Thus, only the emission light reaches the eye or other detector. The resulting fluorescing areas shine against a dark background and the contrast permits detection.

Fluorescence microscopy is an excellent method for studying materials that can be made to fluoresce, either in their natural form (primary fluorescence) or when treated with chemicals capable of fluorescing (secondary fluorescence). Figures 1.5 (a) and (b) (Gonzales and Woods [2001], p.12) show results typical of the capability of fluorescence microscopy.

Figure 1.5 Examples of ultraviolet imaging (a) Normal corn. (b) Corn with smut.
Figure 1.5 (a) shows a fluorescence microscope image of normal corn, and Figure 1.5 (b) shows corn infected by “smut”, a disease of cereals.

1.4.4. Imaging in the Visible and Infrared Bands

The visual band of the electromagnetic spectrum is the most familiar in all our activities. That’s why imaging in this band outweighs by far all the others in terms of scope of application. The infrared band is used in conjunction with visual imaging. We will consider applications in light microscopy, astronomy, remote sensing and industry.

Figure 1.6 (Gonzales and Woods [2001], p.13), shows several examples of images obtained with a light microscope. The examples range from pharmaceutical and
microinspection to materials characterization. Even with just microscopy, the application areas are numerous, so we cannot detail them here. The types of processes that might be applied to these images range from enhancement to measurements.

Another major area of visual processing is remote sensing, which usually includes several bands in the visual and infrared regions of the spectrum. Table 1.1 shows the so-called thematic bands in NASA’s LANDSAT satellite. The primary function of LANDSAT is to obtain and transmit images of the Earth from space, in order to monitor environmental conditions on our planet. The bands are expressed in terms of wavelength, with $1\mu m$ being $10^{-6}$ m.

Table 1.1 Thematic bands in NASA’s LANDSAT satellite.

<table>
<thead>
<tr>
<th>Band No.</th>
<th>Name</th>
<th>Wavelength</th>
<th>Characteristics and Uses</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Visible blue</td>
<td>0.45-0.52</td>
<td>Maximum water penetration</td>
</tr>
<tr>
<td>2</td>
<td>Visible green</td>
<td>0.52-0.60</td>
<td>Good for measuring plant vigor</td>
</tr>
<tr>
<td>3</td>
<td>Visible red</td>
<td>0.63-0.69</td>
<td>Vegetation discrimination</td>
</tr>
<tr>
<td>4</td>
<td>Near infrared</td>
<td>0.76-0.90</td>
<td>Biomass and shoreline mapping</td>
</tr>
<tr>
<td>5</td>
<td>Middle infrared</td>
<td>1.55-1.75</td>
<td>Moisture content of soil and vegetation</td>
</tr>
<tr>
<td>6</td>
<td>Thermal infrared</td>
<td>10.4-12.5</td>
<td>Soil moisture; thermal mapping</td>
</tr>
<tr>
<td>7</td>
<td>Middle infrared</td>
<td>2.08-2.35</td>
<td>Mineral mapping</td>
</tr>
</tbody>
</table>

In order to understand the power of multispectral imaging, consider Figure 1.7 (Gonzales and Woods [2001], p.14), which shows one image for each of the spectral bands in Table 1.1. The area in the image is Washington D.C., which includes features such as buildings, roads, vegetation, and a major river (the Potomac) going through the city. Images of population centers are used routinely (over time) to evaluate population growth and shift
patterns, pollution, and other factors that influence the environment. The differences between visual and infrared image features can be observed in these images. We notice that the river is well defined from its surroundings in Bands 4 and 5.

Some weather observation and prediction are major applications of multispectral imaging from satellites. For example, Figure 1.8 (Gonzales and Woods [2001], p.15) is an image of a hurricane taken by a National Oceanographic and Atmospheric Administration (NOAA) satellite using sensors in the visible and infrared bands. The eye of the hurricane is visible in this image.

A major area of imaging in the visual spectrum is the automated visual inspection of manufactured goods. Figure 1.9 (Gonzales and Woods [2001. p.18), shows some examples. Figure 1.9 (a) is a controller board for a CD-ROM drive. A typical image processing task with this type of product is to detect the missing parts (the black square on the top, right quadrant of the image is a missing component). Figure 1.9 (b) is an image pill container. The objective is to have a machine look for missing pills. Figure 1.9 (c) shows an application in which image processing is used to detect bottles not filled at an acceptable level.
Figure 1.8 Multispectral image of Hurricane Andrew taken by NOAA GEOS (Geostationary Environmental Operational Satellite) sensors.

Figure 1.9 (d) shows a clear-plastic part with a number of air pockets in it. The purpose of industrial inspection is to detect anomalies like these in different products, including wood and cloth. Figure 1.9 (e) shows a batch of cereal during inspection for color and the presence of anomalies such as burned flakes. Finally, Figure 1.9 (f) shows an image of an intraocular implant (replacement lens for the human eye). A “structured” light illumination technique was used to highlight providing easier detection of flat lens deformities toward the center of the lens. The markings at 1 o’clock and 5 o’clock are tweezer damage. Most of the other small speckle detail is debris. This type of inspection aims to find damaged or incorrectly manufactured implants automatically, before the products are packaged.

As a final illustration of image processing in the visual spectrum, consider Figure 1.10 (Gonzales and Woods [2001], p.19). Figure 1.10 (a) shows a thumb print. Images of fingerprints are processed by the computer, either to enhance them or to find features that aid in the automated search of a database for potential matches. Figure 1.10(b) shows an image of paper currency. Applications of image processing in this area include automated counting
Figure 1.9 Some examples of manufactured goods checked using digital image processing. (a) A circuit board controller. (b) Packaged pills. (c) Bottles. (d) Bubbles in clear-plastic product. (e) Cereal. (f) Image of intraocular implant.

and, in law enforcement, the reading of the serial number in order to track and identify bills.

The two vehicle images shown in Figures 1.10 (c) and (d) are examples of automated license plate reading. The light rectangles indicate the area in which the imaging system detected the plate. The black rectangles show the results of automated reading of the plate content by the system. License plate and other applications of character recognition are used extensively for traffic monitoring and surveillance.
1.4.5. Imaging in the Microwave Band

The most important application of imaging in the microwave band is radar. An imaging radar is able to collect data over virtually any region at any time, regardless of weather or ambient lighting conditions. Some radar waves can penetrate clouds, and under certain conditions can also see through vegetation, ice, and dry sand. In many cases, radar is the only modality to explore inaccessible regions of the Earth’s surface. An imaging radar works like a flash camera; it produces its own illumination (microwave pulses) to illuminate an area on the ground and take a snapshot image. Instead of a camera lens, a radar uses an antenna and digital computer processing to record its images. In a radar image, one can see only the microwave energy reflected back toward the radar antenna.
Figure 1.11 (Gonzales and Woods [2001], p.20) shows a spaceborne radar image covering a rugged mountainous area of southeast Tibet, about 90 km east of the city of Lhasa. In the lower right corner is a wide valley of the Lhasa river. Mountains in this area reach about 5800 m (19,000 ft) above sea level, while the valley floors lie about 4300 m (14,000 ft) above the sea level. We notice the clarity and detail of the image, not impeded by clouds or other atmospheric conditions that normally interfere with images in the visual band.

Figure 1.11 Spaceborne radar image of mountains in southeast Tibet.

1.4.6 Imaging in the Radio Band

Similar to the case of imaging at the lower EM end of the spectrum from gamma rays, the major applications of imaging in the radio band are in medicine and astronomy. In medicine, radio waves are used in magnetic resonance imaging (MRI). This technique introduces a patient in a powerful magnet and passes radio waves through his/her body in short pulses. Each pulse causes a responding pulse of radio waves emitted by the patient’s tissues. The computer determinates the origin and the strength of these signals and produces a two-dimensional image of a section of the patient. MRI can produce images in any plane. Figure 1.12 shows an MR image of a human bone (image provided by LSU Health Sciences Center).

In the Figure 1.13(Gonzales and Woods [2001]), the last image shows an image of the Crab Pulsar in the radio band. Images of the same region taken in most of the bands discussed
earlier are also shown. We note that each image gives a different “view” of the Pulsar.

1.4.7 Examples of Using Other Imaging Modalities

Imaging in the electromagnetic spectrum is dominant, but there are a number of other modalities that are also important. We will discuss acoustic imaging, electron microscopy, and synthetic (computer-generated) imaging.

Imaging using “sounds” has application in geological exploration, industry, and medicine. Geological applications use sound in the low end of the sound spectrum (hundreds of Hertz) while imaging in other areas use ultrasound (millions of Hertz). Mineral and oil exploration are the most important commercial applications of image processing in geology.
An important approach used for image acquisition over land is to use a large truck and a large flat steel plate. The plate is pressed on the ground by the truck, and the truck is vibrated through a frequency spectrum up to 100 Hz. The composition of the earth below the surface determines the strength and the speed of the returning sound waves; these are analyzed on the computer and images are generated as a result of the analysis.

In the case of marine acquisition, the energy source consists usually of two air guns towed behind a ship. Hydrophones placed in cables (towed behind the ship, laid on the bottom of the ocean or hung from buoys) detect returning waves. The two air guns are alternately pressurized to ~2000 psi and then set-off. The constant motion of the ship provides a transversal direction of motion. This and the returning sound waves are used to generate a 3-D map of the composition of the Earth below the bottom of the ocean.

Figure 1.14 (Gonzales and Woods [2001], p.22) shows a cross-sectional image of a well-known 3-D model, used to test the performance of seismic imaging algorithms. The arrow points to a hydrocarbon (oil and/or gas) trap. This target is brighter than the surrounding layers. Seismic interpreters look for these “bright spots” to help find oil and gas. Many seismic reconstruction algorithms experience difficulties in imaging this target.

Figure 1.14 Cross-sectional image of a seismic model. The arrow points to a hydrocarbon (oil and/or gas) trap.

Although ultrasound imaging is used routinely in manufacturing, the best known applications of this technique are in medicine, especially in obstetrics. Images of unborn
babies help to evaluate the health of their development. A byproduct of this examination is determining the sex of the baby. Ultrasound images are generated using the following procedure:

- The ultrasound system (a computer, an ultrasound probe, and a display) transmits high-frequency (1 to 5 MHz) sound pulses into the body.
- The sound waves travel into the body and encounter a boundary between tissues (e.g., between soft tissue and bone). Some of the sound waves are reflected back to the probe; some travel on further until they reach another boundary and will be reflected.
- The probe picks up the reflected waves; these are relayed to the computer.
- The machine calculates the distance from the probe to the tissue or organ boundaries using the speed of sound in tissue (1540 m/s) and the time of the each echo’s return.
- The system displays the distances and intensities of the echoes on the screen, and a two-dimensional image is formed.

In a typical ultrasound image, millions of pulses and echoes are sent and received each second. The probe can be moved along the surface of the body and angled to obtain various views. Figure 1.15 (image provided by LSU Health Sciences Center) shows an example of an ultrasound image.

We will consider some examples of electron microscopy. Electron microscopes function as their optical counterparts, except that they use a focused beam of electrons instead of light to image a specimen. The operation of electron microscopes can be described by several basic steps:

- A stream of electrons is produced by an electron source and accelerated toward the specimen using a positive electrical potential. Metal apertures and magnetic lenses are
used to focus this stream into a monochromatic beam.

- This beam is focused onto the sample using a magnetic lens. Interactions occur inside the irradiated sample, modifying the electron beam. These interactions and effects are detected and transformed into an image. These steps are carried out in all electronic microscopes.

A transmission electron microscope (TEM) works out much like a slide projector. A projector transmits a beam of light through the slide; the light passes through the slide, being modified by the contents of the slide. The transmitted beam is projected onto the viewing screen; an enlarged image of the slide is produced. TEMs work the same way, except that they use a beam of electrons through a specimen. The fraction of the beam transmitted through the specimen is projected onto a phosphor screen. The interaction of the electrons with the phosphor produces light and a viewable image. A scanning electron microscope
(SEM) scans the electron beam and records the interaction of beam and sample at each location. This creates one dot on a phosphor screen. A complete image is formed by a raster scan of the beam through the sample, much like a TV camera. The electrons interact with a phosphor screen and produce light. SEMs are suitable for “bulky” samples, while TEMs require thin samples.

Electron microscopes are capable of very high magnification. While light microscopy is limited to magnifications on the order of 1000 times, electron microscopes can achieve magnification of 10,000 times or more. Figure 1.16 (Gonzales and Woods [2001], p.24), shows two SEM images of specimen failures due to thermal overload.

Figure 1.16 (a) Magnified 250 times SEM image of a tungsten filament following thermal failure. (b) Magnified 2500 times SEM image of damaged integrated circuit. The white fibers are oxides from thermal destruction.

We end our discussion of imaging modalities by briefly looking at images generated by computer. Fractals are examples of computer-generated images (Lu [1997]). A fractal is an iterative reproduction of a basic pattern according to some mathematical rules. For instance, tiling is one of the simplest ways to generate a fractal image. A square can be subdivided into four square subregions; each can be divided further into four smaller square regions, and so on. Each subsquare can be filled according to complex rules, so beautiful tile images can be generated.

Shapes such as coastlines and clouds are not very well described by traditional geometry (Barnsley, M.F., Devaney, R.L., Mandelbrot, B.B., Peitgen, H.-O., Saupe, D. and
Voss, R.F., [1988], p.21). These shapes have an invariance under magnification. Another work for invariance under magnification is self similarity. A basic concept related to fractal analysis is the concept of self similarity. This embodies the idea of a set or line being self similar to itself at different scales. Fractal shapes are independent of scale. The statistical self similarity is an essential characteristics of fractals in nature. It is quantified by a fractional dimension number that is not an integer. Natural patterns are not fractals at all scales. Under magnification the parts are not exactly like the larger parts. They are statistically self similar. Natural patterns will exhibit a fractal pattern over a range of scales and not all scales.

Figure 1.17 (Gonzales and Woods [2001], p.25) shows some examples of fractals. The geometry can be arbitrary. For instance, the fractal image could be built radially from a center point. Figure 1.17 (a) shows a fractal grown in this way. Figure 1.17 (b) shows another fractal (a “moonscape”) that provides an analogy to the images of the space used as illustrations in some images of the preceding sections.

Figure 1.17 (a) and (b) Fractal images. (c) and (d) Images generated from 3-D computer models of the objects shown.
Fractal images tend toward artistic, mathematical formulations of “growth” of subimage elements using some rules. They are useful sometimes as random textures. 3-D modeling is a more structured approach to image generation by computer. This is an area considered to be an intersection between image processing and computer graphics and represents the basis for many 3-D visualization systems. Figures 1.17 (c) and (d) show examples of computer-generated images. The original object is created in 3-D; images can be generated in any perspective from plane projections of the 3-D volume. Images of this type can be used for medical training and for a host of other applications, such as criminal forensics and special effects.

1.5 Fundamentals Steps in Digital Image Processing

Digital image processing encompasses processes whose inputs and outputs are images and, in addition, processes that extract attributes from images, up to including the recognition of individual objects. For example, we may consider the area of automated analysis of text. The processes of acquiring an image of the area containing the text, preprocessing that image, extracting (segmenting) the individual characters, describing the characters in a form suitable for computer processing, and recognizing that the individual characters are within the scope of digital image processing. Making sense of the page’s content may be included in the domain of image analysis and even computer vision. Digital image is used successfully in a broad range of areas of exceptional social and economic value.

The image processing techniques can be divided into two broad categories: methods whose input and output are images and methods whose input may be images, but whose outputs are attributes extracted from those images. We summarize this organization in Figure 1.18. The diagram does not imply that that every process is applied to an image. The intention is to convey an idea of all the methodologies that can be applied to images for different purposes and possibly with different objectives.
Outputs of these processes generally are images

- Color image processing
- Wavelets and multiresolution processing
- Compression
- Morphological processing
- Image restoration
- Image enhancement
- Knowledge base
- Segmentation
- Representation & description
- Object recognition

Problem domain

Figure 1.18 Fundamentals steps in digital image processing

**Image acquisition** is the first process shown in Figure 1.18. The discussion in Section 1.4 presented several sources of digital images. The types of images of interest are generated by the combination of an “illumination” source and the reflection or absorption of energy from that source by the elements of the “scene” being imaged. Using quotes for *illumination* and *scene* aims to emphasize the fact that they are considerably more general than the familiar situation in which a visible light source illuminates a common everyday 3-D scene. For example, the illumination may originate from a source of electromagnetic energy such as radar, infrared, or X-ray energy. But it could originate from less traditional sources, such as
ultrasound or even a computer-generated illumination pattern. Similarly, the scene elements could be familiar objects, but they can just as easily be molecules, buried rock formations, or a human brain. Depending on the nature of the source, illumination energy is reflected from, or transmitted through objects. An example in the first category is light reflected from a planar surface. An example falling in the second category is when X-rays pass through a patient’s body generating a diagnostic X-ray film. In some applications, the reflected or transmitted energy is focused onto a photoconverter (e.g. a phosphor screen), which converts the energy into visible light. Electron microscopy and some applications of gamma imaging use this approach.

**Image enhancement** is among the simplest and most appealing areas of digital image processing. The idea behind enhancement is to bring out detail that is obscured, or simply to highlight some features of interest in an image. A familiar example of enhancement is when we increase the contrast of an image because “it looks better”. We have to take into account that enhancement is a very subjective area of image processing.

**Image restoration** is an area that also deals with improving the appearance of an image. However, unlike enhancement, which is subjective, image restoration is objective, in the sense that restoration techniques are based on mathematical or probabilistic models of image degradation. Enhancement is based on human subjective preferences regarding what is considered a “good” enhancement result.

**Color image processing** is an area that has been given importance because of the significant increase in the use of digital image over the Internet. The use of color in image processing is influenced by two main factors. First, color is a powerful descriptor that often simplifies object identification and extraction from a scene. Second, humans can discern thousands of color shades and intensities, compared to about only two dozen shades of gray. This second factor is important in manual (when performed by humans) image analysis.
**Wavelets** are the foundation for representing images in various degrees of resolution. Wavelets provide a powerful and remarkable set of tools for a wide range of problems, such as:

- **Object detection**: What methods should we use to pick out a small image from a larger, more complicated image?
- **Fingerprint compression**: The FBI has more than 25 million fingerprint records. If these fingerprint records were digitized without any compression, they would need more than 250 trillion bytes of storage. Is there a method to compress these records to a manageable size, without losing significant details?
- **Image denoising**: Images formed by electron microscopes and by optical lasers are often contaminated by large amounts of unwanted clutter (referred to as *noise*). Can this noise be removed to clarify the image?
- **Image enhancement**: When an optical microscope image is recorded, it often suffers from blurring. How can the appearance of the objects in these images be sharpened?
- **Image recognition**: How do humans recognize faces? Can we teach machines to do it?
- **Compression deals with techniques for reducing the storage required saving an image, or the bandwidth required transmitting it.** The storage technology has improved significantly over the past decade; this does not apply to the transmission capacity. This is particularly true in uses of the Internet, characterized by significant pictorial content. Image compression is familiar to most users in the form of image file extensions, such as the jpg file extension used in the JPEG (Joint Photographic Experts Group) image compression standard.

**Morphological processing** includes techniques for extracting image components. Morphology is an approach to image processing based upon the shape of structuring elements. Morphology has well defined mathematical properties and the theory helps to
extend the applicability of the methods. The theory evolved from algebra and several basic image processing methods that had proven useful, such as forming the skeleton of objects. Morphological operations preserve the basic properties of the object while removing irrelevant features. The basic shape features of an object are preserved. Morphology is a study of the form and structure of objects.

**Segmentation** procedures partition an image into its constituent parts or objects. In general, autonomous segmentation is one of the most difficult tasks in digital image processing. A robust segmentation procedure that attempts to find a successful solution of imaging problems may be a lasting process that requires individual identification of objects. On the other hand, weak or erratic segmentation algorithms are very likely to fail. In general, the more accurate the segmentation, the more likely recognition is to succeed.

**Representation and description** are operations applied to the output of a segmentation stage. The result of applying the segmentation procedure is raw pixel data that represent either the boundary of a region (i.e., the set of pixels separating one image region from another) or all the pixels in the region itself. In either case, it is necessary to convert the data to a form suitable for computer processing is necessary. Representing the data as a boundary or as a complete region is an important decision. Boundary representation is appropriate when the focus is on external shape characteristics, such as corners and inflections. Regional representation is appropriate when the focus is on internal properties, such as texture or skeletal shape. For some applications, these representations may complement each other. Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computer processing. A method for describing the data is necessary, so that features of interest are highlighted. Description, also called feature selection, extracts attributes that result in some quantitative information of interest or are helpful to differentiate one class of objects from another.
**Recognition** is the process that assigns a label (e.g. “building”) to an object based on its descriptors. The scope of digital image processing includes recognition of individual image regions, which are called objects or patterns.

Knowledge about a problem domain is coded into an image processing system in the form of a knowledge database. This knowledge may be simple: e.g., a map with the regions of the image where the information of interest is known to be located; this limits the search that has to be performed to seek that information. The knowledge base can be quite complex: e.g., an inter-related list of all major possible defects in a material inspection problem or an image database that contains high-resolution satellite images of a region in connection with change-detection application. The knowledge base not only coordinates the operation of each processing module, but also controls the interaction between modules. That’s why, in Figure 1.18, we use the double-headed arrows between the processing modules and the knowledge base.

Viewing the results of image processing can take place at the output of any stage in Figure 1.18. We also note that not all image processing applications require the complexity of interactions implied by Figure 1.18. Not even all modules are needed in some cases. For example, image enhancement for human visual interpretation seldom requires the use of any of the other stages in Figure 1.18. In general, as the complexity of an image processing task increases, so does the number of processes required to solve the problem.

### 1.6 Components of an Image Processing System

In the mid-1980s, numerous models of image processing systems throughout the world were peripheral devices that attached to host computers. Late in 1980s and early in 1990s, the market offered image processing hardware in the form of single boards designed to be compatible with industry standard buses and fit into engineering workstations and personal
computers. In addition to lowering costs, this market shift served as a catalyst for a significant number of new companies that develop software for image processing.

Massive imaging applications such as processing of satellite images may require large-scale image processing systems. The trend is to miniaturize and blend small computers with specialized image processing hardware. Figure 1.19 shows the basic components comprising a typical general-purpose system used for digital image processing.

Figure 1.19 Components of a general-purpose image processing system.
With regard to sensing, two elements are required to acquire digital images. The first is a physical device sensitive to energy radiated by the object to be imaged. The second, called a digitizer, is a device used to convert the output of the physical sensing device into digital form. For example, in a digital video camera, the sensors produce an electrical output proportional to light intensity. The digitizer converts these outputs into digital data.

Specialized image processing hardware usually consists of the digitizer, hardware that performs other primitive operations, such as an arithmetic logic unit (ALU), responsible for arithmetic and logical operations in parallel on entire images. For example, ALU can be used in averaging images as quickly as they are digitized, for the purpose of noise reduction. This type of hardware is sometimes called a front-end subsystem, and speed is one of its features. This unit performs functions that require fast data throughputs (e.g., digitizing and averaging video images at 30 frames/s) that the typical main computer cannot do.

The computer in an image processing system is a general-purpose computer and can range from PC to supercomputer. In specialized applications, sometimes specially designed computers are used to achieve a required level of performance, but our interest here is on general-purpose image processing systems. In these systems, almost any well-equipped PC is suitable for off-line image processing tasks.

Software for image processing consists of specialized modules assigned to specific tasks. A well-designed package also includes a capability for the user to write code in order to use the specialized modules. More sophisticated software packages allow the integration of those modules and general-purpose software commands from at least one computer language.

Mass storage is necessary in image processing applications. An image of 1024×1024 pixels, with each pixel’s intensity being an 8-bit quantity, requires one megabyte of storage if the image is not compressed. When we operate on thousands or even millions of images, providing adequate storage in an image processing system can be a challenge. Digital storage
for image processing applications falls into three main categories: (1) short-term storage for use during processing, (2) on-line storage for relatively fast recall, and (3) archival storage, for the data that is accessed infrequently. Storage is measured in bytes (eight bits), Kbytes (1024 bytes), Mbytes (1024 × 1024 bytes), Gbytes (giga, or \((1024)^3\) bytes), and Tbytes (tera, or \((1024)^4\) bytes).

Short-term storage is computer memory or specialized boards, called frame buffers, that store one or more images and can be accessed rapidly, usually at video rates (i.e., at 30 complete images per second). Using frame buffers allows virtually instantaneous image zoom, as well as scroll (vertical shifts) and pan (horizontal shifts). Frame buffers usually are a specialized image processing hardware unit shown in Figure 1.19. On-line storage includes magnetic disks or optical-media storage. The key feature of on-line storage is frequent access to the stored data. The defining characteristics of archival storage are massive storage requirements and infrequent need for access. Magnetic tapes and optical disks housed in “jukeboxes” are the usual media for archival.

Image displays in use today are mainly color (preferably flat screen) monitors. The outputs of image and graphics display cards are an integral part of the computer system. The display cards available commercially can meet the requirements for most image display applications. In some cases, it is necessary to have stereo displays, implemented in the form of headgear containing two small displays embedded in goggles worn by the user.

Hardcopy devices for acquiring images include laser printers, film cameras, heat-sensitive devices, inkjet units, and digital units, such as optical and CD-ROM disks. Film provides the highest possible resolution, but paper is the medium of choice for written material. For presentations, images are displayed on film transparencies or in a digital medium if image projection equipment is used.
Networking is the default in any computer system in use. Image processing applications manipulate large amounts of data; that’s why a key consideration in image transmission is bandwidth. In dedicated networks, this typically is not a problem, but communications with remote sites via the Internet are not always as efficient. Using optical fiber and other broadband technologies will bring improvements.
Chapter 2

Image Enhancement in the Spatial Domain

A principal objective of enhancement is to process an image so that it has a better presentation than the original image for a specific application. The image enhancement algorithm used depends on the objective to be achieved as well as the application. For example, a method that is quite useful to enhance X-ray images may not be the best approach to enhance images of Mars transmitted by a space probe. Regardless of the method used, image enhancement is one of the most interesting and visually appealing areas of image processing.

Image enhancement falls into two broad categories: spatial domain and frequency domain. The term spatial domain refers to the image plane, and approaches in this category are based on direct manipulation of pixels in an image. Frequency domain processing techniques are based on modifying the Fourier transform of an image. Enhancement techniques based on various combinations of methods from these two categories are not unusual.

There is no known general theory of image enhancement. When an image is processed for visual interpretation, the viewer is the ultimate judge of how well a method works. The visual evaluation of image quality is highly subjective. The definition of a “good image” becomes an elusive standard used to evaluate algorithm performance. When the problem is the processing images for machine perception, this evaluation task is somewhat easier. For example, in a character recognition application, and not considering other issues such as computational requirements, the best image processing method would be the one that yields the best machine recognition results. However, even in situations when a clear criterion of
performance can be imposed on the problem, a certain amount of trial and error usually is required before any image enhancement approach is selected.

2.1. Background

Spatial domain methods are procedures that operate directly on the pixels composing the image. Spatial domain processes will be denoted by the expression

$$g(x, y) = T[f(x, y)]$$

(2.1-1)

where $f(x, y)$ is the input image, $g(x, y)$ is the processed image, and $T$ is an operator on $f$, defined over some neighborhood of $(x, y)$. In addition, $T$ can operate on a set of images, such as performing the pixel-by-pixel sum of $K$ images for noise reduction.

The principal approach in defining a neighborhood of a point $(x, y)$ is to use a square or rectangular subimage area centered at $(x, y)$, as Figure 2.1 shows. The center of the subimage is moved from pixel-to-pixel starting at the top left corner. The operator $T$ is applied to each location $(x, y)$ to yield the output, $g$ at that location. The process utilizes only the pixels in the area of the image spanned by the neighborhood. Square and rectangular

Figure 2.1 A $3 \times 3$ neighborhood about a point $(x, y)$ in an image.
arrays are the most predominant neighborhood shapes because of their ease of implementation.

The simplest form of $T$ is when the neighborhood is of size $1 \times 1$ (a single pixel). In this case, $g$ depends only on the value of $f(x, y)$ and $T$ becomes a gray-level (also called an intensity or mapping) transformation function of the form

$$s = T(r)$$  \hspace{1cm} (2.1-2)

where $r$ and $s$ are variables denoting the gray level of $f(x, y)$ and $g(x, y)$ at any point $(x, y)$. For example, if $T(r)$ has the form shown in Figure 2.2 (a), the effect of this transformation would be to produce an image of higher contrast than the original version by transforming the levels below $m$ into darker levels and the levels above $m$ into brighter levels. This technique is called contrast stretching. The transformation function compresses the values of $r$ below $m$ into a narrow range of $s$, toward black. The opposite effect is obtained for values of $r$ above $m$. In the case depicted in Figure 2.2 (b), $T(r)$ produces a two-level (binary) image. A mapping of this form is called a thresholding function. Some simple processing approaches can be expressed with gray-level transformations. Because enhancement at any point in an image depends only on the gray level at that point, techniques in this category often are referred to as point processing.

Larger neighborhoods allow more flexibility. The general approach uses a function of the values of $f$ in a predefined neighborhood of $(x, y)$ to determine the value of $g(x, y)$. One of the principal approaches is the use of masks (also referred to as filters, kernels, templates or windows). Basically, a mask is a small 2-D array, in which the values of the mask coefficients determine the process, e.g. image sharpening. Enhancement techniques based on this approach are often referred to as mask processing or filtering.
2.2 Gray Level Transformations

Among the simpler image enhancement techniques are the gray-level transformation functions. The values of pixels, before and after processing, will be denoted by \( r \) and \( s \), respectively. As mentioned in the previous section, these values are related by an expression in the form \( s = T(r) \), where \( T \) is a transformation that maps a pixel value \( r \) into a pixel value \( s \). We use digital quantities; that’s why values of the transformation function typically are stored in a one-dimensional array and the mappings from \( r \) to \( s \) are implemented via table lookups. For an 8-bit environment, a lookup table that contains the values of \( T \) will have 256 entries.

Three basic types of function are used frequently for image enhancement: linear (negative and identity transformations), logarithmic (log and inverse-log transformations), and power-law (\( n^{\text{th}} \) power and \( n^{\text{th}} \) root transformations).
2.2.1 Negatives

The negative of a gray level image in the range \([0, L-1]\) is obtained by the negative transformation shown in Figure 2.3, given by the expression:

\[
s = L - 1 - r
\]  
(2.2-1)

Using this formula reverses the intensity levels of an image and produces the equivalent of a negative photograph. This processing is particularly suited for enhancing white or gray detail embedded in dark regions of an image, especially when the black areas are dominant. An example is shown in Figure 2.4 (image provided by LSU Health Sciences Center).

2.2.2 Log Transformations

The general form of a log transformation is

\[
s = c \log(1 + r)
\]  
(2.2-2)

where \(c\) is a constant, and it is assumed \(r \geq 0\). This transformation maps a narrow range of
low gray-level values into a wider range of levels; the opposite is true of the higher values. We could use this transformation to expand the values of dark pixels in an image while low gray-level values into a wider range of levels; the opposite is true of the higher values. We could use this transformation to expand the values of dark pixels in an image while compressing the higher-level values; the opposite is true of an inverse log transformation.

Any curve with the shape of a log function would accomplish spreading/compressing of gray levels in an image. In fact, the power-law transformations are even more versatile than the log transformation. However, the log function compresses the dynamic range of an image that has large variations in pixel intensities. A classic illustration of an application in which pixel values have a large dynamic range is the Fourier spectrum. We may encounter spectrum values that range from 0 to $10^6$ or more. Image display systems are not able to reproduce such a wide range of intensity values. The effect is significant details will be lost in the display of a typical Fourier spectrum. We applied the log transformation given in (2.2-2) with $c = 1$ to the Fourier spectrum in Figure 2.5 and displayed the result in Figure 2.6.

2.2.3 Power-Law Transformations

Power-law transformations are of the form
Figure 2.5 Fourier spectrum

Figure 2.6 Result of applying the log transformation in (2.2-2) to the Fourier spectrum
\[ s = cr^\gamma \text{ or } s = c(r + \varepsilon)^\gamma \text{ (to account for an offset)} \]  \hspace{1cm} (2.2-3)

where \( c \) and \( \gamma \) are positive constants. Offsets typically are an issue of display calibration so they are normally ignored in (2.2-3). Plots of \( s \) versus \( r \) for various values of \( \gamma \) are shown in Figure 2.7. As with log transformations, power-law curves with fractional values \( \gamma \) map a narrow range of darker values into a wider range of output values, with the opposite for higher values. Unlike the log function, we notice a family of possible transformation curves corresponding to different values of \( \gamma \). We see in Figure 2.7 that curves generated for values \( \gamma > 1 \) have the opposite effect compared to those generated with values of \( \gamma < 1 \).

![Figure 2.7 Plots of the equation \( s = cr^\gamma \) for various values of \( \gamma \) (\( c = 1 \) in all cases)](image)

A variety of devices used to capture, print, and display images respond to a power transformation. By convention, the exponent in the power-law equation is referred to as
\( \gamma (\text{gamma}) \). The process used to correct power-law response phenomena is called the \textit{gamma correction}.

Gamma correction is important in displaying an image accurately on a monitor. Images that are not corrected can look either bleached out or too dark. Trying to reproduce colors accurately also requires knowledge of gamma correction because varying the value of gamma changes not only the brightness, but also the ratios of red-to-green-to-blue. Gamma correction has become important, as use of digital images on the Internet has increased. Some computer systems even have built-in partial gamma correction.

In addition to gamma correction, power-law transformations are useful for general-purpose contrast manipulation. Figure 2.8 (image provided by LSU Health Sciences Center)

Figure 2.8 Magnetic resonance (MR) image of a head shows a magnetic resonance (MR) image of a head. Since the given image is predominantly dark, an expansion of gray levels is desirable. This can be accomplished with a power-law transformation with a fractional exponent. The images shown in Figures 2.9-2.11 were
Figure 2.9 Result of applying the transformation in (2.2-3) with $c = 1$ and $\gamma = 0.6$

Figure 2.10 Result of applying the transformation (2.2-3) with $c = 1$ and $\gamma = 0.4$
Figure 2.11 Result of applying the transformation (2.2-3) with $c = 1$ and $\gamma = 0.3$

obtained by processing Figure 2.8 with the power-law transformation function of (2.2-3). The values of $\gamma$ corresponding to images 2.9 through 2.11 are 0.6, 0.4 and 0.3 respectively (the value of $c$ was 1 in all cases). We notice that, as $\gamma$ decreased from 0.6 to 0.4, more detail became visible. A further decrease of $\gamma$ to 0.3 enhanced a little more detail in the background, but began to reduce contrast to the point where the image started to have a very slight “washed-out” look, especially in the background. When we compare all results, we see that the best enhancement in terms of contrasts and discernable detail was obtained with $\gamma = 0.4$

A value of $\gamma = 0.3$ is an appropriate limit below which contrast in this particular image would be reduced to an unacceptable level.

Figure 2.12 (image provided by LSU Health Sciences Center) shows the opposite problem when compared with Figure 2.8. The image to be enhanced now has a washed-out appearance that indicates that a compression of gray levels is desirable. This can be
accomplished with (2.2-3) using values of $\gamma$ greater than 1. The results of processing Figure 2.12 with $\gamma = 3.0$, 4.0, 5.0 are shown in Figures 2.13-2.15. Suitable results were obtained with $\gamma = 3.0$ and $\gamma = 4.0$, the latter having a more appealing appearance because it shows higher contrast. The result obtained with $\gamma = 0.5$ has areas that are too dark, in which some detail is lost. The dark regions in the lower half are examples of such areas.

### 2.2.4 Piecewise-Linear Transformation Functions

Using piecewise linear functions constitutes a complementary approach to the methods previously discussed. The advantage of piecewise linear functions over the types of functions previously discussed is that the piecewise functions can be arbitrarily complex. A practical implementation of some important transformations can be expressed as piecewise functions. The principal disadvantage of piecewise functions is that their specification requires more user input.
Figure 2.13 Result of applying the transformation in (2.2-3) with $c = 1$ and $\gamma = 3.0$

Figure 2.14 Result of applying the transformation in (2.2-3) with $c = 1$ and $\gamma = 4.0$
2.2.4.1 Contrast Stretching

One of the simpler piecewise linear functions is the contrast-stretching transformation. Poor illumination, lack of dynamic range in the imaging sensor, or even an incorrect setting of a lens aperture during image acquisition can produce low-contrast images. The method used in contrast stretching consists of increasing the dynamic range of the gray levels in the image being processed.

Figure 2.16 shows a typical transformation used for contrast stretching. The points \((r_1, s_1)\) and \((r_2, s_2)\) determine the shape of the function that produces the transformation. If \(r_1 = s_1\) and \(r_2 = s_2\), the transformation is a linear function that determines no changes in gray levels.

If \(r_1 = r_2\), \(s_1 = 0\) and \(s_2 = L - 1\), the transformation will be a thresholding function that creates a binary image. Intermediate values of \((r_1, s_1)\) and \((r_2, s_2)\) produce various
degrees of spread in the gray levels of the output image and this will affect the contrast. In general, we assume that \( r_1 \leq r_2 \) and \( s_1 \leq s_2 \), so that the function is single valued and monotonically increasing. This condition preserves the order of gray levels; this prevents introducing intensity artifacts in the processed image.

Figure 2.17 (image provided by LSU Health Sciences Center) shows an 8-bit image with low contrast. Figure 2.18 shows the result of contrast stretching, for the values \((r_1, s_1) = (r_{\min}, 0)\) and \((r_2, s_2) = (r_{\max}, L - 1)\) where \(r_{\min}\) and \(r_{\max}\) denote the minimum and maximum gray levels in the image, respectively. The transformation function stretched the levels linearly from the original range to the full range \([0, L - 1]\). Figure 2.19 shows the result of using the thresholding function with \(r_1 = r_2 = m\), where \(m\) is the mean gray level in the image. The original image is a magnetic resonance image of a human abdomen.

2.2.4.2 Gray-Level Slicing

Highlighting a specific range of gray levels in an image is often desirable. Applications include enhancing features such as water masses in satellite imagery and flaws in X-ray images. There are two basic methods for level slicing. One is to display a high value for all gray levels in the range of interest and a low value for all other gray levels. This
Figure 2.17 A low-contrast image (human abdomen)

Figure 2.18 Result of contrast stretching applied to the image in Figure 2.17
transformation, shown in Figure 2.20 (a), produces a binary image. The other approach, based on the transformation shown in Figure 2.20 (b), brightens the desired range of gray levels but preserves the background and gray-level tonalities in the image. Figure 2.20 (c) (Gonzales and Woods [2001], p.87) shows a grayscale image, and Figure 2.20 (d) shows the result of using the transformation in Figure 2.20 (a).

