Advances in Computer Vision and Pattern Recognition

Mongi A. Abidi
Andrei V. Gribok
Joonki Paik

Optimization Techniques in Computer Vision

Ill-Posed Problems and Regularization

Springer

Advances in Computer Vision and Pattern Recognition

More information about this series at http://www.springer.com/series/4205

Mongi A. Abidi • Andrei V. Gribok • Joonki Paik

Optimization Techniques in Computer Vision

Ill-Posed Problems and Regularization

Springer

Mongi A. Abidi
Department of Electrical and Computer Engineering
University of Tennessee
Knoxville, Tennessee
USA

Andrei V. Gribok
Department of Human Factors, Controls, and Statistics
Idaho National Laboratory
Idaho Falls, Idaho
USA

Joonki Paik
Image Processing and Intelligent Systems Laboratory
Chung-Ang University
Seoul, Korea

ISSN 2191-6586 ISSN 2191-6594 (electronic)
Advances in Computer Vision and Pattern Recognition
ISBN 978-3-319-46363-6 ISBN 978-3-319-46364-3 (eBook)
DOI 10.1007/978-3-319-46364-3

Library of Congress Control Number: 2016955839

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Overview

The advent of the digital information and communication era has resulted in image processing and computer vision playing more important roles in our society today. These roles include creating, delivering, processing, visualizing, and making decisions from information in an efficient and visually pleasing manner.

The term *digital image processing* or simply *image processing* refers to comprehensively processing picture data by a digital computer. The term *computer vision* refers to computing properties of the three-dimensional world from one or more digital images.

The theoretical bases of image processing and computer vision include mathematics, statistics, signal processing, and communications theory. In a wide variety of theories for image processing and computer vision, optimization plays a major role. Although various optimization techniques are used at different levels for those problems, there has not been a sufficient amount of effort to summarize and explain optimization techniques as applied to image processing and computer vision.

The objective of this book is to present practical optimization techniques used in image processing and computer vision problems. A generally ill-posed problem is introduced and it is used to show how this type of problem is related to typical image processing and computer vision problems.

Unconstrained optimization gives the best solution based on numerical minimization of a single, scalar-valued *objective function* or *cost function*. Unconstrained optimization problems have been intensively studied, and many algorithms and tools have been developed to solve them. Most practical optimization problems, however, arise with a set of constraints. Typical examples of constraints include (a) prespecified pixel intensity range, (b) smoothness or correlation with neighboring information, (c) existence on a certain contour of lines or curves, and (d) given statistical or spectral characteristics of the solution.

Regularized optimization is a special method used to solve a class of constrained optimization problems. The term *regularization* refers to the transformation of an objective function with constraints into a different objective function, automatically reflecting constraints in the unconstrained minimization process. Because of its simplicity and efficiency, regularized optimization has many application areas, such as image restoration, image reconstruction, and optical flow estimation.

Optimization-Problem Statement

The fundamental problem of optimization is to obtain the best possible decision in any given set of circumstances. Because of its nature, problems in all areas of mathematics, applied science, engineering, economics, medicine, and statistics can be posed in terms of optimization.

The general mathematical formulation of an optimization problem may be expressed as

$$\min_{x \in R^N} f(x) \quad \text{subject to } x \in C, \tag{1.1}$$

where $f(x)$ represents the objective function, C the constraint set in which the solution will reside, and R^N the N-dimensional real space.

Various optimization problems can be classified based on different aspects. At first, classification based on the properties of the objective function $f(x)$ is:

1. Function of a single variable or multiple variables,
2. Quadratic function or not, and
3. Sparse or dense function.

A different classification is also possible based on the properties of the constraint C as:

1. Whether or not there are constraints,
2. Defined by equation or inequality of constraint functions, i.e., $C = \{x | c(x) = 0\}$ or $C = \{x | c(x) \geq 0\}$, where $c(x)$ is termed the constraint function, and
3. The constraint function is either linear or nonlinear.

If, for example, we want to reduce random noise from a digital image, a simple way is to minimize the extremely high-frequency component while keeping all pixel intensity values inside the range $[0, 255]$. In this case, the objective function of multiple variables represents the high-frequency component. The range of pixel intensity values plays a role in inequality constraints.

Optimization for Image Processing

Various optimization techniques for image processing can be summarized as, but are not limited to, the following:

1. Image quantization: The optimum mean square (*Lloyd*-Max) quantizer design.
2. Stochastic image models: Parameter estimation for auto-regressive (AR), moving average (MA), or auto-regressive-moving average (ARMA) model.
3. Image filtering: Optimal filter design and Wiener filtering.
4. Image restoration: Wiener restoration filter, constrained least squares (CLS) filter, and regularized iterative method.
5. Image reconstruction: Convolution/filtered back-projection algorithms.
6. Image coding: Energy compaction theory and optimum code-book design.

