1 Introduction

The proof is straightforward, and thus omitted

Ja-Chen Lin and Wen-Hsiang Tsai[1]

1.1 Concerning this book

This is an important observation: This book does NOT have enough information to tell you how to implement significant large systems. It teaches general principles. You MUST make use of the literature when you get down to the gnitty gritty.

We have written this book at two levels, the principal level being introductory. "Introductory" does not mean "easy" or "simple" or "doesn't require math." Rather, the introductory topics are those which need to be mastered before the advanced topics can be understood.

In addition, the book is intended to be useful as a reference. When you have to study a topic in more detail than is covered here, in order, for example, to implement a practical system, we have tried to provide adequate citations to the relevant literature to get you off to a good start.

We have tried to write in a style aimed directly toward the student and in a conversational tone.

We have also tried to make the text readable and entertaining. Words which are deluberately missppelled for humorous affects should be ubvious. Some of the humor runs to exaggeration and to puns; we hope you forgive us.

We did not attempt to cover every topic in the machine vision area. In particular, nearly all papers in the general areas of optical character recognition and face recognition have been omitted; not to slight these very important and very successful application areas, but rather because the papers tend to be rather specialized; in addition, we simply cannot cover everything.

There are two themes which run through this book: ***consistency*** and ***optimization***. Consistency is a conceptual tool, implemented as a variety of algorithms, which helps machines to recognize images – they fuse information from local measurements to make global conclusions about the image. Optimization is the mathematical mechanism used in virtually every chapter to accomplish the objectives of that chapter, be they pattern classification or image matching.

[1] Ja-Chen Lin and Wen-Hsiang Tsai, "Feature-preserving Clustering of 2-D Data for Two-class Problems Using Analytical Formulas: An Automatic and Fast Approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(5), 1994.

These two topics, consistency and optimization, are so important and so pervasive, that we point out to the student, in the conclusion of each chapter, exactly where those concepts turned up in that chapter. So read the chapter conclusions. Who knows, it might be on a test.

1.2 Concerning prerequisites

To find out if you meet this criterion, answer the following question: What do the following words mean? "transpose," "inverse," "determinant," "eigenvalue." If you do not have any idea, do not take this course!

The target audience for this book is graduate students or advanced undergraduates in electrical engineering, computer engineering, computer science, math, statistics, or physics. To do the work in this book, you must have had a graduate-level course in advanced calculus, and in statistics and/or probability. You need either a formal course or experience in linear algebra.

Many of the homeworks will be projects of sorts, and will be computer-based. To complete these assignments, you will need a hardware and software environment capable of

You will have to write programs in C (yes, C or C++, not Matlab) to complete this course.

(1) declaring large arrays (256×256) in C
(2) displaying an image
(3) printing an image.

Software and data used in the book can be found at www.cambridge.org/9780521830461.

We are going to insist that you write programs, and that you write them at a relatively low level. Some of the functionality that you will be coding is available in software packages like Matlab. However, while you learn something by simply calling a function, you learn more by writing and debugging the code yourself. Exceptions to this occur, of course, when the coding is so extensive that the programming gets in the way of the image analysis. For that reason, we provide the student with a library of subroutines which allow the student to ignore details like data type, byteswapping, file access, and platform dependencies, and instead focus on the logic of making image analysis algorithms work.

You should have an instructor, and if you do, we strongly recommend that you GO to class, even though all the information you really need is in this book. Read the assigned material in the text, then go to class, then read the text material again. Remember:

A hacker hermit named Dave
Tapped in to this course in his cave.
He had to admit
He learned not a bit.
But look at the money he saved.

And now, on to the technical stuff.

1.3 Some terminology

Students usually confuse machine vision with image processing. In this section, we define some terminology that will clarify the differences between the contents and objectives of these two topics.

1.3.1 Image processing

Many people consider the content of this course as part of the discipline of image processing. However, a better use of the term is to distinguish between image processing and machine vision by the intent. "Image processing" strives to make images look better, and the output of an image processing system is an image. The output of a "machine vision" system is information about the content of the image. The functions of an image processing system may include enhancement, coding, compression, restoration, and reconstruction.

Enhancement

Enhancement systems perform operations which make the image look better, as perceived by a human observer. Typical operations include contrast stretching (including functions like histogram equalization), brightness scaling, edge sharpening, etc.