2.2.4.3 Bit-Plane Slicing

Instead of highlighting gray-level ranges, highlighting the contribution made to the total image appearance by specific bits may be desired. If we assume that each pixel in the image is represented by 8 bits, we may represent the image as eight 1-bit planes, ranging from bit-plane 0 for the least significant bit to bit-plane 7 for the most significant bit. Plane 0 contains all the lowest order bits in the bytes that comprise the pixels in the image and plane 7 all the highest-order bits. Figure 2.21 exhibits these ideas, and Figure 2.23 shows the various bit
planes for the image shown in Figure 2.22 (Gonzales and Woods [2001], p.88). The higher-order bits contain the majority of the visually significant data. The other bit planes contribute to the more subtle details in the image. The process of splitting a digital image into bit planes helps to analyze the importance of each bit of the image and to determine the adequacy of the number of bits necessary to quantize each pixel. Also, this type of decomposition can be used for image compression.
The binary image for bit-plane 7 can be obtained by processing the original image with a thresholding gray-level transformation function that maps all levels of the image \([0, 127]\) to one level (for example 0) and maps all levels \([128, 255]\) to another (for example 255). The binary image for bit-plane 7 in Figure 2.23 was obtained in this manner. If we process the input image with a thresholding gray-level transformation function that maps
all levels in the image $[0, 63] \cup [128, 191]$ to one level (for example 0) and maps all levels $[64, 127] \cup [192, 255]$ to another (for example 255), we will obtain the binary image for bit-plane 6. We can split each sub-interval into two other smaller sub-intervals of the same length, based on the value of the next significant bit and use a similar thresholding gray-level transformation.

2.3 Histogram Processing

The histogram of a digital image with gray levels $[0, L-1]$ is a discrete function

$$ h(r_k) = n_k $$

where $r_k$ is the $k^{th}$ gray level and $n_k$ is the number of pixels in the image having gray level $r_k$. It is a common practice to normalize a histogram by dividing each of its values
by the total number of pixels in the image, denoted by $n$. A normalized histogram is produced by $p(r_k) = \frac{n_k}{n}$, for $k = 0, 1, \cdots, L-1$, where $p(r_k)$ is an estimate of the probability of occurrence of gray level $r_k$.

Histograms are the basis for numerous spatial domain processing techniques. Histogram manipulation can be used effectively for image enhancement. The information contained in the histogram not only provides useful statistics, but is useful in other image processing applications, such as image compression and/or segmentation. Histograms can be calculated with software and lend themselves to economic hardware implementations. They are a popular tool for real-time image processing.

We consider four basic image types: dark, light, low-contrast, high-contrast plus their corresponding histograms in Figures 2.24-2.27. The right side of the figure shows the histograms corresponding to each of these images. The horizontal axis of each histogram plot corresponds to gray level values $r_k$. The vertical axis corresponds to values of $h(r_k) = n_k$ or $p(r_k) = \frac{n_k}{n}$ if the values are normalized. Thus, these histogram plots are plots of $h(r_k) = n_k$ versus $r_k$ or $p(r_k) = \frac{n_k}{n}$ versus $r_k$.

For the dark image, the components of the histogram are concentrated on the lower part of the gray scale. The components of the histogram of the bright image are biased toward the higher part of the gray scale. An image with a low contrast has a histogram that will be narrow and centered toward the middle of the gray scale. The components of the histogram in the high-contrast image cover a broad range of the gray scale and its distribution of pixels is close to uniform, with very few vertical lines being higher than the others. We may conclude that an image whose pixels tend to occupy the entire range of gray levels and, in addition, tend to be uniformly distributed, will appear with high contrast and exhibit a large number of
Figure 2.24 A dark image and its histogram

Figure 2.25 A bright image and its histogram
Figure 2.26 A low-contrast image and its histogram

Figure 2.27 A high-contrast image and its histogram
gray tones. The effect will be an image that shows great gray-level detail and has a high dynamic range. It is also possible to build a transformation function that can produce this effect, based on the histogram of the input image.

### 2.3.1 Histogram Equalization

First, we will consider continuous functions. Let the variable \( r \) represent the gray levels of the image to be enhanced. Initially, we assume that \( r \) has been normalized to the interval \([0, 1]\), with \( r = 0 \) for black and \( r = 1 \) for white. Later, we will consider a discrete formulation with pixel values in the interval \([0, L-1]\).

For any \( r \) that meets these conditions, we consider transformations of the form

\[
s = T(r) \quad 0 \leq r \leq 1
\]

that yield a level \( s \) for every pixel value \( r \) in the original image. We assume that the transformation function \( T(r) \) satisfies the following conditions:

1. \( T(r) \) is single-valued and monotonically increasing in the interval \( 0 \leq r \leq 1 \); and
2. \( 0 \leq T(r) \leq 1 \) for \( 0 \leq r \leq 1 \).

The condition that \( T(r) \) be single-valued is necessary to guarantee that the inverse transformation will exist, and the monotonicity preserves the increasing order from black to white in the output image. A transformation function that is not monotonically increasing could invert at least a section of the intensity range, and would produce some inverted gray levels in the output image. In this case, we do not want this to happen. Condition (2) above guarantees that the output gray levels will be in the same range as the input levels. Figure 2.28 shows an example of a function that satisfies these two conditions. The inverse transformation is

\[
r = T^{-1}(s) \quad 0 \leq s \leq 1
\]
The gray levels in an image can be viewed as random variables in the interval $[0, 1]$. The probability density function (PDF) describes a random variable. Let $p_r(r)$ and $p_s(s)$ denote the probability density functions of random variables $r$ and $s$ respectively; the subscripts indicate that $p_r$ and $p_s$ are different functions. A result from probability theory states that, if $p_r(r)$ and $T(r)$ are known and $T^{-1}(s)$ satisfies condition (1), then the probability density function $p_s(s)$ for the transformed variable $s$ can be obtained using the formula:

$$p_s(s) = p_r(r) \left| \frac{dr}{ds} \right|$$

(2.3-3)

The probability density function of the transformed variable $s$ can be determined by the gray-level PDF of the input image and the chosen transformation function.
A transformation function considered important in image processing is:

\[ s = T(r) = \int_0^r p_s(w)dw \]  \hspace{1cm} (2.3-4)

where \( w \) is a variable for integration. The right hand side of (2.3-4) is the cumulative distribution function (CDF) of random variable \( r \). Probability density functions are always positive. It follows that this transformation function is single valued and monotonically increasing and therefore it satisfies condition (1). The integral of a probability density function for variables in the range \([0, 1]\) is also in the range \([0, 1]\), so condition (2) is satisfied.

Having the transformation function \( T(r) \), we can find \( p_s(s) \) from the (2.3-3). Using Leibniz’s rule, we know that the derivative of a definite integral with respect to its upper limit is the integrand evaluated at that limit:

\[ \frac{ds}{dr} = \frac{dT(r)}{dr} = \frac{d}{dr} \left[ \int_0^r p_s(w)dw \right] = p_s(r) \] \hspace{1cm} (2.3-5)

The result for \( \frac{dr}{ds} \) is substituted into (2.3-3) and since all probability values are positive, we obtain:

\[ p_s(s) = p_s(r) \frac{dr}{ds} = p_s(r) \frac{1}{p_s(r)} = 1 \quad 0 \leq s \leq 1. \] \hspace{1cm} (2.3-6)

We chose \( p_s(s) \) to be a probability density function; then \( p_s(s) \) is zero outside the interval \([0, 1]\) because its integral over all values of \( s \) must equal 1. Using the form of \( p_s(s) \) in (2.3-6), it follows that this is a uniform probability density function. The transformation function (2.3-4) produces a random variable \( s \) characterized by a uniform probability density.
function. \( T(r) \) depends on \( p_r(r) \) (2.3-4) and \( p_s(s) \) is always uniform (2.3-6) and independent of the form of \( p_r(r) \).

For discrete values we will use probabilities and summations instead of the probability density functions and integrals. The probability of occurrence of the gray level \( r_k \) in an image is approximated by

\[
p(r_k) = \frac{n_k}{n} \quad k = 0, 1, \ldots, L-1
\]  

(2.3-7)

where \( n \) is the total number of pixels in the image, \( n_k \) is the number of pixels having gray level \( r_k \), and \( L \) is the total possible gray levels in the image. For discrete values the function defined in (2.3-4) becomes:

\[
s_k = T(r_k) = \sum_{j=0}^{k} p(r_j) = \sum_{j=0}^{k} \frac{n_j}{n} \quad k = 0, 1, \ldots, L-1
\]  

(2.3-8)

A processed image is obtained by mapping each pixel with level \( r_k \) in the input image into a corresponding pixel with level \( s_k \) in the output image via (2.3-8). This transformation is called histogram equalization or histogram linearization.

Using (2.3-8) tends to spread the histogram of the input image so that the levels of the histogram-equalized image will span more of the gray scale automatically. Given an image, the implementation of (2.3-8) (based on information extracted from the image) is analogous to histogram equalization.

The inverse transformation from \( s \) to \( r \) is:

\[
r_k = T^{-1}(s_k) \quad k = 0, 1, 2, \ldots, L-1
\]  

(2.3-9)
The inverse transformation (2.3-9) satisfies the transformation conditions (1) and (2) only if all the levels, $r_k, k = 0, 1, 2, \cdots, L-1$, are present in the input image. The inverse transformation is used in a histogram-matching scheme.

Figures 2.29-2.32 show the result of performing histogram equalization on each of the four images in Figures 2.24-2.27 and the histograms of their equalized images. The first three Figures 2.29-2.31 show significant improvement. Histogram equalization did not produce a visual difference in the fourth image because its histogram spans the full spectrum of the gray scale.

![Figure 2.29](image)

Figure 2.29 (a) The dark image from Figure 2.24. (b) Result of histogram equalization. (c) The equalized histogram.

### 2.3.2 Histogram Matching

Histogram equalization is the transformation that produces an image with a uniform histogram. When automatic enhancement is desired, this is a good approach because the results are predictable and the method is simple to implement. There are other applications in
Figure 2.30 (a) The bright image from Figure 2.25. (b) Result of histogram equalization. (c) The equalized histogram.

Figure 2.31 (a) The low-contrast image from Figure 2.26. (b) Result of histogram equalization. (c) The equalized histogram.
Figure 2.32(a) The high-contrast image from Figure 2.27. (b) Result of histogram equalization. (c) The equalized histogram.

which image enhancement with a uniform histogram is not desirable. It is sometimes better to specify a histogram for an image. To obtain an image with a specified histogram one uses histogram matching or histogram specification.

Even though $r$ and $z$ are continuous gray levels and $p_r(r)$ and $p_z(z)$ are their corresponding continuous probability density functions, we will use $r$ and $z$ to be the gray levels of the input and output (processed) images, respectively. We determine $p_r(r)$ using the input image, and $p_z(z)$ will be the probability density function desired for the output image.

Assume $s$ is a random variable with the property

$$s = T(r) = \int_{0}^{r} p_r(w)dw$$

(2.3-10)
where \( w \) is a variable for integration. This is the continuous version of histogram equalization from (2.3-4). Suppose a random variable \( z \) with the property

\[
G(z) = \int_{w}^{z} p_{z}(t)dt = s
\]

is defined where \( t \) is a variable for integration. Then it follows that \( G(z) = T(r) \) and \( z \) must satisfy

\[
z = G^{-1}(s) = G^{-1}[T(r)]
\]

The transformation \( T(r) \) can be obtained from (2.3-10) after an estimate of \( p_{z}(r) \) is obtained from the input image. The transformation \( G(z) \) is obtained from (2.3-11) because \( p_{z}(z) \) is given.

If \( G^{-1} \) exists and satisfies the transform conditions (1) and (2) in the previous section, (2.3-10), (2.3-11), and (2.3-12) will produce an image with a specified probability density.

We use the following procedure:

1. Obtain the transformation \( T(r) \) with (2.3-10).
2. Use (2.3-11) to obtain the transformation \( G(z) \).
3. Compute the inverse transformation \( G^{-1} \).
4. Obtain the output image with (2.3-12) using all the pixels in the image.

The result produces an image whose gray levels, \( z \), have the specified probability density \( p_{z}(z) \).

In practice, it is seldom possible to obtain an analytical expression for both \( T(r) \) and \( G^{-1} \). This problem is simplified for discrete values. Similarly in histogram equalization, only an approximation to the desired histogram is achievable.
The discrete formulation of (2.3-10) is (2.3-8), which is:

\[
s_k = T(r_k) = \sum_{j=0}^{k} p_j(r_j) = \frac{\sum_{j=0}^{k} n_j}{n} \quad k = 0, 1, \ldots, L-1 \quad (2.3-13)
\]

where \(n\) is the total number of pixels in the image, \(n_j\) is the number of pixels with gray level \(r_j\), and \(L\) is the number of discrete gray levels. The discrete variant of (2.3-11) using the given histogram \(p_x(z_i)\), \(i = 0, 1, \ldots, L-1\) is:

\[
G(z_k) = \sum_{i=0}^{k} p(z_i) = s_k \quad k = 0, 1, \ldots, L-1 \quad (2.3-14)
\]

We are seeking values of \(z\) that satisfy (2.3-14). The discrete version of (2.3-12) is:

\[
z_k = G^{-1}[T(r_k)] \quad k = 0, 1, \ldots, L-1 \quad (2.3-15)
\]

or

\[
z_k = G^{-1}(s_k) \quad k = 0, 1, \ldots, L-1 \quad (2.3-16)
\]

Equations (2.3-13), (2.3-14), (2.3-15) and (2.3-16) are the foundation for histogram matching for digital images. (2.3-13) is a mapping from the original image into corresponding levels \(s_k\) based on the histogram of the original image. (2.3-14) computes the transformation \(G\) from the original histogram \(p_x(z_i)\). Finally, (2.3-15) or (2.3-16) produces the desired intensity levels of an image with this histogram.

Figure 2.33 (a) (Gonzales and Woods [2001], p.100) shows an image from the Mars noon, Phobos, taken by NASA’s Mars Global Surveyor. Figure 2.33 (b) shows the histogram of the image shown in Figure 2.33 (a). The image has large, dark areas, so the histogram is characterized by a large concentration of pixels on the lower part of the scale.

Figure 2.34 (a) shows the histogram equalization transformation ((2.3-8) or (2.3-13)) obtained using the histogram shown on Figure 2.33 (b). We notice that this transformation function increases rapidly from 0 to almost 190. This is due to the large concentration of
Figure 2.33 (a) Image of the Mars moon Phobos taken by NASA’s Mars Global Surveyor (b) Its histogram

Figure 2.34 (a) Histogram equalization transformation of Figure 2.33 (a). (b) Histogram equalized image for Figure 2.33. (c) Histogram of (b)
pixels near 0 in the histogram being used. The effect of this transformation on the levels in
the input image is to map this very narrow interval of pixels into the upper portion of the gray
scale. A large number of pixels in the input have levels in this interval, and their transformed
result will produce a light, washed-out appearance, see Figure 2.34 (b). The histogram of
Figure 2.34 (b) is shown in Figure 2.34 (c). All the gray levels are biased toward the upper
half of the gray scale.

The problem with this transformation was caused by a large pixel concentration in the
original image near 0. We need to modify the original image histogram so that the resulting
image is improved. Figure 2.35 (a) shows a manually specified transformation that preserves
the general shape of the original image histogram, but has a smoother transition of levels for
the lower (darker) region of the gray scale. The desired histogram was produced by dividing
the transformation in Figure 2.35 (a) into 256 equally spaced discrete values. The
transformation \( G(z) \) obtained from this histogram using (2.3-14) is labeled (1) in Figure 2.35
(b). The inverse \( G^{-1}(s) \) from (2.3-16) is labeled (2) in Figure 2.35 (b). If we apply
transformation (2) to the pixels of the histogram-equalized image in Figure 2.34 (b) we obtain
the enhanced image of Figure 2.35 (c). The histogram-specified image is an obvious
improvement of the result obtained by histogram equalization. A small change in the original
histogram produced a significant improvement in the appearance of the result. The histogram
of Figure 2.35 (c) is shown in Figure 2.35(d). Its lower values have been shifted toward the
higher (lighter) region of the gray scale.

### 2.4 Enhancement Using Arithmetic/Logic Operations

Arithmetic/logic operations are performed on a pixel-by-pixel basis between two or more
images (this excludes the NOT operator, performed on a single image). For example, the
subtraction of two images results in a new image whose pixel value for any \((x, y)\) is the
Figure 2.35 (a) Specified histogram. (b) Curve (1) is from Eq. (2.3-14), using the histogram in (a); curve (2) was obtained using Eq. (2.3-16). (c) Enhanced image using mappings from curve (2). (d) Histogram of (c).

difference between the pixels in the same location in the two images being differenced. The mechanics of implementing arithmetic/logic operations can be done sequentially, one pixel at a time, or in parallel, where all operations are performed simultaneously.

Logic operators can also operate on a pixel-by-pixel basis. The operators AND, OR and NOT are functionally complete, i.e. any other logic operator can be implemented by using these three operators). When applying logic operations to gray-scale images, pixel values are processed as binary strings. For example, the NOT operation on a black-8-bit pixel produces a white pixel (a string of eight 1’s). Thus, the NOT operator performs the same function as the negative transformation in (2.2-1). The AND and OR are used in masking;
i.e., for selecting subimages in an image. For AND and OR masks, light is a binary 1 and dark is a binary 0. Masking is sometimes called the region of interest (ROI) processing. In enhancement, masking is used to isolate an area for processing, i.e. to highlight an area and differentiate it from the remainder of the image. Logic operations are frequently used in conjunction with morphological operations.

Of the arithmetic operations, subtraction and addition are the most useful for image enhancement. By multiplying an image by a constant, the average gray level is increased by multiplying an image by a constant. Image multiplication is used in enhancement primarily as a masking operation.

### 2.4.1 Image Subtraction

The difference between two images $f(x, y)$ and $h(x, y)$ can be expressed as:

$$g(x, y) = f(x, y) - h(x, y)$$  \hspace{1cm} (2.4-1)

We compute the difference between all pairs of corresponding pixels from $f$ and $h$. The subtraction can be useful in the enhancement of differences between images. To illustrate this concept, the difference image between a magnetic resonance image of a human bone (Figure 2.36 provided by LSU Health Sciences Center) and its blurred version (Figure 2.37). These two images are nearly identical visually, with the exception of a reduction in the overall contrast in the image on Figure 2.37. The pixel-by-pixel difference between these two images is shown in Figure 2.38. The differences in pixel values are so small that the difference image appears nearly black when displayed with an 8-bit pixel. In order to bring out more detail, a histogram equalization is used (an appropriate power-law transformation would have done this also). The result is shown in Figure 2.39.

One of the most commercially successful and beneficial uses of image subtraction is medical imaging and called mask mode radiography. In this $h(x, y)$, the mask, is an X-ray
Figure 2.36 A magnetic resonance (MR) image of a bone

Figure 2.37 A blurred version of the image in Figure 2.36
Figure 2.38 Difference between images in Figures 2.36 and 2.37

Figure 2.39 Histogram-equalized difference image
image of a region of a patient’s body captured by an intensified TV camera (not by using the traditional X-ray film) located opposite the X-ray source. A contrast medium is injected into the patient’s bloodstream and a series of images of the same anatomical region is taken as \( h(x, y) \). This mask is subtracted from the series of images taken after injection of the contrast medium. The effect of subtracting the mask from each sample in the incoming stream of TV images is that the difference between \( f(x, y) \) and \( h(x, y) \) appear in the output image with enhanced detail. Images are captured at TV rates (30 images/second) ; that’s why the procedure produces a “movie” showing how contrast medium propagates through the various veins in the area being studied.

Figure 2.40 (a) (Gonzales and Woods [2001], p.111) presents an X-ray image of the top of a patient’s head prior to injection of an iodine medium into the bloodstream. The

![Image of X-ray image](image)

Figure 2.40 Enhancement by image subtraction. (a) Mask image. (b) An image (obtained after injection of a contrast medium into the bloodstream) with mask subtracted. The camera used to obtain this image was placed above the patient’s head, looking down. The bright spot in the lower one-third of the image is the core of the spinal column. Figure 2.40
(b) shows the difference between the mask (shown in Figure 2.40 (a)) and an image taken after the medium was introduced into the bloodstream. The bright arterial paths that carry the medium are enhanced in Figure 2.40 (b). These arteries appear bright because they are not subtracted out. The overall background is darker than that in Figure 2.40 (a) because the differences between areas that are similar yield low values, that appear dark after injection of a contrast medium into the bloodstream and with the mask subtracted shades of gray appear in the difference image. The spinal cord is bright in Figure 2.40 (a) but appears dark in Figure 2.40 (b) as a result of mask subtraction.

Before the next section, some comments regarding the implementation are necessary. On off-the-shelf PC monitors, images are displayed with 8 bits (even a 24-bit color image consists of three separate 8-bit channels). We expect image intensity values to be in the interval $[0, 255]$. The intensity values in a difference image ranges from a minimum of -255 to a maximum of 255, so we scale them to display this result. There are two principle scaling methods for a difference image. One is to add 255 to every pixel and then divide by 2; this does not guarantee that the values will span the entire 8-bit interval $[0, 255]$ but all the values will be within this range. This method is fast and easy to implement, but is has limitations; e.g., it does not use the full range of the display and the truncation from the division by 2 will cause a loss in accuracy. The other approach produces more accurate results and it spans the 8-bit range. For this, we determine the minimum difference and add the negative of the minimum difference to all pixels in the difference image. This creates a new image with only non-negative values after which, all the pixels in the image are scaled to span the interval $[0, 255]$ by multiplying each pixel intensity by the $255/Max$, where $Max$ is the maximum pixel intensity value in the modified difference image. This approach is more complex and difficult to implement.
An additional use of pixel intensity change detection via their subtraction is in image segmentation. This type of segmentation attempts to subdivide an image into similar regions based upon the criterion that the intensity “changes”. For instance, the problem of tracking moving vehicles in a sequence of images can be sped-up by subtraction which removes all stationary components in an image. All the moving elements are present in the difference-image.

### 2.4.2 Image Averaging

A noisy image \( g(x, y) \) can be approximated by the addition of noise \( \eta(x, y) \) to the original image \( f(x, y) \); that is,

\[
g(x, y) = f(x, y) + \eta(x, y) \tag{2.4.2}
\]

with the assumption that for every pair of coordinates \((x, y)\) the noise is uncorrelated and has zero average value. The objective of the procedure that follows is to reduce the noise by adding a set of noisy images \( \{g_i(x, y)\} \).

If the noise satisfies the previous approximation, it can be shown that if an image \( \bar{g}(x, y) \) is created by averaging a set of \( K \) different noisy images,

\[
\bar{g}(x, y) = \frac{1}{K} \sum_{i=1}^{K} g_i(x, y) \tag{2.4-3}
\]

then it follows that

\[
E[\bar{g}(x, y)] = f(x, y) \tag{2.4-4}
\]

and

\[
\sigma^2 \bar{g}(x, y) = \frac{1}{\sqrt{K}} \sigma_{\eta(x, y)} \tag{2.4-5}
\]
where \( E[g(x,y)] \) is the expected value of \( g(x,y) \), \( \sigma^2_{\pi(x,y)} \) and \( \sigma^2_{\eta(x,y)} \) are the variances of \( g(x,y) \) and \( \eta(x,y) \), all at coordinates \((x,y)\). The standard deviation at any point in the average image is

\[
\sigma_{g(x,y)} = \frac{1}{\sqrt{K}} \sigma_{\eta(x,y)} \tag{2.4-6}
\]

As \( K \) increases, the variability (noise) of the pixel values at each location \((x,y)\) decreases ((2.4-5) and (2.4-6)). \( \bar{g}(x,y) \) approaches \( f(x,y) \) as the number of noisy images increases in the averaging process (2.4-4). In practice, the images \( g_i(x,y) \) must be registered (aligned) to avoid the blurring and other artifacts in the output image.

An important application of image averaging is in astronomy, where imaging with low light is routine when sensor noise renders single images virtually useless. Figure 2.41 (a) (Gonzales and Woods [2001], p. 114) shows an image of a galaxy pair called NGC 3314, taken by NASA’s Hubble Space Telescope with a wide field planetary camera. NGC 3314 lies about 140 million light-years from Earth, in the southern-hemisphere constellation Hydra. The bright stars that form a pinwheel shape near the center of the galaxy have formed more recently from interstellar gas and dust. Figure 2.41 (b) shows the Figure 2.41 (a) image, corrupted by adding uncorrelated Gaussian noise with zero mean and a standard deviation of its 64 gray levels. This image is useless for practical purposes. Figures 2.14 (c) through (f) show the results of averaging 8, 16, 64, 128 images, respectively. The final result corresponds to \( K = 128 \) in (2.4-3) and is close to the original image in visual appearance.

Addition can be viewed as a discrete formulation of continuous integration. In astronomical observations, a process equivalent to the method described above is to use the
integrating capabilities of sensors for noise reduction by frequently observing the same scene over long periods of time. The net effect is similar to the procedure described.

Adding two or more 8-bit images requires scaling to display the result on an 8-bit display. The values in the sum of \( K \) 8-bit images can range from 0 to \( 255 \times K \). Scaling this back to 8 bits requires dividing the resultant sum by \( K \). Some accuracy will be lost in the process, but the display is limited to 8 bits.
2.5 Spatial Filtering

Some operations work with the pixels separated into neighborhoods or subimages with the same dimensions as the neighborhoods. Each subimage is formed by a filter, mask, kernel, template, or window. The values in a subimage are called coefficients, rather than pixels.

The concept of filtering has its roots in Fourier transforms in signal processing in the frequency domain. The term spatial filtering is used to denote direct filtering on the image pixels. Figure 2.42 illustrates the core idea of spatial filtering. Spatial filtering consists of moving the filter mask from point-to-point in an image. At each point \((x, y)\), the response of the filter is calculated using a predefined relationship. For linear spatial filtering, the response is a sum of products of the filter coefficients and the corresponding image pixels in the area spanned by the filter’s mask. For the \(3 \times 3\) mask in Figure 2.42, the response \(R\) of linear filtering at point \((x, y)\) is:

\[
R = w(-1,-1)f(x-1, y-1) + w(-1,0)f(x-1, y) + \cdots + w(0,0)f(x, y) + \cdots + w(1,0)f(x+1, y) + w(1,1)f(x+1, y+1) \quad (2.5-1)
\]

This is the sum of products of the mask coefficients with the corresponding pixels under the mask. In particular, the coefficient \(w(0,0)\) uses the image value \(f(x, y)\), i.e. the mask is centered at \((x, y)\) when the computation of the sum of products is done. For \(m \times n\) mask, we assume that \(m = 2a + 1\) and \(n = 2b + 1\): \(a\) and \(b\) are nonnegative integers. We only consider masks of odd integer sizes, with the smallest being \(3 \times 3\) (the \(1 \times 1\) mask is excluded).

The linear filtering of a \(M \times N\) image \(f\) with a \(m \times n\) filter mask is obtained by:

\[
g(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x+s, y+t) \quad (2.5-2)
\]

where \(a = \frac{m-1}{2}\) and \(b = \frac{n-1}{2}\). To generate a completely filtered image (2.5-2) must be
Figure 2.42 The spatial filtering process. The magnified drawing shows a $3 \times 3$ mask and the image section directly below it.

applied for $x = 0, 1, \ldots, M - 1$ and $y = 0, 1, \ldots, N - 1$. This assures that the mask processes all pixels in the image.

Linear filtering is similar to the frequency domain concept of convolution; because linear spatial filtering is often referred to as “convolving a mask with an image.” Filter masks are sometimes called convolution masks or convolution kernels.
When we are interested in only the response, $R$, of a $m \times n$ mask at any point $(x, y)$ the notation can be simplified by using the following expression:

$$R = w_1z_1 + w_2z_2 + \cdots + w_mz_m$$

where the $w$’s are the mask coefficients, the $z$’s are the values of the image gray levels corresponding to those coefficients, and $mn$ is the total number of coefficients in the mask.

For the $3 \times 3$ mask shown in Figure 2.43 the response at any point $(x, y)$ in the image is:

$$R = w_1z_1 + w_2z_2 + \cdots + w_9z_9$$

where $w_1, w_2, \ldots, w_9$ are the coefficients of the mask and $z_1, z_2, \ldots, z_9$ are the values of the image gray levels corresponding to those coefficients. The total number of coefficients in the mask is $mn = 9$.

Figure 2.43 A general representation of a general $3 \times 3$ spatial filter mask

This formula is encountered frequently in image processing literature.

Nonlinear spatial filters also operate on neighborhoods, and the operation of a sliding a mask works similarly. In general, filtering is conditionally based on the values of the pixels in the neighborhood under consideration, and does not explicitly use coefficients in the sum-of-products mode in (2.5-2) and (2.5-3). For example, noise reduction can be effectively achieved with a nonlinear filter that computes the median gray-level value in the neighborhood the filter is located. Computation of the median is a nonlinear operation.
With the use of spatial filtering, we have to address what to do when the center of the filter approaches a border of the image. Let us consider a square mask, $n \times n$. At least one edge of this mask will coincide with the border of the image when the center of the mask is at distance $(n - 1)/2$ pixels from the border of the image. If the center of the mask moves any closer to the border, one or more rows or columns of the mask will be outside the image. There are several methods to resolve this situation. One is to limit the position of the center of the mask to be a distance of no less than $(n - 1)/2$ pixels from the border. The filtered image will be smaller than the original, but all pixels in the result will be processed with the full mask. If the same size is required for the result, then another approach typically used is to filter all pixels only with the section of the mask fully contained in the image. Using this approach, bands of pixels near the border will be processed with a partial filter mask. Other approaches include “padding” the image with rows and columns of 0’s (or other constant gray level), or padding by replicating rows and columns. The padding is stripped at the end of the process. This keeps the size of the filtered image the same as the original, but the values that require padding will have an effect near the edges and be more prevalent as the mask size increases. The only way to get a perfectly filtered mask is with a smaller resultant image, i.e. limiting the location of the center of the filter mask to a distance no less than $(n - 1)/2$ pixels from the border of the original image.

2.6 Smoothing Spatial Filters

Smoothing filters are used for blurring and noise reduction. In preprocessing blurring removes small details from an image prior to an object extraction, and to bridge small gaps in lines or curves. Noise reduction can be attained with linear or non-linear smoothing filters.
2.6.1 Linear Smoothing Filters

The output (response) with a linear spatial smoothing filter is the average of the pixels contained in the neighborhood of its mask. These filters are also called averaging or lowpass filters.

If the value of every pixel in an image is replaced by the average of its neighborhood defined by the smoothing filter mask, the resulting image will have the “sharp” transitions in its gray levels reduced. Random noise in medical images typically presents itself as sharp transitions in gray levels, thus an obvious smoothing application would produce noise reduction. However, the edges in a medical image are characterized by sharp transitions in gray levels, i.e. averaging filters have the effect of blurring edges. Another application is the smoothing of “false contours” that result from an inadequate number of gray levels. A major use of averaging filters is to reduce the “irrelevant” detail in an image (“irrelevant” means pixel regions that are small with respect to the filter mask size).

Figure 2.44 shows two 3×3 smoothing filters. Use of the first filter produces the standard average of the pixels under the mask. Substituting the coefficients of the mask into (2.5-4) yields the average of the gray levels of the pixels in the 3×3 neighborhood defined by: \( R = \frac{1}{9} \sum_{i,j} z_{ij} \). The coefficients of the filter of are all 1’s, instead of 1/9, because it is computationally more efficient to use integer coefficients. A \( m \times n \) mask would have a normalizing constant equal to \( 1/mn \). A spatial averaging filter with equal coefficients is called a box filter.

The second mask shown in Figure 2.44 yields a weighted average, because pixels are multiplied by different coefficients, giving more weight to some than others. In the mask shown in Figure 2.44 (b), the pixel in the center of the mask is multiplied by a higher value
Figure 2.44 Two $3 \times 3$ smoothing (averaging) filter masks. The fractional multiplier for each mask is equal to the inverse sum of the values of its coefficients.

than any other, so this pixel has more weight when we compute the average. The other pixels are inversely weighted as a function of their distance from the center of the mask. The diagonal terms are further from the center than the orthogonal neighbors (by a factor of $\sqrt{2}$), and, thus, are weighed less than the neighbors closer to the center pixel. Weighing the center highest and decreasing the weight of the coefficients as a function of distance from the mask’s center reduces blurring in the smoothing process. The values of these weights are not unique. The sum of the coefficients in the mask in Figure 2.44 (b) is 16, an appropriate value for computer implementation because it is an integer power of 2.

The general implementation for filtering an $M \times N$ image with a weighted averaging filter $m \times n$ ($m$ and $n$ odd) is given by:

$$g(x, y) = \frac{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t)f(x+s, y+t)}{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t)}$$  \hspace{1cm} (2.6-1)

The parameters in (2.6-1) are defined in (2.5-1). The complete filtered image is obtained by applying (2.6-1) for $x = 0, 1, \cdots, M - 1$ and $y = 0, 1, \cdots, N - 1$. The denominator in
(2.6-1) is the sum of the mask coefficients, or a fraction computed only once. This scale factor is applied to all the pixels of the output image after the filtering process is completed.

The effects of smoothing as a function of filter size are illustrated in Figures 2.45-2.50 (original image provided by LSU Health Sciences Center). These figures show an original image and the corresponding smoothed results obtained using square averaging filters for \( n = 3, 5, 9, 15, 35 \) pixels, respectively. For \( n = 3 \), note a general slight blurring is present throughout the resultant image. The details are approximately the same size as the filter mask and are affected more. The result for \( n = 5 \) is somewhat similar, with a slight increase in blurring. For \( n = 9 \) there is considerably more blurring. The small gray parts are not as distinct from the background as in the previous three images, illustrating the blending effect that blurring has on image-objects whose gray level is close to that of its neighboring pixels. The results for \( n = 15 \) and \( n = 35 \) are extreme with respect to the size of different parts of this image. The excessive blurring produced by these masks is used to eliminate small objects in an image. The components of the bone image have been blended into the background of the image in Figure 2.50.

An important application of spatial averaging is to blur an image to obtain a gross representation of the objects of interest. The intensity of smaller objects blends with the background and larger objects become "bloblike" and easier to detect. The size of the mask determines the relative size of the objects that will be blended with the background.

### 2.6.2 Order-Statistics Filters

Order-statistics filters are nonlinear spatial filters whose response is based on ordering (ranking) the pixels in the image area encompassed by the filter and replacing the value of the center pixel with the value determined by the ranking. A well-known example is the median
Figure 2.45 A magnetic resonance (MR) image of a bone.

Figure 2.46 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 3$
Figure 2.47 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 5$

Figure 2.48 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 9$
Figure 2.49 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 15$

Figure 2.50 Result of smoothing image in Figure 2.45 with square averaging filter masks with dimension $n = 35$
filter, which replaces the value of each pixel by the median of the gray levels in its neighborhood (the original value of the pixel is included in the computation of the median). Median filters are popular for certain types of random noise because they provide excellent noise-reduction capabilities with less blurring than the linear smoothing filter of similar size. Median filters are effective on impulse noise, also called salt-and-pepper noise because of its appearance as white and black dots superimposed on a gray-scale image.

The median, $\xi$, of a set of values is such that half of the values in the set are less than or equal to $\xi$, and half are greater than or equal to $\xi$. To perform median filtering at a point in an image, first the values of that pixel and its neighbors are sorted, then their median is determined, and this value is assigned to the pixel. For example, in a $3\times3$ neighborhood the median is the 5th largest value, in a $5\times5$ neighborhood the 13th largest value, and so on. When several values in a neighborhood are the same, all equal values are grouped. For example, suppose a $3\times3$ neighborhood has values $(20, 30, 30, 30, 25, 30, 30, 30, 20)$. These values are sorted as $(20, 25, 30, 30, 30, 30, 30, 30, 20)$, which produces a median of 30. The principal function of median filters is to force points with distinct gray levels to have values closer to their neighbors. Isolated clusters of pixels (dark or light) with respect to their neighbors whose area is less than $n^2/2$ (one-half of the filter area), are eliminated by an $n\times n$ median filter where “eliminated” is understood to mean forced to the median intensity of their neighbors. Larger clusters are less affected.

The median filter is a most useful order-statistics filter in image processing, but is not the only one. The median is the 50th percentile of a ranked set of numbers, but ranking leads to other possibilities. For example, using the 100th percentile yields the max filter, which helps to determine the brightest points in an image. The response of a $3\times3$ max filter is given
by \( R = \max \{ z_k \mid k = 1, 2, \ldots, 9 \} \). The 0\(^{th}\) percentile is the \textit{min filter}, used for the opposite purpose.

Figures 2.51 shows a version of the Lena image corrupted by salt-and-pepper noise.

![Figure 2.51 Lena image corrupted by salt-and-pepper noise](image)

To illustrate the superiority of median filtering over average filtering for this degradation, we show in Figure 2.52 the result of processing the noisy image with a 3\(\times\)3 neighborhood averaging mask, and in Figure 2.53 the result of applying a 3\(\times\)3 median filter. The image processed with an averaging filter has less visible noise, but it is significant blurred. The superiority of the median over average filtering in this case is apparent. In general, median filtering is better suited than averaging for the removal of additive salt-and-pepper noise.
Figure 2.52 Noise reduction with a $3 \times 3$ averaging mask

Figure 2.53 Noise reduction with a $3 \times 3$ median filter
2.7. Sharpening Spatial Filters

The principal objective of sharpening is to highlight fine detail in an image or to enhance detail that has been blurred, either in error or as a natural effect of acquisition. Sharpening is used in applications that range from electronic printing and medical imaging to industrial inspection and autonomous guidance systems.

Pixel averaging (Section 2.6) is analogous to integration; therefore, it is logical to conclude that sharpening can be accomplished by spatial differentiation. There are various ways to implement operators for digital differentiation. The response of a derivative operator is proportional to the degree of discontinuity in the image at the point the operator is applied. Thus, image differentiation can enhance edges and other discontinuities (such as noise) and deemphasize areas that have slowly varying gray-levels.

2.7.1 Fundamental Properties of Derivatives

To present detail sharpening filters based on first-order and second-order derivatives, we first consider some fundamental properties of these derivatives in digital context. We are interested in the behavior of these derivatives in the areas with a constant gray level (flat segments), at the onset and end of discontinuities (step and ramp discontinuities), and along gray-level ramps. Also, the behavior of derivatives during transitions into and out of these image features are also of interest.

The derivatives of a digital function are defined in terms of numerical differences. We will use a definition of the first derivatives that satisfies the following (Gonzales and Woods [2001]):

(1) are zero in flat segments (areas of constant gray-level values);
(2) are nonzero at the onset of a gray-level step or ramp;
(3) are nonzero along the ramps.

An acceptable definition of the second derivatives should satisfy:

(1) are zero in flat areas;

(2) are nonzero at the onset and end of a gray-level step or ramp;

(3) are nonzero along ramps of constant slope.

The definition of the first-order derivative of a one-dimensional function \( f(x) \) is:

\[
\frac{df}{dx} \approx \frac{f(x+1) - f(x-1)}{2}
\]

Similarly, the second-order derivative is defined as the difference

\[
\frac{d^2f}{dx^2} \approx \frac{f(x+1) + f(x-1) - 2f(x)}{\Delta x}
\]

where \( \Delta x = 1 \). These two definitions satisfy the conditions stated previously with respect to derivatives of first and second order.

The following properties can be proven (Gonzales and Woods [2001]):

(1) First-order derivatives generally produce thicker edges in an image.

(2) Second-order derivatives have a stronger response to fine detail, such as thin lines and isolated points.

(3) First-order derivatives generally have a stronger response to a gray-level step.

(4) Second-order derivatives produce a “double-edge” effect at step changes in gray level. For similar changes in gray-level values of an image, the response of second-order derivatives is stronger to a line than to a step, and to a point than to a line.

In most applications, the second derivative is better suited than the first derivative for image enhancement because of its sensitivity to details. We will focus our attention on the second derivative for enhancement.
2.7.2. The Second Derivative Used in Enhancement

Two-dimensional, second-order derivatives are used in image enhancement. A discrete second-order derivative is defined to formulate a filter mask. This formulation will be the base for constructing a filter mask. Isotropic filters are desirable because their response is independent of the direction of the discontinuities in the image. Isotropic filters are rotation invariant, i.e., if we rotate an image and apply this filter the same result is obtained as when the filter is applied first and then rotated.