Optimization for Computer Vision

Some examples of optimization techniques for computer vision are summarized as:

1. Feature detection: Optimal edge enhancer, ellipse fitting, and deformable contours detection.
2. Stereopsis: Correspondence/reconstruction procedures, three-dimensional image reconstruction.
3. Motion: Motion estimation, optical flow estimation.
4. Shape from single image cue: Shape from shading, shape from texture, and shape from motion.
5. Recognition
6. Pattern matching

Organization of This Book

This book has five self-contained parts. In Part I, Chap. 1 introduces the scope and general overview of the material. Chapter 1 gives an introduction into ill-posed problems. This chapter also discusses practical reasons why many image processing and computer vision problems are formulated as ill-posed problems. Chapter 1 also presents typical examples of ill-posed problems in image processing and computer vision areas. Chapter 2 discusses different techniques to select regularization parameter.

Part II summarizes the general optimization theory that can be used to develop a new problem formulation. Practical problems are solved using the optimization formulation. Chapter 3 presents a general form of optimization problems and summarizes frequently used terminology and mathematical background. Chapters 4 and 5 describe in-depth formulation and solution methods for unconstrained and constrained optimization problems, respectively. Constrained optimization problems are more suitable for modeling real-world problems. This is true because the desired solution of the problem usually has its own constraints.

In Part III, we discuss regularized optimization, or simply regularization, that can be considered a special form of a general constrained optimization. In Chap. 6, frequency-domain regularization is discussed. Chapters 7 and 8 describe iterative type implementations of regularization and fusion-based implementation of regularization.

Part IV provides practical examples for various optimization technique applications. Chapters 9, 10, 11, and 12 give some important applications of two-dimensional image processing and three-dimensional computer vision.

Appendices summarize commonly used mathematical background.

Knoxville, TN — Mongi A. Abidi
Idaho Falls, ID — Andrei V. Gribok
Seoul, Republic of Korea — Joonki Paik

About the Authors

Mongi A. Abidi received the Principal Engineering degree in electrical engineering from the National Engineering School of Tunis, Tunisia, in 1981 and the M.S. and Ph.D. degrees in electrical engineering from the University of Tennessee, Knoxville, in 1985 and 1987, respectively. He is a Professor with the Department of Electrical and Computer Engineering, the University of Tennessee, where he directs activities in the Imaging, Robotics, and Intelligent Systems Laboratory as an Associate Department Head. He conducts research in the field of 3D imaging, specially, in the areas of scene building, scene description, and data visualization.

Andrei V. Gribok is a Principal Research Scientist at Idaho National Laboratory, Department of Human Factors, Controls, and Statistics. He received his Ph.D. in Physics from Moscow Institute of Biological Physics in 1996 and his B.S. and M.S. degrees in systems science/nuclear engineering from Moscow Institute of Physics and Engineering in 1987. Dr. Gribok worked as an instrumentation and control researcher at the Institute of Physics and Power Engineering, Russia, where he conducted research on software fault detection systems for nuclear power plants. He also was an invited research scientist at Cadarache Nuclear Research Center, France, where his research focus was on ultrasonic visualization systems for liquid metal reactors. He also holds the position of Research Associate Professor with the Department of Nuclear Engineering, University of Tennessee, Knoxville. Dr. Gribok was a member of a number of international programs including IAEA coordinated research program on acoustical signal processing for the detection of sodium boiling or sodium-water reaction in LMFRs and large-scale experiments on acoustical water-in-sodium leak detection in LMFBR. Dr. Gribok's research interests include reliability and soft computing in nuclear engineering, nuclear systems' instrumentation and control, prognostics and diagnostics for nuclear power plants, uncertainty analysis of software systems, and inverse and ill-posed problems in engineering. Dr. Gribok taught classes on soft computing at the University of Tennessee, Knoxville. Dr. Gribok is an author and coauthor of three book chapters, 37 journal papers, and numerous peer-reviewed conference papers.

Joonki Paik received his B.S. degree in control and instrumentation engineering from Seoul National University in 1984. He received the M.S. and the Ph.D. degrees in electrical engineering and computer science from Northwestern University in 1987 and 1990, respectively. From 1990 to 1993, he joined Samsung Electronics, where he designed the image stabilization chip sets for consumer camcorders. Since 1993, he has joined the faculty at Chung-Ang University, Seoul, Korea, where he is currently a Professor in the Graduate school of Advanced Imaging Science, Multimedia and Film. From 1999 to 2002, he was a visiting Professor at the Department of Electrical and Computer Engineering at the University of Tennessee, Knoxville.

Contents

Part I

Chapter 1
Ill-Posed Problems in Imaging and Computer Vision

1.1 Introduction

In many image processing problems, we need to estimate the original complete data sets, generally from incomplete and, most often, from degraded observations. One simplified example is to estimate the original pixel intensity value, which has been attenuated in the imaging system, without any correlation with neighboring pixels. If we know the nonzero attenuation factor for the imaging system, we can easily estimate the original value by multiplying by this attenuation factor. Figure 1.1 shows the corresponding attenuation and the restoration processes.