Coding

Coding is the process of finding efficient and effective ways to represent the information in an image. These include quantization methods and redundancy removal. Coding may also include methods for making the representation robust to bit-errors which occur when the image is transmitted or stored.

Compression

Compression includes many of the same techniques as coding, but with the specific objective of reducing the number of bits required to store and/or transmit the image.

Restoration

Restoration concerns itself with fixing what is wrong with the image. It is unlike enhancement, which is just concerned with making images look better. In order to "correct" an image, there must be some model of the image degradation. It is common in restoration applications to assume a deterministic blur operator, followed by additive random noise.

Reconstruction

Reconstruction usually refers to the process of constructing an image from several partial images. For example, in computed tomography (CT),[2] we make a large number, say 360, of x-ray projections through the subject. From this set of one-dimensional signals, we can compute the actual x-ray absorption at each point in the two-dimensional image. Similar methods are used in positron emission tomography (PET), magnetic resonance imagery (MRI), and in several shape-from-X algorithms which we will discuss later in this course.

1.3.2 Machine vision

Machine vision is the process whereby a machine, usually a digital computer, automatically processes an image and reports "what is in the image." That is, it recognizes the content of the image. Often the content may be a machined part, and the objective is not only to locate the part, but to inspect it as well. We will in this book discuss several applications of machine vision in detail, such as automatic target recognition (ATR), and industrial inspection. There are a wide variety of other applications, such as determining the flow equations from observations of fluid flow [1.1], which time and space do not allow us to cover.

The terms "computer vision" and "image understanding" are often also used to denote machine vision.

Machine vision includes two components – measurement of features and pattern classification based on those features.

Measurement of features

The measurement of features is the principal focus of this book. Except for Chapters 14 and 15, in this book, we focus on processing the elements of images (pixels) and from those pixels and collections of pixels, extract sets of measurements which characterize either the entire image or some component thereof.

Pattern classification

Pattern classification may be defined as the process of making a decision about a measurement. That is, we are **given** a measurement or set of measurements made on an unknown object. From that set of measurements with knowledge about the possible *classes* to which that unknown might belong, we make a decision. For

[2] Sometimes, CT is referred to as "CAT scanning." In that case, CAT stands for "computed axial tomography." There are other types of tomography as well.

example, the set of possible classes might be men and women and one measurement which we could make to distinguish men from women would be height (clearly, height is not a very good measurement to use to distinguish men from women, for if our decision is that anyone over five foot six is male we will surely be wrong in many instances).

Pattern recognition

Pattern recognition may be defined as the process of assigning unknowns to classes just as in the definition of pattern classification. However, the definition is extended to include the process of making the measurements.

1.4 Organization of a machine vision system

Fig. 1.1 shows schematically, at the most basic level, the organization of a machine vision system. The unknown is first measured and the values of a number of *features* are determined. In an industrial application, such features might include the length, width, and area of the image of the part being measured. Once the features are measured, their numerical values are passed to a process which implements a *decision rule*. This decision rule is typically implemented by a subroutine which performs calculations to determine to which class the unknown is most likely to belong based on the measurements made.

As Fig. 1.1 illustrates, a machine vision system is really a fairly simple architectural structure. The details of each module may be quite complex, however, and many different options exist for designing the classifier and the feature measuring system. In this book, we mention the process of classifier design. However, the process of determining and measuring features is the principal topic of this book.

The "feature measurement" box can be further broken down into more detailed operations as illustrated in Fig. 1.2. At that level, the organization chart becomes more complex because the specific operations to be performed vary with the type of image and the objective of the tasks. Not every operation is performed in every application.

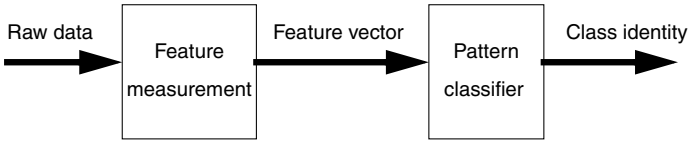

Fig. 1.1. Organization of a machine vision system.