2.7.2.1 Method Description

It can be proven (see Rosenfeld and Kak [1982]) that the simplest isotropic derivative operator is the Laplacian. For a function (image) \( f(x, y) \) of two variables, the Laplacian is defined as:

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

Derivatives of any order are linear operations, so the Laplacian is a linear operator.

We want this equation in its discrete form. The definition we are using satisfies the properties in Section 2.7.1. The notation for the partial second-order derivative in the

\[ x \]-direction:

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

and similarly in the \( y \)-direction:

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

These two components are summed to obtain the two-dimensional Laplacian:
Equation (2.7-4) can be implemented using the mask shown in Figure 2.54 (a). The implementation details are given in (2.5-1) and illustrated in Section 2.6-1 for linear smoothing filters.

Figure 2.54 (a) Filter mask used to implement the Laplacian, as defined in (2.7-4). (b) Mask used to implement an extension that includes diagonal neighbors. (c) and (d) Other implementations of the Laplacian (sign changed)

The diagonal directions can be included in the definition for the Laplacian by adding two terms to (2.7-4), one for each of the two diagonal directions. The form of each new term is the same as either (2.7-2) or (2.7-3), but the coordinates are along diagonals. Each diagonal term requires a $-2f(x,y)$ term and the total contribution is $-8f(x,y)$. This mask is shown in Figure 2.54(b). This mask yields isotropic results for rotations of $45^\circ$. The two masks shown in Figure 2.54 are used frequently. They are based on a sign change in the definition of the Laplacian used in (2.7-1).
The Laplacian operator’s use highlights gray-level discontinuities in an image and decreases the emphasis of the regions with slowly varying gray levels. This will create an image with grayish edges and other discontinuities, superimposed on a dark, featureless background. Background features can be “recovered” while preserving the sharpening effect of the Laplacian by adding the original and Laplacian images. If the definition used has a negative center coefficient, then we subtract the Laplacian to obtain a sharpened result. In summary, we use the Laplacian for image enhancement as follows:

\[
g(x, y) = \begin{cases} 
  f(x, y) - \nabla^2 f(x, y) & \text{if Laplacian center coefficient is negative} \\
  f(x, y) + \nabla^2 f(x, y) & \text{if Laplacian center coefficient is positive}
\end{cases}
\] (2.7-5)

Figure 2.55 shows a magnetic resonance image of a human bone. Figure 2.56 shows the result of filtering this image with the Laplacian mask in Figure 2.54 (b). The Laplacian contains both positive and negative values, so the scaling method described in Section 2.4-1 is used.

The image shown in Figure 2.57 is scaled in the manner described for display. Notice the dominant features of the image are edges and sharp gray-level discontinuities of various gray-levels. The background, previously black, is gray due to the scaling. Figure 2.58 shows the result obtained using (2.7-5). The detail in this image is clearer and sharper than in the original image. Adding the image to the Laplacian restores the overall gray level variations in the image and increases the contrast at gray-level discontinuities. The result is an image with its small details enhanced and the background tonality was perfectly preserved. Such results have made Laplacian-based enhancement a fundamental tool for sharpening digital images.

2.7.2.2. Simplified Version

The previous Section 2.7.2.1 implemented (2.7-5) by first computing the Laplacian-filtered image and subtracting it from the original image. In practice, (2.7-5) is usually implemented
Figure 2.55 A blurred image of a bone

Figure 2.56 Laplacian filtered image
Figure 2.57 Laplacian image scaled for display purposes

Figure 2.58 Image enhanced using Eq. (2.7-5)
with one pass of a single mask. The coefficients in the single mask are obtained by substituting (2.7-4) for \( \nabla^2 f(x, y) \) in the first line of (2.7-5):

\[
g(x, y) = f(x, y) - [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] + 4f(x, y)
\]

\[
= 5f(x, y) - [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)]
\]

Equation (2.7-6) can be implemented using the mask in Figure 2.59 (a). The mask in Figure 2.59 (b) would be used if the diagonal neighbors were included to compute the Laplacian. Identical masks would have been obtained if we had substituted the negative of (2.7-4) into the second line of (2.7-5).

![Composite Laplacian mask](image)

Figure 2.59 (a) Composite Laplacian mask. (b) A second composite Laplacian mask.

Figure 2.60 is a blurred version of a magnetic resonance image of a human bone. The results achieved with a mask that contains the diagonal terms are slightly sharper than those obtained with the basic mask of Figure 2.59 (a). This property is illustrated by the Laplacian-filtered images shown in Figures 2.61 and 2.62, obtained with the masks in Figures 2.59 (a) and (b) and the image in Figure 2.60. If we compare the filtered images with the original image shown in Figure 2.60, both masks produced effective enhancement, but the result using the mask in Figure 2.59 (b) is visibly sharper.
2.7.2.3 Unsharp Masking and High-Boost Filtering

A process used for many years to sharpen images consists of subtracting a blurred version of an image from the image itself. This process, called *unsharp masking*, is expressed as

\[ f_s(x, y) = f(x, y) - \tilde{f}(x, y) \]  

(2.7-7)

where \( f_s(x, y) \) is the sharpened image obtained by unsharp masking, and \( \tilde{f}(x, y) \) is a blurred version of \( f(x, y) \). The origin of unsharp masking is in dark-room photography, where it consists of clamping together a blurred negative to a corresponding positive film and then using this combination to produce a sharper image.

A further generalization of unsharp masking is called *high-boost filtering*. A high-boost filtered image, \( f_{hb} \), is defined at any point \((x, y)\) as

\[ f_{hb}(x, y) = Af(x, y) - \tilde{f}(x, y) \]  

(2.7-8)
Figure 2.61 Result of filtering with the mask in Figure 2.59 (a)

Figure 2.62 Result of filtering with the mask in Figure 2.59 (b)
where $A \geq 1$ and, as before, $\tilde{f}$ is a blurred version of $f$. This equation may be written as:

$$f_{hb}(x, y) = (A - 1)f(x, y) + f(x, y) - \tilde{f}(x, y) \quad (2.7-9)$$

By using Eq. (2.7-7), we obtain

$$f_{hb}(x, y) = (A - 1)f(x, y) + f_s(x, y) \quad (2.7-10)$$

and this is the expression used to compute a high-boost-filtered image.

Equation (2.7-10) can be applied in general and does not state explicitly how the sharp image was obtained. If we use the Laplacian, then we know the sharpened image can be determined using Eq. (2.7-5). In this case, Eq. (2.7-10) becomes:

$$f_{hb} = \begin{cases} 
Af(x, y) - \nabla^2 f(x, y) & \text{if Laplacian center coefficient is negative} \\
Af(x, y) + \nabla^2 f(x, y) & \text{if Laplacian center coefficient is positive}
\end{cases} \quad (2.7-11)$$

High-boost filtering can be implemented with one pass using either mask shown in Figure 2.63. Notice that if $A = 1$, high-boost filtering becomes Laplacian sharpening. As $A$ increases,

<table>
<thead>
<tr>
<th>0</th>
<th>-1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>$A + 4$</td>
<td>-1</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>-1</th>
<th>-1</th>
<th>-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>$A + 8$</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

Figure 2.63 The high-boost filtering method can be implemented with either mask.
the contribution of the sharpening process becomes less and less important. If $A$ is large enough, the high-boost image will be approximately equal to the original image multiplied by a constant.

We may use boost filtering when an image is darker than desired. By changing the boost coefficient, it is possible to get an overall increase in average gray level of the image, resulting in a brighter image. Figures 2.64-2.67 show such an application. Figure 2.64 is a magnetic resonance image of a skull (image provided by LSU Health Sciences Center). Figure 2.65 shows the Laplacian using the mask in Figure 2.63(b), with $A = 0$. Figure 2.66 was obtained using the mask in Figure 2.63 (b) with $A = 1$. As expected, the image is sharper, but dark like the original. Figure 2.67 shows the result of using $A = 1.7$. This is better because

![Figure 2.64 A dark magnetic resonance image of a human head](image_url)

the average gray level is increased. The image in Figure 2.67 looks lighter and more suitable for viewing.
Figure 2.65 Laplacian of image in Figure 2.64, computed with the mask in Figure 2.63(b) using $A = 0$

Figure 2.66 Laplacian enhanced image using the mask in Figure 2.63 (b) with $A = 1$
2.7.3 First Derivatives for Enhancement

First derivatives in image enhancement are implemented using the magnitude of the gradient. For a function \( f(x, y) \), the gradient of \( f \) at coordinates \((x, y)\) is defined as the two-dimensional column vector

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

(2.7-12)

The magnitude of this vector is:

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

(2.7-13)
The components of the gradient vector are linear operators, but the magnitude of the gradient vector is not a linear operator. The partial derivatives in (2.7-12) are not rotation invariant (isotropic), but the magnitude of the gradient vector is. The magnitude of the gradient vector is often referred to as the gradient. We will use this term and explicitly refer to the vector or its magnitude when the meaning is not clear.

Implementing (2.7-13) over the entire image is not a simple computational task; it is common in practice to approximate the magnitude of the gradient as:

\[ \nabla f = |G_x| + |G_y| \]  

(2.7-14)

This equation is simple to compute and preserves relative changes in the gray level, but is not in general isotropic. The isotropic properties of the digital gradient are kept only for a number of rotational increments depending on the masks used to approximate the derivatives. The most popular masks for gradient approximation give the same result only for vertical and horizontal edges; thus, the isotropic properties of the gradient are only preserved for multiples of 90°.

We will use the previously presented approximations and construct the corresponding filter masks. As specified in Section 2.7-1, the simplest approximation of a first order derivative that satisfies conditions (1), (2) and (3) in Section 2.7-1 are

\[ G_x = f(x+1, y) - f(x, y) \] and \[ G_y = f(x, y+1) - f(x, y). \] Other definitions proposed (Roberts[1965]) in the early stages of development of image processing use cross differences:

\[ G_x = f(x+1, y+1) - f(x, y) \] and \[ G_y = f(x, y+1) - f(x, y+1). \]  

(2.7-15)

If we choose (2.7-13) the gradient is:

\[ \nabla f \approx \left[ (f(x+1, y+1) - f(x, y))^2 + (f(x+1, y) - f(x, y+1))^2 \right]^{1/2} \]  

(2.7-16)

If we use absolute values, the following approximation is obtained:
This can be implemented with the two masks shown in Figures 2.68 (a) and (b). These masks are called *Roberts’ cross-gradient operators*.

![Roberts cross-gradient operators](image)

Masks of even size are not convenient for implementation. We need filter masks being at least $3 \times 3$. An approximation that uses absolute values, at point $f(x, y)$, and a $3 \times 3$ mask, is given by:

$$\nabla f \approx |(f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1)) - (f(x-1, y-1) + 2f(x-1, y) + f(x-1, y+1))| + |(f(x-1, y) + 2f(x, y+1) + f(x+1, y+1)) - (f(x-1, y-1) + 2f(x, y-1) + f(x+1, y-1))|$$

(2.7-18)

The difference between the third and first rows of the $3 \times 3$ image region centered on $f(x, y)$ approximates the derivative in the $x$-direction, and the difference between the third and first columns approximates the derivative in the $y$-direction. The masks shown in Figures 2.69 (a) and (b) are called *Sobel operators* and can be used to implement (2.7-18) via (2.5-1). A value of 2 is used to achieve smoothing by assigning more importance to the center point. Notice that the coefficients in the masks in Figures 2.68 and 2.69 sum to 0. This proves they have a response of 0 on a constant gray level area.

Figure 2.70 shows a medical image of a knee (image provided by LSU Health Sciences Center). Figure 2.71 presents the gradient obtained using (2.7-14) with the two Sobel masks in Figure 2.69. The edges are quite visible in this image.
Figure 2.69 Sobel operators

Figure 2.70 A medical image of a knee
The gradient is frequently used in industrial inspections, either as an aid to humans when required to detect defects or as a preprocessing step in automated inspections. Finally, let’s consider an example of the use of a gradient to enhance the appearance of defects and eliminate slowly changing background features. In this example, the enhancement used as a preprocessing step for an automated inspection.

Figure 2.72 (a) (Gonzales and Woods [2001], p. 137) shows a contact lens, illuminated by lighting designed to highlight its imperfections. We can see two edge defects in the lens boundary at 4 and 5 o’clock. Figure 2.72(b) shows the gradient of the lens image using (2.7-14) with the Sobel masks, Figures 2.69 (a) and (b). The edge defects are also visible in this image, but have the advantage of constant or slowly varying shades of gray being eliminated. This considerably simplifies the computational task required for automated inspection. We can also see that the gradient highlighted small specs that are not visible in the
original gray-scale image. The capability to enhance small discontinuities in a flat gray field is an important feature of the gradient.

Figure 2.72 (a) Optical image of contact lens (with defects on the boundary at 4 and 5 o’clock). (b) Sobel gradient.
Chapter 3

Image Enhancement in the Frequency Domain

The previous chapter presented spatial techniques for image enhancement. To have a thorough understanding of this area, one needs to understand the Fourier transform and how its frequency domain can be used for image processing. This will emphasize image characteristics and the mathematical tools used to represent them. This chapter develops and presents the Fourier transform and the frequency domain, and how they are used in image enhancement.

We begin the discussion with a brief outline of the origins of the Fourier transform and its impact on different branches of mathematics, science, and engineering. Next, an introduction to the Fourier transform, its frequency domain, and the reasons why these are useful for image enhancement will be presented.

3.1 Background

The French mathematician Jean Baptiste Joseph Fourier published the important results for which he is remembered in his 1822 book “La Theorie Analitique de la Chaleur”; this was translated into English “The Analytic Theory of Heat” (Freeman [1878]). Fourier found that any periodic function can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient (a Fourier series). It does not matter how complicated the function as long as it is periodic and meets some minor/known mathematical conditions; it can be represented by such a sum.

Even functions that are not periodic (but whose span is finite) can be expressed as the integral of sines and/or cosines multiplied by a weighing function. This formulation is the Fourier transform, and has use in many practical problems. A function, expressed by either a Fourier series or transform, can be completely reconstructed (recovered) via an inverse
process, with no loss of information. This is one of the most important characteristics of these representations because work in the “Fourier domain” can be transformed into the original domain of the function without losing any information.

Entire industries and academic disciplines have used Fourier’s ideas. The advent of digital computation and the “discovery” of a fast Fourier transform (FFT) algorithm revolutionized signal processing. These two core technologies produced for the first time practical processing and interpretation of signals of exceptional human and industrial importance, e.g. medical monitors, scanners, modern electronic communications, etc.

3.2. Introduction to the Fourier Transform and the Frequency Domain

This section is devoted to Fourier transform in one and two dimensions. The discrete formulation of the continuous transform will be discussed in detail.

3.2.1 The One-Dimensional Fourier Transform and Its Inverse

The Fourier transform, $F(u)$, of a single variable, continuous function, $f(x)$ is

$$F(u) = \int_{-\infty}^{\infty} f(x)e^{-i2\pi ux} dx$$

(3.2-1)

where $i = \sqrt{-1}$. Conversely, given $F(u)$, we can obtain $f(x)$ by the inverse Fourier transform

$$f(x) = \int_{-\infty}^{\infty} F(u)e^{i2\pi ux} du$$

(3.2-2)

These two equations comprise the Fourier transform pair. They indicate that a function can be recovered from its transform. These equations are extended to two variables, $u$ and $v$:

$$F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y)e^{-i2\pi(ux+vy)} dxdy$$

(3.2-3)

and similarly for the inverse transform,

$$f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v)e^{i2\pi(ux+vy)} dudv$$

(3.2-4)
We are interested in discrete functions because digital images are discrete. The Fourier transform of a discrete function of one variable, \( f(x), \ x = 0, 1, \ldots, M - 1 \) is given by the equation:

\[
F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x)e^{-i2\pi xu/M} \quad \text{for} \ u = 0, 1, \ldots, M - 1. \tag{3.2-5}
\]

This is the discrete Fourier transform (DFT). Similarly, given \( F(u) \), we can obtain the original function using the inverse DFT:

\[
f(x) = \sum_{u=0}^{M-1} F(u)e^{i2\pi xu/M} \quad \text{for} \ x = 0, 1, \ldots, M - 1 \tag{3.2-6}
\]

To compute \( F(u) \) in (3.2-5), start with \( u = 0 \) in the exponential term and sum for all values of \( x \); then with \( u = 1 \) in the exponential and repeat the summation over all values of \( x \); and repeat this process for all \( M \) values of \( u \) to obtain the Fourier transform. It takes approximately \( M^2 \) summations and multiplications to compute a discrete Fourier transform. Like \( f(x) \), the transform is a discrete quantity, and has the same number of components as \( f(x) \); similar computations are required for the inverse Fourier transform.

An important property of the discrete transform pair is that, unlike the continuous case, the discrete Fourier transform and its inverse always exist. This can be shown by substituting either (3.2-5) or (3.2-6) into the other and making use of the orthogonality of the exponential terms. Thus, for digital image processing the existence of either the discrete transform or its inverse is not an issue.

The concept of the frequency domain follows directly from Euler’s formula:

\[
e^{i\theta} = \cos \theta + i \sin \theta. \tag{3.2-7}
\]

Substituting this expression into (3.2-5), and using the property \( \cos(-\theta) = \cos \theta \) we obtain:

\[
F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x)[\cos 2\pi ux/M - i \sin 2\pi ux/M] \quad u = 0, 1, \ldots, M - 1. \tag{3.2-8}
\]
Each term of the Fourier transform is composed of the sum of all values of \( f(x) \). The values of \( f(x) \), in turn, are multiplied by sines and cosines of various frequencies. The domain (values of \( u \)) over which the values of \( F(u) \) range is called the frequency domain, because \( u \) determines the frequency of the components of the transform. Each of the \( M \) terms of \( F(u) \) is called a frequency component of the transform.

In general, we see from (3.2-5) or (3.2-8) that the components of the Fourier transform are complex. It is convenient to express \( F(u) \) in polar coordinates:

\[
F(u) = |F(u)|e^{-i\phi(u)}
\]

where

\[
|F(u)| = \left[R^2(u) + I^2(u)\right]^{1/2}
\]

is called the magnitude or spectrum of the Fourier transform and

\[
\phi(u) = \tan^{-1}\left(\frac{I(u)}{R(u)}\right)
\]

is called the phase angle or phase spectrum of the transform. In (3.2-10) and (3.2-11), \( R(u) \) and \( I(u) \) are the real and imaginary parts of \( F(u) \). In image enhancement we are concerned primarily with the properties of the spectrum. The power spectrum is defined as the square of the Fourier spectrum:

\[
P(u) = |F(u)|^2 = R^2(u) + I^2(u)
\]

The term spectral density is also used for the power spectrum.

Let us consider a simple one-dimensional example of the discrete Fourier transform. Figure 3.1(a) shows a function and Figure 3.1(b) shows its Fourier spectrum. Both \( f(x) \) and \( F(u) \) are discrete quantities, but the points in the plot are linked so that they can be easily viewed. In this example, we have
Figure 3.1 (a) A discrete function of M points, and (b) its Fourier spectrum. (c) A discrete function with twice the number of nonzero points and (d) its Fourier spectrum.
\[ f(x) = \begin{cases} 
0, & \text{if } x = 1, 2, \ldots, 16 \\
1, & \text{if } x = 9, 10, \ldots, 256 
\end{cases} \]

We notice that the spectrum is centered about \( u = 0 \). The next two figures depict the Fourier spectrum of the function:

\[ f(x) = \begin{cases} 
0, & \text{if } x = 1, 2, \ldots, 32 \\
1, & \text{if } x = 17, 18, \ldots, 256 
\end{cases} \]

We observe that (1) the height of the spectrum doubled as the area under the curve in the \( x \)-domain doubled and (2) the number of zeros in the spectrum in the same interval doubled as the length of the function doubled. This “reciprocal” nature of the Fourier transform pair is most useful when we interpret results of image processing in the frequency domain.

In (3.2-5), the function \( f(x) \) for \( x = 0, 1, \ldots, M-1 \) represents \( M \) samples from its continuous counterpart. These samples are not always integer values of \( x \) in the interval \([0, M-1]\), but they are equally spaced. This is represented by \( x_0 \) denoting the first (arbitrarily chosen) point in the sequence and the first value of the sampled function is \( f(x_0) \).

The next sample is a fixed interval unit \( \Delta x \), i.e. \( f(x_0 + \Delta x) \). The \( k^{th} \) sample gives \( f(x_0 + k\Delta x) \), and the final sample is \( f(x_0 + [M-1]\Delta x) \). In the discrete case, \( f(k) \) is shorthand notation for \( f(x_0 + k\Delta x) \) and \( f(x) \) is understood to mean

\[ f(x) = f(x_0 + x\Delta x). \]  

The variable \( u \) has a similar interpretation, but the sequence always starts at zero frequency.

Thus, the sequence for the values of \( u \) is \( 0, \Delta u, \ldots, [M-1]\Delta u \) and \( F(u) \) is understood to mean

\[ F(u) = F(u\Delta u), \quad \text{for } u = 0, 1, \ldots, M-1. \]  

This shorthand notation simplifies equations and is easier to follow.
In the given inverse relationship between a function and its transform, \( \Delta x \) and \( \Delta u \) are related by expression

\[
\Delta u = \frac{1}{M\Delta x}
\]  \hspace{1cm} (3.2-15)

### 3.2.2 The Two-Dimensional DFT and Its Inverse

The discrete Fourier transform of a function \( (M \times N) f(x,y) \) is given by

\[
F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-i2\pi (ux/M + vy/N)}
\]  \hspace{1cm} (3.2-16)

This expression must be computed for values of \( u = 0, 1, \cdots, M-1 \), and also for \( v = 0, 1, \cdots, N-1 \).

Similarly, given \( F(u,v) \), we obtain \( f(x,y) \) via the inverse Fourier transform

\[
f(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{i2\pi (ux/M + vy/N)}
\]  \hspace{1cm} (3.2-17)

for \( x = 0, 1, \cdots, M-1 \) and \( y = 0, 1, \cdots, N-1 \). (3.2-16) and (3.2-17) comprise the two-dimensional, discrete Fourier transform (DFT) pair. The variables \( u \) and \( v \) are transform (frequency) and \( x \) and \( y \) are spatial (image).

The Fourier spectrum, phase angle and power spectrum are defined:

\[
|F(u,v)| = \left[ R^2(u,v) + I^2(u,v) \right]^{1/2}
\]  \hspace{1cm} (3.2-18)

\[
\phi(u,v) = \tan^{-1} \left[ \frac{I(u,v)}{R(u,v)} \right]
\]  \hspace{1cm} (3.2-19)

and

\[
P(u,v) = |F(u,v)|^2 = R^2(u,v) + I^2(u,v)
\]  \hspace{1cm} (3.2-20)

where \( R(u,v) \) and \( I(u,v) \) are the real and imaginary parts of \( F(u,v) \), respectively.

The transform at \( (u,v) = (0,0) \) is, from (3.2-16),

\[
F(0,0) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)
\]  \hspace{1cm} (3.2-21)
which is the average of $f(x, y)$. If $f(x, y)$ is an image, the value of the Fourier transform at the origin is the average gray-level of the image. Because both frequencies are zero at the origin, $F(0,0)$ is sometimes called the DC component of the spectrum, where “DC” signifies a current with zero frequency from electrical engineering.

In the 2-D case, we have the following relationships between samples in the spatial and frequency domains:

$$
\Delta u = \frac{1}{M\Delta x} \quad (3.2-22)
$$

and

$$
\Delta v = \frac{1}{N\Delta y} \quad (3.2-23)
$$

Figure 3.2 (image provided by LSU Health Sciences Center) presents a 256×256 pixel medical image. The Fourier transform of Figure 3.2, its spectrum is displayed using the log transformation in (2.2-2) to enhance the detail on Figure 3.3.

Figure 3.2 A 256x256 pixel medical image (a bone) on a black background
3.2.3 Filtering in the Frequency Domain

The frequency domain is defined by values of the Fourier transform and its frequency variables \((u, v)\). We can attach “meaning” to the frequency domain, to relate it to image processing.

3.2.3.1 Basic Properties of the Frequency Domain

We observe in (3.2-16) that each term of \(F\) contains all values of \(f\), multiplied by the exponential terms. Thus, with the exception of trivial examples, it usually is impossible to make direct associations between specific components of an image and its transform.

However, we can make some statements related to the frequency components of the Fourier transform and spatial characteristics of an image. For instance, since frequency is directly related to rate of change, the frequencies in the Fourier transform can be associated with

![Fig. 3.3 Centered Fourier spectrum shown after log transformation (2.2-2)](image)
patterns of intensity variations in an image. We specified in the previous section that the frequency component \((u = v = 0)\) corresponds to the average gray-level of an image. As we move from the origin of the transform, the low frequencies correspond to the slowly varying components of an image. As we move further from the origin, the higher frequencies begin to correspond with faster and faster gray-level changes in the image. These are the edges of objects and other components of an image characterized by abrupt changes in gray-level, e.g. noise.

We will illustrate these ideas. The image shown in Figure 3.4 (Gonzales and Woods [2001], p.157) is a scanning electron microscope image of an integrated circuit, magnified approximately 2500 times. We note as main features: strong edges that run approximately at \(\pm 45^\circ\), and the two white oxide protrusions resulted from thermally induced failure. The Fourier spectrum in Figure 3.5 shows prominent components along the \(\pm 45^\circ\) directions that correspond to the edges. If we observe carefully the vertical axis, we see a vertical component off-axis slightly to the left. This component was caused by the edges of the oxide protrusions. We note the off-axis angle of the frequency component corresponding to the inclination off horizontal of the long white element, and the zeros in the vertical frequency

![Figure 3.4 Scanning electron microscope image of a damaged integrated circuit](image-url)
component element, and the zeros in the vertical frequency component, corresponding to the narrow vertical span of the oxide protrusions.

Figure 3.6 shows a simple image with three blocks (Brayer [2003], p.6), with prominent edges and Figure 3.7 shows its Fourier spectrum. We notice a bright line going to high frequencies perpendicular to the strong edges in the image. Anytime an image has a strong contrast, sharp edge the gray values must change very rapidly. It takes high frequency power to follow such an edge so there is usually such a line in its magnitude spectrum.

These examples are typical of the types of associations that can be made in general between the frequency and spatial domains. These types of associations, along with the relationships mentioned previously between frequency content and rate of change of gray levels in an image, can lead to useful enhancement results.
3.2.3.2 Basics of Filtering in the Frequency Domain

Filtering in the frequency domain consists of the following steps:

1. Compute $F$, the DFT of the image.

2. Multiply $F$ by a filter function $H$.
3. Compute the inverse DFT of the result in step 2.

4. Obtain the real part of the result in step 3.

$H$ is called a filter (the term filter transfer function is also used) because it suppresses certain frequencies in the transform while leaving others unchanged.

Let $f$ represent the input image in Step 1 and $F$ its Fourier transform. The Fourier transform of the output image is:

$$G = HF$$

(3.2-24)

The multiplication of $H$ and $F$ involves two-dimensional functions, being defined on an element-by-element basis. In general, the components of $F$ are complex quantities, but the filters are real. In this case, each component of $H$ multiplies both the real and imaginary parts of the corresponding component in $F$. Such filters are called zero-phase-shift filters. These filters do not change the phase of the transform, because in (3.2-19) the multiplier of the real and imaginary part will cancel.

If we denote by $\mathcal{F}[ ]$ the Fourier transform of the argument, the filtered image is obtained by using the inverse Fourier transform of $G$:

$$\text{Filtered Image}=\mathcal{F}^{-1}[G]$$

(3.2-25)

The final image is obtained by taking the real part of this result. The inverse Fourier transform is, in general, complex. However, when the input image and the filter function are real, the imaginary components of the inverse transform should be zero. In practice, the inverse DFT generally has parasitic imaginary components due to computational round-off errors. These components will be ignored.

The filtering detailed on Figure 3.8 has a more general form that includes pre- and post-processing stages. Examples of other processes might include cropping of the input image to even dimensions (required for transform centering), gray-level scaling, conversion to floating point on input, and conversion to an 8-bit integer format on the output. Multiple
filtering stages and other pre- and post-processing functions are also possible. The filtering is based on modifying the transform of an image via a filter function, and then computing the inverse of the result to obtain the processed output image.

### 3.2.3.3 Basic Filters

We have the foundation for filtering in the frequency domain and now will use some specific filters to see how they operate on images. The expression for $F(0,0)$ given in (3.2-21) will help to introduce an example of filtering. Suppose we are interested in having the average value of an image at zero. According to (3.2-21), the average value of an image is computed as $F(0,0)$. We set this term as zero in the frequency domain and take the inverse transform. The resulting image will have the average value zero. We multiply all values of $F$ by the filter function:

$$H(u,v) = \begin{cases} 0 & \text{if } (u,v) = (0,0) \\ 1 & \text{otherwise} \end{cases}$$

(3.2-26)
This filter will set \( F(0,0) \) to zero and leave all other frequency components of the Fourier transform unchanged. The processed image (with zero average value) can be obtained by computing the inverse Fourier transform of \( HF \) as specified in (3.2-25). Both the real and imaginary parts of \( F \) are multiplied by the filter function \( H \).

This filter is called a notch filter because it is a constant with a hole (notch) at the origin. Figure 3.9 shows a standard test image, known as Lena, which appears frequently in image processing. The Lena image contains various combinations of image properties, such as a large number of curved edges and textures in her hair and the feathers on her hat, and combinations of light and dark regions of shading in the background and in her face. The

![Figure 3.9 Lena image.](image)

result of processing this image with the notch filter is shown in Figure 3.10. Note that the overall average gray level has been decreased. This is a result of forcing the average to zero. Notice also the by-product result of making prominent edges stand out. Actually, the average
of the displayed image cannot be zero because the image would have to contain negative gray-level values and displays cannot handle negative quantities. Figure 3.10 was displayed in the “standard” way, with its most negative value as 0, or black, and all other values scaled up from 0. Notch filters are useful when it is possible to identify spatial image effects caused by specific, localized frequency domain components.

Low frequencies in the Fourier transform give the gray-level appearance of a smooth image, while high frequencies show detail, such as edges and noise. A filter that attenuates high frequencies while “passing” low frequencies is called a lowpass filter, and a filter with the opposite characteristics is called a highpass filter. We would expect a lowpass-filtered image to have less sharp detail than the original because the high frequencies have been attenuated. Similarly, a highpass-filtered image would have less gray level variations in smooth areas and emphasized gray-level detail. A highpass filtered image will be “sharper”.

Figures 3.12-3.15 illustrate the effect of lowpass and highpass filtering the image in Figure
Figure 3.11 A magnetic resonance brain image

describe” the filters and the Figures 3.13 and 3.15 show the results of filtering with the procedure described in Figure 3.8. The filters \( H \), shown in Figures 3.12 and 3.14 are both circularly symmetric. They were multiplied by the centered Fourier transform, as outlined in (3.2-24) and (3.2-25) and Figure 3.8. The real part of the result yields the images in Figures 3.13 and 3.15. The image in Figure 3.13 is blurred, and the image in Figure 3.15 is sharp, with very little smooth gray level detail because the \( F(0,0) \) term was set to zero. This usually happens when a highpass filter in the frequency domain is used. A procedure often used is to add a constant to the filter so that it will not completely zero \( F(0,0) \). The result of using this procedure is shown in Figure 3.16. This is clearly an improvement when compared with the image in Figure 3.15.
Figure 3.12 A two-dimensional lowpass filter function

Figure 3.13 Result of lowpass filtering the original image
Figure 3.14 A two-dimensional highpass filter function

Figure 3.15 Result of highpass filtering the original image
Figure 3.16 Result of highpass filtering the image in Figure 3.11 with the filter in Figure 3.14, modified by adding a constant of one-half of the filter height to the filter function

3.2.4 Filtering in the Spatial and Frequency Domains and Their Relationship

In Chapter 3, forms of various spatial filters were presented. These can be derived using intuition and/or mathematical formulation, such as the Laplacian. This section will help us to establish a link between some spatial filters and their frequency domain counterparts.

The relationship between the spatial and frequency domain is described by a well-known result called the convolution theorem. The basic concepts and formulas of convolution in the spatial domain were introduced in Chapter 2, Section 2.5. The process of moving a mask from pixel to pixel in an image and computing a predefined quantity at each pixel is an explanation of the convolution process. The discrete convolution of two functions $f$ and $h$ of size $M \times N$, denoted by $f * h$ is defined as:
\[(f \ast h)(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n)h(x - m, y - n) \]  

(3.2-27)

With some differences in the leading constant, the minus sign, and the limits of the summation, this expression is similar in form to (2.5-1), Chapter 2, Section 2.5. The minus signs are needed because the function \( h \) is mirrored about the origin. The convolution of two functions can be summarized in the following procedure:

1. flip one function about the origin;
2. shift that function with respect to the other by changing the values of \((x, y)\);
3. compute a sum of products over all values of \( m \) and \( n \), for each displacement \((x, y)\).

Let \( F \) and \( H \) denote the Fourier transforms of \( f \) and \( h \), respectively. One-half of the convolution theorem states that \( f \ast h \) and \( FH \) constitute a Fourier transform pair. This result can be expressed as:

\[ f \ast h \Leftrightarrow FH \]  

(3.2-28)

The double arrow indicates that the expression on the left (spatial convolution) can be obtained taking the inverse Fourier transform of the product \( F(u, v)H(u, v) \) in the frequency domain. The expression on the right can be obtained as the result of applying the forward Fourier transform of the expression on the left. An analogous result is that convolution in the frequency domain reduces to multiplication in the spatial domain, and vice versa:

\[ fh \Leftrightarrow F \ast H \]  

(3.2-29)

The results expressed in (3.2-28) and (3.2-29) compose the convolution theorem.

We will consider one more concept for the relationship between the spatial and frequency domain. An impulse function of strength \( A \), located at coordinates \((x_0, y_0)\), is denoted by \( A \delta_{x_0,y_0} \) and is defined by the expression
This equation states that the summation of a function \( s(x, y) \) multiplied by an impulse is the value of that function at the location of the impulse, multiplied by the strength of the impulse. The limits of the summation are the same as the limits spanned by the function. We point out that \( A\delta_{y_0,y_0} \) also is an \( M \times N \) image. It contains zeros at all coordinates, except the point \((x_0, y_0)\), where the value of the image is \( A \).

If we let \( h \) in (3.2-27) to be an impulse function and use the definition in (3.2-30), we get:

\[
\begin{align*}
    f(x_0, y_0) * A\delta_{y_0,y_0}(x_0, y_0) &= \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n)A\delta_{y_0,y_0}(x_0 - m, y_0 - n) \\
    &= \frac{A}{MN} f(x_0, y_0)
\end{align*}
\]

This equality means that convolution of a function with an impulse produces the value of that function at the location of the impulse. This characteristic is called the shifting property of the impulse function. We consider the unit impulse located at the origin, denoted as \( \delta_{0,0} \). In this case,

\[
\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} s(x, y)\delta_{0,0}(x, y) = s(0,0)
\]

We can derive a very interesting relationship between filtering in the spatial and frequency domain. Using (3.2-16) and (3.2-32), we compute the Fourier transform of a unit impulse at the origin:

\[
F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \delta_{0,0}(x, y)e^{-i2\pi(x/M+y/N)} = \frac{1}{MN}
\]

Thus, the Fourier transform of an impulse at the origin of the spatial domain is a real constant.
Assume $f = \delta_{0,0}$ and compute for every pair $(x, y)$ the convolution defined in (3.2-27). Using (3.2-32) allows:

$$f(x, y) * h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \delta(m, n) h(x - m, y - n)$$

$$= \frac{1}{MN} h(x, y) \quad \text{(3.2-34)}$$

Combine the results of (3.2-33) and (3.2-34) with (3.2-28) and:

$$f * h \leftrightarrow FH$$

$$\delta * h \leftrightarrow \mathcal{F}[\delta]H$$

$$h \leftrightarrow H \quad \text{(3.2-35)}$$

Using the properties of the impulse function and convolution theorem, we have proven that filters in the spatial and frequency domains constitute a Fourier transform pair. Given a filter in the frequency domain, we can obtain the corresponding filter in the spatial domain by computing the inverse Fourier transform. The reverse is also true.

All functions involved in the previous calculations were $M \times N$. In practice, using a filter in the frequency domain and the inverse transform to get an equivalent spatial domain filter of the same size does not help from a computational point of view. If both filters are the same size, it is computationally efficient to apply the filtering in the frequency domain. On the other hand, we use smaller filters in the spatial domain. It makes sense to filter in the spatial domain with small filter masks. With (3.2-35) we can specify filters in the frequency domain, take their inverse transform, and use the resulting filter in the spatial domain as a guide to smaller spatial filter masks.

Filters based on Gaussian functions are important because they are easily specified and both the forward and inverse Fourier transforms of a Gaussian function are also real Gaussian functions. To simplify the notation, we begin our use with functions of one variable.
Let \( H(u) \) denote a Gaussian function in the frequency domain:

\[
H(u) = A e^{-u^2/2\sigma^2} \tag{3.2-36}
\]

where \( \sigma \) is the standard deviation of the Gaussian curve. The corresponding filter in the spatial domain is:

\[
h(x) = \sqrt{2\pi\sigma} A e^{-x^2/2\sigma^2} \tag{3.2-37}
\]

These two functions constitute a Fourier transform pair; both are Gaussian and real. They have reciprocity i.e. if \( H \) has a broad profile (for large values of \( \sigma \)) then \( h \) has a narrow profile and vice versa. This type of reciprocal behavior was encountered in Section 3.2.1, regarding Figure 3.1. A plot of a Gaussian filter in the frequency domain is shown in Figure 3.17 (a). This is a lowpass filter. The corresponding lowpass filter in the spatial domain is shown in Figure 3.17 (c). We observe that all the values are positive in both domains. So, we conclude that one can implement lowpass filtering in the spatial domain by using a mask with all positive coefficients, as used in our examples in Section 2.6.1. Two of the masks from that section are presented in Figure 3.17 for reference. The reciprocal relationship is important. The more narrow the frequency domain filter, the more it will attenuate the low frequencies; this will result in increased blurring. In the spatial domain, this results in a wider filter, which implies a larger mask, as illustrated in Section 2.6.1 in connection with Figures 2.45-2.50.

We may use the Gaussian filter (3.2-36) to construct more complex filters. For instance, we can consider a highpass filter as the difference of two Gaussians:

\[
H(u) = A e^{-u^2/2\sigma_1^2} - B e^{-u^2/2\sigma_2^2} \tag{3.2-38}
\]

with \( A \geq B \) and \( \sigma_1 > \sigma_2 \). The corresponding filter in the spatial domain is:

\[
h(x) = \sqrt{2\pi\sigma_1} A e^{-x^2/2\sigma_1^2} - \sqrt{2\pi\sigma_2} B e^{-x^2/2\sigma_2^2} \tag{3.2-39}
\]

The plots of these two functions are shown in Figures 3.17 (b) and (d). We note the reciprocity in width, and the spatial filter has both negative and positive values. Two of the
masks used in Section 2.7.2.3, (see Figure 2.63) for highpass filtering are shown in Figure 3.17 (d).

![Figure 3.17](image)

Figure 3.17 (a) Gaussian frequency domain lowpass filter. (b) Gaussian frequency domain highpass filter. (c) Corresponding lowpass spatial filter. (d) Corresponding highpass spatial filter.