As a second example, suppose we estimate the original pixel intensity value by averaging this pixel with eight neighboring pixels as shown in Fig. 1.2. This example is frequently used in simplifying the out-of-focus blur in an imaging system. As shown in Fig. 1.2a, a pixel intensity value is distributed into the neighborhood on the imaging plane. As a result, a pixel value in the imaging plane is determined by integrating partial contributions of neighboring pixels in the object plane, as shown in Fig. 1.2b. In order to have mathematical representation of the two-dimensional (2D) signals and systems, we use the row-ordered (or lexicographically ordered) vector-matrix notation [jain89]. If we assume that both the input image contained in the object plane and the observed image contained in the imaging plane are $N \times N$, the pixel averaging process can be represented as

$$Du = f, \tag{1.1}$$

where u and f represent $N^2 \times 1$ input and output images, respectively, and D the $N^2 \times N^2$ matrix, such that

M.A. Abidi et al., *Optimization Techniques in Computer Vision*, Advances in Computer Vision and Pattern Recognition, DOI 10.1007/978-3-319-46364-3_1

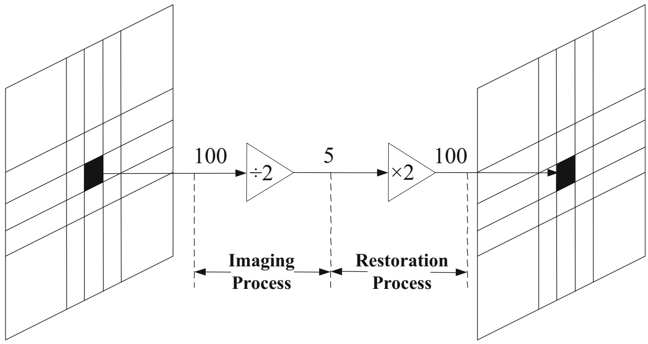

Fig. 1.1 A diagram illustrating pixel attenuation followed by restoration

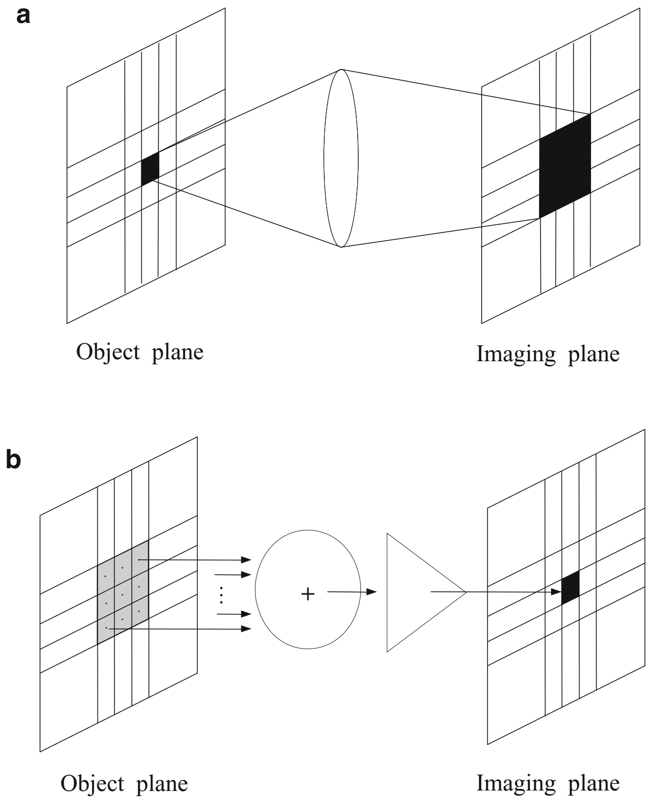

Fig. 1.2 (**a**) Simplified out-of-focus process and (**b**) the corresponding model obtained by averaging pixels in the neighborhood

$$D = \begin{bmatrix} D_1 & D_1 & 0 & \cdots & 0 \\ D_1 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & D_1 \\ 0 & \cdots & 0 & D_1 & D_1 \end{bmatrix}, \text{ where } D_1 = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \\ 1 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & 1 & 1 \end{bmatrix}. \tag{1.2}$$

See Appendix A for more details of this formulation.

To estimate u from f, we can multiply D^{-1} to the left-hand side of f, such that $\hat{u} = D^{-1}f$, where $\hat{u}$ denotes a calculated estimate of u. Keep in mind that in order to compute the inverse of a matrix, the matrix must be nonsingular. One also will recall that a nonsingular matrix has a nonzero determinant. It is desirable for the matrix D to be well conditioned. This means that a bounded perturbation in the observed data f results in a bounded error in the estimated solution $\hat{u}$. In this example, $\hat{u}$ is a good estimate, in some sense, because D is assumed to have an inverse and assumed to be well conditioned [golub96]. However, D becomes ill conditioned as the number of averaged pixels increases. Even when D has an inverse, the estimate from observed data, being possibly corrupted by noise, is not reliable if this matrix is ill conditioned. This means bounded perturbations in the observed data may result in unbounded errors in the estimated solution.

This second example deals with a simple estimation problem in which the observed data and the solution have the same dimensions. This results in the estimation process outcome being equivalent to multiplying the inverse of the distortion matrix D by the observed data f. In many engineering problems, however, the observed data and the solution generally have different dimensions and characteristics. For this reason, we need a more general description of the characteristics of inverse problems.

1.2 The Concepts of Well Posedness and Ill Posedness

J. Hadamard introduced the concept of a well-posed problem, resulting from physical mathematical models, to clarify the most natural boundary conditions for various types of differential equations in the early 1900s [tikhonov77]. As mentioned previously, a linear equation with a well-conditioned matrix is a good example of a well-posed problem. A formal definition of well posedness is now presented.