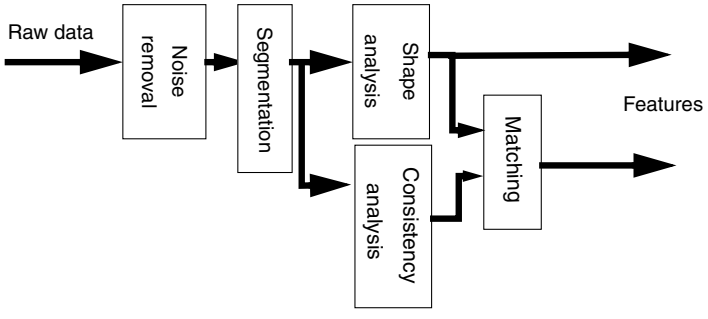

Fig. 1.2. Some components of a feature characterization system. Many machine vision applications do not use every block, and information often flows in other ways. For example, it is possible to perform matching directly on the image data.

1.5 The nature of images

We will pay much more attention to the nature of images in Chapter 4. We will observe that there are several different types of images as well as several different ways to represent images. The types of images include what we call "pictures," that is, two-dimensional images. In addition, however, we will discuss three-dimensional images and range images. We will also consider different representations for images, including iconic, functional, linear, and relational representations.

1.6 Images: Operations and analysis

Some equivalent words.

We will learn many different operations to perform on images. The emphasis in this course is "image analysis," or "computer vision," or "machine vision," or "image understanding." All these phrases mean the same thing. We are interested in making measurements on images with the objective of providing our machine (usually, but not always, a computer) with the ability to recognize what is in the image. This process includes several steps:

- *denoising* – all images are noisy, most are blurred, many have other distortions as well. These distortions need to be removed or reduced before any further operations can be carried out. We discuss two general approaches for denoising in Chapters 6 and 7.
- *segmentation* – we must segment the image into meaningful regions. Segmentation is covered in Chapter 8.
- *feature extraction* – making measurements, geometric or otherwise, on those regions is discussed in Chapter 9.

- *consistency* – interpreting the entire image from local measurements is covered in Chapters 10 and 11.
- *classification and matching* – recognizing the object is covered in Chapter 12 through Chapter 16.

So turn to the next chapter. (Did you notice? No homework assignments in this chapter? Don't worry. We'll fix that in future chapters.)

Reference

[1.1] C. Shu and R. Jain, "Vector Field Analysis for Oriented Patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(9), 1994.

2 Review of mathematical principles

Everything, once understood, is trivial

W. Snyder

2.1 A brief review of probability

Let us imagine a statistical experiment: rolling two dice. It is possible to roll any number between two and twelve (inclusive), but as we know, some numbers are more likely than others. To see this, consider the possible ways to roll a five.

We see from Fig. 2.1 that there are four possible ways to roll a five with two dice. Each event is **independent**. That is, the chance of rolling a two with the second die (1 in 6) does not depend at all on what is rolled with die number 1.

Independence of events has an important implication. It means that the *joint probability* of the two events is equal to the product of their individual probabilities and the conditional probabilities:

$$Pr(a|b)P(b) = Pr(a)Pr(b) = Pr(b|a)Pr(a) = Pr(a, b). \tag{2.1}$$

In Eq. (2.1), the symbols a and b represent **events**, e.g., the rolling of a six. $Pr(b)$ is the probability of such an event occurring, and $Pr(a \mid b)$ is the **conditional probability** of event a occurring, given that event b has occurred.

In Fig. 2.1, we tabulate all the possible ways of rolling two dice, and show the resulting number of different ways that the numbers from 2 to 12 can occur. We note that 6 different events can lead to a 7 being rolled. Since each of these events is equally probable (1 in 36), then a 7 is the most likely roll of two dice. In Fig. 2.2 the information from Fig. 2.1 is presented in graphical form.

In pattern classification, we are most often interested in the probability of a particular measurement occurring. We have a problem, however, when we try to plot a graph such as Fig. 2.2 for a continuously-valued function. For example, how do we ask the question: "What is the probability that a man is six feet tall?" Clearly, the answer is zero, for an infinite number of possibilities could occur (we might equally well ask, "What is the probability that a man is (exactly) 6.314 159 267 feet tall?"). Still, we know intuitively that the likelihood of a man being six feet tall is higher than the likelihood of his being ten feet tall. We need some way of quantifying this intuitive notion of likelihood.

Sum		Number of ways
0		**0**
1		**0**
2	1–1	**1**
3	2–1, 1–2	**2**
4	1–3, 3–1, 2–2	**3**
5	2–3, 3–2, 4–1, 1–4	**4**
6	1–5, 5–1, 2–4, 4–2, 3–3	**5**
7	3–4, 4–3, 2–5, 5–2, 1–6, 6–1	**6**
8	2–6, 6–2, 3–5, 5–3, 4–4	**5**
9	3–6, 6–3, 4–5, 5–4	**4**
10	4–6, 6–4, 5–5	**3**
11	6–5, 5–6	**2**
12	6–6	**1**

Fig. 2.1. The possible ways to roll two dice.