In the frequency domain, we take advantage of the correspondence between frequency content and image appearance. Some enhancement tasks that would be difficult or impossible to formulate directly in the spatial domain become almost trivial in the frequency domain. Once we select a specific filter via experimentation in the frequency domain, the implementation of the method usually is performed in the spatial domain. One approach would be to construct small spatial masks that attempt to capture the “essence” of the full filter function in the spatial domain, as shown on Figure 3.17.
3.3 Smoothing Frequency-Domain Filters

In Section 3.2.3 the edges and other sharp transitions (such as noise) in the gray levels of an image were shown to contribute significantly to the high-frequency content of its Fourier transform. Hence smoothing (blurring) in the frequency domain can be obtained by attenuating a specified range of high-frequency components in the transform of an image. We will use (3.2-24) for filtering in frequency domain. For convenience, we write the equation:

\[ G = HF \]  

(3.3-1)

where \( F \) is the Fourier transform of the image we intend to smooth. Our objective is to select a filter transfer function \( H \) that will attenuate the high-frequency components of \( F \). The filtering in this section is based on the procedure described in Section (3.2-3), including the zero-phase-shift filters.

We consider three types of filters: ideal, Butterworth, and Gaussian. These filters cover the range from very sharp (ideal) to very smooth (Gaussian). The Butterworth filter has a parameter, called the filter order. For high values of this parameter the Butterworth filter approaches the ideal filter. For lower-order values, the Butterworth filter has a smooth form similar to the Gaussian filter.

3.3.1 Ideal Lowpass Filters

The simplest lowpass filter removes all high-frequency components of the Fourier transform at a value greater than a specified value \( D_0 \) from the origin of the centered transform. Such a filter is called a two-dimensional (2-D) ideal lowpass filter (ILPF). Its transfer function is:

\[ H(u, v) = \begin{cases} 
1 & \text{if } D(u, v) \leq D_0 \\
0 & \text{if } D(u, v) > D_0 
\end{cases} \]  

(3.3-2)

where \( D_0 \) is a given nonnegative number and \( D(u, v) \) is the distance from point \((u, v)\) to the center of the frequency rectangle. If the image size is \( M \times N \), then its transform is the same
size, and the center of the frequency rectangle is \((M/2, N/2)\). The distance from any point \((u,v)\) to the center (origin) of the Fourier transform is:

\[ D(u,v) = \left[ (u - M/2)^2 + (v - N/2)^2 \right] \]  

(3.3-3)

Figure 3.18 shows a 3-D perspective of \(H(u,v)\) as a function of \(u\) and \(v\). Figure 3.19 is the projection of \(H(u,v)\) onto the \((u,v)\) plane. An ideal filter indicates that all frequencies inside a circle of radius \(D_0\) are passed without attenuation, whereas all frequencies outside this circle are completely attenuated. Filters that are radially symmetric about the origin can be specified with a cross section extending as a function of distance from the origin along a radial line as Figure 3.20 shows. The complete transfer function can be visualized by rotating the cross section \(360^\circ\) about the origin.

Figure 3.18 Perspective plot of an ideal lowpass filter transfer function
For an ideal lowpass filter cross section, the point of transition between $H(u,v) = 1$ and $H(u,v) = 0$ is called the cutoff frequency. In the Figure 3.20, the cutoff frequency is $D_0$.

We can compare the lowpass filters in this section by their behavior as a function of cutoff frequencies. A set of cutoff frequencies is generated by circles that enclose specified amounts of the total image power $P_T$. To generate this we sum the components of the power spectrum at each point $(u,v)$ for $u = 0, 1, \ldots, M-1$ and $v = 0, 1, \ldots, N-1$:

$$P_T = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} P(u,v)$$  \hspace{1cm} (3.3-4)

where $P(u,v)$ is the result of (3.2-20). If the transform is centered then a circle of radius $r$ with origin at the center of the frequency rectangle encloses $\alpha$ percent of the power where

$$\alpha = 100 \left( \sum_{u} \sum_{v} P(u,v)/P_T \right)$$  \hspace{1cm} (3.3-5)
and this summation is over the values of $(u,v)$ inside the circle and on its boundary.

Figure 3.21 shows an image of a brain. The Fourier spectrum of this image is shown in Figure 3.22. The circles superimposed on the spectrum have radii 5, 15, 30, 50 and 80 pixels (the circle of radius 5 is not easily visible). The percent values of the image power enclosed in these circles are displayed in Table 3.1. The spectrum decreases rapidly, with only 88.34% of the total power contained in the relatively small circle of radius 5.

Figure 3.23 shows the results of applying ideal lowpass filters with cutoff frequencies at the radii shown in Figure 3.22. Figure 3.23 (b) is of no value, unless the objective of blurring is to eliminate all detail in the image, except for the “blobs” that represent large objects. The severe blurring in this image indicates that most of the sharp detail of this picture is contained in the 12% power removed by the filter. As the filter radius increases, less power
Figure 3.21 An image of size $256 \times 256$ pixels

Figure 3.22 The Fourier spectrum of the image in Figure 3.21
Table 3.1 Values of percent of the image power included in circles with different radii

<table>
<thead>
<tr>
<th>Radius</th>
<th>$\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>88.3435</td>
</tr>
<tr>
<td>15</td>
<td>94.4298</td>
</tr>
<tr>
<td>30</td>
<td>98.3659</td>
</tr>
<tr>
<td>50</td>
<td>99.5492</td>
</tr>
<tr>
<td>80</td>
<td>99.9472</td>
</tr>
</tbody>
</table>

is lost; this results in less blurring. A careful analysis of the result for $\alpha = 99.94$ shows minimal blurring, but this image is quite close to the original. This demonstrates that little edge information is contained in the upper 0.1% of the spectrum power.

Using these examples, we may conclude that ideal lowpass filtering is not very practical. However, because ideal filters can be implemented with a computer, it is useful to study their behavior.

### 3.3.2 Butterworth Lowpass Filters

The transfer function for a Butterworth lowpass filter (BLPF) of order $n$, with cutoff frequency $D_0$ from the origin, is defined as:

$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}}$$  \hspace{1cm} (3.3-6)

where $D(u,v)$ is (3.3-3). Figure 3.24 shows a perspective plot of the Butterworth lowpass filter transfer function. Figure 3.25 is the Butterworth filter projected onto the $(u,v)$ plane. Figure 3.26 presents the radial cross section of this filter $D(u,v)$ vs. $H(u,v)$.

Unlike the ideal lowpass filter, the Butterworth lowpass filter transfer function does not show a sharp discontinuity that defines a clear cutoff between passed and filtered frequencies.
Figure 3.23 (a) Original image. (b)-(f) Results of ideal lowpass filtering with cutoff frequencies set at radii values of 5, 15, 30, 50, 80
Figure 3.24 Perspective plot of a Butterworth lowpass filter transfer function

Figure 3.25 Butterworth lowpass filter projected onto the $(u, v)$ plane
Figure 3.26 Butterworth lowpass filter radial cross section of orders 1 through 4

Figure 3.27 shows the results of applying the BLPF given by (3.3-6) to Figure 3.27 (a) with $n = 2$ and $D_0$ equal to 5, 15, 30, 50, 80. One can see a smooth transition in blurring as the cutoff frequency increases.

### 3.3.3 Gaussian Lowpass Filters

We introduced Gaussian lowpass filters (GLPFs) in Section 3.2.4 and explored some important relationships between the spatial and frequency domains. In two dimensions, these filters are defined by:

$$H(u, v) = e^{-D^2(u, v)/2\sigma^2}$$

(3.3-7)

where $D(u, v)$ is the distance from an origin of the Fourier transform, shifted to the center of the frequency rectangle. In this, we do not use a constant with the filter as in Section 3.2.4 but to be consistent with the other filters discussed, which have a value of 1 at the origin. The parameter $\sigma$ measures the spread of the Gaussian curve. If $\sigma = D_0$, we can express this filter
Figure 3.27 (a) Original image. (b)-(f) Results of filtering with BLPFs of order 2, with cutoff frequencies at radii of 5, 15, 30, 50, 80
as:

$$H(u, v) = e^{-D^2(u,v)/2D_0^2}$$  \hspace{1cm} (3.3-8)

where $D_0$ denotes the cutoff frequency. When $D(u, v) = D_0$, the filter is 0.607 of its maximum value.

In Section 3.2.4, the inverse Fourier transform of the Gaussian lowpass filter also is itself Gaussian. The advantages of this property in analysis were presented in Section 3.2.4. Figures 3.28, 3.29 and 3.30 show a perspective plot, projection onto $(u, v)$ plane and radial cross sections of a GLPF function.

Figure 3.28 shows the result of applying the GLPF (3.3-8) to Figure 3.31 (a), with $D_0$ equal to 5, 15, 30, 50, 80. Using BLPF (Figure 3.27), we observe a smooth transition in blurring as the cutoff frequency increases. The GLPF does not produce as much
Figure 3.29 Gaussian lowpass filter projected onto \((u, v)\) plane

Figure 3.30 Gaussian lowpass filter radial cross section for various values of \(D_0\)
Figure 3.31 (a) Original image. (b)-(f) Results of filtering with Gaussian lowpass filters with cutoff frequencies set at radii values of 5, 15, 30, 50, 80
smoothing as the BLPF of order 2 for the same cutoff frequency, as observed by comparing Figures 3.27 (c) and 3.31 (c). This occurs because the profile of the GLPF is not as “tight” as the profile of the BLPF of order 2. However, the results are comparable in general. For applications that require tight control of the transition between low and high frequencies about the cutoff frequency, the BLPF could be a more appropriate choice.

3.4 Sharpening Frequency-Domain Filters

In the previous section images are blurred by attenuating the high-frequency components of their Fourier transform. Edges and other abrupt changes in gray levels are associated with high-frequency components. Thus, we can sharpen an image in the frequency domain by using a highpass filter, which attenuates the low-frequency but does not affect the high-frequency in the Fourier transform. Only zero-phase filters that are radially symmetric are considered.

The filters in this section have the reverse operation of the ideal lowpass filters described in the previous section. Thus, the transfer function of the highpass filters in this section is:

\[
H_{hp}(u,v) = 1 - H_{lp}(u,v)
\]

where \(H_{lp}(u,v)\) is the transfer function of the corresponding lowpass filter. This relationship shows that when the lowpass filter attenuates frequencies, the highpass filter passes them and vice-versa.

We will consider ideal, Butterworth and Gaussian highpass filters to illustrate their characteristics in the frequency domain.

3.4.1 Ideal Highpass Filters

A 2-D ideal highpass filter (IHPF) is defined:

\[
H(u,v) = \begin{cases} 
0 & \text{if } D(u,v) < D_0 \\
1 & \text{if } D(u,v) \geq D_0
\end{cases}
\]

(3.4-2)
where $D_0$ is the cutoff value measured from the origin of the frequency rectangle and $D(u,v)$ is (3.3-3). This filter is the opposite of the ideal lowpass filter; it zeros all frequencies inside a circle of radius $D_0$ and passes all frequencies outside this circle.

Figure 3.32 shows a 3-D perspective plot of $H(u,v)$ as a function of $u$ and $v$ and Figure 3.33 shows $H(u,v)$ displayed as a projection onto the $(u,v)$ plane. A cross section as a function of distance from the origin along a radial line is sufficient to specify the filter, as shown on Figure 3.34.

Figure 3.35 (a) shows a sharp image of a brain. Figures 3.35 (b)-(d) show various IHPF results of applying ideal highpass filters to the image in Figure 3.35 (a) with $D_0$ set to 15, 30, 50 pixels respectively. We chose a sharp version of the image shown in Figure 3.21 to see the contours more clearly. The edges of the image shown in Figure 3.35 (b) appear as solid white and thicker than the edges of the image shown in Figures 3.35 (c) and
Figure 3.33 Ideal highpass filter displayed as a projection onto the \((u, v)\) plane

Figure 3.34 Ideal highpass filter radial cross section
(d). For $D_0 = 30$, we observed that the edges are also clearly visible and one can see the results of filtering smaller objects. The result for $D_0 = 50$ shows thinner edges and that the smaller objects have been filtered properly.

### 3.4.2 Butterworth Highpass Filters

The transfer function for the Butterworth highpass filter (BHPF) of order $n$ with a cutoff frequency at $D_0$ is expressed as:

$$H(u, v) = \frac{1}{1 + \left[D_0 / D(u, v)\right]^{2n}}$$  \hspace{1cm} (3.4-3)
where $D(u,v)$ is defined in (3.3-3). (3.4-3) can be derived from (3.4-1) and (3.3-6). Figure 3.36 shows a perspective plot of the Butterworth highpass filter transfer function, Figure 3.37 shows a projection of Figure 3.36 onto the $(u,v)$ plane and Figure 3.38 a cross section of a BHPF function.

![Figure 3.36 Perspective plot of a Butterworth highpass filter transfer function](image)

Similar to the case of lowpass filters, we expect Butterworth highpass filters to have a smoother behavior compared to IHPFs. Figures 3.39 (b)-(d) show the results of highpass filtering the image in Figure 3.39 (a) (the same image as in Figure 3.35 (a)) using a BHPF of order 2 with $D_0$ set to the same values as in Figure 3.35. The boundaries can be seen more clearly than in Figures 3.35 (b)-(d). The performance of the two filters in terms of filtering small objects is similar.
Figure 3.37 Butterworth highpass filter displayed as a projection onto the \((u, v)\) plane

Figure 3.38 Butterworth lowpass filter radial cross section of order 2
Figure 3.39 (a) Original image. (b)-(d) Results of highpass filtering the image in (a) using a BHPF of order 2 with $D_0 = 15, 30, 50$ respectively.

### 3.4.3 Gaussian Highpass Filters

The transfer function for the Gaussian highpass filter (GHPF) with cutoff frequency $D_0$ is given by

\[ H(u, v) = 1 - e^{-D^2(u,v)/2D_0^2} \]  \hspace{1cm} (3.4-4)

where $D(u, v)$ is given in (3.3-3). This expression follows from (3.4-1) and (3.3-8). Figure 3.40 shows a perspective plot of the GHPF function. Figure 3.41 presents the projection of the GHPF onto the $(u, v)$ plane and Figure 3.42 presents a cross section of the GHPF.
Figure 3.40 Perspective plot of a GHPF transfer function

Figure 3.41 Gaussian highpass filter displayed as a projection onto the \((u, v)\) plane
Figures 3.43 (b)-(d) show the results of highpass filtering the image in Figure 3.43 (a) (the same image as in Figures 3.35 (a) and 3.39 (a)) using a GHPF with \( D_0 \) set to the same values as in Figure 3.35. The results obtained with the GHPF are smoother than those obtained with the previous two filters. The filtering of smaller objects and thin bars is cleaner with the Gaussian highpass filter.

As mentioned in Section 3.2-4, other highpass filters can be constructed as the difference of Gaussian lowpass filters. These difference filters have more parameters; therefore, the filter shape can be determined more precisely. The simple filter given by (3.4-4) is very useful in practice and can be easily formulated.
3.4.4 The Laplacian in the Frequency Domain

The following is a property of the Fourier transform:

\[ \mathcal{F}\left[ \frac{d^n f(x)}{dx^n} \right] = (i\omega)^n \mathcal{F}(f) \]  

(3.4-5)

Using this property:

\[ \mathcal{F}\left[ \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \right] = (i\omega)^2 \mathcal{F}(f(u, v)) + (i\omega)^2 \mathcal{F}(f(u, v)) \]

\[ = -(u^2 + v^2)\mathcal{F}(f(u, v)) \]  

(3.4-6)
In (2.7-1), the Laplacian is defined by the expression in the brackets on the left hand side of (3.4-6). Thus, (3.4-6) can be written:

\[ \mathcal{F}\left[\nabla^2 f(x, y)\right] = -\left(u^2 + v^2\right)F(u, v). \quad (3.4-7) \]

Equation (3.4-7) allows the Laplacian in the frequency domain using the filter to be implemented as:

\[ H(u, v) = -\left(u^2 + v^2\right). \quad (3.4-8) \]

The filtering is done after the origin of \( F(u, v) \) has been centered. If \( f \) and \( F \) are \( M \times N \), the center of the transform is shifted so that \((u, v) = (0, 0)\) becomes \((M/2, N/2)\) in the frequency rectangle. Thus, the center of the filter function is shifted:

\[ H(u, v) = -\left[\left(\frac{u}{2}\right)^2 + \left(\frac{v}{2}\right)^2\right]. \quad (3.4-9) \]

The Laplacian-filtered image in the spatial domain is obtained with the inverse Fourier transform of \( H(u, v)F(u, v) \):

\[ \nabla^2 f(x, y) = \mathcal{F}^{-1}\left\{ -\left[\left(\frac{u}{2}\right)^2 + \left(\frac{v}{2}\right)^2\right]F(u, v)\right\}. \quad (3.4-10) \]

If the Laplacian in the spatial domain is computed using (2.7-1) and then the Fourier transform is applied, we obtain \( F(u, v)H(u, v) \). This relationship can be expressed as:

\[ \nabla^2 f(x, y) \Leftrightarrow -\left[\left(\frac{u}{2}\right)^2 + \left(\frac{v}{2}\right)^2\right]F(u, v) \quad (3.4-11) \]

Figure 3.44 shows the 3-D plot of \( H(u, v) \) and Figure 3.45 is the projection of \( H(u, v) \) onto the \((u, v)\) plane. The Laplacian is centered at \((M/2, N/2)\) and has negative values, except for the value at its maximum which is zero.

Using the rule described in (2.7-5), an enhanced image \( g(x, y) \) is obtained by subtracting the Laplacian from the original image:

\[ g(x, y) = f(x, y) - \nabla^2 f(x, y) \quad (3.4-12) \]
Figure 3.44 3-D plot of the Laplacian in the frequency domain

Figure 3.45 The Laplacian in the frequency domain projected onto the \((u, v)\) plane
In the spatial domain we also obtained an enhanced image with a single mask. Similarly, the entire operation in the frequency domain can be performed with only one filter, defined as $H(u,v) = [1 + ((u-M/2)^2 + (v-n/2)^2)]$. In this case a single inverse transform operation is used for the enhanced image:

$$g(x,y) = f(x,y) * [1 + ((u-M/2)^2 + (v-N/2)^2)]F(u,v)$$ (3.4-13)

Figure 3.46 shows the same image as Figure 2.55. Figure 3.47 shows the result of filtering this image in the frequency domain using (3.4-10). The Laplacian-filtered images contain both positive and negative values of comparable magnitudes, so scaling was necessary. Figure 3.48 shows the image scaled (for display purposes), so that the most negative value was scaled to zero and the maximum positive value was scaled to maximum value of the gray level range. Figure 3.49 shows the enhanced image obtained after using (3.4-12). For the image in Figure 3.49, the small feature detail is sharper, as expected after
Figure 3.47 Laplacian filtered image

Figure 3.48 Laplacian image scaled
using the Laplacian. The sequence of images shown in Figures 3.46-3.49 can be compared with the images displayed in Figures 2.55—2.58, obtained by applying the Laplacian filter in the spatial domain. The results are similar for all practical purposes.

### 3.4.5 Unsharp Masking, High-Boost Filtering and High-Frequency Emphasis Filtering

The average background intensity of all the filtered images displayed in Sections 3.4.1 -3.4.3 has been reduced to near black. This happened because the highpass filters applied to those images eliminate the zero-frequency component of their Fourier transforms. In section 3.2.3 this phenomenon was discussed. This can be avoided by adding a portion of the image back to the filtered result. Enhancement using the Laplacian added back the entire image to the filtered result. Sometimes it is desirable to increase the contribution of the original image to the overall filtered result. This approach, called high-boost filtering, is a generalization of
unsharp masking. These procedures were discussed in Section 2.7.2. We repeated them here using frequency domain notation.

Unsharp masking generates a sharp image by subtracting a blurred version of the image from the original image. A highpass-filtered image is created by subtracting a lowpass filtered version of the image from the image itself. We express this procedure as:

\[ f_{hp}(x, y) = f(x, y) - f_{lp}(x, y) \]  
\[ f_{lp}(x, y) \]


High-boost filtering generalizes this procedure by multiplying \( f(x, y) \) by a constant \( A \geq 1 \):

\[ f_{hh}(x, y) = Af(x, y) - f_{lp}(x, y) \]  
\[ f_{lp}(x, y) \]

High-boost filtering allows an increase to the contribution made by the image in the enhanced result. The previous equation can be written as:

\[ f_{hh}(x, y) = (A - 1)f(x, y) + f(x, y) - f_{lp}(x, y) \]  
\[ f_{hp}(x, y) \]

Using (3.4-14), we obtain:

\[ f_{hp}(x, y) = (A - 1)f(x, y) + f_{hp}(x, y) \]  
\[ f_{hp}(x, y) \]

In this equation, a highpass filter is used rather than a lowpass filtered version of the image. For \( A = 1 \), high-boost filtering becomes regular highpass filtering. For \( A > 1 \), the contribution of the image is more dominant.

From (3.4-14) we obtain \( F_{hp}(u, v) = F(u, v) - F_{lp}(u, v) \). If \( H_{lp} \) is the transfer function of a lowpass filter, we express \( F_{lp} \) as \( F_{lp}(u, v) = H_{lp}(u, v)F(u, v) \). Therefore, unsharp masking can be implemented in the frequency domain by using the filter

\[ H_{hp}(u, v) = 1 - H_{lp}(u, v) \]  
\[ H_{lp}(u, v) \]

This result agrees with (3.4-1). Similarly, high-boost filtering can be implemented using the filter:

\[ H_{hh}(u, v) = (A - 1) + H_{lp}(u, v) \]  
\[ H_{hp}(u, v) \]
with $A \geq 1$. This filter can be multiplied by the centered transform of the input image and the inverse transform of this product is obtained. The real part of this result is the high-boost filtered image $f_{hb}(x,y)$ in the spatial domain.

Figures 3.50-3.53 show the same sequence as Figures 2.64-2.67, using frequency domain computations. Figure 3.50 is the input image and Figure 3.51 is the highpass filtered image. We used the Laplacian as a highpass filter (3.4-10). Here, the Laplacian was used instead of a composite mask to produce the scaling operation.

![Figure 3.50 A dark magnetic resonance image of a human head](image)

The image in Figure 3.52 was obtained using (3.4-17) with $A = 2$. As in Figure 2.66 this image is sharper, but still too dark. We used $A = 2.7$ to obtain the image displayed in Figure 3.53 (the input image was multiplied by 1.7 before subtracting the Laplacian from it). This image is an improved result.
Figure 3.51 Laplacian of the image in figure 3.50

Figure 3.52 Image obtained using (3.4-17) with $A = 2$
Sometimes the purpose is to emphasize the contribution to enhancement made by the high-frequency components of an image. In this case, multiply the highpass filter function by a constant and add a offset so that the zero frequency term is not excluded by the filter. This process is called high-frequency emphasis. The filter transfer function is:

\[ H_{hp}(u, v) = a + bH_{hp}(u, v) \]  

(3.4-20)

where \( a \geq 0 \) and \( b > a \). Typically, \( a \in [0.25, 0.5] \) and \( b \in [1.5, 2] \). Using (3.4-19), we observe that high-frequency emphasis reduces to high-boost filtering when \( a = (A - 1) \) and \( b = 1 \). For \( b > 1 \), the high frequencies are emphasized, so the procedure is called high-frequency emphasis.

Figure 3.54 shows a blurred version of a chest X-ray image. Our objective is to sharpen the image. X-rays cannot be focused in the same way that lenses are focused, so the resulting image could be blurred.
Figure 3.54 A blurred version of a chest X-ray image

Figure 3.55 shows the result of highpass filtering using a Butterworth filter of order 2 and a value of $D_0 = 12$ (5% of the image vertical dimension). The filtered result is rather featureless, but it shows faintly the principal edges in the image. The advantage of high-emphasis filtering (with $a = 0.5$ and $b = 2$) is shown in Figure 3.56. The image is still dark, but the gray-level tonality due to the low frequency component was not lost.

As specified in Section 2.3, any image with gray levels in a narrow range of gray scale is a candidate for histogram equalization. This method was used and the result is displayed in Figure 3.57. The bone structure and other details not visible in the result is displayed in Figure 3.57. The bone structure and other details not visible in the previous three images, can be observed in this image. The final enhanced image is a little noisy, but this is typical of X-ray images when their gray scale is expanded. We used a combination of high-frequency
Figure 3.55 Result of Butterworth highpass filtering

Figure 3.56 Result of high-frequency emphasis filtering
Figure 3.57 Result of performing histogram equalization on the image in Figure 3.56 emphasis and histogram equalization and the result is superior to the result that would have been obtained with either method alone.
Chapter 4
Processing Image Structure

There have been three principal approaches to the study of image-based spatial processing within vision science: computational, psychophysical, and physiological. They use fundamentally different methods but have related goals.

The physiological approach of spatial image processing came first historically; it resulted from the application of recently developed single-cell recording techniques in the 1950s and 1960s.

4.1 Physiological Mechanisms

The first knowledge about the biological underpinnings of vision was created as a result of the studies of impaired perception due to large-scale brain damage from strokes, tumors, and gunshot wounds (Ferrier, [1878]; review in Glickstein, [1988]). The authors discovered that the brain areas that are dedicated to visual abilities are in or near the occipital cortex at the back of the head. Very little was known about the specific neural events that resulted in vision until the single-cell recording techniques in the 1950s (see Section A.4.3 of Appendix A). This allowed the vision scientists to understand how neural information processing works and to explore the mechanisms of vision neuron by neuron.

4.1.1 Retinal and Geniculate Cells

In this section our concern is how the spatial processing in the retina. In Section A.2 of Appendix A we mentioned several different layers of retinal cells that did some spatial and temporal processing before the neural signals left the eye. They are important in the earliest
operations of visual processing; their discovery made possible Hubel and Wiesel’s landmark studies of cortical mechanisms of spatial vision.

4.1.1.1 Ganglion Cells

The ganglion cells were the first retinal cells whose spatial properties yielded to scientific analysis. Stephen Kuffler and Horace Barlow recorded the firing rates from individual ganglion cells using a micro-electrode (Kuffler, [1953] and Barlow, [1953]). Different images were presented to the animal’s retina to determine which ones made the cell fire and which ones did not. Kuffler and Barlow discovered that the firing rate was highest for a spot of light of a particular size at a particular position on the retina. If the size of the spot was either increased or decreased, the firing rate diminished (Figure 4.1, Palmer [1999], p. 147).

Researchers mapped the complete receptive fields of these ganglion cells and found two distinct types: on-center cells and off-center cells. On-center cells produced spike discharges when the light at the center of the respective field was turned on (Figure 4.2 A). Off-center cells caused discharges when the light at the center was turned off (Figure 4.2 B). On-center cells have an excitatory response to light at the center, whereas off-center cells have an inhibitory response to light at the center. The area that surrounds the central region has the opposite characteristic: on-center cells have off-surrounds and off-center cells have on-surrounds (Figure 4.2, Palmer, [1999], p. 148). The characteristic structure of on-center cells is shown in Figure 4.2. The excitation/inhibition of the cell to light is plotted as the height of the graph. This type of receptive field is known as a “Mexican hat” because its three-dimensional structure, obtained by rotating the graph in depth around its central axis resembles a Mexican hat. Off-center, on-surround cells (B) have the opposite characteristics; they have an “inverted Mexican hat” receptive field structure.
Figure 4.1 Response of on-center, off-surround ganglion cells. As the size of a spot of light increases, the response of an on-center, off-surround ganglion cell first increases (A), reaches a maximum when it covers the excitatory center (B), and then decreases as the spot falls on the inhibitory surround (C and D). Minimum response (E) occurs when light falls on the inhibitory surround.

4.1.1.2 Bipolar Cells

Ganglion cells are the first cells in the visual system to produce spike discharges. Receptors and bipolar cells produce graded potentials—continuous changes in electrical potential. Later studies proved that bipolar cells, like ganglion cells, have circularly symmetric receptive fields with antagonistic relations between center and surround (Dacey, [1996]; Werblin, [1969]).

The neural architecture underlying the receptive fields of bipolar cells is illustrated in Figure 4.3A (Palmer, [1999], p. 149). Retinal receptor cells synapse directly onto both bipolar
Figure 4.2  Receptive field structure of ganglion cells. On-center, off-surround cells (A) fire to light onset and stop at offset in their excitatory center, but they stop firing to light onset and begin firing at offset in their inhibitory surround. Their 2-D receptive field is in the center. The “Mexican hilt” response profile is obtained when firing rate is plotted as a function of horizontal position through the center of the receptive field. Off-center, on-surround cells (B) have the opposite characteristics.

cells and horizontal cells. The horizontal cells synapse onto bipolar cells; an indirect pathway from receptors to bipolar cells is formed. The direct pathway from receptors to a given bipolar cell can be either excitatory or inhibitory; the indirect path is the opposite. Figure 4.3 shows a smaller central excitatory region from the direct pathway and a broader inhibitory region from the indirect ones. These two regions of excitation and inhibition are summed by the bipolar cell to produce the Mexican hat receptive field structure, as Figure 4.3B shows.

4.1.1.3 Lateral Geniculate Nucleus

Axons from the ganglion cells leave the eye through the optic nerve, pass through the optic chiasm, where some of them cross to the opposite side and synapse onto cell bodies in the lateral geniculate nuclei (LGN) of the thalamus. LGN cells have center/surrounded receptive
fields similar to those of retinal ganglion cells but larger and with a stronger inhibitory surround.

![Diagram of neural inputs to an on-center, off-surround bipolar cell.](image)

**Figure 4.3** Neural inputs to an on-center, off-surround bipolar cell. The direct path (black connections) comes from receptors in the center of the receptive field. The indirect path of opposite polarity (gray connections) comes from receptors throughout the receptive field via horizontal cells.

The cellular architecture of the LGN differs from that of the last retinal layer. The ganglion cells form a two-dimensional sheet parallel to the receptor surface of the retina and receive input from nearby cells in their own eye. In contrast, the LGN has a three-dimensional structure and receives input from both eyes. Each individual LGN is monocular, firing response to stimulation from one eye.

Studies of the internal architecture of the LGN have revealed important information about a crucial distinction in the visual system. The LGN is laminar (or layered); it consists
of many 2-D sheets of neurons. Each LGN has six distinct layers of cells folded as shown in Figure 4.4 (Palmer, [1999], p.149). The lower two layers are called the magnocellular (from the Latin magnus that means large) layers because they contain large cell bodies; the upper four layers are called parvocellular (from the Latin parvus that means small), because they have small cell bodies. These cells are different with respect to their physiology as well as their anatomy. The magnocellular cells are sensitive to differences in contrast, are not very selective to color, have relatively large receptive fields, and show a transient response to changes in retinal stimulation that start and end quickly. The parvocellular cells are relatively insensitive to contrast, highly selective to color, have relatively small receptive fields and yield a more sustained response to changes in retinal stimulation.

These differences between “mango” and “parvo” cells are presented in Table 4.1. Most of these differences are statistical; there is substantial overlap between mango and parvo cells. This specialization of function gave rise to the idea that the mango cells are a specialized neural pathway used to process motion and depth information, whereas the parvo
cells constitute a distinct pathway to process color and shape (DeYoe and Van Essen, [1988]; Livingstone and Hubel, [1988]).

Table 4.1 Functional differences between Magno and Parvo LGN cells

<table>
<thead>
<tr>
<th></th>
<th>Parvo</th>
<th>Magno</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color sensitivity</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Contrast sensitivity</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Spatial resolution</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Temporal resolution</td>
<td>Slow</td>
<td>Fast</td>
</tr>
<tr>
<td>Receptive field size</td>
<td>Small</td>
<td>Large</td>
</tr>
</tbody>
</table>

The researchers discovered that there are two different kinds of ganglion cells in the monkey retina that project selectively to the magnocellular and parvocellular cells in LGN. They are called M ganglion cells and P ganglion cells (Shapley and Perry, [1986]). P ganglion cells are more sensitive to color than to black and white, whereas the M ganglion cells are more sensitive to black and white than to color. P cells receive input from cones, and M cells receive input from both rods and cones.

Although the LGN as a whole gets input from both eyes, each layer receives signals from one eye. The four parvocellular layers alternate between left and right eye input, as well as the two magnocellular layers; the top and bottom layers receive input from the eye on the opposite side (Figure 4.5, Palmer [1999], p.150). Each layer is arranged spatially like the retina of the eye from which he receives input. This is called retinotopic mapping or topographic mapping; it preserves the relative location of cells from retina to LGN: nearby regions on the retina project to adjacent regions of the LGN. This type of spatial mapping is a common feature of higher levels of the visual nervous system.
4.1.2 Striate Cortex

Striate cortex is a thin sheet of neurons (about 2 mm thick); yet it is the single largest cortical area in primates, with approximately 200 million cells. This is more than 100 times larger than the number of LGN cells (about 1.5 million) or retinal ganglion cells (about 1 million); much complex visual processing takes place at this level. It was known as the primary cortical region for vision from early lesion studies, but its function was not understood in detail until the late 1950s.

Hubel and Wiesel applied the receptive field mapping techniques (discovered by Kuffler and Barlow) to striate cortex. Hubel and Weisel identified several different types of cortical cells having different receptive field characteristics. They classified them into three types: simple cells, complex cells and hypercomplex cells.

4.1.2.1 Simple Cells

Hubel and Wiesel used the term simple cells for certain neurons whose responses to complex stimuli could be predicted from their responses to individual spots of light. The receptive
field of a simple cell can be mapped by finding its response to a small spot of light at each position on the retina. Inside an excitatory region, a spot of light will cause the cell to fire more intensely than its spontaneous background firing rate. Within an inhibitory region, a spot of light will cause the cell to fire at less than its spontaneous rate. A simple cell’s response to a more complex pattern of stimulation can be approximated by summing its responses to the set of small spots of light that form that pattern. The approximate linearity of response is one of the most important characteristics of simple cells.

Hubel and Wiesel identified several different subtypes of simple cells. Most of them have an elongated structure; they fire most intensely to a line or an edge at a specific retinal position and orientation. Figure 4.6 (Palmer [1999], p. 152) shows some receptive field types

Figure 4.6 Receptive fields and response profiles of vertically oriented simple cells in area V1. The receptive fields (diagrams) and response profiles (graphs) for four types of simple cells are shown: light line detectors (A), dark line detectors (B), dark-to-light edge detectors (C), and light-to-dark edge detectors (D).
for simple cells. For some of them, the area of excitation is on one side and the area of inhibition is on the other side. These cells are called edge detectors because they have a stronger response to a luminance edge in the proper orientation and of the proper polarity (light-to-dark or dark-to-light). The receptive fields of other simple cells have a central elongated region being either excitatory or inhibitory. These cells respond to bright or dark lines, so they are called line detectors or bar detectors.

A new view of image processing has been developed as a result of Hubel and Wiesel’s discoveries: finding the lines and the edges in the image is an early step in spatial image processing. The line and edge detector theory also supports the atomistic view of vision discussed in Section A.3 of Appendix A; the atoms used in spatial vision are lines and edges at particular positions, orientations and contrasts. Higher-level properties, such as shapes and orientations of objects might be constructed using the local edges and lines identified by detector cells in V1. Although the correctness of this view is still an open question, it has dominated thinking about the initial stages of visual processing for several decades.

More recent studies have revealed that the receptive fields of simple cells are more complicated than Hubel and Wiesel described them. Many simple cells have additional lobes of excitation and inhibition to the sides of primary ones (De Valois and De Valois, [1988]). A wide variety of different receptive field sizes have also been found; large ones respond to coarse spatial structure and small ones respond to fine spatial structure. Russell De Valois and Karen DeValois have advanced a different interpretation of the functional role of these cells (DeValois and DeValois, [1980], [1988]).
4.1.2.2 Complex Cells

The most common cell type in striate cortex is complex cells. They also have elongated receptive fields. There are several important aspects that make them different from simple cells (Palmer, [1999]):

1. Nonlinearity. Complex cells are highly nonlinear and exhibit weak response to small stationary spots. That’s why the orientation tuning of their receptive fields cannot be mapped by measuring their response to individual spots in each retinal location.

2. Motion sensitivity. Complex cells have a high response to moving lines or edges in their receptive field. Often the motion sensitivity is specific to a direction of movement.

3. Position insensitivity. Complex cells are not very sensitive to the position of certain stimuli. Small differences in the location of a bar, for example, will not change their response rates.

4. Spatial extension. Complex cells tend to have larger receptive fields, on average, than simple cells.

Hubel and Wiesel asserted that complex cells were constructed by integrating the responses of many simple cells, as shown in Figure 4.7 (Palmer [1999], p. 154)

4.1.2.3 Hypercomplex Cells

Hubel and Wiesel distinguished a third type of striate cell: the hypercomplex cells. They have even more selective receptive fields than complex cells. As an important characteristic of these cells, we note that extending a line or edge beyond a certain length causes them to fire
Figure 4.7 Possible neural wiring of a complex cell from simple cells. Complex cells receive input from several simple cells whose receptive fields have the same orientation but different positions.

less vigorously than they do to a shorter line or edge; that’s why they are often called end-stopped cells. Researchers now believe that hypercomplex cells are end-stopped simple or complex cells.

4.2 Psychophysical Channels

The functional interpretation of the cells discovered by Hubel and Wiesel is subject of debate: what are these cells doing? As we mentioned in Section 4.1.2, the line and edge detector hypothesis would offer one possible answer to the controversy over their functional significance. There is an alternative hypothesis that suggests a different view of spatial processing in area V1.

The second approach to image-based processing was proposed within a branch of sensory psychology known as psychophysics.

“Psychophysics is the study of quantitative relations between people’s conscious experiences (their psyche) and properties of the physical world (physics) using behavioral methods. Calling the method “behavioral” indicates that psycho-physicists, unlike physiologists, do not record electrical events in neurons or directly measure other aspects of neural activity. Instead, they measure people’s performance in specific perceptual tasks and try to infer something about underlying mechanisms from behavioral measurements.” (Palmer, [1999]).
The spatial frequency theory dominates psychophysical theories of spatial vision. This theory can explain important results from psychophysical experiments.

4.2.1 Spatial Frequency Theory

The ability of men and other animals to perceive the details of objects and scenes is determined to a large extent by how well their visual system can discern contrasts: the differences in brightness of adjacent areas. The size of the visual image on the retina also plays an important role in the perception of detail. We all know from experience that as an object recedes from us and becomes smaller, details with low contrast become difficult to perceive. The reason for this loss in contrast perception is not that the relative brightness of adjacent areas changes but rather that the visual system is less sensitive to contrast when the spacing of the contrast areas decreases. If the spacing of the contrasting areas is regular, it can be called a spatial frequency. It is a remarkable fact that the visual system is much more sensitive to contrast at certain spatial frequencies than it is to contrast at other spatial frequencies, just as the ear is more sensitive to certain frequencies of sound than it is to others.

Like the line and edge detector theory of image processing, the spatial frequency theory of image processing is based on an atomistic assumption: that the representation of any image, no matter how complex, is an assemblage of many primitive spatial “atoms”. The primitives of spatial frequency theory are spatially extended patterns called sinusoidal gratings: a two-dimensional pattern whose luminance varies according to a sine wave over one spatial dimension and is constant over the perpendicular dimension. Figure 4.8 (Palmer [1999], p.159) shows examples of different sinusoidal gratings. The graph shows the luminance profile of the sinusoidal grating.
The simplest sound signal is a pure sine wave. In vision the equivalent is a grating pattern whose brightness varies in a simple sinusoidal manner. This one is called a sinusoidal grating because the intensity of its light and dark bars changes gradually, in a sinusoidal fashion. Each primitive sinusoidal grating can be characterized completely by four parameters: its spatial frequency, orientation, amplitude and phase. In Figure 4.8 we observe how these parameters change the grating.