We consider a solution of any quantitative problem, where the solution u is to be estimated from given data f and the operator A, which relates u and f, such as $A: u \to f$. We shall consider u and f as elements of metric spaces U and F with metrics $\rho_U(u_1, u_2)$ for $u_1, u_2 \in U$ and $\rho_F(f_1, f_2)$ for $f_1, f_2 \in F$.

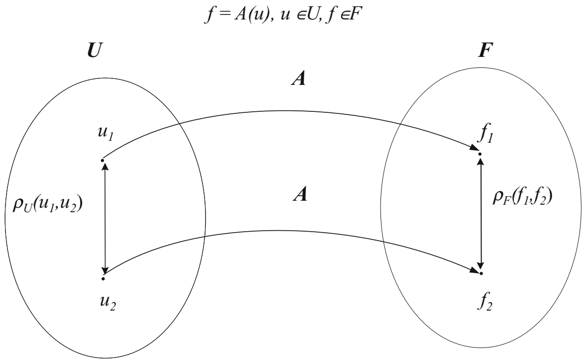

Fig. 1.3 Relationships among spaces, elements, the operator, and metrics

Figure 1.3 shows the relationships among spaces, elements, the operator, and metrics. Frequently, the Euclidean distance is used for metrics ρ_U and ρ_F. U and F are then assumed to be Hilbert spaces.

The problem of determining the solution u in the space U from the given data f in the space F is said to be well posed on the pair of metric spaces (U, F) if the following three conditions are satisfied:

1. For every element $f \in F$, there exists a solution u in the space U.
2. The solution is unique.
3. The problem is stable on the spaces (U, F).

The problem is otherwise ill posed. For a long time, it was assumed that a mathematical problem must satisfy the above conditions and that an applied problem should be formulated in the same manner [courant62]. This assumption, however, was revealed to be invalid after many physical phenomena were studied.

1.3 Ill-Posed Problems Described by Linear Equations

In general, ill-posed problems arise in a wide variety of applied physics and engineering areas such as nuclear physics, plasma physics, radiophysics, and geophysics as well as electrical, nuclear, and mineral engineering. In order to generalize the solution of the linear equation (1.1), we need the following operator equations of the first class[1]:

$$Au = f, \tag{1.3}$$

[1] If u is defined on the discrete space, each element of f in (1.3) represents a weighted sum, and we call it the linear or the first order operation. On the other hand, if u is defined on the real space, the corresponding operation is called the *first class* instead of the *first order*.

where f represents the given, incomplete, or distorted data, u the complete or original data to be estimated, and A the operator which can have various forms.

Suppose $u(t)$ is an unknown function in space U, $f(t)$ a known function in space F, and that operator A is defined by

$$Au = \int_a^b K(t,s)u(s)\,ds. \tag{1.4}$$

Equation (1.3) then becomes the Fredholm integral equation of the first kind, such that

$$\int_a^b K(t,s)u(s)\,ds = f(t), \quad \text{for} \quad c \le t \le d, \tag{1.5}$$

where $[c, d]$ represents the domain on which f is defined.

Equation (1.5) is almost always an ill-posed problem. This is true because of the instability resulting from a bounded perturbation in the given data causing an unbounded value of the integration in Eq. (1.5). Detailed analysis about the existence and stability of a solution for Eq. (1.5) can be found in [tikhonov77].

In this section we will describe ill-posed problems by analyzing a discrete counterpart of Eq. (1.5). This discrete counterpart better represents digital image processing and computer vision problems.

In Eq. (1.3), suppose that u is an unknown vector, f is a known vector, and the A operator represents a square matrix D, with elements d_{ij}. Equation (1.3) then becomes a linear equation having an $N \times N$ square matrix D, such that

$$Du = f. \tag{1.6}$$

When U and F represent Hilbert spaces ($u \in U$ and $f \in F$), the problem denoted by Eq. (1.6) is well posed if there exists a unique least square solution which continuously depends on the data [nashed81].

If we consider Eq. (1.6) as the discretization of an ill-posed continuous problem given in Eq. (1.5), quantization error and noise may be added to the data, as in Eq. (1.7).

$$Du + v = f, \tag{1.7}$$

where the D matrix has a number of zero, or very small singular values. As a result, ill posedness of the continuous problem translates into an ill-conditioned matrix D, with additive quantization error v [kats91]. Various types of difficulties exist in solving an ill-conditioned linear equation as discussed below.

- Singularity check: If D is nonsingular, a unique solution vector u exists. If D is singular, a solution which is not unique only exists when a special condition is satisfied. Because we want to have a unique solution, it is generally necessary to check the singularity of D before beginning to solve the linear equation. A singularity check for the $N \times N$ square matrix can be made by investigating the determinant of D. This requires approximately N^3 operations. When the dimension of the system becomes large, a singularity check is a significant burden.