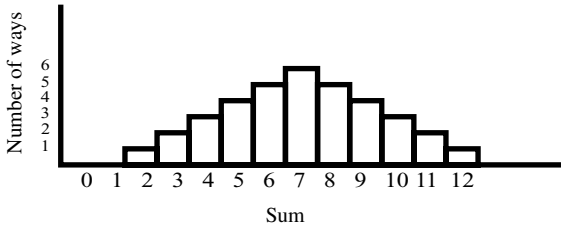

Fig. 2.2. The information of Fig. 2.1, in graphical form.

One question that does make sense is, "What is the probability that a man is **less than** six feet tall?" Such a function is referred to as a **probability distribution** function

$$P(x) = Pr(z < x) \tag{2.2}$$

for some measurement, z.

Fig. 2.3 illustrates the probability distribution function for the result of rolling two dice.

When we asked "what is the probability that a man is less than x feet tall?" we obtained the probability distribution function. Another well-formed question would be "what is the probability that a man's height is between x and $x + \Delta x$?" Such a question is easily answered in terms of the density function:

$$Pr(x \leq h < x + \Delta x) = Pr(h < x + \Delta x) - Pr(h < x) = P(x + \Delta x) - P(x)$$

Dividing by Δx and taking the limit as $\Delta x \to 0$, we see that we may define the **probability density function** as the derivative of the distribution function:

$$p(x) = \frac{d}{dx} P(x). \tag{2.3}$$

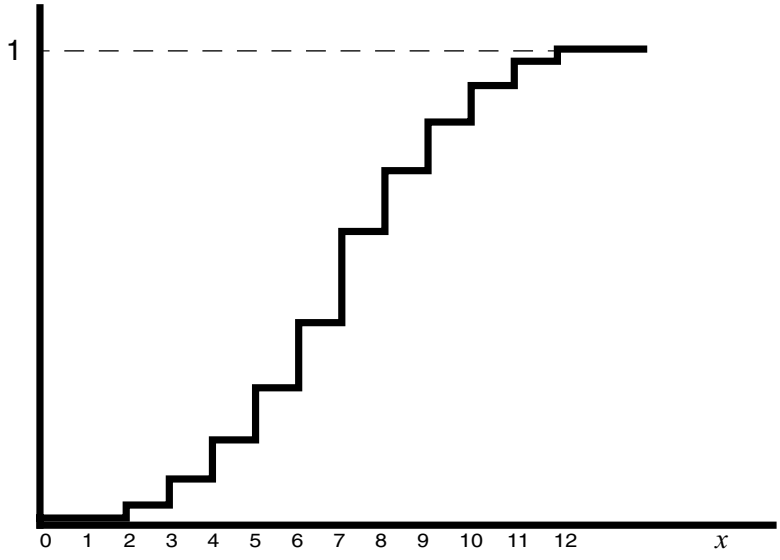

Fig. 2.3. The probability distribution of Fig. 2.2, showing the probability of rolling two dice to get a number LESS than x. Note that the curve is steeper at the more likely numbers.

$p(x)$ has all the properties that we desire. It is well defined for continuously-valued measurements and it has a maximum value for those values of the measurement which are intuitively most likely.

Furthermore:

$$\int_{-\infty}^{\infty} p(x)\,dx = 1, \tag{2.4}$$

which we must require, since **some** value will certainly occur.

2.2 A review of linear algebra

This section will serve more as a reference than a teaching aid, since you should know this material already.

In this section, we very briefly review vector and matrix operations. Generally, we denote vectors in boldface, scalars in lowercase Roman, and matrices in uppercase Roman.

Vectors are always considered to be column vectors. If we need to write one horizontally for the purpose of saving space in a document, we use transpose notation. For example, we denote a vector which consists of three scalar elements as:

$$\boldsymbol{v} = [x_1 \quad x_2 \quad x_3]^{\mathrm{T}}.$$

The inner product of two vectors is a scalar, $\boldsymbol{v} = \boldsymbol{a}^{\mathrm{T}}\boldsymbol{b}$. Its value is the sum of products