Figure 4.8 Sinusoidal gratings. (A) Standard grating, (B) Grating of a lower spatial frequency, (C) Grating of a different orientation, (D) Grating with a different amplitude, (E) Grating with a different phase

1. The spatial frequency is usually specified in terms of the number of light/dark cycles per degree of visual angle, a quantity that varies inversely with stripe width (Campbell, [1974]). In Figure 4.8, grating B has a higher spatial frequency than the other gratings.
2. The orientation of the grating refers to the angle of its light and dark bars as specified in degrees counterclockwise from vertical. In Figure 4.8, grating C has a horizontal orientation; the others have vertical orientation.

3. The amplitude (or contrast) of the grating refers to the difference in luminance between the lightest and darkest parts, which corresponds to the difference in height between the peaks and the valleys in its luminance profile. Contrast is specified as a percentage of the maximum possible amplitude difference, so 0% contrast is a uniform gray field and 100% contrast varies from the brightest white to the darkest black. In Figure 4.8, grating D has a lower amplitude than the other gratings.

4. The phase of a grating refers to the position of the sinusoid relative to some reference point. Phase is specified in degrees, such that a grating whose positive-going inflection point is at the reference point is said to have a phase of 0° (called sine phase), one whose peak is at the reference point has a phase of 90° (cosine phase), one whose negative-going inflection point is at the reference point has a phase of 180° (anti-sine phase), and one whose valley is at the reference point has a phase of 270° (anti-cosine phase). In Figure 5.8, grating E differs from others in its phase.

4.2.1.1 Fourier Analysis, as a Method of Decomposing Images

It might seem odd to consider sinusoidal gratings as primitives or atomic elements for spatial vision. After all, we do not consciously experience sinusoidal gratings when we look at naturally occurring scenes. If conscious perception of visual elements were a necessary condition, the bars and edges would be preferred to sinusoidal gratings. At least we see bars and edges in natural scenes. However, we do not have a reason to suppose that primitive elements in early spatial vision need to be conscious.
There is a good theoretical reason for choosing sinusoidal gratings as primitives, but it is a formal mathematical reason rather than an experiential one. The rationale is based on a well-known and widely used mathematical result called Fourier’s theorem. Fourier analysis is a method used to analyze any two-dimensional luminance image into the sum of a set of sinusoidal gratings that differ in spatial frequency, orientation, amplitude and phase. In Figures 4.9 and 4.10 (Palmer [1999], p.161-162), we can see how sinusoidal gratings can be combined to form more complex images. In Figure 4.9 a series of sinusoidal gratings of the same orientation at spatial frequencies of $f$, $3f$, $5f$, … are added together in the proper amplitude and phase relationship to obtain a square wave that has sharp edges. Figure 4.10 shows how two such square waves at different orientations can be added together to produce a plaid pattern.

Fourier analysis is not limited to these simple, regularly repeated patterns. It can be applied to complex images of objects, people and even whole scenes. We can identify what kind of spatial information is carried by different ranges of spatial frequencies. Figure 4.11 (Palmer [1999], p.163) shows a picture of Groucho Marx together with two different versions of it that contain only low and high spatial frequencies, respectively. We notice that low spatial frequencies in the middle picture carry the coarse spatial structure of the image (the large black and white areas), whereas the high spatial frequencies in the right picture carry the fine spatial structure (the sharp edges and small details).

The Fourier analysis of an image results in two parts: the power spectrum and the phase spectrum. The power spectrum determines the amplitude of each constituent grating at a particular spatial frequency and orientation, whereas the phase spectrum specifies the phase of each grating at a particular spatial frequency and orientation. The original image can be reconstructed by adding up all of these gratings at the proper phases and amplitudes. Thus,
Figure 4.9 Constructing a square wave by adding sinusoidal components. (A) A grating at the fundamental frequency \( f \) of the square wave together with its luminance profile. (B) A grating at the third harmonic \( 3f \) with one-third the amplitude is shown. Adding these two gratings results in the grating and luminance profile in C. Adding the fifth harmonic \( 5f \) at one-fifth the amplitude gives the result shown in D. Adding all the odd harmonics in the proper amplitudes and phases gives the square wave shown in E.

Fourier analysis is a general method of decomposing complex images into primitive components. Fourier analysis can be “inverted” through the process called Fourier synthesis so that the original image can be reconstructed from its power and phase spectra. The invertibility of Fourier analysis proves that these spectra contain all the information in the original image.
A

B

C

Figure 4.10 Constructing a plaid grating by adding square wave gratings at different orientations.

Figure 4.11 Spatial frequency content of a complex image. The picture of Groucho Marx on the left has been decomposed into its low-frequency information (middle) and high-frequency information (right). Low frequencies are responsible for the global pattern of light and dark; high frequencies carry the local contrast information at the edges of objects.

4.2.1.2 Using the Fourier Analysis to Evaluate Image Quality

A function of two variables has a Fourier transform that is a function of two variables as well. These latter two variables can be taken to equal the spatial frequency $f_x$ (the number of
cycles per unit distance in the direction perpendicular to the bars) and orientation $\rho$ (the degrees of rotation from vertical) of a sinusoidal component. Then the amplitude characteristic of the two-dimensional Fourier transform at any particular pair $(f_s, \rho)$ will tell us how much of a sinusoid at that spatial frequency and orientation we would need to use in adding up sinusoids to form the original function. The phase characteristic will tell us in what phase that sinusoid will need to be.

Rather than expressing these two variables as spatial frequency and orientation, it is often more convenient to express them as “horizontal” and “vertical” spatial frequencies $f_x$ and $f_y$ - the frequency variables corresponding to $x$ and $y$ of the original function. The relationship between the pair of variables $f_s$ and $\rho$ (spatial frequency and orientation) and the pair $f_x$ and $f_y$ (horizontal and vertical spatial frequencies) is straightforward: spatial frequency and orientation are the polar coordinates corresponding to Cartesian coordinates $f_x$ and $f_y$. To convert from $(f_x, f_y)$ to $(f_s, \rho)$ or vice versa:

$$\rho = \arctan \frac{f_y}{f_x} \quad \text{and} \quad f_s = \left(f_x^2 + f_y^2\right)^{1/2}$$

$$f_y = f_s \times \sin \rho \quad \text{and} \quad f_x = f_s \times \cos \rho$$

In general, for any grating the horizontal spatial frequency is the frequency along any horizontal line and the vertical spatial frequency is the frequency along any vertical line.

We considered a set of medical images having the details visible as original images. We applied blurring and sharpening operations for each “good” image. We applied the Fourier transform (3.2-16) and the log transformation ((2.2-2) with $c=1$). After scaling the results, we obtained the images presented in Figure 4.12, Figure 4.13 and Figure 4.14. The
low-frequency component is displayed in the middle. The horizontal and vertical components of the spatial frequency increase as we depart from center. We notice that, for the blurred image, the low-frequency component is prevailing. The sharp version of the image has a better local contrast, so the high-frequency component is emphasized in the corresponding image (Figure 4.14).

![Figure 4.12 The magnitude of the Fourier transform for a clear medical image](image)

In other experiments, we used the Discrete Cosine Transform (DCT), as a real valued-to-real valued transform, that gives us more information about the image frequency content than the magnitude of the Fourier transform coefficients. We computed average values of the spatial frequency’s amplitudes over components of the DCT matrix for the whole image. The plot contained in Figure 4.15 shows that a sharpened image has a high-frequency component well-represented in comparison with the original image or its blurred version.
Figure 4.13 The magnitude of the Fourier transform for the blurred version of the same medical image.

Figure 4.14 The magnitude of the Fourier transform the sharpened version of the same medical image as before.
Figure 4.15 Average values of the amplitudes of spatial frequency, computed over high-frequency components of the DCT coefficient matrix for a normal image (blue), its blurred (red) and sharpened (green) versions.

4.2.1.3 Spatial Frequency Channels

The spatial frequency theory of image-based vision proposes that early visual processing can be understood in terms of a large number of overlapping psychophysical channels at different spatial frequencies and orientations.

The concept of a psychophysical channel is a hypothetical mechanism in the visual system—whose physiological substrate is not yet known—that is selectively tuned to a limited range of values within some continuum. Each channel is defined by the spatial frequency and orientation of the gratings which are maximally sensitive.

The spatial frequency approach to image processing states that the visual system consists of many overlapping channels selectively tuned to different ranges of spatial frequencies and orientations. The spatial frequency theory of vision was launched by British
psychophysicists Colin Blakemore and Fergus Campbell (Blakemore and Campbell, [1969]). They reported the results of an experiment that supported the existence of spatial frequency channels in vision. The goal of the experiment was to prove that when people viewed a sinusoidal grating for a long time, their visual systems adapt selectively to gratings at the presented orientation and frequency but not others, as measured by psychophysical techniques.

4.2.1.4 Contrast Sensitivity Functions

Blakemore and Campbell studied the effects of adapting an observer to a particular spatial frequency grating by measuring their sensitivity to such gratings both before and after adaptation. The standard measurement of how sensitive observers are to gratings at different frequencies is called the contrast sensitivity function (CSF). The lowest contrast at which the observer can detect the difference between sinusoidal grating and a uniform gray field is a measure of the contrast sensitivity of a particular observer; in other words, contrast sensitivity is the threshold at which a very low-contrast grating stops looking like a uniform gray field and starts to look striped. This threshold is measured for gratings at many different spatial frequencies from low (wide fuzzy stripes) to high (narrow fuzzy stripes).

The method of adjustment is the fastest procedure for measuring contrast thresholds. Each subject controls the contrast of the grating at a particular spatial frequency on a monitor to the value that allows him to detect its stripped appearance. This adjustment procedure is repeated for many gratings at different spatial frequencies. The results of such an experiment can be described in a graph in which the contrast at a threshold is plotted as a function of spatial frequency, as shown in Figure 4.16 (Palmer [1999], p. 164). The reciprocal of this graph-made by flipping it upside down-defines the contrast sensitivity function over the
spatial frequency continuum, since threshold is high when sensitivity is low and vice versa.

The CSF obtained using this procedure typically looks like the one shown in Figure 4.16 B.

![Figure 4.16 Contrast sensitivity functions (Campbell [1974]). (A) Minimum contrast at threshold plotted as a function of spatial frequency. (B) Contrast sensitivity plotted as a function of spatial frequency, the inverse of graph A. (C) Contrast sensitivity functions for adult humans, macaque monkeys, and infants at several ages.]

We can determine the overall shape of our own CSF by looking at Figure 4.17 (Palmer [1999], p. 164). It contains sinusoidal stripes of increasing spatial frequency along the horizontal axis and decreasing contrast along the vertical axis. At threshold contrast, our ability to detect the gratings disappears; our sensitivity to gratings at the given spatial frequency is the height at which we no longer see the stripes but just a gray background. If we hold the image about 30 inches from our eyes, the outline of the striped portion of Figure 4.17 should look very much like the CSF plotted in Figure 4.16B.

The CSF shows that people are most sensitive to intermediate spatial frequencies at about 4-5 cycles per degree of visual angle. Figure 4.16C presents other CSFs for comparison. We notice that babies are much less sensitive at birth, especially at high frequencies (Atkinson, Braddick, Moar [1977]). If the CSF is measured under low-light conditions in humans, sensitivity to all frequencies drops significantly, especially at the highest frequencies. This means that at night, when just rods are active, human vision lacks
the high acuity available in daylight. This is primarily because there are no rods in the fovea, the area of greatest visual acuity under high-lighted conditions.

4.2.1.5 Selective Adaptation of Channels

After Blakemore and Campbell measured each subject’s CSF, they asked the subject to scan back and forth over a grating of a particular spatial frequency, to evaluate the subject’s adaptation. Then they measured again thresholds at each spatial frequency. The extended exposure to the grating determined the time it took for the visual system to adapt. The visual system became less sensitive after the prolonged viewing experience, but only near the particular spatial frequency and orientation of the adapting grating. The postadaptation CSF,
showed as a dotted curve in Figure 4.18 (Palmer [1999], p. 165) indicates how selective the change in sensitivity is for the spatial frequency of the adapting grating. Test gratings having much lower or higher frequencies were not affected by adapting to the grating. Figure 4.18 B contains the plot of the difference between the original CSF and the adapted CSF. Each psychophysical channel adapts to a degree that reflects its sensitivity to the adapting stimulus.

A theory based on spatial frequency channels explains the results of this experiment. According to this theory, the broad-band CSF originally obtained is an expression of the combined contribution of many overlapping narrow-band channels, each channel being sensitive to a different range of spatial frequencies, as showed in the upper graph in Figure 4.19 (Palmer [1999], p.166). When the adapting grating is observed for an extended period,
the channels that are sensitive to that spatial frequency get tired and respond less vigorously. The lower graph in Figure 4.19 displays a lowered sensitivity in channels near the frequency of the adapting grating. The overall CSF after adaptation has a “notch” around the adapting grating because, after adaptation, the channels that cause the perception of the gratings in this frequency range are less sensitive to the same or similar stimuli.

![Graph showing contrast sensitivity before and after adaptation](image)

**Figure 4.19** The multiple spatial frequency channels hypothesis. The contrast sensitivity function (dashed curve in the upper graph) is hypothesized to be the overall envelope of many overlapping spatial frequency channels (solid curves). The dip in contrast sensitivity following adaptation (dashed curve in the lower graph) is assumed to occur due to selective adaptation by channels near the adapting frequency.

### 4.2.2 Physiology of Spatial Frequency Channels

Psychophysical channels are hypothetical mechanisms determined from behavioral measures rather than directly observed biological mechanisms of the nervous system. These channels are information processing constructs at Marr’s algorithmic level of description rather than at the implementation level; if they are real, they must be implemented in the visual nervous system. The questions are: how and where are the psychophysical channels implemented?
To find the answers to these questions, a second theory about the function of the cells discovered by Hubel and Wiesel in striate cortex has been proposed. There is substantial evidence that these cells may be responsible for a local spatial frequency analysis of incoming images. The analysis performed by these cells is local because the receptive fields of striate cells are spatially limited to a few degrees of visual angle (or less in the fovea). This is restricted compared to the theoretically infinite extent of the sinusoidal gratings, the basis of Fourier analysis. It is more restricted than the large grating stimuli (10’ or more) used in psychophysical studies. A local spatial frequency analysis can be performed through many small patches of sinusoidal gratings that decrease in intensity with distance from the center of the receptive field, as shown in Figure 4.20 (Palmer [1999], p. 170). This type of receptive field structure –called a Gabor function is constructed by multiplying a global sinusoidal grating by a bell-shaped Gaussian envelope. Figure 4.20 A contains the one-dimensional luminance profile of this function and a display indicating how light intensity varies over space according to a Gabor function.

As mentioned in Section 4.1.2, Russell De Valois, Karen De Valois and their colleagues have mapped the receptive fields of V1 cells and have discovered multiple lobes of excitation and inhibition. Such receptive fields look similarly to the profiles of Gabor functions. To emphasize the connection between these cells and local spatial frequency theory, De Valois, Albrecht and Thorell measured the spatial frequency tuning of both simple and complex cells (De Valois, Albrecht and Thorell, [1982]). Many cells were sharply tuned to small frequency ranges, as would be expected if they were the biological implementations of local spatial frequency channels in the brain.
Figure 4.20 Gabor functions. A Gabor function is obtained by multiplying a sinusoidal function by a Gaussian function as in A. The resulting luminance pattern can be seen in B.

Some cortical cells are very sharply tuned and others are broadly tuned (De Valois et al. [1982]). In general, cells that are tuned to high spatial frequencies have narrower tuning than the cells that are tuned to low spatial frequencies. Simple cells tend to be more narrowly tuned than complex cells. Some cells respond to gratings that are close to a specific orientation, whereas others respond almost equally to gratings in any orientation. Cells that are broadly tuned for spatial frequency are also broadly tuned for orientation, and cells that are narrowly tuned for spatial frequency are also narrowly tuned for orientation (De Valois and De Valois, [1988]).

Although the results of the experiments were in the favor of the hypothesis that simple and complex cells in area V1 may perform a local spatial frequency analysis, this hypothesis is not universally confirmed to hold. Nevertheless, local spatial frequency theory has
facilitated important discoveries about the properties of V1 cells and is considered an alternative to the line and edge detector theory suggested by Hubel and Wiesel.

These two theories have different functional implications. Spatial frequency theory suggests that these cells do not detect lines and edges, but are filters than that decompose the image into a set of primitives. This hypothesis does not preclude the existence of line and edge detector cells in the visual system. It locates them at a higher level, where Gabor filters could specify the line or edge. The local spatial frequency theory could be compatible with the line and edge detector theory but not with the claim that these detectors are implemented in the cells of area V1.

4.3 Computational Approaches

Computational theorists have explored the nature of image processing from a number of different perspectives. The majority were interested in effective techniques for detecting image features such as edges and lines in gray-scale images. The members of the vision group at M.I.T., including David Marr, Tomasso Poggio, Ellen Hildredth, Shimon Ullman are the best known practitioners of this approach, considered a traditional one. The main purpose of their research was to obtain a computer implementation of Marr’s raw primal sketch. This group has produced computational and algorithmic descriptions of edge and line detector problems. Their results are related to Hubel and Wiesel’s conjecture that striate cortical cells are responsible for detecting edges and lines.

Although this approach dominated the research in computer vision, alternative computational views have been considered. One advocates a filtering approach to vision, based on the spatial frequency theory of early vision described in the previous section. Filtering theorists such as Adelson and Bergen (Adelson and Bergen, [1985]), Heeger
(Heeger, [1988]), Koenderink and Van Doorn (Koenderink and Van Doorn, [1976a]) and Malik (Jones and Malik, [1992]; Malik and Perona, [1990]) explored the computational advantages of operating with multiorientation, multiscale filters (such as the Gabor functions) as the spatial primitives used in higher level processes.

Another group of computational theorists, the connectionists consider a different approach to the problem of understanding how image processing works. They use computational learning techniques (e.g. back propagation) that make the neural networks “program themselves” to perform a perceptual task. The network is trained to perform the task adequately and the researchers evaluate the results. One of the networks used in this endeavor contains neuronlike units with “receptive fields” that are similar to the cells Hubel and Wiesel found in striate cortex.

4.3.1 Marr’s Primal Sketches

Visual physiologists discovered the cells that responded to bright and dark lines unexpectedly. Unlike visual physiologists, computer vision researchers knew from the start that detecting edges and lines would be an important goal in image processing. From the computer vision perspective, the first operation to be performed on the real image of a scene is to find the lines and edges present in that image. Even in Roberts’s early computer vision program, the first step was to find local edges and lines (Roberts, [1965]).

David Marr’s proposals about the structure of primal sketches influenced the computational view of image processing. In his interpretation, he outlined two primal sketches: the raw primal sketch and the full primal sketch. Marr’s ideas regarding the image-based representation in terms of primal sketches are not widely held, but they played an important role in the development of computational theories of low-level vision.
Marr’s raw primal sketch details the first step in the transition from the gray-scale image to a symbolic representation of image-based features. He formulated four different types of image features in the primal sketch: edges, lines (or bars), blobs, and terminations. These primitives are called “symbolic” because they form discrete classes of image-based features.

The raw primal sketch represents continuous information about several parameters for each image symbol (edge, bar, blob or termination): its position, size, orientation and contrast.

The full primal sketch is a more complex version of the raw primal sketch. It includes the results of different operations: linking short line segments and edges into longer ones, grouping similar elements together, determining the texture of areas in the visual field, etc.

4.3.2 Edge Detection

Of the spatial primitives of Marr, special consideration was given to the processes required to detect luminance edges.

When a three-dimensional scene of objects reflects light onto a two-dimensional surface such as the retina or an array of sensors in a video camera, the changes in luminance produced along a uniformly colored smooth surface tend to be gradual. The changes made across a transition from one surface to another are more abrupt, and they form luminance edges. These abrupt edges are important because they correspond to either a change in the reflectance of the surface (e.g., from one material to another), a change in the amount of light (e.g., due to a shadow), or a change in surface orientation relative to the light source. Based on these reasons, luminance edges are considered important image-based features.
4.3.2.1 Edge Operators and Convolution

In the early attempts to detect luminance edges automatically, researchers used different edge operators to see which ones accomplished the task well. An edge operator is a computational scheme that operates over the gray-scale values of a neighborhood of adjacent pixels in an image and computes a number that represents the likelihood that there is a luminance edge at that location in the image. Figure 4.21 (Palmer [1999], p. 173) shows several simple edge operators used to detect vertical and horizontal edges.

**First-Order Edge Operators**

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>+1</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

A. Vertical  B. Horizontal

**Second-Order Edge Operators**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>+2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

C. Vertical  D. Horizontal  E. Omnidirectional

Figure 4.21 Simple edge operators. First-order operators for vertical (A) and horizontal (B) edges compute the difference of adjacent pixels. Second-order operators for vertical (C), horizontal (D) and omnidirectional (E) edges compute differences of differences.

Edges are detected by computing the convolution of an edge operator with an image. Convolution is the mathematical operator corresponding to the operation performed by a sheet of cortical cells to an incoming image, if their receptive fields have the form of an edge operator. Each cell computes the sum of its excitatory inputs plus the sum of its inhibitory inputs (being negative, so this is subtracted) and obtains the final results as its output. The output for the entire sheet of cells regarding the image is the convolution of the cell’s receptive field (the edge operator) with the image.
The procedure of computing convolutions of an image with an edge operator is a sequential version of what a sheet of cells would compute in parallel if their receptive fields corresponded to the weighting scheme of the edge operator. Convolution is not inherently sequential, but it is implemented this way on serial computers. The visual system performs convolutions much faster by implementing them in parallel, using the connections between layers of neurons to compute the value of the convolution at each location.

Everywhere there is a vertical edge in the image, the value of the convolution is either highly positive or highly negative. Thus, extreme values for this edge operator indicate the likely presence of a vertical edge. If the convolution operation between the image and the horizontal edge operator (Figure 4.21 B) is performed, one will find that horizontal edges determine extreme output values in the corresponding locations.

These local edge operators are called first-order differential operators because they take the simple difference between adjacent pixels. They compute the slope of the luminance function along a particular direction. More complex local edge operators can be constructed by taking the difference between adjacent first-order operators to produce second-order differential operators, as we can see in Figures 4.21 C, 4.21 D and 4.21 E. Second-order edge operators work differently from first-order ones; edges are detected by zero values with extreme values flanking them. Figure 4.21 E contains a second-order edge operator that can detect vertical, horizontal, left-diagonal, and right-diagonal edges simultaneously.

4.3.2.2 The Marr-Hildreth Zero-Crossing Algorithm

Marr and Hildreth formulated a theory of edge detection within the framework of Marr’s three levels of description of an information processing system (Marr and Hildreth, [1980]). Although this algorithm is no longer considered a the state-of-the-art edge detection algorithm, having been replaced by more recent algorithms (Canny, [1986]; Deriche, [1987],
Spacek, [1985]), it is of historical interest as one of the first edge detection schemes that attempted to match its computational structure with the physiology of the mammalian visual system.

Marr and Hildreth were interested in finding some characteristics of a one-dimensional luminance edge (Figure 4.22 A). They located such an edge by finding the position of maximum slope in its luminance function. The luminance profile of a one-dimensional luminance edge can be represented as a curve, as illustrated in Figure 4.22 B.

First-order differential operators compute the slope of luminance functions. The slope of the luminance function is represented in Figure 4.22 B by the orientation of the rectangles drawn along the curve. Figure 4.22 C plots the slope function (the first derivative) as a change in intensity as a function of position. The location of the luminance edge can be determined by detecting the maximum in the first derivative of the luminance function, which corresponds to the maximum of a first-order edge operator. This approach was not computationally efficient because first-order edge operators must be computed at every possible orientation.

Marr and Hildreth proposed a more efficient algorithm to find edges: to detect zero-crossings of the second derivative of the luminance function. The second derivative of a luminance function is the slope of the first derivative of the luminance function. In the second derivative function, the position of the luminance edge corresponds to the zero value in between a highly positive and a highly negative value. The local operators used to compute the second derivative are the second-order differential operators specified earlier (Figure 4.21 C, 4.21 D and 4.21 E).

The zero-crossing of the second derivative is equivalent to the maximum of the first derivative. The second-order operators are symmetrical about their midpoint (Figures 4.21 C
Figure 4.22 First and second derivatives of luminance edges. A luminance edge (A) can be seen in a graph (luminance profile) that plots light intensity (luminance) as a function of spatial position (B). The first derivative (or slope) of this function (C) has a peak at the center of the luminance edge. The second derivative (E) has a zero crossing at the center of the luminance edge (Palmer [1999], p. 176).

and 4.21 D); thus, the second derivative of a two-dimensional image can be computed in all orientations at once by the single two-dimensional operator depicted in Figure 4.21 E and 4.23 A. This edge operator is the combination of second-order operators at vertical, horizontal, left-diagonal, and right-diagonal orientations. In continuous space, the two-dimensional second-order operator would look like Figure 4.23 B. This edge operator looks familiar, for its center/surrounded (Mexican hat) receptive field structure is similar to that of retinal ganglion and LGN cells presented in Section 4.1.1

Figure 4.24 illustrates the application of the zero-crossing algorithm to the image of a plant. Figure 4.24 A shows the gray-scale image. Figure 4.24 B shows the convolution of image A with the second-order operator in gray-scale form, so zero values are gray, positive
Figure 4.23 Discrete (A) versus continuous (B) versions of a second-order omnidirectional edge operator. The upper diagram shows the receptive field of the operators on the imaging surface. The lower graph shows its response profile across the horizontal dimension (Palmer [1999], p. 176)

values are lighter, and negative values are darker. Figure 4.24 C shows the binary version of the image in 4.24 B; zero-crossings are represented as black/white edges. Figure 4.24 D shows the zero-crossings found by the zero-crossing algorithm.

The computational theories of edge detection have to take into account an issue: luminance edges occur on different size scales. Some edges represent slow changes in intensity over broad regions of space and others are rapid changes over tiny regions. To face
Figure 4.24 Finding edges by detecting zero-crossings. The gray-scale image (part A) is first convolved with a second-order omnidirectional edge operator to obtain the image in part B, where zero is represented as neutral gray, positive values as lighter gray and negative values as darker gray. Part C shows positive values as white and negative values as black. The zero-crossings in B have been found in D to obtain the edges.

For this problem, Marr and Hildreth used second-order edge operators of three different sizes: large ones to detect low-resolution (coarse) edges, medium ones to detect medium-resolution edges, and small ones to find high-resolution (fine) edges. Figure 4.25 (Palmer, [1999], p. 178 shows a discretely sampled digital approximation of the face of Abraham Lincoln (Harmon and Julesz, [1973]). Marr and Hildreth’s algorithm detected the zero-crossings at fine (B), medium (C) and coarse (D) levels of resolution. The edges that make the Lincoln’s face distinguishable are located at low resolution levels.

For this image, few edges at high and low levels of resolution are the same. In natural images, most edges found at low levels of resolution will also be found in the same general
Figure 4.25 Edge detection at different spatial scales. The quantized image of Lincoln’s face (A) has been analyzed for edges at fine (B), medium (C), and coarse (D) levels of resolution using the Marr-Hildreth zero-crossing algorithm.

location at higher levels of resolution. Finding edges at multiple levels of resolution in the same location proves that it is likely that a real edge (rather than noise) is present.

4.3.2.3 Neural Implementation

Marr and Hildreth were interested in implementing their edge and line detection algorithm in neural hardware. They performed a detailed analysis of correspondence between their algorithm and the types of cells already known in the mammalian visual nervous system. They proposed that the convolution of the image with the second-order edge operator was accomplished by ganglion and LGN cells; these cells have receptive fields with a rotationally symmetric center/surrounded organization with the Mexican hat shape. The result of this neural information processing would be similar to Figure 4.24 B. In this representation, zero crossings of the second-order differential operator are present but have not been explicitly detected.

As mentioned in the previous section, zero-crossings in the output of the second-order operator occur between highly positive values and highly negative values. Therefore, Marr and Hildreth proposed that zero-crossings could be detected by locating these positive-to-
negative (or negative-to-positive) changes in the output of the center/surrounded receptive fields. An operator performing a logical AND on adjacent on-center and off-center geniculate cells can be used for this purpose. Such edge detector unit will fire strongly only if both LGN cells are highly active, as shown in Figure 4.26 A. Many aligned zero-crossing detectors can be combined to construct edge detectors sharply tuned to specific orientations, as illustrated in Figure 4.26 B.

Figure 4.26 A neural model for the Marr-Hildreth zero-crossing algorithm. A zero-crossing detector (A) can be obtained by taking the logical AND and the outputs of an on-center and off-center receptive field. Several units will be highly selected to edges at a specific orientation. (Palmer [1999], p. 179)

This proposal is similar to Hubel and Wisel’s conjecture about summing the outputs of LGN cells to construct “edge detector” simple cells in the striate cortex. Marr and Hildreth
propose that oriented “edge detector” simple cells in a striate cortex are actually zero-crossing detectors.

More recent, Canny proposed a more effective edge detection algorithm for computer vision using complex analyses to maximize performance in edge detection under certain conditions (Canny [1986]). Canny used first-order differential operators at several different orientations. Canny considered an optimal detector for two-dimensional luminance “step edges”, a close approximation to an idealized Hubel-Wiesel edge detector, as shown in Figure 4.27. Figure 4.28 illustrated the excellent performance of this operator. Other effective edge detection algorithms have been proposed by Deriche (Deriche, [1987]) and Spacek (Spacek, [1985]).

4.3.3 Alternative Computational Theories

Detection of image-based features such as edges, bars and blobs is not the only computational theory that attempted to explain the function of the cells discovered by Hubel and Wiesel. We have already explored the local spatial frequency analysis, but still other functions have been proposed. One alternative advocates the idea that these cells are an important link in the analysis of texture information. Another approach states that they mediate the recovery of the curvature of a surface from shading information.

4.3.3.1 Texture Analysis

Malik and Perona proposed another computational theory with implications for the possible function of the cells Hubel and Wiesel discovered (Malik and Perona, [1990]). Their theory of texture analysis concerns the process by which the visual system defines regions that differ in the statistical properties of spatial structure. Malik and Perona advocate the idea that V1
Figure 4.27 Edge operators for Canny’s algorithm. The edge operators are defined by the cross sections parallel to the edge (A) and perpendicular to the edge (B). Several oriented edge operators of this form at different orientations are shown in C.

Figure 4.28 An application of Canny’s edge algorithm. A gray-scale image (A) is shown with the edges detected by Canny’s algorithm (B).
cells provide an initial stage in the segregation of regions according to texture information.

Figure 4.29 (Palmer [1999], p. 184) demonstrates texture segregation because the difference between the concentric squares is given by their textural properties rather than in their color or overall luminance.

![Image](image.png)

Figure 4.29 Demonstration of texture analysis. The different regions in this image have the same average luminance but can be defined by their different textures. Malik and Perona’s theory of texture segmentation is based on the output of receptive fields like those found in area V1 of visual cortex.

The gist of their proposal is that many of the results on texture perception can be explained by a theory that takes into account an initial stage of convolving the visual image with receptive fields like those measured in area V1. Differences in both the orientation and scale (spatial frequency) of receptive fields are crucial in their analysis.

### 4.3.3.2 Structure from Shading

Lehky and Sejnowski advanced a different computational theory about the possible role of the cells in area V1 (Lehky and Sejnowski [1988], [1990]). They studied the structure from shading: how information about the curvature of surfaces can be derived from changes in luminance due to depth structure in the image. Figure 4.30 demonstrates the important role of shading information. The unshaded version in Figure 4.30 A looks flat and the content cannot be recognized (except for an oddly shaped blob). When shading information is added in
Figure 4.30 B, the object incorporates a 3-D quality and becomes recognizable. It is obvious that shading information is important in how one perceives this figure.

Figure 4.30 Demonstration of shading analysis. The difference between perception of the outline version (A) and the shaded version (B) is due to information about relative depth from gray-scale shading.

Lehky and Sejnowski investigated how a three-layer neural network, such that in Figure 4.31, might learn to compute the curvature of surfaces from image luminance. They used the back propagation to change the connection weights so that the hidden layer could map the input representation to the shape-from-shading representation. Lehky and Sejnowski used as the input layer the Mexican hat-shaped center/surrounded receptive fields of LGN
cells (Figure 4.2), and they specified to the output layer the characteristics needed to code the curvature of local surface regions. They employed the back propagation algorithm that adjusted the weights of the connections to and from the middle layer to map the LGN input representation to the output representation of surface curvature.

![Figure 4.31](image.png)

Figure 4.31 The architecture of a three-layer feedforward network used to learn shading analysis. There were 122 input units that simulated LGN cells with center/surrounded receptive fields, 27 hidden units, and 24 output units to represent surface curvature information.

After the neural network was trained to extract surface curvature, many of the hidden neurons produced receptive fields that looked similar to those of cells in cortical area V1. The analysis of the similarity between these receptive fields and those of real cells is suggestive. The computational structure of these simulated cells is related to the kinds of shadows and shadings one sees on various curved surfaces.

A formal mathematical analysis of structure from shading by Pentland has proved that under certain restricted conditions, a set of Gabor filters can recover the approximate shape of the surface from its projected image (Pentland, [1989]). The set of filters contained a sampling of different spatial frequencies and orientations in both sine and cosine phase at different positions. This is the set of receptive fields researchers believed to compose the cells
in area V1 (Daugman, [1980], De Valois and De Valois, [1988]). Researchers may assume that the output of these cells could be used as the basis of a later process that reconstructs shape from shading, at least for small regions of the retinal image, as Lehky and Sejnowski’s results suggested.

4.3.4 A Theoretical Synthesis

As we already mentioned, there are several theories that aim to explain the function of the cortical cells discovered by Hubel and Wiesel more than 40 years ago. These cells are assumed to have a role in image-level processing, but this role has not yet been determined, although an important amount of work has been directed at understanding them.

4.3.4.1 Local Spatial Frequency Filters

From the psychophysical perspective, these cells seem to implement the spatial frequency channels that Campbell and Robson proposed. An important difference is that each cell’s receptive field is localized in a small region of the retina rather than present over the entire visual field. Simple cells are tuned to different ranges of spatial frequency and orientation at a particular retinal position.

The observations about the spatial frequency tuning of cortical cells were not very clear in determining their functional role. According to the psychophysical view, these cells describe images in a piecewise, local Fourier analysis. Marr and Hildreth claimed that these cells implement the edge detection mechanisms at different spatial scales. Malik and Perona suggested that these cells have a role in texture analysis. Lehky and Sejnowski believed that these cells may do intermediate shape-from-shading analysis. Having all these theories about the role of the cortical cells discovered by Hubel and Wiesel makes the task of determining the real function of these cells a difficult one.
These different views are not as incompatible as they seem at the first. These descriptions of spatial processing may be appropriate for different levels of the visual system, as presented in Figure 4.32. The psychophysical view that these cells perform a local spatial frequency analysis is quite general: the cells’ receptive fields have a spatial luminance structure that is characteristic of Gabor functions—local patches of sinusoidal gratings that decrease in contrast when the distance from the center of the receptive field increases. This view is compatible with the possibility that the outputs of these cells could be involved later in detecting edges, computing shape from shading, or to segregate textures.

From this perspective, edge detectors would be constructed from the output of local spatial frequency analyzers. Edge detectors would be located at least one level beyond the cells in V1, after the results at different spatial scales have been combined. Lehky and Sejnowski’s description of surface curvature from shading information could be interpreted in a similar way. The units that are sensitive to the curvature of surfaces would exist at the level beyond the “Hubel-Wiesel” cells, seen as an intermediate representation that enable the
higher-level surface curvature analyzers to integrate the appropriate patterns of activity. This view makes the edge and surface curvature processing algorithms compatible with each other: the same intermediate representation (obtained in terms of local spatial frequency components) may be used in different ways by different higher-level processes. The fact that all of the cortical processing of visual information in primates goes through area V1 and is sent directed to other cortical areas is compatible with this view. The additional visual properties that might be computed from the output of these cells in extrastriate areas have to be determined. Recent theoretical investigations by Malik and his colleagues have suggested that the output of cells tuned to local spatial frequency components are efficient in solving computationally difficult problems such as the perception of motion (Weber and Malik, [1995]) and stereoscopic depth (Jones and Malik, [1992]).

The present view implies that the representation of image structure in V1 may be a kind of unified image-based representation as suggested by Marr, although its structure now appears to be different from his primal sketch proposal. Marr suggested that the primal sketch was a symbolic representation composed of bars, edges, blobs and terminators, V1 seems to represent the visual image in terms of the continuous output of filters at different positions, orientations, spatial scales and phases.

If this theory is sound, Marr’s assumption that the visual system “goes symbolic” very early in visual processing is not true. No categorical types of naturalistic image features would be detected so early. Having detected edges, lines, blobs and terminators in this stage could eliminate the gradual changes in luminance that reflect shading due to surface curvature and are used in depth and texture perception. A representation that preserves all the structure in the image in a more efficient form would be preferred. This is important in V1 because all further cortical processing depends on its output. If information were lost at this
crucial stage, it could not be recovered by later processes. Local spatial frequency analysis with Gabor or wavelet filters has the advantage of preserving all the information, without assuming that the visual system represents uses a set of symbolic primitives.

4.3.4.2 Exploiting the Structure of Natural Images

Why does the visual system represent image-based information in this fashion? This representation does not correspond to our conscious perception of the environment; any visual representation using Gabor filters appears to be completely unconscious. Such a representation exploits structure in natural images projected onto our retinas (Barlow [1961], [1983]).

“The term “natural images” refers to retinal images that arise from natural environments under natural viewing conditions. The term “exploiting structure” refers to eliminating redundancies so that the information from such natural images can be represented more efficiently.” (Palmer, [1999])

Researchers that explored the statistical structure of natural images proved that interpreting images in terms of the outputs of Gabor filters were more efficient than the outputs of other types of receptive field structures (Field, [1993], [1994]) and that the receptive fields of V1 cells may be optimized for extracting this information according to certain computational principles (Olshausen and Field, [1996]).

To understand the idea of statistical structure in natural images and the manner of exploiting it to create an efficient representation, we will explain the concept of the state space of a receptor array.

“Given an array composed of \( n \) receptors, each of which can represent any value within a range of luminances, every possible image that can be represented in that array corresponds to a single point in an \( n \)-dimensional space, called its state space. Each dimension of the state space corresponds to the output of one receptor (or, equivalently, the luminance of the corresponding pixel). The complete set of luminance values for every pixel in an image therefore specifies a single point in this space, and that point corresponds to a particular image that has that particular pattern.
of luminance values for its pixels. The state space therefore represents any image that can be registered by that receptor array as a single point, and the entire state space represents the set of all possible images that the array can encode.” (Palmer, [1999]).

The set of natural images represents only a small fraction of the set of logically possible images. Figure 4.33 B shows unnatural images in contrast with the natural images in Figure 4.33 A. It would be very unlikely to encounter such images in natural environments. The set of natural images is a small subset of all possible images, so it will be associated with a restricted set of locations in the state space. The key issue is how the natural images are distributed over the state space. If the subset of natural images occupies random locations, then there would be no statistical structure in natural images for the visual system to exploit. However, if natural images tend to be in restricted regions of the state space, then the visual system could use this structure to process these images more efficiently. Natural images have statistical structure. The images in Figure 4.33 B have a different statistical structure so they look obviously different from those in Figure 4.33 A and less natural.