- Observation and numerical errors: In practical problems, we cannot avoid observation errors incorporated with data. With observation errors, the linear equation can be rewritten as $\widetilde{D}\widetilde{u} = \widetilde{f}$, where $\| \widetilde{D} - D \| \leq \delta$ and $\| \widetilde{f} - f \| \leq \delta$. In this case, the symbol $\sim$ is used to represent observed or measured data. Even if D is nonsingular, $\widetilde{D}$ may become singular or near singular. Furthermore, the error in the solution $\| \widetilde{u} - u \|$ may not be bounded by a sufficiently small value. In either case, or both, a stable and meaningful solution cannot be obtained.

1.4 Solving Ill-Posed Problems

It is not practical to calculate the solution of ill-posed problems by the direct method. In this section we introduce a selection method which serves as a general approach for estimating an approximate solution to the ill-posed problem. In the case of the ill-posed problem, approximate solutions are determined that are stable under small changes in the initial data based on the use of supplementary information. The concept of the selection method is to obtain multiple solutions of well-posed problems that are near the original ill-posed problem, in some sense. Next, one additional problem is formulated based on additional information from prior well-posed problem solutions.

Example 1.1 Consider the relationship

$$Du = f, \tag{1.8}$$

where $D = \begin{bmatrix} 2 & 1 \\ 2 & 2 \end{bmatrix}$ and $u = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$. The operation of Du yields the vector, $f = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$. Now suppose that we desire the solution u, given D and f. Since matrix D is invertible, the problem is well posed. We can solve for u directly by calculating D^{-1}, that is, $u = D^{-1}f = \begin{bmatrix} 1 & -0.5 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$.

Alternatively, assume we know that the solution for u exists inside the circle of radius 2 and centered at $[1, 1]^{\mathrm{T}}$. If we have three candidate solutions in the circle, $a = [0, 0]^{\mathrm{T}}$, $b = [1, 0]^{\mathrm{T}}$, and $c = [2, 0]^{\mathrm{T}}$, then calculate $Da = [0, 0]^{\mathrm{T}}$, $Db = [2, 2]^{\mathrm{T}}$, and $Dc = [4, 4]^{\mathrm{T}}$. Finally, we choose c as the approximate solution based on the fact that Dc is the closest to f in the Euclidean distance sense. This selection process is depicted in Fig. 1.4.

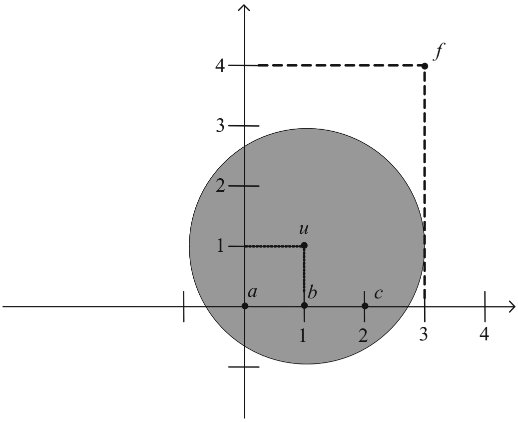

Fig. 1.4 An example of the selection method for solving an approximated linear equation with supplementary information

To describe the selection method for generally ill-posed problems, we consider an operator equation of the first kind

$$Au = f, \tag{1.9}$$

where u and f belong to metric spaces U and F, respectively, and A represents an operator that maps U onto F. Given subclass $M \subset U$, the general selection method is defined by calculating a class of Au for $u \in M$ and then taking u_0 such that

$$\rho_F(Au_0, f) = \inf_{u \in M} \rho_F(Au, f), \tag{1.10}$$

where u_0 is determined such that $\rho_F\ (Au,f)$ is minimized over $u \in M$.

The selection method can be accepted for solving a wide variety of ill-posed problems due to the use of the following proposition.

Proposition 1.1 *If, for $u_n \in M$, $\rho_f\ (Au_n,f)$ approaches zero as n becomes infinitely large, then $\rho_U(u_n, u_T) \rightarrow 0$ also approaches zero with infinitely large n, where u_T represents the exact solution. This proposition can be proved by using the following lemma.*

Lemma 1.1 *Suppose that a compact subset U of a metric space U_0 is mapped onto a subset F of a metric space F_0. If the mapping $U \rightarrow F$ is continuous and one to one, the inverse mapping $F \rightarrow U$ is also continuous.*

Proof of this lemma can be found in [tikhonov77]. An element of $\widetilde{u} \in M$ minimizing the functional $\rho_F\ (Au,f)$ on the set M is called a *quasisolution* of Eq. (1.9).

1.5 Image Restoration

Any image acquired by optical, electronic, or numerical means cannot be free from degradation due to the inherent errors produced in sensing, transmitting, and processing the image. Image restoration is concerned with estimating an image and making it as close as possible to the original image prior to degradation. The degradation may occur in the form of sensor noise, out-of-focus blur, motion blur, random atmospheric turbulence, and so forth. Most forms of image degradation can be represented, or approximated, by the linear equation

$$y = Dx + \eta, \tag{1.11}$$

where y and x represent vectors whose elements are sets of two-dimensionally distributed pixel intensity values of the observed and original images, respectively. Matrix D represents a degradation operation. Noise vector η plays a role in incorporating sensor noise and numerical errors into the degradation process [jain89, kats91]. In the previous section, Fig. 1.2b and Eqs. (1.1) and (1.2) showed a simple example for image degradation.