How the visual system may exploit this structure? Two possible answers have been explored: compact coding and sparse distributed coding.

“Compact coding assumes that the output of the receptor array should be recoded so that the number of units needed to represent the image is minimized, as illustrated in Figure 4.34 A. Sparse distributed coding assumes that the receptor output should be recoded so that the number of active units is minimized, as illustrated in Figure 4.34 B.” (Palmer, [1999]).

One method that produces compact coding is principle component analysis (PCA) (Linsker, [1988], Sanger, [1989]). Principle component analysis is a method that chooses a reduced set of orthogonal vectors (called basis functions) that capture the maximum variance of the subset of points used to represent the set of natural images in the state space. These vectors are considered the dimensional axes and images are recoded.
Figure 4.33 Natural versus unnatural images. The images in A are sampled from the natural environment. They have different structure from those in B, unlikely to be encountered in natural environment. (Palmer [1999], p. 188)

Under the assumption that natural images have nonrandom statistical structure, PCA can be used to derive more compact encodings of natural images. Because natural images have this type of structure, theorists conjectured that the receptive fields of V1 cells might be learned from extensive viewing of natural images, using the compact coding. Researchers used unsupervised neural network learning algorithms to test this hypothesis. These learning algorithms determined optimal receptive fields according to the compact coding principle. The weights of the connections from the receptor array to the recoding units were adjusted so the recoding increases its compactness as learning is in progress. PCA was applied on small
Figure 4.34 Compact versus sparse coding. Compact coding (A) represents the image using the minimum number of neural units, whereas sparse coding (B) represents it using minimum number of active units. (Palmer [1999], p. 190)

patches (8×8 pixels) of natural images; the set of receptive fields is shown in Figure 4.35. Some of the receptive fields resemble those of simple cells only at the lowest spatial frequencies. Another issue is that the receptive fields learned via PCA do not simulate the local nature of receptive fields of V1 cells; they span the entire image patch with strong connections to all receptors. Although the results seem to be on the right track, they do not correspond close enough to actual receptive fields in V1 cells to offer a plausible explanation.

Another method used in recoding natural images to exploit their statistical structure is sparse coding (Field, [1993], [1994]). In a sparse coding framework, the number of active units is minimized to obtain an efficient representation.
Recent experiments that employed unsupervised learning algorithms to shape the receptive fields units to achieve sparse coding have shown promising results (Olshausen and Field, [1996]). Many natural image patches (16×16 pixels) were given to a network of 192 recoding units. The learning part adjusted the connection weights from the receptor units to the recoding units by a method that penalized dense (nonsparse) representations. Figure 4.36 shows the resulting receptive fields for the recoding units. We notice that these receptive fields show localized, oriented structure at different spatial scales. They look similar to Gabor/wavelet functions that describe V1 simple cells. As shown in Figure 4.37 A, these receptive fields contain many small and a few large receptive fields. Field (Field, [1993]) has predicted in this theory that a system based on the wavelet will exhibit this property (Figure 4.37 B).

The results of these experiments demonstrate that learning algorithms that impose two theoretical constraints can generate receptive fields like those found in area V1. The first
Figure 4.36 Receptive fields learned from exposure to natural images using sparse coding constraints. Each 16×16 pixel square represents the strength of connections to the receptor array. Light pixels correspond to excitatory connections and dark pixels to inhibitory connections. (Palmer [1999], p. 192).

Figure 4.37 Representing image structure via a pyramid of local oriented filters at different spatial frequencies. A shows the distribution of positions and orientations of receptive fields at different spatial frequency ranges (high, medium and low). B shows a similar representation in terms of Gabor filters at different positions and spatial frequencies, for a single orientation (Palmer, [1999], p. 193).

constraint specifies that the information in the image has to be preserved by the output of the recoding units. Thus, the output is sufficient to recover the original image with a high degree of accuracy. In other words, the recoding should lose as little information as possible. The redundancies in natural images are not informative so they can be removed in recoding and
the image will not be distorted. The second constraint is that the recoding be sparse. The structure of any image is represented by activity in relatively few of the units. Field has mentioned three reasons of having sparse codings in the visual system (Field, [1994]):

1. Signal-to-noise ratios. Sparse codings increase signal-to-noise ratios because only a few cells will be active and many inactive. Sparse coding concentrates activation in a relatively few cells.

2. Feature detection. Sparse codings can be useful in later recognition processes based on the detection of specific features because relatively few feature detectors will be active.

3. Storage and retrieval from associative memory. Research into associative connectionist memories has proved that sparse codings enable networks to store more memories and to retrieve them more effectively than dense codings (Palm [1980]).

Although the research on methods of capturing the statistical structure of natural images had promising results and reinforces current beliefs about the neural coding of images in simple cells of area V1, these experiments are only in the beginning. Researchers do not have a method to determine the optimal sparse code for a given subset of images, for example, and they have to devise a neurally plausible learning mechanism that implements sparse coding. However, the research confirms the hypothesis that the structure of the visual system is sensitive to the structure in natural images.
Chapter 5

Multiscale Contrast Enhancement

When we look at images, we see regions of similar texture and gray levels that combine to form objects. If the objects are small or their contrast low, we examine them at high resolutions; if they are large or high in contrast, a coarse view is required. If small and large objects (or low and high contrast objects) are present simultaneously, it can be advantageous to study them at several resolutions. This offers a motivation for multiresolution processing.

Images are often composed of features of many sizes, with no particular scale or spatial frequency. Therefore a visual system, whether natural or artificial, should offer a certain uniformity in the representation and processing of visual information over multiple scales.

Primate visual systems exhibit multiscale characteristics. The decomposition of the image into a set of spatial frequency tuned responses is critical to vision in nature, and has been found to be very useful in many artificial settings.

5.1 Image Pyramids

A powerful, but conceptually simple structure for representing images at more than one resolution is the image pyramid (Burt and Adelson, [1983]). Originally devised for machine vision and image compression applications, an image pyramid is a collection of decreasing resolution images arranged in the shape of a pyramid.

A pyramid is a multiscale representation built through a recursive method that leads naturally to self-similarity. The basic idea is presented in Figure 5.1 (Wolfram Research [2005]), which shows a Gaussian pyramid on the left and a Laplacian pyramid on the right described by Burt and Adelson. To build a Gaussian pyramid, the original image is convolved with a lowpass filter and subsampled with a factor of two; the filter-subsample operation is
repeated recursively to produce the sequence of images shown. This pyramid can be useful for operations that require access to information about low frequencies. The pyramid can be computed in a highly efficient way.

The right part of Figure 5.1 shows a Laplacian pyramid of the same image, in which a bandpass filter is used rather than a lowpass filter. A Laplacian pyramid is a complete representation of an image, in the sense that one can perfectly reconstruct the original image given the coefficients in the pyramid. The reconstruction process is straightforward: we simply interpolate (expand) each image up to the full size of the original image using an interpolation filter, and then sum all of the interpolated images.

Figure 5.1 Examples of Gaussian Pyramid (left) and Laplacian Pyramid (right)
The pyramid structure is very useful for representing images. The pyramid is built by using multiple copies of the image. Each level in the pyramid has a smaller size than the previous level. The lowest level has the highest resolution, and the highest level has the lowest resolution.

### 5.1.1 Method Description

A Gaussian-like weighting function is used to compute the predicted value for each pixel. This function is centered on the pixel itself. The weighting function is convolved with the image. The predicted values of all pixels are obtained as a result of the convolution operation. The lowpass filtered image is subtracted from the original.

Let \( g_0(i, j) \) be the original image, and \( g_1(i, j) \) be the result of applying the convolution operation to \( g_0 \). The prediction error \( L_0(i, j) \) is then given by

\[
L_0(i, j) = g_0(i, j) - g_1(i, j).
\]  

(5.1-1)

Rather than encode \( g_0 \), we encode \( L_0 \) and \( g_1 \). \( L_0 \) is decorrelated and may be represented with fewer bits than \( g_0 \). \( g_1 \) is lowpass filtered and may be encoded at a reduced sample rate. These two facts can yield data compression. Further data compression is achieved by iterating this process. We apply the convolution operation to \( g_1 \) and denote the result \( g_2 \). We compute again the difference:

\[
L_1(i, j) = g_1(i, j) - g_2(i, j).
\]  

(5.1-2)

We may apply the convolution repeatedly and obtain a sequence of two-dimensional arrays \( L_0, L_1, \cdots, L_n \) where \( L_1 \) is smaller than its predecessor by a scale of \( 1/2 \). These arrays could be stored in a pyramid data structure. The value at each node in the pyramid
represents the difference between two Gaussian-like functions convolved with the original image. This is referred as the Laplacian pyramid (Burt and Adelson, [1983]).

5.1.2 The Gaussian Pyramid

First, a lowpass filter is applied to the original image \( g_0 \). The result is image \( g_1 \) which is a “reduced” version of \( g_0 \). The process for \( g_1 \) is repeated to produce a new image \( g_2 \). At each step, the current image is convolved with a symmetric weighting function. The sequence of images \( g_0, g_1, \ldots, g_n \) is called the Gaussian pyramid (Burt, Adelson [1983]). Figure 5.2 contains a symbolic representation for the Gaussian pyramid.

![Figure 5.2 The first levels of a Gaussian Pyramid](image)

5.1.2.1 Gaussian Pyramid Generation

Suppose the image is represented initially by the array \( g_0 \), which contains \( C \) columns and \( R \) rows of pixels. Each pixel represents the light intensity at the corresponding image point by an integer \( I \) between 0 and \( K - 1 \). This image will be the bottom or zero level of the
Gaussian pyramid. The result of filtering the original image \( g_0 \), using a lowpass filter is stored in pyramid level 1. Each value within level 1 is computed as a weighted average of values in level 0 within a \( 5 \times 5 \) window. Each value within level 2, representing \( g_2 \), is then obtained from values within level 1 by applying the same pattern of weights.

The level-to-level averaging process is performed by the function REDUCE:

\[
g_k = REDUCE(g_{k-1})
\]

which means, for levels \( 0 < l < N \) and nodes \( i, j, 0 \leq i < C_l, 0 \leq j < R_l \),

\[
g_l(i, j) = \sum_{m=-2l}^{2l-2} \sum_{n=-2l}^{2l-2} w(m, n) g_{l-1}(2i+m, 2j+n)
\]

\( N \) is the number of levels in the pyramid, while \( C_l \) and \( R_l \) are the dimensions of the \( l^{th} \) level. The dimensions of the original image are appropriate for the pyramid construction if integers \( M_c, M_R \) and \( N \) exist such that \( C = M_c 2^N + 1 \) and \( R = M_R 2^N + 1 \). The dimensions of \( g_j \) are \( C_j = M_c 2^{N-j} + 1 \) and \( R_j = M_R 2^{N-j} + 1 \).

### 5.1.2.2 The Generating Kernel

The same \( 5 \times 5 \) pattern of weights \( w \) is used to generate each pyramid array from its predecessor. This weighting pattern is called the generating kernel. We will choose \( w \) separable:

\[
w_2(m, n) = w(m) \times w(n)
\]

The one-dimensional, length 5, function \( w \) is normalized

\[
\sum_{m=-2}^{2} w(m) = 1
\]

and symmetric \( w(i) = w(-i) \) for \( i = 0, 1, 2 \). An additional constraint is called equal contribution. This means that all nodes at a given level must contribute the same total weight.
(1/4) to nodes at the next higher level. If we consider \( w(0) = a, w(-1) = w(1) = b \) and \( w(-2) = w(2) = c \), equal contribution requires that \( a + 2c = 2b \). These constraints are satisfied when \( w(0) = a, w(-1) = w(1) = 1/4, w(-2) = w(2) = 1/4 - a/2 \). Figure 5.3 shows the content of the Gaussian Pyramid for \( a = 0.4 \).

![Figure 5.3](image)

Figure 5.3 depicts the images contained in the Gaussian pyramid generated for \( a = 0.4 \). The original image, with the dimension 257 \times 257 \ becomes level 0 on the pyramid. Each higher level is approximately half as large in each dimension as its predecessor.

### 5.1.2.3 Weighting Functions

Iterative pyramid generation is equivalent to convolving the image \( g_0 \) with a set of “equivalent weighting functions” \( h_i \):

\[
g_i = h_i \otimes g_0 \quad \text{or} \quad g_i(i, j) = \sum_{m=-M_i}^{M_i} \sum_{n=-M_i}^{M_i} h_i(m, n) g_0(2^i m + j2^i + n). \tag{5.1-7}
\]

The size \( M_i \) of the equivalent weighting function doubles from one level to the next, as does the distance between samples. The effect of convolving an image with one of the weighting functions \( h_i \) is to blur, or low-pass filter, the image. This is a very fast algorithm, requiring fewer computational steps to compute a set of filtered images than are required by the FFT to compute a single filtered image (Burt [1981]).
5.1.2.4 Gaussian Pyramid Interpolation

We consider a function EXPAND as the reverse of REDUCE. It expands an \((M + 1) \times (N + 1)\) array into an \((2M + 1) \times (2N + 1)\) array by interpolating new node values between the given values. EXPAND applied to array \(g_t\) of the Gaussian pyramid produces an array \(g_{t+1}\) which is the same size as \(g_{t-1}\).

Let \(g_{l,n}\) be the result of expanding \(g_l\) \(n\) times. Then

\[
g_{l,0} = g_l \quad \text{and} \quad g_{l,n} = \text{EXPAND}(g_{l,n-1}).
\]  

(5.1-8)

By EXPAND we mean, for levels \(0 < l \leq N\) and \(0 \leq n\) and nodes \(i, 0 \leq i < C_{l-n}, 0 \leq j < R_{l-n}\),

\[
g_{l,n}(i, j) = \frac{1}{2} \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m, n) \times g_{l,n-1} \left( \frac{i - m}{2}, \frac{j - n}{2} \right).
\]  

(5.1-9)

Only terms for which \((i - m)/2\) and \((j - n)/2\) are integers are included in this sum. If we apply EXPAND \(l\) times to image \(g_l\), we obtain \(g_{l,l}\), which is the same size as the original image \(g_0\). The EXPAND operation is presented in Figure 5.4.

5.1.3 The Laplacian Pyramid

The reduced image \(g_1\) may serve as a prediction for pixel values in the original image \(g_0\). To obtain a compressed representation, we encode the error image \(L_0 = g_0 - g_1\). \(L_0\) becomes the bottom level of the Laplacian pyramid. Encoding \(g_1\) in the same way generates the next level.

5.1.3.1 Laplacian Pyramid Generation

The Laplacian pyramid is a sequence of error images \(L_0, L_1, \ldots, L_N\). Each is the difference between two levels of the Gaussian pyramid. Thus, for \(0 \leq l < N\),
Figure 5.4 We can expand the subsampled image \( n \) times to recover what the image is supposed to look like in the original size.

The value at each node in the Laplacian pyramid is the difference between the convolutions of two equivalent weighting functions \( h_l, h_{i,l} \) with the original image.

We may view the Laplacian pyramid as a set of bandpass filtered copies of the image. If we denote \( L_{l,n} \) the result of expanding \( L_l \) \( n \) times, we notice that \( L_{l,1} \) is the size of the original image. The expanded Laplacian pyramid levels for the Lenna image are shown in Figure 5.5.

### 5.1.3.2 Decoding

The original image can be recovered exactly by expanding, then summing all the levels of the Laplacian pyramid:

\[
g_0 = \sum_{l=0}^{N} L_{l,l}
\]  

(5.1-11)
Figure 5.5 Image features such as edges and bars appear enhanced in the Laplacian Pyramid. Enhanced features are segregated by size: fine details are prominent in $L_{0,0}$; coarser features are prominent in the higher level images.

It is more efficient to expand $L_N$ once and add it to $L_{N-1}$, then expand this image once and add it to $L_{N-2}$, and so on until level 0 is reached and $g_0$ is recovered. This procedure simply reverses the steps in Laplacian pyramid generation. From (5.1-10) we see that

\[ g_N = L_N \text{ and for } i = N - 1, N - 2, \ldots, 0 \quad g_i = L_i + \text{EXPAND}(g_{i+1}) \]  

(5.1-12)

### 5.1.4 Using Laplacian Pyramid for Contrast Enhancement

In medical image processing, multiscale methods have been used for many purposes, e.g., in the context of segmentation (Qian, Clarke, Zheng [1995]), registration (Unser, Thenevaz and Lee [1995]), noise reduction (Weaver, Yansun, Healy and Cromwell [1991]) or compression of images(Saipetch, Ho, Panwar, Ma and Wei [1995]; Manduca [1995];Yang, Kallergi, DeVore, Lucier, Qian, Clark and Clarke [1995]). Only within the last decade, multiscale methods have been applied to contrast enhancement of medical images. Two types of
multiscale methods have been used in this context: the Laplacian Pyramid and the wavelet methods.

By contrast enhancement we do not mean image enhancement via reduction or suppression of noise, but rather contrast amplification of the structures of interest. While enhancement by means of the Laplacian Pyramid was applied to X-ray images in general, wavelet-based methods were used in the context of mammography.

The basic idea in multiscale enhancement is to decompose the image into components which represent individual details, and to improve the contrast by immediately operating on these components rather than on the original image.

In our research, the image is decomposed according to the Laplacian pyramid transform. Contrast improvement is achieved by modifying the coefficients of the Laplacian pyramid. Small coefficients represent subtle details. These are amplified to improve the visibility of the corresponding details. The strong density variations have a major contribution to the overall dynamic range, and these are represented by large coefficient values. They can be reduced without risk of information loss, and by compressing the dynamic range, overall contrast resolution will improve.

We improved medical images using the Laplacian Pyramid decomposition and modification of the transform coefficients. We built the Laplacian Pyramid decomposition for medical images, multiplied each level of the pyramid with some coefficients and reconstructed the image using the new levels stored in the Laplacian Pyramid. The result is displayed in Figure 5.6.
5.2 Contrast Enhancement by Wavelet Analysis

“Wavelet theory” is the result of a multidisciplinary effort that brought together mathematicians, physicists and engineers...this connection has created a flow of ideas that goes well beyond the construction of new bases or transforms.” (Mallat, [1998])

5.2.1 Preview

The past decade has witnessed the development of wavelet analysis, a new tool which emerged from mathematics and was quickly adopted by diverse fields in science and engineering. In the brief period since its creation (around 1987), it has reached a certain level of maturity as a well-defined mathematical discipline, with its own conferences, journals, research monographs and textbooks.

Wavelet analysis has started to play an important role in a broad range of applications, including signal processing, data and image compression, solution of partial differential
equations, modeling multiscale phenomena, and statistics. There seem to be no limits to the subjects they can be applied.

Fundamental ideas often appear at about the same time to many researchers in widely separated disciplines. The ideas behind wavelet analysis constitute an example. Most fundamental and scientific concepts also have antecedents that helped to discover the ultimately successful approach. Wavelet analysis has a rich collection of precedents that were interesting but narrowly focused; these probably helped the researchers to identify the class of problems that would benefit from wavelet analysis.

Although the Fourier transform has dominated the transform-based image processing since the late 1950s, the wavelet transform is making even easier the compression, transmission and analysis of images. The basis functions for the Fourier transform are sinusoids; wavelet transforms are based on small waves, called wavelets, of varying frequency and limited duration. The wavelets provide the equivalent of a musical score for an image; they reveal not only what notes (or frequencies) to play but also when to play them. Conventional Fourier transforms provide only the notes or frequency information; temporal information is not preserved in the transformation process.

In 1987, Mallat presented a new approach to signal processing and analysis, called multiresolution theory (Mallat, [1987]), founded on wavelets. Multiresolution theory incorporates techniques from different disciplines, including subband coding from signal processing, quadrature mirror filtering from digital speech recognition, and pyramidal image processing. As its name suggests, multiresolution theory represents and analyzes signals (or images) at more than one resolution. The advantage of this approach is obvious: features that could not be detected at one resolution may be identified at another. Although the interest of imaging community in multiresolution analysis was limited until the late 1980s, the number
of papers, thesis and books devoted to the subject has increased significantly in the last two decades.

5.2.2 Wavelet Applications

Wavelets have scale aspects and time aspects, consequently every application has scale and time aspects.

5.2.2.1 Scale Aspects

As a complement to the spectral signal analysis, new signal forms appear. They are less regular signals than the usual ones.

The wavelet techniques of regularity are applied in the domains such as:

- Biology for cell membrane recognition, to distinguish the normal from the pathological membranes;
- Metallurgy for the characterization of rough surfaces;
- Finance for detecting the properties of quick variation of values;
- In Internet traffic description, for designing the services size.

5.2.2.2 Time Aspects

For time aspects, we mention as main goals:

- Rupture and edges detection
- Study of short-time phenomena as transient processes

The application domains include:

- Industrial supervision of gear-wheel
- Checking undue noises in craned or dented wheels, and more generally in nondestructive control quality processes
- Detection of short pathological events as epileptic crises or normal ones as evoked potentials in EEG (medicine)
• Automatic target recognition

• Intermittence in physics

5.2.2.3 Wavelet Decomposition as a Whole

Many applications employ the wavelet decomposition considered as a whole. The common goals include the signal or image clearance and simplification, which are parts of de-noising or compression.

Many papers in oceanography and earth studies use wavelet decomposition techniques.

One of the most popular successes of the wavelets is the compression of FBI fingerprints.

Several thousand papers related to wavelet transforms were written within the last 15 years; that’s why the task of classifying the applications by domain becomes very difficult. Medicine is a very productive field in terms of wavelet applications. We can find studies on micro-potential extraction in EKGs, on time localization of electrical heart activity, in ECG noise removal, etc. In EEGs, a quick transitory signal is drowned in the usual one. The wavelets help to determine if a quick signal exists, and if so, can localize it. There are attempts to enhance mammograms to discriminate tumors from calcifications.

Another prototypical application is a classification of Magnetic Resonance Spectra. The researchers study the influence of the fat we eat on our body fat. The type of feeding is the basic information and the study avoids taking a sample of the body fat. Each Fourier spectrum is encoded by some of its wavelet coefficients; the most interesting of its properties can be coded using a few coefficients. The classification is performed on the coded vectors.
5.2.3 Fourier Analysis versus Wavelet Analysis

Fourier analysis breaks down a signal into constituent sinusoids of different frequencies. Also, Fourier analysis is a mathematical technique that transforms our view of the signal from time-based to frequency-based (see Figure 5.7).

For many signals, Fourier analysis is extremely useful because the signal's frequency content is of great importance. Why do we need other techniques, like wavelet analysis?

Fourier analysis has a serious drawback: time information is lost when the signal is represented in the frequency domain. When one is looking at a Fourier transform of a signal, he cannot tell when a particular event took place. If the signal properties do not change much

![Fourier Analysis](image)

Figure 5.7 Fourier analysis produces a representation of the signal in the frequency domain over time (that is, if the signal is stationary) this constraint is not very important. However, most interesting signals contain numerous nonstationary or transitory characteristics: drift, trends, abrupt changes, and beginnings and ends of events. These characteristics are often the most important part of the signal, and Fourier analysis is not suited to detecting them.

In order to correct this deficiency, Dennis Gabor proposed a technique called windowing the signal, a variant of the Fourier transform that analyzes only a small section of the signal at a time (Gabor, [1946]). Gabor’s adaptation, called the Short-Time Fourier Transform (STFT), maps a signal into a two-dimensional function of time and frequency (Figure 5.8).
The STFT can be seen as a compromise between the time-based and frequency-based views of a signal. It provides some information about both when and at what frequencies a signal event occurs. However, one can only obtain this information with limited precision, determined by the size of the window.

The drawback of STFT is that once a particular size for the time window is chosen, that window is the same for all frequencies. Many signals require a more flexible approach - one that allows the variation of the window size to determine more accurately either time or frequency.

Wavelet analysis is a windowing technique with variable-sized regions. With wavelet analysis, one can observe long time intervals for precise low-frequency information, and shorter regions for high-frequency information (see Figure 5.9). We notice that wavelet analysis does not use a time-frequency region, but rather a time-scale region.

Figure 5.9 Wavelet analysis is a windowing technique with variable-size regions

Figure 5.10 shows, comparatively, the time-based, frequency-based, and STFT views of a signal:
One major advantage afforded by wavelets is the ability to perform local analysis - that is, to analyze a localized area of a larger signal.

As an example, we may consider a sinusoidal signal with a small discontinuity as displayed in Figure 5.11. Such a signal easily could be generated in the real world, perhaps by a power fluctuation or a noisy switch.

A plot of the Fourier coefficients of this signal shows a flat spectrum with two peaks representing a single frequency. By contrast, a plot of wavelet coefficients clearly shows the exact location in time of the discontinuity (Figure 5.12).
Wavelet analysis is able to reveal aspects of data that other signal analysis techniques miss: trends, breakdown points, discontinuities in higher derivatives, and self-similarity. It can often compress or de-noise a signal without appreciable degradation.

5.2.4 What is Wavelet Analysis?

A wavelet is a waveform of effectively limited duration that has an average value of zero.

We may compare wavelets with sine waves, which are the basis of Fourier analysis. Sinusoids do not have limited duration; they extend from minus to plus infinity. Where sinusoids are smooth and predictable, wavelets tend to be irregular and asymmetric.

Fourier analysis decomposes a signal into sine waves of various frequencies. Similarly, wavelet analysis breaks up a signal into shifted and scaled versions of the original (or mother) wavelet.

By observing pictures of wavelets and sine waves, one can see intuitively that signals with sharp changes might be better analyzed with an irregular wavelet than with a smooth sinusoid (see Figure 5.13). Also, local features can be described better with wavelets that have local extent.

Wavelet analysis can be applied to one-dimensional data (signals), two-dimensional data (images) and, in principle, to higher dimensional data.
5.2.5 The Continuous Wavelet Transform

Mathematically, the process of Fourier analysis is represented by the Fourier transform:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

The results of the transform are the Fourier coefficients $F(\omega)$; these coefficients, multiplied by a sinusoid of frequency $\omega$ produce the constituent sinusoidal components of the original signal.

Similarly, the continuous wavelet transform (CWT) is defined as the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet function $\psi$:

$$C(\text{scale, position}) = \int_{-\infty}^{\infty} f(t) \psi(\text{scale, position, } t) dt$$

The results of the CWT are many wavelet coefficients $C$, which are a function of scale and position.

Scaling a wavelet simply means stretching (or compressing) it. The scale factor, denoted by $a$, is a measure of the stretching. Figure 5.14 shows the effects of the scale factor on sinusoids.

The scale factor works exactly the same with wavelets. The smaller the scale factor, the more "compressed" the wavelet.

The diagrams show that, for a sinusoid $\sin(\alpha t)$, the scale factor $a$ is related (inversely) to the radian frequency $\omega$. Similarly, with wavelet analysis, the scale is related to the frequency of the signal (Figure 5.15). A low scale is a characteristic of compressed...
wavelet, that describes rapidly changed details of the signal, corresponding to high frequency $\sigma$. Higher scales correspond to the more stretched wavelets that measure the coarser signal features; these have low frequency $\sigma$.

Shifting a wavelet simply means delaying (or hastening) its onset. Mathematically, delaying a function $f(t)$ by $k$ is represented by $f(t-k)$ (Figure 5.16).
Figure 5.16 Shifting a wavelet

Any signal processing performed on a computer using real-world data must be performed on a discrete signal. The continuous wavelet transform is also operating in discrete time. So what exactly is “continuous” about it?

The set of scales and positions at which CWT operates, distinguishes it from the discrete wavelet transform.

Unlike the discrete wavelet transform, the CWT can operate at every scale, from that of the original signal up to some maximum scale.

The CWT is also continuous in terms of shifting: during computation, the analyzing wavelet is shifted smoothly over the full domain of the analyzed function.

5.2.6 The Discrete Wavelet Transform

Calculating wavelet coefficients at every possible scale is a fair amount of work, and it generates a lot of data.

If we choose scales and positions based on powers of two so called dyadic scales and positions -then our analysis will be much more efficient and just as accurate. The discrete wavelet transform (DWT) performs such an analysis.

An efficient way to implement this scheme using filters was developed in 1988 by Mallat (Mallat, [1989]). The Mallat algorithm is a classical scheme known in the signal processing community as a two-channel subband coder (Strang and Nguyen [1996]).
This very practical filtering algorithm yields a fast wavelet transform -- a box into which a signal passes, and out of which wavelet coefficients quickly emerge. Let's examine this in more depth.

The low-frequency content contains the identity of signals and images. The high-frequency content, on the other hand, imparts flavor or nuance.

In wavelet analysis, we obtain approximations (or trends) and details (or fluctuations). The approximations are the high-scale, low-frequency components of the signal. The details are the low-scale, high-frequency components. Figure 5.17 shows the filtering process:

![Figure 5.17 The first stage of DWT](image)

The original signal, S, passes through two complementary filters (Lo_D and Hi_D in Figure 5.17) and emerges as two signals. By performing this operation on a real digital signal, we obtain twice as much data as we started with.

If the original signal S consists of N samples of data, then the resulting signals will each have N samples. The wavelet decomposition can be done in a more subtle way. We may keep only one point out of two in each of the two samples to get the complete information. This process is called downsampling. It discards every second data point. Although this operation introduces aliasing in the signal components (see Strang, Nguyen, [1996], p. 91), it turns out this issue can be resolved later in the process.

Given a signal s of length N, the DWT consists of \( \log_2 N \) stages at most. Starting from s, the first step yields two sets of coefficients: approximation coefficients \( cA_1 \), and detail
coefficients $cD_j$. These vectors are obtained by convolving $s$ with the low-pass filter $Lo_D$ for approximation, and with the high-pass filter $Hi_D$ for detail, followed by dyadic decimation, as shown in Figure 5.17.

The next step splits the approximation coefficients $cA_j$ in two parts using the same scheme, replacing $s$ by $cA_j$ and producing $cA_2$ and $cD_2$, and so on.

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components. This is called the wavelet decomposition tree (see Figure 5.19).

So the wavelet decomposition of the signal $s$ analyzed at level $j$ (as shown in Figure 5.18) has the following structure: $[cA_j, cD_j, \ldots, cD_1]$.

5.2.7 Wavelet Reconstruction

The discrete wavelet transform can be used to analyze, or decompose, signals and images. This process is called decomposition or analysis. Those components can be assembled back into the original signal without loss of information. This process is called reconstruction, or synthesis. The synthesis is performed by the inverse discrete wavelet transform (IDWT).
Where wavelet analysis involves filtering and downsampling, the wavelet reconstruction process consists of upsampling and filtering. Upsampling is the process of lengthening a signal component by inserting zeros between samples.

The choice of filters is crucial in achieving perfect reconstruction of the original signal.

It is remarkable that perfect reconstruction is possible. The downsampling of the signal components performed during the decomposition phase introduced a distortion called aliasing. It turns out that by carefully choosing filters for the decomposition and reconstruction phases that are closely related (but not identical), the effects of aliasing can be eliminated (see Strang, Nguyen, [1996], p. 347). The low- and high-pass decomposition filters (Lo_D and Hi_D), together with their associated reconstruction filters (Lo_R and Hi_R), form a system of what is called quadrature mirror filters.

As in Figure 5.20, the IDWT starts from \( cA_j \) and \( cD_j \) and reconstructs \( cA_{j-1} \). The decomposition step is inverted by inserting zeros and convolving the results with the reconstruction filters.
5.2.8 Two-Dimensional Wavelet Transform

So far, we have considered one-dimensional signals, but wavelet analysis can be done in any number of dimensions. The essential ideas, however, are revealed in two dimensions. In this section we will detail the particular aspects of 2D wavelet analysis. Many of the basic ideas are similar to the 1D case; we will not repeat similar formulas but rather focus on the ideas needed for the 2D case. We will describe applications of wavelet analysis to image enhancement.

A 2D wavelet transform of a discrete image can be performed whenever the image has an even number of rows and an even number of columns. A one-level wavelet transform of an image \( f \) is defined, using the 1D wavelet transforms already presented, by performing the following two steps:

![One-Dimensional IDWT Diagram]

**Figure 5.20 IDWT performs wavelet reconstruction**

- **Reconstruction Step**

  - **Upsample**
  - **Low-pass**
  - **High-pass**
  - **Insert zeros at odd-indexed elements.**
  - **Convolve with filter X.**
  - **Take the central part of U with the convenient length.**

- **Level j**

- **Level j-1**

- **where**

  - **cA_j**
  - **cD_j**
  - **wkeep**

- **cA_{j-1}**
1. Perform a 1-level, 1D wavelet transform, on each row of $f$; this operation will yield a new image.

2. On the new image produced in step 1, perform the same 1D wavelet transform on each of its columns.

Steps 1 and 2 could be done in reverse order and the result would be the same. A 1-level wavelet transform of an $M \times N$ image $f$ can be described as:

$$f \rightarrow \begin{pmatrix} A^1 & H^1 \\ V^1 & D^1 \end{pmatrix}$$

where the subimages $A^1$, $H^1$, $V^1$ and $D^1$ each have $M/2$ rows and $N/2$ columns.

The subimage $A^1$ is created by computing trends along rows of $f$ followed by computing trends along columns; it is an averaged, lower resolution version of the image $f$. Figure 5.21 contains the image of a woman (Barb image). Figure 5.22 contains the subimages produced by 1-level Daub 4 wavelet transform. The subimage $A^1$ appears in the upper left quadrant of the transform and it is a lower resolution version of the original image. A 1D trend computation is $\sqrt{2}$ times an average of successive values in a signal and the 2D trend subimage $A^1$ was computed from trends along both rows and columns; so each value of $A^1$ is equal to $2$ times an average of a square that contains adjacent values from the image $f$.

The $H^1$ subimage is created by computing trends along rows of the image $f$ followed by computing fluctuations (or details) along columns. Consequently, the fluctuations along columns. The fluctuations along columns can detect the horizontal edges in an image. The subimage $H^1$ appears in the upper right quadrant of the transform and emphasizes the horizontal edges. This subimage is referred as first horizontal fluctuation.
Figure 5.21 Image of a woman (Barb image)

Figure 5.22 One-level transform of an image of a woman
The subimage $V^1$ is similar to $H^1$, except that the roles of horizontal and vertical are reversed. This subimage is shown in the lower left quadrant in Figure 5.22. This image shows the vertical edges prominently. $V^1$ is referred as first vertical fluctuation.

The first diagonal fluctuation $D^1$ tends to emphasize diagonal features, because it was created from fluctuations along both rows and columns. This subimage is shown in lower right quadrant in Figure 5.22.

The fluctuation values are generally much smaller than the trend values. In the wavelet transform shown in Figure 5.22, for instance, the fluctuation subimages $H^1$, $V^1$ and $D^1$ have significantly smaller values than the values in the trend subimage $A^1$.

Similar to 1D, multiple levels of 2D wavelet transforms are defined by repeating the 1-level transform of the previous trend (approximation). Figure 5.23 shows, generically, the decomposition process at level $j$.

Similar to the 1D wavelet transform, the trend and the fluctuations at one level can be assembled to reconstruct the trend at the previous level. As we can see in Figure 5.24, the IDWT uses the trend coefficients $cA_{j+1}$ and the coefficients $cD_{j+1}^{(h)}$, $cD_{j+1}^{(v)}$, $cD_{j+1}^{(d)}$, corresponding to horizontal, vertical and diagonal fluctuations to reconstruct $cA_j$.

### 5.2.9 Using Wavelet Analysis for Contrast Enhancement

We considered a set of blurred medical images and applied wavelet analysis to enhance the appearance of edges in each of these images. In order to sharpen the edges, we used the following algorithm:

**Edge Enhancement Method**

1. Perform a wavelet transform of the original image.
2. Multiply fluctuation values by a constant larger than 1, but not the trend values.
3. Perform an inverse transform of modified image from step 2.
3. Perform an inverse transform of modified image from step 2.

Figures 5.25 show a blurred medical image on the left. We used 1-level Coiflet wavelets (the function “coif3” in MATLAB) in step 1, and multiplied the first-level fluctuation values by the constant 3 in step 2. The right part of Figure 5.25 contains the resulting image. Figures 5.26 and 5.27 show the results of using the wavelet transform and multiplying the fluctuation values by constants 10 and 20, respectively.

Figure 5.23 Two-dimensional DWT on level $j$

Comparing the two images of a human bone in Figure 5.25, we can see that the edge enhanced image is a sharper image than the original. Some details can be observed more
clearly in the enhanced image, such as the boundaries of the bone in the central part of the image.

A related procedure would be to perform a multiple level transform in step1, and to use different constants at each fluctuation level. Figure 5.28 shows a blurred medical image on the left and the result of performing the multiple-level Haar wavelet transform and multiplying the fluctuation values by constants greater than 1 on the right. We observe that, by using the edge enhancement method on different levels, different parts of the bone have become more distinguishable.

Figure 5.24 Two-dimensional IDWT on level $j$

5.2.10 Quantitative Measures of Error

There are several ways of measuring the amount of error between a noisy image and the original image. All of these measures aim to prove that noise removal is effective.
Figure 5.25 Improving a blurred medical image with multiplier=3

Figure 5.26 Improving a blurred medical image with multiplier=10
Figure 5.27 Improving a blurred medical image with multiplier=20

Figure 5.28 Improving a blurred medical image using multiple-level wavelet analysis
One measure is Root Mean Square Error (RMS). The Root Mean Square Error of the
noisy image \( g \) compared with the original \( f \) is defined as:

\[
RMS \quad Error = \sqrt{\frac{\left( f_{1,1} - g_{1,1} \right)^2 + \left( f_{1,2} - g_{1,2} \right)^2 + \cdots + \left( f_{M,N} - g_{M,N} \right)^2}{M \times N}}
\]

We computed the RMS for Figures 5.25, 5.26 and 5.27 (see Tables 5.1, 5.2 and 5.3). These
values show improvement but do not seem to reflect the increase of the contrast in the
reconstructed image versus the blurred image.

Another measure is the relative 2-norm difference, \( D(f, g) \) defined as:

\[
D(f, g) = \sqrt[2]{\sqrt{\left( f_{1,1} - g_{1,1} \right)^2 + \left( f_{1,2} - g_{1,2} \right)^2 + \cdots + \left( f_{N,M} - g_{N,M} \right)^2}}
\]

where \( f \) is the original image and \( g \) is a noisy image. The errors calculated using \( D \) for the
Figures 5.25, 5.26 and 5.27 are summarized in Tables 5.1, 5.2 and 5.3 for different values of
the constant that multiplies the first-level fluctuation values. Again, as with RMS error, these
values show improvement but this measure of error \( D \) does not seem to reflect the
enhancement of the contrast in the resulting image.

| Table 5.1 Error measurements for Figure 5.25 (constant=3) |
|---------------------------------|-----------------|-----------|----------|
| Image                          | RMS             | \( D(f, g) \) | SNR      | \( \|f - g\|_\infty \) |
| Blurred Image                  | 10.5885         | 0.1246     | 18.0881  | 3.7047    |
| Enhanced Image                 | 10.6053         | 0.1248     | 18.0744  | 3.7587    |

| Table 5.2 Error measurements for Figure 5.26 (constant=10) |
|---------------------------------|-----------------|-----------|----------|
| Image                          | RMS             | \( D(f, g) \) | SNR      | \( \|f - g\|_\infty \) |
| Blurred Image                  | 10.5885         | 0.1246     | 18.0881  | 3.7047    |
| Enhanced Image                 | 10.8270         | 0.1274     | 17.8947  | 4.1661    |
Table 5.3 Error measurements for Figure 5.27 (constant=20)

<table>
<thead>
<tr>
<th>Image</th>
<th>RMS</th>
<th>$D(f, g)$</th>
<th>SNR</th>
<th>$|f - g|_w$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blurred Image</td>
<td>10.5885</td>
<td>0.1246</td>
<td>18.0881</td>
<td>3.7047</td>
</tr>
<tr>
<td>Enhanced Image</td>
<td>11.5529</td>
<td>0.1360</td>
<td>17.3310</td>
<td>5.0296</td>
</tr>
</tbody>
</table>

A third measure, commonly used in image processing, is Signal to Noise Ratio (SNR). If $f$ is the original image and $g$ is a noisy image, then the SNR measure of error is defined by

$$SNR = 20\log_{10}\left[1/D(f, g)\right]$$

One reason for using SNR is that human visual systems respond logarithmically to light energy. The results of applying the SNR measure to our blurred medical image and the result of using the edge enhancement method are summarized in Tables 5.1, 5.2 and 5.3 for different values of the constants as multipliers for step 2. An increase in SNR represents a decrease in error. But, as with other measures discussed, the SNR also does not seem to accurately reflect the increase in contrast in the images displayed on the right part of Figures 5.25, 5.26 and 5.27 respectively.

The measures of error enumerated above present deficiencies for the perceptions of our visual systems. It is recognized that they have remained in use, despite their deficiencies, because they have been suitable for the type of mathematics used in image processing. However, because of the numerous applications of wavelets in image analysis, a measure of error that fits well into a wavelet analysis framework has been proposed. This measure, denoted by $\|f - g\|_w$, is defined using a given wavelet transform. Suppose that $\{\hat{f}_{j,k}\}$ are the wavelet transform values for the image $f$ using one of the Daubechies wavelet transforms, and suppose that $\{\hat{g}_{j,k}\}$ are transform values for the image $g$ using the same wavelet transform. The quantity $\|f - g\|_w$ is defined by
\[ \| f - g \|_s = \frac{\left| \hat{f}_{1,1} - \hat{g}_{1,1} \right| + \left| \hat{f}_{1,2} - \hat{g}_{1,2} \right| + \cdots + \left| \hat{f}_{N,M} - \hat{g}_{N,M} \right|}{M \times N} \]

For the Figures 5.25, 5.26 and 5.27, we used a one-level Coiflet wavelets ("coif3" in MATLAB) to compute \( f - g \). The results are also displayed in Tables 5.1, 5.2 and 5.3 for different values of the constants that multiply the first-level fluctuation values. These results show the improvement that can be observed in these figures.
Chapter 6

Summary

6.1 Conclusion

In this dissertation research different approaches to image enhancement in the spatial and frequency domain were explored. Spatial domain refers to the image plane, and this category is based on direct manipulation of the pixels in the image. Frequency domain techniques modify the Fourier transform of the image.

This dissertation research presents an image enhancement method that uses multiscale approaches for contrast manipulation. The method is based on the Laplacian Pyramid and 2D wavelets. The basic idea in multiscale enhancement decomposes the image into components that represent individual details, and these results are used to improve the contrast with these components rather than operating on the original image. In this research, the image is decomposed with the Laplacian pyramid transform and the wavelet transform.

A pyramid is a multiscale representation built through a recursive method that leads naturally to self-similarity. To build a Gaussian pyramid, the original image is convolved with a lowpass filter and subsampled with a factor of two; the filter-subsample operation is repeated recursively to produce a sequence of images. To obtain a Laplacian pyramid of the same image, a bandpass filter is used rather than a lowpass filter. A Laplacian pyramid is a complete representation of an image, in the sense that one can perfectly reconstruct the original image given the coefficients in the pyramid. The reconstruction process is straightforward: each image is expanded up to the full size of the original image using an interpolation filter, and then all of the interpolated images are summed.

Contrast improvement is achieved by modifying the coefficients of the Laplacian pyramid. Small coefficients represent subtle details. These are amplified to improve the
visibility of the corresponding details. The strong density variations have a major
collection of the different frequencies in an image. In particular, the plots for human visual frequency indicate that some frequencies are more
visible than the others, and some are not important at all. Removing certain frequencies can
help emphasize the others (keeping the total image “energy” the same), and improving the
quality of the image.

The spatial frequency theory and the Fourier analysis are methods used to analyze any
two-dimensional luminance image into the sum of a set of sinusoidal gratings. Fourier
analysis can be applied to complex images of objects, people and scenes; it allows us to
identify what kind of spatial information is carried by different ranges of spatial frequencies. The Fourier and Discrete Cosine Transforms were employed to evaluate image quality. Based
on statistical measures (such as medians computed on sets of selected coefficients), we were
able to differentiate good images from blurred or extremely sharp images.

For optimizing the method, a set of medical images having visible details was used. Based on these, we have constructed and quantified an ideal image frequency profile, i.e. a
frequency profile that corresponds to the most “balanced-natural” image.

The effects of the most common artifacts (such as blurring, noise) on frequency
content of each image were studied. The multiscale decomposition corresponding to a blurred
or sharp medical image and that corresponding to an “ideal” image were compared (using
statistical parameters). Based on this comparison, the values of the coefficients to be applied
to the components of the blurred or extremely sharp image in different frequency bands were
computed. For each image, a Laplacian Pyramid representation was built and multiplied at
each level of the pyramid with pre-selected coefficients and finally, we have reconstructed
the image using the new levels stored in the Laplacian Pyramid. Thus, in this study, we have
demonstrated that Laplacian Pyramid can be used to produce distinguishable image
frequency profiles and have applied them for contrast enhancement of the medical images. To
the best of our knowledge, this approach is unique to this dissertation research. Also, we
employed discrete wavelet transforms to decompose medical images obtained with different
modalities (such as MRI, CT, X-ray, ultrasound). We presented a comparison of the results of
enhancing MR images using both Laplacian Pyramid and wavelet transforms.

Appendix A presents briefly the overall structure of the part of the nervous system
known to be involved in processing visual information; also, it enumerates classical theories
of vision, information processing theory and stages of visual perception. In order to devise a
mathematical method to assess image quality, we have to understand how the human visual
system interprets the information contained in natural images.

Frequently, a given enhancement task will require application of several
complementary enhancement techniques in order to achieve an acceptable result. We
illustrated in several examples how to combine different approaches developed in the spatial
or frequency domain to address this difficult enhancement task.
6.2 Implementation and Future Work

The research in this dissertation has been focused on the development of algorithms to enhance the contrast in medical images. The enhanced image is intended to assist a radiologist in a more accurate reading of the image for diagnostic purposes.

As mentioned in Chapter 2, researchers have not developed a general theory of image enhancement. The visual evaluation of image quality can and often does differ from viewer-to-viewer. One of the tasks in this research was to determine the profile of the “ideal” brain image. To do this, a diverse set of brain images was assembled and used with the intent of developing this profile. Different people were asked to evaluate this assembled diverse set of brain images. Each person was to rate the image quality of each of these brain images. Each person was to assign a numerical grade in the interval \([-5, 5]\) to classify each image as being good, blurred or sharp. A blurred image was to be assigned a score in the interval \([-5, 0)\). An extremely blurred image was to be assigned -5. A “perfect” image was to be assigned a 0-value. A sharp image from this set was to be assigned a value in the interval \((0,5]\). The ratings for the diverse set of brain images were accumulated from each person. Then the accumulated ratings were reviewed to determine the “ideal” brain image. This review of the accumulated ratings led to an immediate result: some viewers would rate an image as good (the most “balanced-natural” image) and other observers would rate the same image as being blurred or being sharp. Our finding was that each person’s classifications for this diverse set of brain images was essentially unique and differed from the other’s classifications. The conclusion leads to the assumption that each radiologist will have to decide if the “medical image of interest” needs to be made to look “better.”

If the initial image is affected by noise, the radiologist should have an option to denoise the image, as a processing stage. Noise removal is an essential element of image processing. One reason for this is that many images are acquired under less than ideal
conditions and consequently are contaminated by significant amounts of noise. This is the case with medical images. Another reason is that several important image processing operations, such as contrast enhancement, histogram equalization, and edge enhancement work much better if the random noise is removed.

There is a legal liability exposure if a medical image has been changed: the radiologist can temporarily modify a medical image to achieve an image that is more clear and allows a more accurate analysis and/or diagnosis. It is necessary that only the radiologist the user intervene during this process. For a particular image, the radiologist will decide if an image needs improvement or not. The deblurring operation can be applied repeatedly, as long as the viewer continues to be satisfied with the result. If noise was introduced during a deblurring operation, the result has also to be denoised.

We can describe the enhancement procedure using the pseudocode language as follows:

Observe the input image $I_0$.

If $I_0$ is acceptable, exit

else if $I_0$ is blurred, then $I_1 = DEBLUR(I_0)$

else $I_1 = DENOISE(I_0)$;

End If;

Do While ($I_1$ needs improvement)

If $I_1$ is blurred, then $I_2 = DEBLUR(I_1)$

Else $I_2 = DENOISE(I_1)$;

$I_1 = I_2$;

End While;
MATLAB was used for the implementation of the filter functions. After careful optimization, the program produces the enhancement almost instantly. It is so fast it is suitable for real-time applications, i.e. a radiologist can analyze each improved image in seconds. We have tested our enhancement methods on MR brain and bone images, but we expect the method to work on images obtained with other modalities (e.g., CT, PET, X-ray, ultrasound, etc.). More testing on the images acquired using modalities different from MR would be required. However, our method is not applicable to every distorted image. For example, we cannot recover useful information from an image that has had its frequency components drastically altered, such as a “totally black” image.

In a telecommunication connected medical environment, it is important to understand the role of a Picture Archiving and Communications System (PACS) is to store and display the digital images acquired for the radiology department. The PACS would have the information and images related to this research. This includes the images themselves along with the information relating to how the image was acquired, the equipment used in the acquisition, and general information about the image pixels. The PACS may also function as an image tracking and display system for the digital exam/study/image by location/patient.

In the future, the enhancement methods presented in this research may be incorporated in the UniPACS Viewing Workstation- a joint project launched by Health Sciences Center and the Department of Computer Science- to view, read, analyze, archive and exchange digital images in DICOM format.

A functional image enhancement module has to address the individual radiologists sensitivity to contrast. Each radiologist’s preferences have to be assessed. This requires the collection of data that depends upon each radiologist. Each radiologist’s profile can be analyzed and constructed over time. Based on individual profiles, an automatic image enhancement procedure unique to each radiologist can be constructed. However, we could
also construct a profile based on the results from the majority of the radiologists and use this profile to enhance images in the future.
References


Williams, L.R.”Subband Coding –Laplacian Pyramids” at
http://www.cs.unm.edu/~williams/cs432/project4s00.html

Wolfram Research [2005] “Burt-Adelson Pyramid”, at :
http://library.wolfram.com/examples/pyramid/

Yang, Z., Kallergi, M, DeVore, R.A., Lucier, B., Qian, W., Clark, R.A., and Clarke, L.P.
[1995]“Effects of Wavelet Bases on Compressing Digital Mammograms”, IEEE Engineering


(5755), p. 412-418.

Appendix

Theoretical Approaches to Vision

A.1 Introduction to Visual Perception

Most of us take for granted our ability to see the world around us. How we see it seems to be no mystery: we open our eyes and look! We perceive a complex array of meaningful objects, as components of 3-D space. For example, Figure A.1 (Palmer [1999], p.4) shows a typical scene on the Berkeley campus of the University of California: students walking through Sather Gate, with trees and the Campanile bell tower in the background. We perceive this quickly and effortlessly, so it is difficult to imagine anything complicated about it. When we view the visual perception as an ability to be explained, it becomes incredibly complex and it seems almost a miracle that we can do it at all.

![Figure A.1 A real-world scene on Berkeley campus](image)

The visual experience resulting from viewing natural scenes like Figure A.1 happens when the neural tissues at the back of our eyes are stimulated by a two-dimensional pattern of light that includes bits and pieces of the objects that are perceived. Most of the Campanile is hidden behind the trees and parts of the trees are not visible due to the towers of the gate. We
do not perceive the Campanile as floating in the air or the trees as having tower-shaped holes in them. Even the objects that seem fully visible, such as the gate towers and the students can be seen only in part, because the far sides are occluded by the near sides. How are we able so quickly to perceive the meaningful, 3-D scene from the incomplete, 2-D pattern of light received by our eyes?

This is a fundamental question of vision and the researchers are looking for an answer from a scientific point of view. The first step in any scientific endeavor is to ask questions about processes or phenomena that are normally taken for granted. Many other questions are also important. A few of them are listed (Palmer [1999]):

- Why do objects appear colored?
- How can we determine whether an object is large and distant or small and close?
- How do we perceive which regions in a visual image are parts of the same objects?
- How do we know what the objects we see?
- How can we tell whether we are moving with respect to the objects in the environment or they are moving relative to us?
- Do newborn babies see the world in the same way we do?
- Can people “see” without being aware of what they see?

Different parts of the answers to these questions come from a variety of different disciplines—biology, psychology, computer science, neuropsychology, linguistics, and cognitive anthropology—all being part of the emerging field of cognitive science. The premise
of cognitive science is that the problems of cognition will be solved more quickly and completely if the scientists approach them from different perspectives.

The modern study of vision is suitable for this interdisciplinary approach. It is becoming an integrated field at the intersection of related disciplines. Vision science, as part of cognitive science, generates excitement among the scientists who study vision and it may hold a promise for reaching a new understanding about how we see.

Visual perception is defined (Palmer [1999]) as “the process of acquiring knowledge about environmental objects and events by extracting information from the light they emit or reflect”. From this definition, we can infer the following:

- Visual perception concerns the acquisition of knowledge. Vision is a cognitive activity, distinct from optical processes. There are some similarities between eyes and cameras in terms of optical phenomena, but there are no similarities in terms of perceptual phenomena. Cameras have no perceptual capabilities at all; they do not know anything about the scenes they record.

- The knowledge achieved by visual perception is related to objects and events in the environment. Perception is not about an observer’s subjective visual experiences.

- Visual knowledge about the environment is obtained by extracting information. An information processing approach to understanding visual perception and cognition allows vision scientists to talk about how people see in the same terms used to explain how computers might be programmed to see.

- The information processed in visual perception comes from the light emitted or reflected by objects. Optical information is the foundation of all vision. It is a result of the way in which physical surfaces interact with light in the environment. This restructuring of light
determines what information about objects is available for vision, so it is the appropriate starting point for any systematic analysis of vision (Gibson [1950]). Most of the early problems understanding vision are created from the difficulty of undoing what happens when light is projected from a three-dimensional world onto the two-dimensional surfaces at the back of the eyes. The study of the information contained in these projected images is an important frontier of research in vision science, explored by the computational theorists.

A.2 Visual Systems

We will present, briefly, the overall structure of the part of the nervous system known to be involved in processing visual information.

A.2.1 The Human Eye

Although it has been known since antiquity that eyes are the sensory organs of vision, an accurate understanding of how they work is a recent achievement. The Greek philosopher Plato (427-347 B.C.) believed that an “inner fire” produced rays that emanated from the eye toward perceived objects. Epicurus (341-270 B.C.) believed that tiny replicas of objects were transmitted rapidly into the mind through the eyes. Galen (A.D. 130-200) proposed that after the rays emanated from the eye, they interacted with the object and then returned to the eye. He believed that, in the lens of the eye, the rays interacted with a “visual spirit” that flowed from the brain to the eye and back.

The modern era of physiological optics started when the Arabic philosopher Alhazen (A.D. 965-1040) proposed the idea that the eye is like a pinhole camera and vision occurs when light from external sources is reflected from surfaces of objects and enters the eye. However, an accurate understanding of the optics of the eye became possible after the
invention of lenses. The astronomer Johannes Kepler (1571-1630) assembled these elements into an approximation of the modern theory of physiological optics.

A.2.1.1 Eye and Brain

Although Galen’s theory about vision was not correct, he was right in believing that both eyes and brain are essential (Figure A.2, Palmer, [1999], p. 25). Optical information from the eyes is transmitted to the primary visual cortex in the occipital lobe at the back of the head, as shown in Figure A.2. This information is sent to other visual centers located in the posterior temporal and parietal cortex. The complete visual system includes much of the brain as well.

Figure A.2 The human visual system. Visual processing begins in the eyes and relayed to the brain by the optic nerve. The primary visual pathway then goes from the lateral geniculate nucleus to occipital cortex via the optic radiations. From there, visual information travels to other parts of the brain. A secondary path goes from the optic nerve to the superior colliculus and then to other brain centers.
as the eyes. The eyes must collect and register information collected in light, and the brain must process that information.

A.2.1.2 Anatomy of the Eye

Humans have two eyes, approximately spherical in shape. They are located at the horizontal midline of the head and sit in hemispherical holes in the skull, called the eye sockets. Each eye is moved by six small muscles, called the extraocular muscles, controlled by specific areas in the brain. Eye movements allow us to scan different regions of the visual field without having to turn the entire head and for focusing on objects at distance. Eyelids and eyelashes protect the eyes, and tears make them moist and clean.

A.2.1.3 Physiological Optics

The eyes have two important optical functions: to gather light reflected from surfaces in the world and to focus it in a clear image on the back of the eye. If insufficient light is admitted, the image will be dim and ineffective for vision. If the image is not well focused, fine-grained optical information will be lost and spatial perception will be affected.

The eye has many parts that accomplish different optical functions (Figure A.3, Palmer, [1999], p. 27). First, light enters the cornea, a transparent bulge on the front of the eye. There is a cavity filled with a clear liquid, called aqueous humor behind the cornea. The light passes through the pupil, a sized opening in the iris, which gives the eye the external color. The lens is located behind the iris; its shape is controlled by ciliary muscles attached to its edge. The vitreous humor fills the central chamber of the eye. Finally, the light reaches the retina, the curved surface at the back of the eye. The retina is covered with over 100 million light-sensitive photoreceptors, which convert light into neural activity. This information is then sent to the visual centers of the brain.
288

Figure A.3 A cross section of the human eye. Light enters the eye through the cornea, aqueous humor, lens, and vitreous humor before it encounters the light-sensitive receptors of the retina, where light is converted into electrochemical signals that are sent to the brain via the optic nerve.

A.2.2 The Retina

After the optics of the eye have finished their task, the next function of the eye is to convert light into neural activity so that the brain can process the optical information. To understand how this happens, we must explore the basic building blocks of the brain and the way they work.

A.2.2.1 Neurons

The main functional component of the brain is the neuron: a specialized type of cell that integrates the input activity of other neurons connected to it and propagates that output activity to other neurons. The process of integration and transmission is done by a series of biochemical events within the neuron. The parts of the neuron are depicted in Figure A.4 (Palmer, [1999], p. 29) and their function is presented in the following:

1. The dendrites are thin protrusions from the cell body that receive chemical signals from other neurons and convert them into electrical activity along the thin membrane that surrounds the cell. This electrical activity is a graded potential: an electrical difference between the inside and outside of the dendrite whose value can vary within a range.
2. The cell body contains the nucleus and cellular machinery. The membrane around the cell body integrates the electrical signals received from the dendrites, coded as a graded potential and converts it into a series of electrical potentials (called action potentials, nerve impulses, or spikes) that are propagated along the axon.

3. The axon is a long projection of the neuron that facilitates the propagation of action potentials to other neurons. Most of the axon is covered by the myelin sheath, which speeds the conduction of action potentials. The strength of the integrated signal transmitted by the axon is encoded in its firing rate: the number of electrical impulses generated in a given amount of time (e.g., spikes per second).

4. The terminals are the branching ends of the axon where the electrical activity of the axon is converted back into a chemical signal, used to stimulate another neuron. This is done by releasing a neurotransmitter into the small gap between the terminal and the dendrite of the next neuron.
5. The synapse is the small gap between the terminals of one neuron and the dendrites of another. The neurotransmitter released into the synapse crosses the gap and affects the next neuron’s dendrite.

Neurons receive input from other neurons and send their input to other neurons. The energy from the environment must be converted into the form needed by the neurons. In the visual system, this function is accomplished by the photoreceptors-specialized retinal cells stimulated by light energy. Once the optical information is coded into neural responses, initial processing is carried out within the retina by several other types of neurons, including the horizontal, bipolar, amacrine and ganglion cells (see Figure A.5, Palmer [1999], p. 30). The axons of the ganglion cells carry information out of the eye through the optic nerve and on to the visual centers of the brain.

A.2.2.2 Photoreceptors

There are two distinct classes of photoreceptors cells in the retina: rods and cones. Their names describe their shapes, as shown in the scanning electron micrograph in Figure A.6 (Palmer, [1999], p. 31). Rods are longer, with untapered (rodlike) ends, whereas cones are shorter, thicker, with tapered (conelike) ends. Rods are more numerous (about 120 million), sensitive to light, located everywhere in the retina except its center (Figure A.7, Palmer [1999]). They perform the vision functions at low light levels (scotopic conditions): at night, at twilight, or in dimly lighted rooms. Cones are less numerous (8 million), less sensitive to light, concentrated in the center of the retina, with some of them scattered throughout the periphery (Figure A.7, Palmer, [1999], p. 31). They are used for vision under normal lighting conditions (photopic conditions) and for our experiences of color. There is a small region, called the fovea, in the center of the retina that contains packed cones (Figure A.7). The fovea covers a visual angle of about 2 degrees, but it makes both color and spatial vision more acute.
Figure A.5 The human retina. The retina consists of five major types of neurons: receptors (the rods and three kinds of cones), bipolar cells, ganglion cells, horizontal cells, and amacrine cells.

The process of changing the electromagnetic energy of photons into neural activity in photoreceptors is well understood. Both rods and cones have two parts: the inner segment that contains the nucleus and other cellular components and the outer segment with billions of light-sensitive pigment molecules.
Figure A.6 Scanning electron micrograph of rods and cones. The outer segments of rods have an untapered cylindrical shape, and those of cones have a tapered conical shape.

Figure A.7 Distribution of rods (solid curve) and cones (shaded region) in the human retina. The cones are almost exclusively present in the fovea and rods are more plentiful than cones in the periphery.

The pigment in rods is called rhodopsin. When a photon strikes a rhodopsin molecule and is absorbed by it, the molecule changes its shape in a way that is modifies the flow of electric current in and around the pigment molecule. The result of this biochemical reaction is
to produce electrical changes in the outer membrane of the receptor. These changes are propagated down the outer membrane to the synaptic region of the receptor, where chemical transmitters affect the next neuron.

The electrical changes resulted from many photons absorbed within the same receptor are integrated in the response of its outer membrane. The overall change in the electrical potential between inside and outside of the cell is graded, being continuous.

This complex chain of events in the outer segment is called pigment bleaching because the change in the molecular shape brought by light also causes the molecule to change color. Before a rhodopsin molecule is bleached by absorbing light, it appears to be deep purple; afterwards, it is almost transparent.

A.2.2.3 Pathways to the Brain

The axons of the ganglion cells leave the eye via the optic nerve which leads to the optic chiasm: the fibers from the nasal side of the fovea in each eye cross over to the opposite side of the brain while the others remain on the same side. The mapping from external visual fields to the cortex is completely crossed: all the information from the left half of the visual field is processed by the right half of the brain, and all the information from the right visual field is sent to the left half of the brain.

From the optic chiasm, there are two separate pathways into the brain on each side. The smaller one is transmitted to superior colliculus, a nucleus in the brain system. This visual center processes information about the location of things and has a function in the control of eye movements. The larger pathway goes first to the lateral geniculate nucleus (LGN) of the thalamus and then to the occipital cortex (or primary visual cortex).

A.2.3. Visual Cortex

The human cortex is a layered sheet of neurons, whose surface is convoluted and folded. It is divided into two halves, or cerebral hemispheres, approximately symmetrical.
A.2.3.1 Localization of Function

One of the first questions about the brain was whether or not its functions are localized: are mental faculties located in different anatomical regions or are all functions spread throughout the entire brain? Phrenology, the study of the shape of people’s skulls, was the source of pseudoscientific support for the localization hypothesis. Phrenologists claimed that the size of the lumps, bumps and bulges on someone’s skull is an indicator for the size and development of the brain structures underneath. They tried to find correspondences between skull measurements and assessments of mental attributes, such as “ambition”, “calculation” and “spirituality”. They created phrenological maps, that show the position of such functions. These correlations were unfounded, and scientific support for the concept of localization of function was developed when the effects of brain damage were studied systematically during the late nineteenth century.

The physicians began to perform postmortem analyses of the brains of the patients who had mental disabilities during their lifetime from strokes or head injuries. The results of these analyses provided the scientific basis of localization of function. The physicians discovered that certain types of deficits were strongly correlated with damage to certain regions. They proved that visual dysfunction occurred when there was damage to the posterior parts of the brain, mainly in the occipital lobe (Figure A.8, Palmer [1999], p. 26). It is now established that the occipital lobe is the primary cortical receiving area for visual information and that there are other cortical areas similarly specialized for other sensory modalities: audition, taste, touch, and smell.

A.2.3.2 Occipital Cortex

The physicians gathered data about visual cortex from old-world monkeys, such as the macaque. Behavioral analyses showed that their visual abilities are similar to those of humans (De Valois and De Valois, [1988]). The cellular exploration of visual cortex began in the
Figure A.8 Visual areas of the human cortex. From primary visual cortex in the occipital lobe, visual information separates into two major pathways: a lower (ventral) one that goes to the inferior regions of the temporal lobe and an upper (dorsal) one that goes to the parietal lobe.

1960s with the pioneering studies of Hubel and Wiesel. The vision scientists came to a better understanding of both the anatomy and physiology of various mammalian visual systems. The structure and function of distinct visual areas have been explored, and their interrelationships have been determined using a variety of techniques.

The first steps in cortical processing of visual information are performed in the striate cortex. This part of the occipital lobe receives the input from the lateral geniculate nucleus (LGN) on the same side of the brain. The visual input of striate cortex is crossed: the left visual field projects to the right striate cortex, and the right visual field projects to the left striate cortex. Both sides are activated by the central vertical strip having 1 degree of visual angle that separates the two sides of the visual field.

The mapping from retina to striate cortex is topographical—nearby regions on the retina project to nearby regions in striate cortex. This transformation maintains qualitative spatial relations but distorts quantitative ones. As a result of this transformation, the central
area of the visual field receives much greater representation in the cortex than the periphery does. This is called the cortical magnification factor. We have more detailed spatial information about objects in the central region of the retina than about those in the peripheral regions.

**A.2.3.3 Parietal and Temporal Cortex**

The physiologists Mortimer Mishkin and Leslie Ungerleider brought evidence of a difference between the function of the visual areas in the temporal versus parietal lobes of the monkey’s cortex (Figure A.9, Palmer [1999], p. 38). The inferior temporal centers in the lower (ventral) system seem to be responsible for identifying objects, whereas the parietal centers in the upper (dorsal) system seem to have a role in locating objects. These two pathways are known as the “what” system and the “where” system, respectively.

![Two visual pathways in the monkey cortex](image)

**Figure A.9** Two visual pathways in the monkey cortex. The lower (ventral) pathway goes from occipital cortex to the temporal lobe and is believed to be specialized for object recognition (the “what” system). The upper (dorsal) pathway goes from occipital cortex to parietal cortex and is believed to be specialized for object location (the “where” system).

**A.2.3.4 Mapping Visual Cortex**

An important part of the visual cortex in humans and related primates is hidden within the folds of the cortex. Figure A.10 (Palmer, [1999], p. 40) shows an anatomically correct description of the location of some of the principal areas of visual cortex (areas V1 through V5) in the brain of macaque.
Figure A.10 The location of primary visual cortex in macaque monkeys. The anatomical positions of striate cortex (area V1) and several prestriate areas (V2 through V5) are shown in a horizontal slice through the brain. The cellular structure of these areas is shown in the inset (A) with the transition between V1 and prestriate areas indicated by an arrow.

monkeys. These areas are part of the highly convoluted sheet of cortical neurons. These are only a few of the visual areas; others lie far from primary visual cortex.

The first cortical stage of visual processing-called striate cortex, primary visual cortex or area V1- is the largest, being located at the back of the occipital lobe. This area receives the most part of ascending projections from the LGN and performs the first few operations of
visual processing. The anatomy and physiology of this area is known better than any other area in the brain.

Initially, the physiologists thought that there might be a strict serial ordering of visual processing, each area projecting to the next. That hypothesis was rejected as the researchers found more anatomical connections among visual areas. It is known that processing takes place in parallel across different areas, each region projecting fibers to several other areas.

The cerebral cortex has a laminar structure, being constructed in layers. Visual cortex has six major anatomically defined layers, with several more sublayers defined by physiological evidence. Of these, the fourth seems to be the input layer for “forward” or “ascending” projections from lower parts of the nervous system. This is certainly true in area V1, where the ascending fibers from LGN synapse in layer 4 are usually located. For other cortical areas, it is less clear which projections are “forward” and which are “backward” or “descending”. In one direction, the projections originate in the superficial layers of cortex and terminate in layer 4 (Figure A.11 (a), Palmer, [1999], p. 42). These are called forward projections by analogy with those from LGN. Projections in the opposite direction—feedback or backward connections—have their originate and terminate outside of layer 4 (Figure A.11 (b), Palmer, [1999], p. 42).

A.2.3.5 The Physiological Pathways Hypothesis

A possible relation between anatomical structure and physiological function has emerged in the last decade. The hypothesis is that there are separate neural pathways for processing information about different visual properties such as color, shape, depth and motion. Several studies suggested that different areas of cortex were specialized for processing different properties (Zeki [1978],[1980]). It became plausible that this specialization was rooted in the visual system. Livingstone and Hubel presented this anatomical, physiological, and
perceptual evidence and proposed that the four different visual properties (color, shape, depth and motion) are processed in different neural pathways from the retina. They traced differences from two classes of retinal ganglion cells (one for color and form, the other depth and motion) to the LGN and from there to different regions of V1, V2 and beyond.

Using single cell recordings, they concluded that color, form, motion and stereoscopic depth information are processed in distinct subregions of V1 and V2, as shown in Figure A.12 (Palmer, [1999], p. 43). These areas then project to distinct higher-level areas of cortex: movement and stereoscopic depth information to area V5 (called MT, Medial Temporal cortex), color to area V4, and form through intermediate centers to area IT (Infero Temporal cortex). From these areas, the form and color pathways may project to the ventral “what” system for object identification and the depth and motion pathways to the dorsal “where” system for object localization.

The nature of visual processing in higher levels of the cortex is less clear than in area V1. The nature of spatial processing that takes place between V1 and IT is not well known.
Some theorists believe that color, shape, motion, and depth are processed independently in the visual system. This diagram contains a simplified form of the theory.

The motion analysis in area MT provides output to area MST (Medial Superior Temporal cortex) and other parietal areas. The researchers do not have the information about specific processing done in these centers.

The understanding of visual areas in the human cortex is not complete. Researchers in human neuropsychology have found interesting correlations between locations of strokes and tumors and the visual deficits they cause, but the evidence is complex and difficult to evaluate. The use of brain imaging techniques (PET, fMRI) has provided useful information about localization of visual functions in humans. These new imaging technique seem very promising as tools employed in the research in human neuropsychology.

A.3 Classical Theories of Vision

Vision scientists aim to understand how knowledge of the environment can arise from light that enters into the eyes. One might think that such a question can be answered by
discovering the relevant facts. But a scientific understanding of a complex domain such as visual perception implies more than just getting facts; it requires a theory.

“A theory is an integrated set of statements (called hypotheses) about underlying mechanisms or principles that not only organizes and explains known facts, but also makes predictions about new ones.

A theory is an internally consistent set of hypotheses or assumptions from which one can derive explanations of known facts and testable predictions of new facts.” (Palmer [1999], p. 46).

The distinguished Gestalt psychologist Kurt Koffka (Koffka, [1935]) asked the simple question “Why do things look as they do?” Different answers have been proposed, based on three “classical” issues, the core of psychological theories of visual perception:

1. Environment versus organism. One possible answer to Koffka’s question would be “Because the world is the way it is.” This approach analyzes external stimulus conditions as the proper way to understand perception. It suggests that one should examine the parts of the information in the proximal stimulus that correspond to the perceived information in the distal stimulus. An alternative answer would be “Because we are the way we are” or more precisely “Because our visual nervous systems are the way they are.” This approach emphasizes the nature of the organism rather than the external nature of the world around it.

2. Empiricism versus nativism. A different type of answer to Koffka’s question would be “Because we have learned to see them that way.” This is the **empiricist** view; it states that we see the way we do because of knowledge accumulated through personal interactions with the world. An alternative answer would be “Because we were born to see them that way”. This is the **nativist** view; it asserts that we do not need knowledge acquired during our lifetime because through evolution neural mechanisms have been created. According to the nativist view, the whole species has
learned by experience through evolution, so that each individual organism does not have to start from scratch as a “blank state”.

3. Atomism versus holism. Another possible response to Koffka’s question would be “Because of the way in which each small piece of the visual field appears.” According to the atomistic approach, perception of the whole visual field can be obtained by putting together the bits of visual experience in each local region. By contrast, the answer “Because of the way in which the whole visual field is organized.” This holistic view considers that how one component of the field appears perceptually will be influenced by other parts of the field. It is not enough to add local bits; these bits have to be globally integrated. According to the holistic approach, the visual system organizes stimuli so that the properties of the whole objects or whole scenes supersede those of local regions of the visual field.

4. Introspection versus behavior. This issue related to theories of perception is methodological. It takes into account two different methods used to derive the perceptual theory:

- from phenomenological observations of one’s own conscious experience (introspection)

- from objective measurements of human performance (behavior).

We will consider four psychological theories of visual perception and the theorists who proposed them. Three of these theories – known as structuralism, Gestaltism, and ecological optics- can be identified by the stands they take on the four key issues (see Table 4.1). They also differ in the theoretical analogies to which they appeal. The fourth psychological theory
of perception—called constructivism—can be viewed as a psychological theory of perception—called constructivism—can be viewed as a mixture of the other three. It represents the

Table A.1 Three psychological theories of visual perception

<table>
<thead>
<tr>
<th>THEORY</th>
<th>NATIVISM vs. EMPIRICISM</th>
<th>ATOMISM vs. HOLISM</th>
<th>ORGANISM vs. ENVIRONMENT</th>
<th>PRINCIPAL ANALOGY</th>
<th>METHOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structuralism</td>
<td>Empiricism</td>
<td>Atomism</td>
<td>Organism</td>
<td>Chemistry</td>
<td>Trained Introspection</td>
</tr>
<tr>
<td>Gestaltism</td>
<td>Nativism</td>
<td>Holism</td>
<td>Organism</td>
<td>Physical Field Theory</td>
<td>Naive Introspection</td>
</tr>
<tr>
<td>Ecological Optics</td>
<td>Nativism</td>
<td>Holism</td>
<td>Environment</td>
<td>Mechanical Resonance</td>
<td>Stimulus Analysis</td>
</tr>
</tbody>
</table>

currently dominant approach to perceptual theory within psychology and leads into the modern information processing view.

A.3.1 Structuralism

The earliest psychological approach to perceptual theory is known as structuralism. Its roots can be traced down in the views of a philosophical school called British empiricism, particularly in the writings of Locke, Berkeley and Hume. Wilhelm Wundt introduced these ideas into the new field of psychology in Germany.

In the structuralist view, perception arises from a process in which basic sensory atoms evoke memories of other sensory atoms that have been associated in memory through repeated prior joint occurrences. Such associations could occur whenever sensory experiences were close enough in time and space over a sufficient number of presentations. In case of vision, the sensory atoms were represented by visual experiences of color in each
tiny localized region of the visual field, resulting from the activity of individual photoreceptors in the retina. Presumably, these local sensations combined in perceptions by concatenation.

Visual experiences were assumed to arouse memories from other sensory modalities by association. Perception was thought to occur by rapid and unconscious associative processes that accessed memories acquired through extensive experience with the world. According to structuralism, as observers learn more and more about the world through associations, their perceptions become richer, more accurate and more complex.

Structuralist theory relied on a theoretical analogy to chemistry—namely, that the relation between simple sensations and complex perceptions is the same as the relation between primitive atoms and more complex molecules in chemistry. The mechanism that holds sensations together in more complex percepts was considered to be associations resulting from their spatial and temporal contiguity in past experiences. The structuralist theory is based on atomism and empiricism.

Trained introspection constitutes a foundation for the structuralist approach to perception. In the structuralists’ opinion, one could discover the elementary units of perception by turning one’s mind inward (introspecting) and observing one’s own experiences. However, they considered that a person could achieve this only if he or she was first trained in the method of introspection by a qualified expert. Unfortunately, the method of training often had strong influences on the results obtained (Boring, [1953]).

Structuralism was a transitional phase between an early philosophical period in the history of perceptual theory to a more sophisticated period.
A.3.2 Gestaltism

Historically, the Gestalt movement occurred as a reaction against structuralism. The leaders of this movement- Max Wertheimer, Wolfgang Kohler and Kurt Koffka- rejected the theoretical assumptions of atomism and empiricism the chemical analogy and the method of trained introspection assumed to be true in structuralism.

A.3.2.1 Holism

Gestalt is a German word that means, roughly, “whole form” or “configuration”. The Gestaltists rejected the structuralist idea that perceptions were built out of local sensory atoms by simple concatenation. The Gestalt theory of form perception- “The whole is different from the sum of its parts”- conveyed the belief that perceptions had their intrinsic structure as wholes that could not be reduced to parts or piecewise relations among parts.

To explain holism, Gestaltists pointed to examples in which configurations with emergent properties that are not shared by any of their local parts. This idea can be understood by examining a line made up of many different dots (Figure A.13). Each dot has

![Figure A.13 Emergent properties of a configuration. The arrangement of several dots in a line generates properties such as length, orientation, and curvature, that are different from the properties of the dots that form it.](image)

three perceptual properties: color, size and position. But when many dots are arranged in a line, the whole configuration has additional properties such as length, orientation and curvature. These properties emerge from the configuration when points form a line, because they do not reside in the individual parts.
After considering examples like this, Gestaltists rejected structuralism because the simple concatenation of parts can seldom capture the perceived structure of the whole. Gestaltists studied the aspects of perception that depend on qualities of whole figures or configurations. They were interested in the way in which the structure of a whole figure organizes its subparts.

The Gestaltists also rejected the classical chemical analogy of structuralism as too atomistic. They thought of mental processes as analogous to force fields in physics, such as magnetic or electrical fields. An important aspect of physical fields observed by Gestaltists was their holistic nature.

As a further affirmation against structuralism, Gestalt theorists rejected empiricism as the basis for much of perception. For instance, they believed that the mechanisms of perceptual organization did not require learning from experience, but arose from the interaction of the brain structure with stimulus structure. The Gestaltists rejected the structuralist view that experience played the dominant role in perception and claimed that unlearned processes were more important.

**A.3.2.2 Psychophysiological Isomorphism**

Although Gestaltists were mainly concerned with the structure of the stimulus image, they appealed to explanations based on brain mechanisms inside organisms. Gestaltists expressed their opinion about the relation between mind and brain in their doctrine of psychophysiological isomorphism, that stated that one’s perceptual experiences are structurally isomorphic to the underlying brain events (Wertheimer, [1912]; Kohler, [1947]).

The opponent process theory is an example of the doctrine of psychophysiological isomorphism (Hering, [1964]). This states that there are six psychologically primary colors
structured into three pairs of opposites: red versus green, blue versus yellow, and black versus white. This analysis was based on insightful observations about the nature of color experiences.

The Gestalt doctrine of psychophysiological isomorphism suggests that there should be some corresponding opponent structure in the neural events that underlie color perception. In fact, there is now good evidence that this is the case. There are three types of neurons within the human visual system that code color into the opposing pairs of red/green, blue/yellow, and black/white, as Hering’s analysis of color experiences proposed (DeValois, Jacobs [1968]). This correspondence between opponent color experiences and opponent neural events supports the Gestalt theory of psychophysiological isomorphism.