Accuracy of the solution of Eq. (1.11) depends on both the conditioning of the matrix D and the amount of noise η. Intuitively, the conditioning of a matrix (i.e., the accuracy of the represented information) becomes less reliable as the size and bandwidth of the matrix increases. Due to the nature of the general image degradation model, matrix D is almost always ill conditioned. In other words, Eq. (1.11), which deals with a large-size image and severe distortion, is an ill-posed problem.

Because matrix D is either singular or ill conditioned, an approximate solution of the transformed, well-posed equation is used to replace the original solution. One popular such transformation is given by

$$b = Tx, \tag{1.12}$$

where $T = D^{\mathrm{T}}D + \lambda C^{\mathrm{T}}C$ and $b = D^{T}y$. If we assume that λ is equal to zero, we can easily see that the solution for Eq. (1.12) is identical to the solution for Eq. (1.11) by ignoring noise η. However, matrix $D^{\mathrm{T}}D$ is better conditioned than D because the energy of diagonal elements increases by multiplying the matrix by itself. If λ has an appropriate value, the energy of T is further concentrated on the diagonal elements with a suitable choice of matrix C, which serves as a high-pass filter in many signal restoration problems. See Sects. 1.9, 1.10, and 1.11 for a more detailed analysis and implementation of matrix C.

In transforming the ill-posed problem into the well-posed one, supplementary or a priori information about the solution is incorporated by using the filtering term $C^{\mathrm{T}}C$. By appropriately selecting the value λ, we can make matrix T as well conditioned as desired.

Figure 1.5 shows the entire process of image degradation and the corresponding restoration. The 256×256 8-bit gray-level image is used for the original data x. As

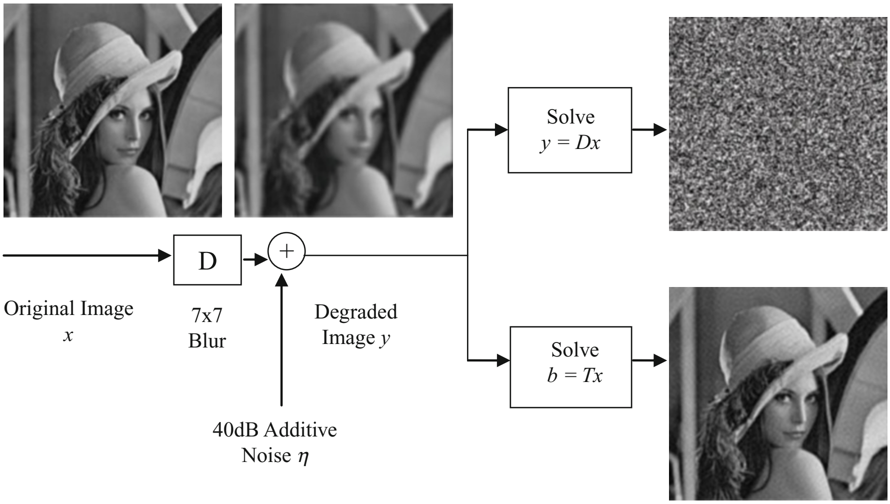

Fig. 1.5 General image degradation and restoration processes

the degradation operation, a 7×7 pixel averaging (blurring) and 40 dB additive noise are used. When we directly solve the ill-posed problem, we usually get a very poorly estimated image. On the other hand, when we solve the translated well-posed equation, we get an acceptably well-estimated image. In transforming the ill-posed problem into a well-posed one, we use supplementary smoothness information, which assumes that the original image does not have abruptly changed data patterns.

1.6 Image Interpolation

Image interpolation is used to obtain a higher-resolution image from a low-resolution image. Image interpolation is very important in high-resolution or multi-resolution image processing. More specifically, it can be used in changing the format of various types of images and videos and in increasing the resolution of images in a purely digital or numerical method.

Let $x_C\ (p, q)$ represent a two-dimensional spatially continuous image, and let $x(m, n)$ represent the corresponding digital image obtained by sampling $x_C\ (p, q)$, with sampling size $N \times N$, such that

$$x(m, n) = x_C(mT_\mathrm{v}, nT_\mathrm{h}), \quad \text{for } m, n = 0, 1, \ldots, N - 1 \tag{1.13}$$

where T_v and T_h represent the vertical and the horizontal sampling intervals, respectively. In a similar way, the image with four times lower resolution in both horizontal and vertical directions can be represented as

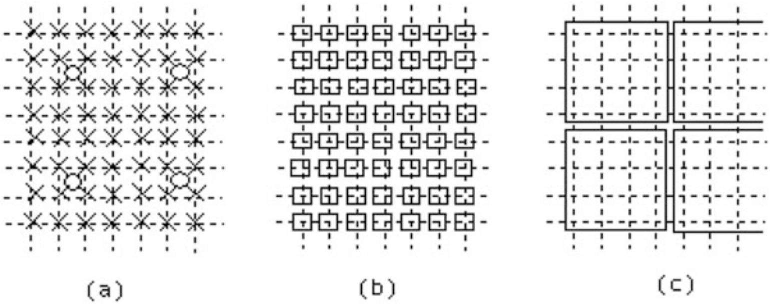

Fig. 1.6 Relationship between two images with different resolutions: (**a**) Sampling grids for two different images; "O" for $x_{1/4}$ and "X" for x, (**b**) an array structure of photodetectors for $x(m, n)$ and (**c**) an array structure of photodetectors for $x_{1/4}(m, n)$

$$x_{1/4}(m,n) = \frac{1}{16}\sum_{i=0}^{3}\sum_{j=0}^{3} x(4m+i, 4n+j), \text{ for } m,n = 0,1,\ldots,\frac{N}{4}-1, \quad (1.14)$$

where the subscript 1/4 represents four times downsampling.

Two different sampling grids for images in Eqs. (1.13) and (1.14) are shown in Fig. 1.6a. Array structures of photodetectors corresponding to the images in Eqs. (1.13) and (1.14) are shown in Fig. 1.6b, c, respectively.

A discrete linear space-invariant degradation model for an $\frac{N}{4} \times \frac{N}{4}$ low-resolution image, which is obtained by subsampling the original $N \times N$ high-resolution image, can be given as

$$y = Hx + \eta, \quad (1.15)$$

where the $N^2 \times 1$ vector x represents the lexicographically ordered high-resolution image and $\left(\frac{N}{4}\right)^2 \times 1$ vectors y and η the subsampled low-resolution and noise images, respectively. The $\left(\frac{N}{4}\right)^2 \times N^2$ matrix H represents the series of low-pass filtering and subsampling and can be written as

$$H = H_1 \otimes H_1, \quad (1.16)$$

where $\otimes$ represents the Kronecker product and the $\left(\frac{N}{4}\right) \times N$ matrix H_1 represents the one-dimensional low-pass filtering and subsampling by a factor of 4, such as

$$H_1 = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 & 1 & 1 \end{bmatrix}. \quad (1.17)$$

To estimate the original high-resolution image from the low-resolution observation given in Eq. (1.15), we need to solve the corresponding inverse problem. Because H is not a square matrix, a unique solution does not exist for the inverse problem. Instead, we may obtain the solution that minimizes

$$f(x) = ||y - Hx||^2, \tag{1.18}$$

which is clearly an ill-posed problem because the solution is not unique. To select one good solution from the set of feasible solutions, we must use additional information about the solution, such as smoothness or finite bandwidth. One typical way to incorporate the additional information into the minimization process is to minimize

$$f(x) = ||y - Hx||^2 + \lambda||Cx||^2, \tag{1.19}$$

where C represents a filter that extracts a certain kind of frequency component, and λ the scalar value that controls the utilization of additional information.

1.7 Motion Estimation

Motion estimation is one of the fundamental problems in image processing and computer vision. In approaching this problem, two-dimensional image plane motion estimation serves as a theoretical basis of motion-compensated video coding standards. Three-dimensional object motion estimation is used for reconstructing three-dimensional shapes and object tracking systems.

Consider a pair of similar images which can be obtained from either two adjacent image frames in a moving image sequence, stereo image pairs, or a pair of synthetic aperture radar (SAR) images. Those images can be depicted as in Fig. 1.7. The motion estimation problem can be posed as the estimation of *image*

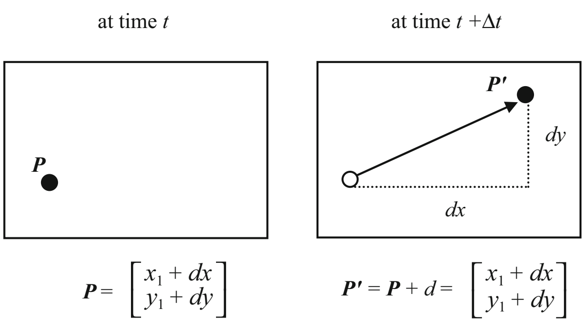

Fig. 1.7 A pair of images in which the point $\boldsymbol{P}$ in the *left image* moves onto $\boldsymbol{P'}$ in the *right image*

plane correspondence vectors or simply *motion vectors* denoted by $d = [dx, dy]^T$. Let $I(x, y, t)$ be the intensity value of the (x, y) -th pixel at time t. We can then write

$$I(x + dx, y + dy, t + \Delta t) = I(x, y, t), \tag{1.20}$$

where we assume that there is no intensity variation between two corresponding points.

Because the correspondence vector relates two different positions in each image frame, the motion estimation problem can be considered as a correspondence problem.

Among various motion vector estimation methods, the block-matching algorithm is one that minimizes the sum of absolute difference in the neighborhood of the point of interest, such as

$$\text{minimize } f(dx, dy) = \sum_{(x,y)\in S} |I(x + dx, y + dy, t + \Delta t) - I(x, y, t)|, \tag{1.21}$$

where S represents the neighborhood for the point of interest.

It is well known that motion estimation cannot be free from the following three problems [tekalp95].

1. Existence occluded problem: On the region, no correspondence can be established.
2. Uniqueness problem: If, for example, a line moves along a certain direction, many different displacements can match the two lines. It is called an aperture problem.
3. Continuity problem: Motion estimation is highly sensitive to the presence of noise. A small amount of noise may result in a large deviation in the estimated motion vector.

Because of the above-stated problems, motion estimation is an ill-posed problem. Therefore, the estimated motion vector, without proper constraints, cannot be a good approximation for the real motion.