A.3.3 Ecological Optics

James J. Gibson of Cornell University was the founder of the theory of ecological optics, another classical theory of visual perception. Gibson opposed structuralism and was influenced by the Gestalt movement, especially by its emphasis on holism. Gibson’s rejection of structuralism went even further. He rejected the idea that organismic structure should be the basis for perceptual theory. His idea was that perception could be better understood by analyzing the structure of the organism’s environment, called its ecology. Gibson’s approach was “Ask not what’s inside your head, but what your head’s inside of” (Mace, [1977]). Gibson’s theory of ecological optics is about the informational basis of perception in the environment, rather than about its mechanistic basis in the brain.

A.3.3.1 Analyzing Stimulus Structure

James J. Gibson used the concept ambient optic array (AOA) to define the optical information available in light. The AOA refers to the light that comes toward a given point of
observation from all directions. The term “ambient” is used because the observation point is surrounded by light that converges to it from all directions. If the observer’s eye were at this point, either light reflected from environmental surfaces or light emitted directly from the radiant source would be available from every direction. Vision is possible at that observation point because surfaces in the environment structure the light in the AOA in complex, but lawful ways.

The external world has three spatial dimensions. The objects in the 3-D space are illuminated and the light is reflected by surfaces into the observer’s eye along straight lines. Photons pass into the eye to form a 2-D, upside-down image on its back surface. The object in the external world is referred to as the distal stimulus (distant from the observer), and its optical image at the back of the eye as the proximal stimulus (close to the observer). The size of an object’s image in the eye is usually specified by its visual angle: the number of degrees subtended by the image from its extremes to the focal point of the eye. This angle measures the spatial dimensions of the proximal stimulus, not the distal stimulus. The same external object will subtend a smaller angle when it is farther away and a larger angle when it is closer to the observer’s eye.

Gibson’s ecological theory of perception aimed to specify how the world structures light in the AOA such that people are able to perceive the environment by sampling that information. Gibson was interested in finding out what characteristics of the proximal stimulus provide information about distal stimulus. He asserted that the whole pattern of proximal stimulation provided much more information about the distal stimulus than it was previously estimated. Gibson’s theory of ecological optics emphasizes perceiving as the active exploration of the environment and the informational consequences of this fact. When the observer moves, the spatial pattern of simulation on the retina changes over time. Gibson
noticed the importance of this optic flow on the retina in understanding perception and invented the concept of dynamic AOA to refer to it. In the earlier work on perception, the subjects sat with their heads immobilized while viewing stationary stimuli under artificial conditions. Gibson and his followers argued that perceptual systems evolved in organisms on the move. The theory of ecological optics tries to specify what information related to the environment is available to the eyes of an observer that moves.

Gibson believed that the information available in retinal stimulation allowed an exploring organism to perceive the environment unambiguously. He named the operation that takes place in the brain to accomplish this skill information pickup. Although he was less interested in understanding the biological mechanisms of perception than in understanding their informational support in the stimulus, he proposed a resonance metaphor for how the process of information pickup might occur. Gibson compared the brain to a system of tuning forks and suggested that the process that occur in the brain’s activity was analogous to a process of mechanical resonance (Gibson, [1966], [1979]). Information in the stimulus determines the appropriate neural structures in the brain to fire; similarly, mechanical vibration of a specific frequency in the air causes a tuning fork of the same characteristic to vibrate. Unfortunately, Gibson never developed this analogy much further.

A.3.3.2 Direct Perception

Gibson proposed the concept of direct perception, i.e. visual perception in the environment is fully specified by the optical information available at the retina of a moving, actively exploring organism without any mediating processes or internal representations. He rejected the earlier theories that perception is possible only by making “unconscious inferences” that go beyond the information given in the sensory stimulation.
A.3.4 Constructivism

Constructivism, considered the dominant classical approach to visual theory, combines aspects of the three approaches previously presented. It is not characterized by strong stands on the four major issues used to characterize other psychological theories. Modern constructivism is a theory about internal mechanisms of perception rather than one about the external environment. The central idea in constructivism is that global percepts are constructed from local information. It also acknowledges the importance of emergent properties of lines, edges, angles and whole figures, as discussed by Gestalt theorists. Modern constructivism is neutral with respect to the controversy over nativism versus empiricism. Some aspects of perceptual abilities must be innate, but others are learned through interaction with the world. Most modern constructivists are methodological behaviorists, i.e. they study quantitative measures of human or animal behavior rather than introspecting about the contents of perceptual consciousness. In this matter, they contrast with both structuralists and Gestaltists, who used phenomenological observations. Introspective analysis is an important first step used by constructivists in drawing their theories. The difference is that constructivists then collect behavioral measures to test their ideas objectively.

A.3.4.1 Unconscious Inference

The founder of constructivist theory was Hermann von Helmholtz, a brilliant German physicist, mathematician, and physiologist who was contemporary of Wundt and the structuralists. His main ideas, published in his 1867 book “Treatise on Physiological Optics” have not changed much into modern times.

The most central contribution that Helmholtz made to the understanding of visual perception is the idea that perception depends on a process of unconscious inference.
Helmholtz acknowledged the logical gap between the optical information obtained directly from retinal stimulation and the perceptual knowledge derived from it. He proposed to fill this gap by using hidden “assumptions” in conjunction with retinal images to obtain perceptual “conclusions” about the environment. He claimed that vision requires a process of “inference” to transform insufficient 2-D optical information into a perceptual interpretation of the 3-D environment. The process of perceptual inference is unconscious. Unlike inferential processes involved in problem solving, people are not aware of how or when or why they are using visual inferences. Other constructivists, such as Richard Gregory (Gregory, [1970]), Julian Hochberg (Hochberg, [1964]) and Irvin Rock (Rock, [1983]) have continued Helmholtz’s ideas.

It is important to understand the basis used by visual system to make inferences about the nature of the environment from optical information in retinal stimulation. Helmholtz proposed the idea that vision chooses the interpretation that is the most likely state of affairs in the external world that was the cause of the retinal stimulation. This proposal is called the likelihood principle. It is a probabilistic view of perception; Helmholtz assumed that the visual system computes the interpretation with the highest probability given in retinal stimulation.

The likelihood principle contrasts with the Gestalt principle of Pragnanz (called also the minimum principle), which states that the simplest alternative is selected among possible interpretations. It is not easy to distinguish between these two different theoretical positions. There is a high correlation between what is likely and what is simple. The events that are most likely to yield observed patterns of simulation also tend to be simpler than the alternative events that might have produced the same optical image.
A.3.4.2 Heuristic Interpretation

Stephen Palmer (Palmer, [1999]) developed the idea that “the visual system transcends the available optical information by implicitly making a number of highly plausible assumptions about the nature of the environment and the conditions under which it is viewed”. These assumptions, combined with the sensory data in the incoming image result in a heuristic interpretation process in which the visual system makes inferences about the most likely environmental condition that could have generated the image. The process is heuristic because it uses the probabilistic rules of thumb that are usually, but not always true.

The likelihood principle is related to the constructivist view that perceptions uses heuristic assumptions. The heuristic assumptions allow the vision scientists to understand in what sense the visual system goes beyond the information provided to solve complex problems in effective ways.

A strict interpretation of unconscious inference would draw the conclusion that perception is accomplished by applying the symbolic logic’s rules or by solving mathematical equations. These are the processes considered by Helmholtz when he proposed his notion of unconscious inference. There are now more plausible interpretations within the computational framework that can be explained by using the general inferential scheme advocated by constructivists. For example, connectionist networks can reach perceptual conclusions based partly on incoming sensory data and partly on additional assumptions contained in the pattern of interconnections among its neuron-like elements. Such networks can “make inferences” based on heuristic assumptions without using either symbolic logic or mathematical equations. The behavior of such networks is also related to the dynamic behavior of Kohler’s physical Gestalts and to Gibson’s metaphor of mechanical resonance for information pickup.
A.4 A Brief History of Information Processing

The modern era in vision science began in the 1950s and the 1960s. The use of computer simulations to model cognitive processes, the application of information processing ideas in psychology and the idea that the brain is a processor of information the understanding of vision. These developments have influenced the evolution of vision science as an interdisciplinary field.

A.4.1 Computer Vision

An important breakthrough in the development of vision science was the idea that digital computers could be used to simulate complex perceptual processing. Previously, vision and other forms of perception and cognition had been considered abilities of living organisms. Scientists had to investigate a working system by developing theories and testing them on living beings, which is an expensive and difficult endeavor. The use of computer simulation techniques changed this situation radically. These techniques allowed vision scientists to build synthetic systems with well-known principles of operation and to compare their behavior with that of sighted organisms.

A.4.1.1 Early Stages of AI and Computer Vision

The modern computer age was born in the 1930s when the English mathematician Alan Turing defined a class of hypothetical machines known today as universal Turing machines. These machines could be programmed to process information automatically in a theoretically infinite variety of ways. The machine described by Turing was a mathematical abstraction, but it was not long until engineers began to build such machines. Mathematician John von Neumann coordinated the team that built the first digital computer, ENIAC, in 1946 at the University of Pennsylvania.
In the 1940s, Turing envisioned the possibilities of this computing machine to simulate intelligent thought (Turing, [1950]). Thus, the field of artificial intelligence (AI) was born, as a branch of computer science where computer programs simulate intelligent behavior. Originally, AI theorists focused their research on simulating difficult intellectual tasks such as playing chess and proving mathematical theorems (Newell, Shaw, and Simon, [1958], Newell and Simon, [1963]). Later, they took into account the alternative of programming computers to perceive the environment visually. This endeavor made possible the birth of computer vision, whose objective is to program computers to extract information about the environment from optical images.

Computer vision promoted two important developments that produced an important change in the theoretical branch of vision science:

-Real images. Theories of vision simulated on computers can be applied to gray-scale images obtained from video cameras that recorded real-world scenes. Classical theories of visual perception were designed for stimulus conditions that never exist in real situations: perfect, noiseless, line drawings of ideal objects. Computer vision made it possible to test theories on real images of real objects.

-Explicit theories. Before computer simulations, theories of perception were vague, informal, and incomplete, stressing large conceptual issues. Computer simulations changed this, because computer programming requires the theorist to make everything explicit.

One of the conclusions derived from these endeavors was that vision is extremely difficult. It turns out to be hard to get computers to “see” even simple things. Processes taken for granted by psychological theorists (like detecting edges, finding regions, understanding which regions are part of the same 3-D object) required complex computational efforts. The human visual system accomplishes these tasks with speed and accuracy and without a special
effort. Even state-of-the-art computer programs running on the most powerful computers failed to obtain the speed, accuracy and flexibility of human perceivers.

**A.4.1.2 Blocks World**

Early computer vision theories strived to understand scenes from blocks world, a “microworld” whose objects were simple, uniformly colored, geometrical solids, like a child’s set of blocks. The task was simplified enough so that computers succeeded to some extent. But if a cup of coffee or a piece of crumbled paper was introduced in the scene, the computer program would fail.

Roberts (Roberts, [1965]) produced one of the first significant simulations of blocks world vision. His program could recognize visual scenes from gray-scale images, but only for nonoverlapping polyhedral blocks constructed from a limited set of known prototypes, as shown in Figure A.14 (a) (Palmer, [1999], p.61). The program had two main stages. In the first stage, the goal was to construct a clean line drawing from the video image (Figure A.14 (e)). This was done using local luminance edges. Edges were detected at four orientations at 45° intervals (Figure A.14 (c)), and were linked together to produce smooth lines at the contours (Figure A.14 (d)). In the second phase, the purpose was to fit a model of the known set of volumetric primitives to the line drawing to get a geometrical description of its shape (Figure A.14 (f)). Shapes of planar regions were used in the line drawing to detect what type of 3-D primitives might be in the scene; the shapes that fit together were selected to form a description of the blocks.

**A.4.1.3 Computational Variants of Ecological Optics**

More recent advances in computer vision were obtained from formal analyses of the information existent in optical images under less restrictive conditions. Gibson’s earlier
theory on ecological optics (Gibson [1966], [1979]) advocated this approach. The mathematical analysis of the reflection of the environmental structure in image structure allowed the theorists to figure out modalities to obtain more complete information about the visual scene directly from the image.

The Dutch psychophysicists Jan Koenderink and Andrea Van Doorn were the first who considered this approach. They applied complex mathematical methods from differential geometry to problems such as motion perception from optical flow (Koenderink and Van Doorn, [1976a]), depth perception from stereoscopic information (Koenderink and Van Doorn, [1976b]), the 3-D orientation of surfaces from shading information (Koenderink, Van
Doorn, and Kappers, [1992]) and other similar topics. They did not create computer vision programs, but their work laid the foundation for writing such programs.

The mathematical approach successful in obtaining computer vision programs was proposed by David Marr and his colleagues at Massachusetts Institute of Technology (M.I.T.). They analyzed the way the luminance structure in two-dimensional images provides information about the structure of surfaces and objects in three-dimensional space (Marr, [1982]); this is a continuation of Gibson’s ecological approach. Their results dominated computer vision research for much of the past two decades.

A.4.1.4. Connectionism and Neural Networks

The most recent development in computer simulations of vision was created as the result of the great interest in connectionist network and neural network models (e.g., Feldman, [1981]; Feldman and Ballard, [1982]; Grossberg, [1982]; Hinton and Anderson, [1981]). These models were created on the assumption that human vision depends on the parallel structure of neural circuits in the brain. Such models are complex networks of many interconnected elements; one element functions like a simplified neuron. The element’s current state is determined by an activation level. Activation is transmitted through the network by connections that are either excitatory or inhibitory, similar to the synapses that allow the communication between neurons (Figure A.15, Palmer [1999], p. 62). These connectionist models can be expressed mathematically, but their behavior is described by nonlinear equations that are not solved analytically.

A.4.2 Information Processing Psychology

An important development in the evolution of the modern vision science was the information processing approach in psychology. The behaviorist movement dominated the psychology
from the 1920s onward. Behaviorists believed that studying the directly observable behavior would be the proper approach to psychology. This approach banished the introspective methods used by structuralists and Gestaltists. The psychological theories of any internal processes that referred to “mentalism” or “consciousness” were purged. In its extreme form, this would imply removing any reference to perception from psychological theories, because perception is a kind of internal experience of the external world.

The dominance of behaviorism continued during the 1940s and 1950s, more in areas related to learning. Toward the end of the 1950s, a new approach, that rejected the behaviorist dogma against considering internal states and processes gained more acceptance. The new idea was that mental processes could be viewed as information processing events, by using the new concepts developed in the fields of computer science and information theory. The psychological researchers were able to state their new theories of vision within a more precise language, closely related to the programs. New experimental methods were also developed to test these information processing theories (Sternberg, [1966], [1969]). The information
processing approach was well established in cognitive psychology and became more important for understanding visual perception and other aspects of mental activity.

A.4.3 Biological Information Processing

The emergence of the information processing paradigm was influenced by the invention of physiological techniques for studying neural activity in the visual system. These new methods were employed by the scientists to understand how visual information is processed in the retina and visual centers of the brain. The neuron was considered to be the appropriate unit of analysis in the visual system; vision was explained in terms of the firing pattern that resulted from the interactions among individual neurons (Barlow, [1972]). The methods for studying the activity of the individual neurons allowed the scientists to trace out a functional wiring diagram of the entire visual system, neuron by neuron, that could determine the operation accomplished by each neuron.

The research into biological information processing was influenced by these new techniques. The new developed techniques for imaging the structure and function of living brains have expanded the possibilities for physiological research into visual perception.

A.4.3.1 Early Experiments

Before the 1950s, the brain was not considered an information processing device; it was seen as a biological organ with opaque mechanisms. It took years of study and debate before biologists realized that neurons were not directly connected to each other, but distinct entities that communicated through chemical transmissions across synaptic clefts. The idea that brains were processing information was not widely accepted until the analogy between brains and computers was proposed.
At first, brain function was studied in lesion experiments in which areas of animals’ brains were surgically removed or destroyed. Such experiments produced fascinating discoveries about the localization of function in the brain, as mentioned in Section 4.2. Electrical brain stimulation techniques were also used with good results. In this case, an electrode is inserted into the brain and mild electrical current is sent to neurons surrounding its tip to see what behavior is obtained. Both of these techniques allowed the researchers to understand the large-scale structure of visual centers in the brain, but neither is appropriate to study information processing events in a normally functioning brain because they do not measure the electrochemical behavior of individual neurons.

A.4.3.2 Single-Cell Recording

During the 1950s, a physiological technique called single-cell was developed to explore the information processing performed by individual neurons. Extremely thin electrodes are positioned close to a neuron’s axon so that they can register the small changes in electrical potential produced each time a spike of neural activity passes along the axon. The output of the electrode can be recorded and analyzed to determine the stimulus conditions that activate the neuron. In vision, this is discovered by projecting patterns of light onto the animal’s retina; this allows us to find out whether their presence makes the neuron fire more or less than it does in their absence.

The most important early discoveries were made by Stephen Kuffler, David Hubel, and Torsten Wiesel at Harvard University. Kuffler used single-cell recording techniques to determine the receptive field of retinal ganglion cells: the region in the retina that influences the firing rate of the target neuron by increasing it (excitation) or decreasing it (inhibition) (Kuffler [1953]). It turned out that the optimal stimulus pattern to make ganglion cells fire vigorously was either a bright spot in the center of the receptive field with a dark disk
surrounding it or the reverse pattern. This knowledge formed the basis for the neural wiring diagrams that would explain the observed results.

Nobel laureates Hubel and Wiesel used similar techniques to map out the more complex receptive fields of cells in the visual cortex (Hubel and Wiesel, [1959], [1962]). They found elongated receptive fields that contain information about orientation as well as position (Figure A.16, Palmer [1999], p.67). An expansion of research into neural mechanisms was determined by their discoveries. As a result, portions of the visual cortex are well understood.

Single-cell recording techniques have some significant limitations. They are confined to the study of individual cells. Many electrodes are necessary to find out what many different cells are doing simultaneously. The number of cells that can be recorded simultaneously is limited. Such constraints make mapping the overall structure of cortical areas with single-cell techniques laborious.

Figure A.16 Receptive fields in cortical cells. Cells in the first area of visual cortex have elongated receptive fields that more vigorously if they are stimulated by an edge or line at a particular orientation and position.
A.4.3.3 Autoradiography

More efficient physiological methods to study the overall architecture of cortex have been developed. One of these techniques is autoradiography. To obtain an autoradiogram, an animal is injected with a dose of radioactively labeled substance that accumulates within the active neurons. After the chemical is injected, a visual stimulus with specific properties (such as lines of a single orientation) is shown to the animal. Radioactive sugar accumulates rapidly in the cells that fire most actively to the lines of that orientation. The animal is sacrificed, and thin slices of brain tissue are placed in contact with photographic paper sensitive to the level of radioactivity. Thus, a picture that displays the spatial distribution of the thousands of cells activated by the image is obtained. Single-cell recording techniques and autoradiographic methods have been useful in understanding the architecture of the visual cortex.

A.4.3.4 Brain Imaging Techniques

The most exciting new tools for studying the neural mechanisms that make vision possible are noninvasive methods to obtain images of the human brain: computer tomography (CT), positron emission tomography (PET) and magnetic resonance imagery (MRI). These techniques were discovered in the 1970s and have produced a revolution in modern medicine because they allow physicians to examine bodily tissues, including the brain, without breaking the skin. The processes employed in these new techniques were presented in detail in Sections 1.3 and 1.4.

A.5 Information Processing Theory

The information processing paradigm considers the nature of the human mind as a computational process. It has been applied successfully not only to visual perception, but also
to a wide range of cognitive phenomena in auditory perception, memory, language, judgment, thinking, and problem solving.

A.5.1 The Computer Analogy

The evolution of visual theories has been influenced by available research techniques. The invention of computers made a significant contribution to the information processing theories. First, computers allowed testing new theories of visual processing on real images. As mentioned in Section A.4, computer vision is a field within computer science that was founded by this approach. Its goal is to program computers to understand the world around similar to the way people do.

Second, computers served as theoretical analogy for mental processes within the information processes paradigm. Analogically, the relation between mental processes (such as visual perception) and the brain is the same as the relation between programs and computers used for their execution. In other words, minds can be seen as programs than run on brains; “minds are the “software” of biological computation, and brains are the “hardware”” (Palmer, [1999], p. 71). Due to this analogy, many perceptual theories over the past 30 years have been implemented as computer programs and others have been expressed within the framework of information processing without having been implemented as programs.

The computer analogy has replaced the theoretical analogies presented in Section A.2: the chemical analogy that inspired structuralist theory, the field theoretical analogy used in Gestalt theory and the resonance analogy that laid the foundation for the Gibson’s theory of information pickup. The computer analogy is compatible with the inferential analogy of constructivism. Some cognitive scientists believe that deep similarities exist between brains
and computers and that the brain is a biological computer. If this assertion is true, a computer may be able to perceive its environment with the same results like a person.

The analogy between mind/brain and program/computer can lead to the idea that a well programmed “seeing” computer would have conscious visual experiences. This form of the relation between the computer programs and mental events is sometimes called “strong AI” (Searle, [1980]). This is a strong claim for artificial intelligence: a properly programmed machine is capable of mental processes, including conscious experiences. This view contrasts with the concept of “weak AI”, that claims that a machine is only able to simulate events, conscious or not. The assertions claimed to be true in strong AI opened serious debate (Searle, [1980]).

A.5.2 Levels of Information Processing

In his 1982 book “Vision”, David Marr presented three different levels to be considered when complex information systems are analyzed:

- the computational level,

- the algorithmic level, and

- the implementational level.

Marr explained that there are important differences between these levels and all of them are essential for understanding vision as information processing.

A.5.2.1 The Computational Level

The computational level is the most abstract description proposed by Marr. This level was defined as the informational constraints used to map input information to output information.
This level specifies what computation should be performed and what information is involved, but it does not specify how it is accomplished.

**A.5.2.2 The Algorithmic Level**

Algorithmic descriptions are more specific than computational descriptions. They specify how a computation is executed in terms of operations of information processing. The algorithmic level corresponds to the program in computer science.

In order to construct an algorithm for a computational task, one must choose a representation for the input and output information and construct a set of processes that will obtain the output representation given the input representation. A representation can be thought as a way to encode information about something and a process can be seen as a way to change one representation into another.

**A.5.2.3 The Implementational Level**

The implementational level specifies how an algorithm is incorporated as a physical process within a physical system. Analogical to the execution of the same program on computers with different physical construction, one algorithm can be implemented on brains as well as various computers.

**A.5.3 Assumptions of Information Processing**

Palmer and Kimchi analyzed the information processing paradigm from a psychological perspective (Palmer and Kimchi, [1986]). Their analysis is related to that presented by Marr.

Palmer and Kimchi analyzed the assumptions that form the foundation of information processing theories in cognitive psychology. We will discuss the most important of these assumptions in the following sections.
A.5.3.1 Informational Description

Mental events can be functionally described as informational events; one such event consists of three parts:

- the input information
- the operation performed on the input
- the output information.

This assumption states that mental events, including visual perception transform the input information into output information. This can be illustrated in an information flow diagram (Figure A.17). If the correspondence between input and output is well defined, the operation can be specified such that knowing the input and the operation produces the output. A cognitive theory of this level of abstraction corresponds to Marr’s computational level, because it specifies what information is transformed from input to output but it does not specify how the transformation is accomplished.

A.5.3.2 Recursive Decomposition

Informational description is a necessary condition for an information processing theory, but is not sufficient. Gibson’s theory of information pickup specifies informational correspondences
between input (the dynamic ambient optic array) and output (perception of the environment), so it satisfies the assumption of informational description. But Gibson’s theory is not an information processing theory because it does not analyze the internal representation of the processes to obtain the mapping. Palmer and Kimchi specify another requirement of information processing theories as the assumption of recursive decomposition.

Any complex informational event at one level can be decomposed into a number of component informational events and a flow diagram that shows the temporal ordering relations among the components. A black box can be defined in terms of a number of smaller black boxes inside it, and the way they are interconnected. These smaller boxes are often called stages; each stage is assumed to be independent of other stages to some degree.

The decomposition is recursive, i.e. it can be performed again on the results of previous decompositions. The decomposed flow diagrams of an information processing system correspond to Marr’s algorithmic level, because when a system is decomposed into simpler components, an algorithm for the higher-level computation is specified. Marr viewed this algorithm as a single entity, whereas Palmer and Kimchi decomposed it into hierarchically nested levels.

### A.5.3.3 Physical Embodiment

Palmer and Kimchi considered the connection between the informational and physical levels in the assumption of physical embodiment.

“In a physical system whose behavior is being described as informational events, information is carried by states of the system (called representations), while operations that use this information is carried out by changes in state (called processes)” (Palmer and Kimchi, [1986]).

This assumption explains the difference between the abstract functional level of information and operations and the actual structure of a real physical system (or
implementation, as called by Marr). In this view, information and operations are entities in
the information processing descriptions, whereas representations and processes are entities in
the physical world, viewed as embodiments of information and operations.

The three levels—computational, algorithmic, and implementational—give a
classification of the research work in vision science. Currently, the researchers in computer
vision try to identify the optical information provided by retinal images that makes possible
perception of the external environment; this work is done at a computational level. These
theorists analyze the mathematical relations between the proximal and the distal stimulus.
Thus, they follow Gibson’s approach of ecological optics. At the algorithmic level, both
computer scientists and psychologists explore the ways of decomposing complex
computational problems into sets of simpler components and the flow of information among
them. Computer scientists are more interested to find out how well the algorithm works to
obtain useful perception; psychologists are more concerned to evaluate how well the
algorithm models human performance. At the implementational level, computer scientists
translate the algorithm in electronic devices; physiologists and psychologists want to
understand how brains process visual information through its neurons. In this
interdisciplinary point of view, the central belief is that the problem of vision has to be
approached at all three levels simultaneously to gain an adequate understanding.

A.5.4 Representation

Representations and processes are fundamental components of an information processing
system. A representation was defined as a physical entity that contains information about
something and process was understood as a physical transformation that change one
representation into the next. Some questions about the nature of the information processing
theories are important:
• What kind of information does a visual representation carry and how can accomplish this task?

• What type of processes are performed by a visual information processing system?

“A representation refers to a state of the visual system that stands for an environmental property, object, or event: it is a model of what it represents. A representation occurs only as part of a larger representational system that includes two related but distinct worlds: the represented world outside the information processing system (usually called the external world or environment) and the representing world within the information processing system (usually called the internal representation or simply the representation)” (Palmer, [1978]).

An internal world can represent an external world if the internal representation preserves information about the structure of the external world. This happens when the structure of the two worlds is the same to some extent. A representational system can be seen as a homomorphism: a mapping from objects in one domain (the external world) to objects in another domain (the internal representation) such that relations between objects in the external world have corresponding relations among corresponding objects in the representation (Tarski, [1964]). Figure A.18 shows a homomorphic mapping.

Figure A.18 Representation as a homomorphic mapping. External (represented) objects are mapped to internal (representing) objects such that relations among external objects are reflected by corresponding relations among internal objects.
A.5.5 Processes

“Processes are the active components in an information processing system that transform or operate on information by changing one representation into the next. In other words, processes are the dynamic aspect of the information processing system that actually cause informational transformations to occur.” (Palmer, [1999]).

A.5.5.1 Implicit versus Explicit

Processes transform implicit information in the input representation into explicit information in the output representation (and vice versa). Processes cannot create information about the environment; they can transform it by moving it and changing its form. All the information exists either in the optical structure projected from the environment onto retina or from internal sources within the observer. Processes collect and combine information properly to construct new representations.

A.5.5.2 Processes as Inferences

According to the information processing view, implicit information can be made explicit by mapping one representation into another. Processes that produce such transformations can be understood as inferences, as Helmholtz proposed. In the classic logical syllogism, the initially explicit information is given in the form of premises (e.g., “All people are mortal”, and “Socrates is a person”), and using logical rules the information that is implicit in the premises becomes explicit in the conclusion (“Therefore, Socrates is mortal”).

When vision scientists extend this inferential view of information processing to vision, they consider the “premises” to consist of the retinal image plus the stored knowledge or prior assumptions of the perceiver in the course of perceptual processing.

We presented the formal similarity between logical inferences and visual processing. As differences, we notice that the logical inferences made in solving syllogisms are
deliberate, slow, verbal, and conscious, whereas visual inferences are effortless, rapid, nonverbal and unconscious.

There are two types of inferences: deductive and inductive. The example considered before (“Socrates is mortal”) is a classic example of deductive inference. The standard mathematical operations are also deductive inferences. For this class of inferences, the conclusions are certain as long as the premises are true.

In contrast, inductive inferences are inherently uncertain and probabilistic. A classic example of inductive inference is the conclusion that all people are mortal.

The light reflected from the 3-D world produces 2-D images at the back of the eye where vision begins. This process of image formation is completely determined by the laws of optics, so for a given scene with well-defined light conditions and a point of observation, the 2-D image can be determined accurately.

The early stages of visual perception attempt to solve the inverse problem: how to obtain from optical images of scenes the objects that gave rise to them. The most obvious solution is for vision to attempt to invert the process of image formation by undoing the optical transformations applied during image formation.

Unfortunately, it becomes very difficult, because the mathematical relation between the environment and its projective image is not symmetrical. The projection from environment to image goes from three dimensions to two and is a well-defined function. The inverse mapping from image to environment goes from two dimensions to three, and this is not a well-defined function.
The inverse problem is underspecified (or under-constrained) by the sensory data in the image. There is no easy way around this problem, and that is why visual perception is so complex.

Most inferences in visual processing are inductive. They are not guaranteed to be true, because of the underconstrained and probabilistic nature of the inverse problem that they attempt to solve. Key processes in vision are inductive rather than deductive inferences, but they can be treated as deductive inferences by making hidden assumptions (Cutting, [1991]).

By considering the problem of visual processing in terms of deductive versus inductive inferences, Cutting reopened the debate between direct (Gibsonian) and indirect (Helmholtzian) theories of perception. In his interpretation, Gibson’s position that perception is direct can be identified with the assumption that perception is deductively inferred from the information contained in the image and therefore certain. Cutting identifies Helmholtz’s position of unconscious inference with the claim that perception is inductively obtained from the image and can be accomplished with the addition of further assumptions.

**A.5.5.3 Top-Down versus Bottom-Up Processes**

Another important distinction in the processing of perceptual information is made in terms of “direction”: bottom-up or top-down. The analogy underlying this distinction can be understood as a flowchart of visual processing that shows the retinal image at the bottom and subsequent interpretations farther along the pathway located at higher levels (Figure A.19, Palmer [1999], p.85). Bottom-up processing - also called data-driven processing- include processes that take a “lower-level” representation as input and create a “higher-level” representation as output. Top-down processing–also called hypothesis-driven or expectation-driven processing- refers to processes that start with a “higher-level” representation as input and produce a “lower-level” representation as output.
Figure A.19 Bottom-up versus top-down processes. The two directions of processing are known as bottom-up (or data driven) from lower to higher levels of processing and top-down (or hypothesis driven) from higher to lower levels of processing.

Intuitively, vision can be seen as a bottom-up process. It starts with the sensory information in the retinal images and continues with perceptual and conceptual interpretations. Most scientists adhere to the idea that the early stages of visual processing are indeed bottom-up. But there are arguments that prove that this cannot be true for the entire visual perception. For example, Palmer argues that

“perception of the present state of affairs produces expectations about the future. These expectations imply a top-down component to visual processing, because they suggest that prior higher-level interpretations influence current processing at lower levels.” (Palmer [1999])

The point at which top-down processes complete bottom-up processes is currently a controversial issue. Some theorists believe that this happens early in visual processing; others believe that it happens late.
A.6 Stages of Visual Perception

Some of the concepts introduced in the information processing approach can be applied to vision. The visual perception can be decomposed at the algorithmic level into four major stages beyond the retinal image, as illustrated in Figure A.20 (Palmer, [1999], p. 85).

![Diagram of visual perception stages]

Figure A.20 Four stages of visual processing. There are four stages of visual processing: image-based, surface-based, object-based, and category-based processing.

Each stage is defined by a different kind of output representation and the processes that compute it from the input representation. Different theorists use different names to refer to these stages. A generic labeling scheme in which each stage is named for the kind of information it represents is proposed by Palmer (Palmer, [1999]): the image-based, surface-based, object-based, and category-based stages of perception. The theoretical framework is based on the influential writings of David Marr and his colleagues at M.I.T. Other schemes have been considered, but these four stages provide a robust framework for understanding vision as a computational process.

A.6.1 The Retinal Image

The proximal stimulus for vision is the pair of 2-D images projected from the environment to the viewer’s eyes. The optical image received in the retina is completely continuous, but its
registration by the retinal receptors is discrete. The set of firing rates in all receptors in both eyes form the first representation of optical information within the visual system.

In formal and computational theories of vision, the retinal representation is approximated as a homogeneous, two-dimensional array of receptors. The spatial locations of these receptors is identified by their coordinates \( (x, y) \) in an integer plane whose center is in the middle of fovea (a small region in the center of the retina) and whose \( x \) and \( y \) axes are aligned with the retinally defined horizontal and vertical. These square image elements – called pixels – are the primitive, indivisible visual unit of information in the input image.

**A.6.2 The Image-Based Stage**

Most vision scientists agree that the initial registration of the images in the eyes is not the only representation based on a 2-D retinal organization. The additional representations and processes constitute the image-based stage. This includes image-processing operations such as detecting local edges and lines, linking local edges and lines together, matching up corresponding images in the left and right eyes, defining 2-D regions in the image. These 2-D features of images determine their structure before they are interpreted as properties of 3-D scenes.

Marr called the representations produced by image-based processes primal sketches and used the following classification:

- raw primal sketch, including the results of elementary detection processes that locate edges, bars, blobs and line terminations
- full primal sketch, including global grouping of the local features detected in the raw primal sketch.
The structure of an image-representation is defined by the following characteristics:

- **Image-based primitives.** The primitive elements contain information about the 2-D structure of the luminance image (such as edges and lines defined by differences in light intensity) rather than information about the objects in the external world (such as surface edges or shadow edges).

- **Two-dimensional geometry.** The geometry of spatial information in image-based representation is 2-D and can be represented using 2-D arrays.

- **Retinal reference frame.** The coordinate system that allows the location of 2-D features is relative to the retina, in the sense that the axes are aligned with the eye.

### A.6.3 The Surface-Based Stage

The second stage of visual processing, the surface-based stage, deals with recovering the intrinsic properties of visible surfaces in the external world that generated the features found in the image-based stage. In the surface-based stage, the information about the external world is represented as a layout of visible surfaces in three dimensions.

Gibson proposed the notion that the visual system perceives the surface layout—the spatial distribution of surfaces in the 3-D environment (Gibson, [1950]). His idea was not accepted by other theorists until computer vision theorists began to discuss it later (Marr, [1978]; Marr and Nishihara, [1978]). Gibson’s idea was difficult to be understood, because he did not provide a specific representation for this surface layout or a set of processes that could generate it from retinal images.

The concept of a surface-based representation as an intermediate stage in vision gained acceptance when it was described quantitatively by computer vision theorists and
implemented in computer simulations. Surface-based representations and algorithms to construct them from gray-scale images were proposed by Marr (Marr, [1978]) and Barrow and Tennenbaum (Barrow and Tennenbaum, [1978]). Marr used the term 2.5 D sketch to suggest that his surface-based representation is between the 2-D structure of image-based representations and the true 3-D structure of object-based representations. Barrow and Tennenbaum called their surface-based representations intrinsic images to emphasize the idea that they represent intrinsic properties of surfaces in the external world rather than features of the input image.

Getting a surface-based representation is the first step in recovering the third spatial dimension from 2-D images. It contains information about the surfaces that are visible from the current viewpoint. Visible surfaces are a source of information about their distance from the viewer and their slant. They cannot be computed from the retinal images without additional assumptions because this is an underconstrained inverse problem, as discussed in Section A.5.5.2.

We mention the most important properties of a surface-based representation:

- Surface primitives. The primitive elements of the surface-based representation are local patches of 2-D surface at some particular slant located at some distance from the viewer within 3-D space. Each patch of surface can be further specified by its color and texture.

- Three-dimensional geometry. Although the surfaces themselves are locally 2-D, their spatial distribution is represented within a 3-D space.
• Viewer-centered reference frame. The coordinate system used to represent the 3-D layout of surfaces is established in terms of the direction and distance from the observer’s standpoint to the surface rather than in terms of the retina.

**A.6.4 The Object-Based Stage**

The representation of the visible surfaces is not the final point in visual perception. We have expectations about partly and completely hidden surfaces. This proves that there is some form of true 3-D representation that includes at least some occluded surfaces in the visual world. In the object-based stage, the visual representation includes 3-D information. The visual system is able to manage using further hidden assumptions about the nature of the visual world, because the inferences include information about unseen surfaces or parts of surfaces. This stage is called object-based because explicit representations of the whole objects in the environment make possible the inclusion of unseen surfaces.

An object-based representation can be obtained using a boundary approach or a volumetric approach. In the boundary approach, the surface-based representation is extended to include unseen surfaces within a 3-D space. In the volumetric approach, the objects are seen as 3-D entities, represented as arrangements of a set of primitive 3-D shapes.

The volumetric approach dominated theories of object-based processing for many years, due to the work on 3-D shape primitives in computer vision by Agin and Binford (Agin and Binford [1976]) and Marr and Nishihara (Marr and Nishihara, [1978]).

The main features of the object-based representation are the following:

• Volumetric primitives. The primitive elements of the object-based representation may be descriptions of 3-D volumes, that include information about unseen
surfaces of the object.

- Three-dimensional geometry. The space within which the volumetric primitives are located is fully 3-D.

- Object-based reference flames. The coordinate system used to represent the spatial relations among volumetric primitives may be defined in terms of the intrinsic structure of the volumes themselves.

A.6.5 The Category-Based Stage

The ultimate goal of perception is to allow the perceiving organism to obtain accurate information about the environment. The final stage of the perception should be able to recover the functional properties of objects. This processing is called the category-based stage.

The categorization (or pattern recognition) approach to perceiving relevant function assumes that two operations are performed. First, the visual system classifies an object as a member of one of known categories according to its visible properties, such as size, shape, color and location. Second, this classification gives access to stored information about this type of object, including its function and expected behavior.

Gestalt theorists suggested that the visual system might be able to perceive an object’s function by registering functional properties of objects directly from their visible characteristics without categorizing them. They called these properties physiognomic characters.

It is possible that people perceive the function both directly and indirectly. For some objects, such as chairs and cups, their functional properties are tied to their visible structure,
so one might not need to categorize them to know their usage. The functions of other objects, such as computers and telephones, cannot be described from their visual characteristics, so these objects need to be categorized first.

These four stages of visual processing –image-based, surface-based, object-based and category-based –constitute the most widely accepted theory about the overall structure of the visual perception. They were presented in the order in which they must logically be initiated, but it is not necessary that each stage is completed before the next begins. The arrows going backward in Figure A.20 suggest that later processes may influence the earlier ones.
Vita

Alina Monica Trifas was born on June 3, 1972, in Ploiesti, Romania. She received her bachelor’s degree in mathematics from the University of Bucharest in 1996. She has also received her Master of Science in Computer Science from University of Bucharest in 1999. In 2001, she began graduate studies in computer science at Louisiana State University. Her research interests include digital image processing, high performance computing, artificial intelligence, bioinformatics.