1.8 Volumetric Models for Shape Representation

Fundamental problems in computer vision can be stated as follows:

1. Two-dimensional images are inadequate for defining the three-dimensional world.
2. More than one three-dimensional scene can produce identical two-dimensional images [ber88].

Therefore, recovery of a three-dimensional scene from two-dimensional images is an ill-posed problem. The relationship between the three-dimensional real world

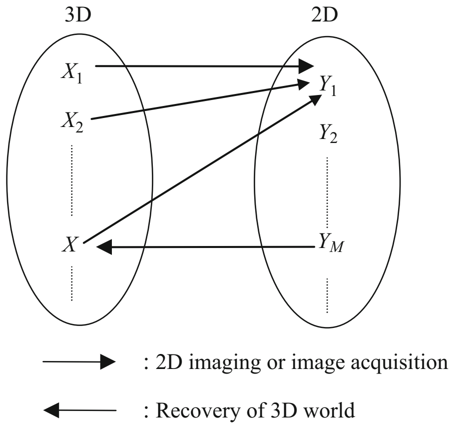

Fig. 1.8 The relationship between the real-world and two-dimensional images

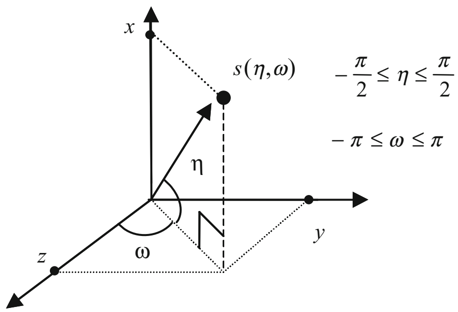

Fig. 1.9 Representation of a point in three-dimensional space

and two-dimensional images can be depicted in Fig. 1.8. Right-hand arrows represent two-dimensional image formulation or acquisition, while the left-hand arrows represent recovery of the three-dimensional world.

Due to its ill-posed nature, three-dimensional scene or shape recovery requires additional knowledge that may be incorporated in models that represent shapes. The reasons for using models for shape recovery can be summarized as follows:

1. Data compression becomes possible by using a small number of model parameters.
2. The resulting shape can be immunized against erroneous data.
3. Higher-level description is possible.
4. We can deduce occluded or missing data depending on the model.

In this section, we will introduce the superquadrics as a volumetric model for three-dimensional scene recovery [solina90]. Figure 1.9 shows a point on the superquadric surface in the three-dimensional coordinate. The point $s(\eta, w)$ can be described mathematically as

$$s(\eta, w) = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_1 \cos^{\varepsilon_1}(\eta) \cos^{\varepsilon_2}(w) \\ a_2 \cos^{\varepsilon_1}(\eta) \sin^{\varepsilon_2}(w) \\ a_3 \sin^{\varepsilon_1}(\eta) \end{bmatrix}, \tag{1.22}$$

where a_1, a_2, and a_3 represent the size of the superquadric in the x, y, and z directions, respectively. ε_1 and ε_2 represent the measure of squareness in the latitude and longitude, respectively. According to the signs and magnitudes of ε_1 and ε_2, the corresponding superquadrics may have the form of a sphere, a cylinder, a parallelepiped, etc.

From Eq. (1.22), any point on the surface of a superquadric satisfies the following relationships:

$$(x/a_1)^{1/\varepsilon_2} = (\cos\eta)^{\varepsilon_1/\varepsilon_2} \cos w, \tag{1.23}$$

$$(y/a_2)^{1/\varepsilon_2} = (\cos\eta)^{\varepsilon_1/\varepsilon_2} \sin w, \tag{1.24}$$

and

$$(z/a_3)^{1/\varepsilon_1} = \sin\eta \tag{1.25}$$

On the surface of superquadric, it is also true that

$$F(x, y, z) = \left[\left\{ \left(\frac{x}{a_1}\right)^{2/\varepsilon_2} + \left(\frac{y}{a_2}\right)^{2/\varepsilon_2} \right\}^{\varepsilon_2/\varepsilon_1} + \left(\frac{z}{a_3}\right)^{2/\varepsilon_1} \right]^{\varepsilon_1} = 1, \tag{1.26}$$

where we call $F(x, y, z)$ the inside-outside function.

If the coordinate on which the superquadric is defined, rotated, and translated by (ϕ, θ, φ) and (p_x, p_y, p_z) from the three-dimensional world coordinate system, (x, y, z) can be expressed by the following homogeneous transformation:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}, \tag{1.27}$$

where R represents the rotation matrix.

Finally, the model of a superquadric surface is represented by the 11 parameters, that is, three-scale parameters (a_1, a_2, a_3), two measure of squareness parameters $(\varepsilon_1, \varepsilon_2)$, three rotation parameters (ϕ, θ, φ), and three translation parameters (p_x, p_y, p_z). For simplicity, let $\{b_1, b_2, \ldots, b_{11}\}$ represent the 11 parameters.

Given N three-dimensional surface points, such as (x_w^i, y_w^i, z_w^i), for $i = 1, \ldots, N$, the optimum superquadric that fits the data points can be obtained from the following optimization problem: