CAMBRIDGE

Cambridge University Press
978-1-107-01179-3 - Computer Vision: Models, Learning, and Inference
Simon J. D. Prince
Excerpt
More information

Chapter 1

Introduction

The goal of computer vision is to extract useful information from images. This has proved a surprisingly challenging task; it has occupied thousands of intelligent and creative minds over the last four decades, and despite this we are still far from being able to build a general-purpose "seeing machine."

Part of the problem is the complexity of visual data. Consider the image in Figure 1.1. There are hundreds of objects in the scene. Almost none of these are presented in a "typical" pose. Almost all of them are partially occluded. For a computer vision algorithm, it is not even easy to establish where one object ends and another begins. For example, there is almost no change in the image intensity at the boundary between the sky and the white building in the background. However, there is a pronounced change in intensity on the back window of the SUV in the foreground, although there is no object boundary or change in material here.

We might have grown despondent about our chances of developing useful computer vision algorithms if it were not for one thing: we have concrete proof that vision is possible because our own visual systems make light work of complex images such as Figure 1.1. If I ask you to count the trees in this image or to draw a sketch of the street layout, you can do this easily. You might even be able to pinpoint where this photo was taken on a world map by extracting subtle visual clues such as the ethnicity of the people, the types of cars and trees, and the weather.

So, computer vision is not impossible, but it is very challenging; perhaps this was not appreciated at first because what we perceive when we look at a scene is already highly processed. For example, consider observing a lump of coal in bright sunlight and then moving to a dim indoor environment and looking at a piece of white paper. The eye will receive far more photons per unit area from the coal than from the paper, but we nonetheless perceive the coal as black and the paper as white. The visual brain performs many tricks of this kind, and unfortunately when we build vision algorithms, we don't have the benefit of this preprocessing.

Nonetheless, there has been remarkable recent progress in our understanding of computer vision, and the last decade has seen the first large-scale deployments of consumer computer vision technology. For example, most digital cameras now have embedded algorithms for face detection, and at the time of writing the Microsoft Kinect (a peripheral that allows real-time tracking of the human body) holds the Guinness World Record

Figure 1.1 A visual scene containing many objects, almost all of which are partially occluded. The red circle indicates a part of the scene where there is almost no brightness change to indicate the boundary between the sky and the building. The green circle indicates a region in which there is a large intensity change but this is due to irrelevant lighting effects; there is no object boundary or change in the object material here.

for being the fastest-selling consumer electronics device ever. The principles behind both of these applications and many more are explained in this book.

There are a number of reasons for the rapid recent progress in computer vision. The most obvious is that the processing power, memory, and storage capacity of computers has vastly increased; before we disparage the progress of early computer vision pioneers, we should pause to reflect that they would have needed specialized hardware to hold even a single high-resolution image in memory. Another reason for the recent progress in this area has been the increased use of machine learning. The last 20 years have seen exciting developments in this parallel research field, and these are now deployed widely in vision applications. Not only has machine learning provided many useful tools, it has also helped us understand existing algorithms and their connections in a new light.

The future of computer vision is exciting. Our understanding grows by the day, and it is likely that artificial vision will become increasingly prevalent in the next decade. However, this is still a young discipline. Until recently, it would have been unthinkable to even try to work with complex scenes such as that in Figure 1.1. As Szeliski (2010) puts it, “It may be many years before computers can name and outline all of the objects in a photograph with the same skill as a two year old child.” However, this book provides a snapshot of what we have achieved and the principles behind these achievements.

Organization of the book

The structure of this book is illustrated in Figure 1.2. It is divided into six parts.

The first part of the book contains background information on probability. All the models in this book are expressed in terms of probability, which is a useful language for describing computer vision applications. Readers with a rigorous background in engineering mathematics will know much of this material already but should skim these chapters to ensure they are familiar with the notation. Those readers who do not have this background should read these chapters carefully. The ideas are relatively simple, but they underpin everything else in the rest of the book. It may be frustrating to be forced to read fifty pages of mathematics before the first mention of computer vision, but please trust me when I tell you that this material will provide a solid foundation for everything that follows.

The second part of the book discusses machine learning for machine vision. These chapters teach the reader the core principles that underpin all of our methods to extract useful information from images. We build statistical models that relate the image data to the information that we wish to retrieve. After digesting this material, the reader should understand how to build a model to solve almost any vision problem, although that model may not yet be very practical.

The third part of the book introduces graphical models for computer vision. Graphical models provide a framework for simplifying the models that relate the image data to the properties we wish to estimate. When both of these quantities are high-dimensional, the statistical connections between them become impractically complex; we can still define models that relate them, but we may not have the training data or computational power to make them useful. Graphical models provide a principled way to assert sparseness in the statistical connections between the data and the world properties.

The fourth part of the book discusses image preprocessing. This is not necessary to understand most of the models in the book, but that is not to say that it is unimportant. The choice of preprocessing method is at least as critical as the choice of model in determining the final performance of a computer vision system. Although image processing is not the main topic of this book, this section provides a compact summary of the most important and practical techniques.

The fifth part of the book concerns geometric computer vision; it introduces the projective pinhole camera – a mathematical model that describes where a given point in the 3D world will be imaged in the pixel array of the camera. Associated with this model are a set of techniques for finding the position of the camera relative to a scene and for reconstructing 3D models of objects.

Finally, in the sixth part of the book, we present several families of vision models that build on the principles established earlier in the book. These models address some of the most central problems in computer vision including face recognition, tracking, and object recognition.

The book concludes with several appendices. There is a brief discussion of the notational conventions used in the book, and compact summaries of linear algebra and optimization techniques. Although this material is widely available elsewhere, it makes the book more self-contained and is discussed in the same terminology as it is used in the main text.

At the end of every chapter is a brief notes section. This provides details of the related research literature. It is heavily weighted toward the most useful and recent papers and

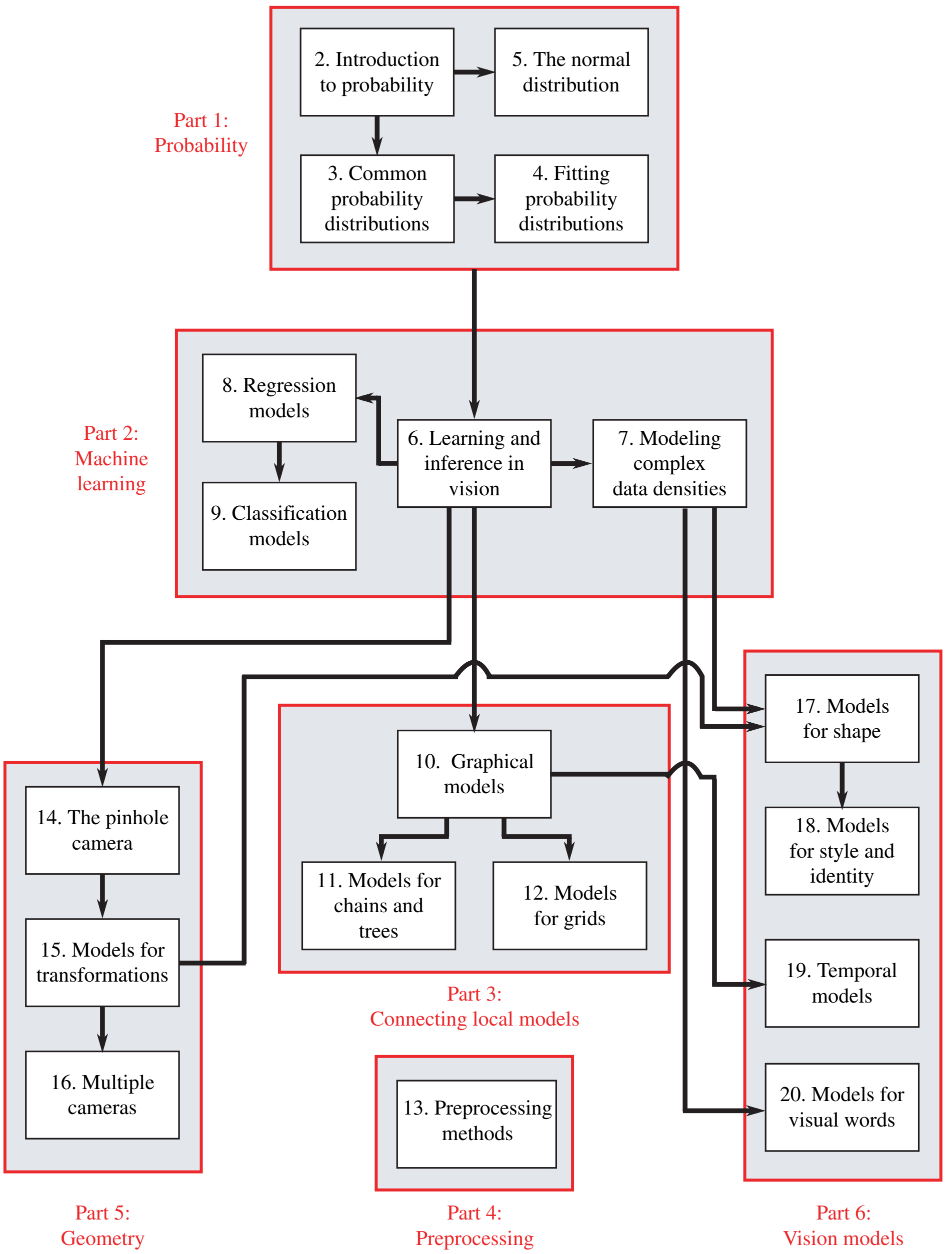

Figure 1.2 Chapter dependencies. The book is organized into six sections. The first section is a review of probability and is necessary for all subsequent chapters. The second part concerns machine learning and inference. It describes both generative and discriminative models. The third part concerns graphical models: visual representations of the probabilistic dependencies between variables in large models. The fourth part describes preprocessing methods. The fifth part concerns geometry and transformations. Finally, the sixth part presents several other important families of vision models.

does not reflect an accurate historical description of each area. There are also a number of exercises for the reader at the end of each chapter. In some cases, important but tedious derivations have been excised from the text and turned into problems to retain the flow of the main argument. Here, the solution will be posted on the main book Web site (http://www.computervisionmodels.com). A series of applications are also presented at the end of each chapter (apart from Chapters 1–5 and Chapter 10, which contain only theoretical material). Collectively, these represent a reasonable cross-section of the important vision papers of the last decade.

Finally, pseudocode for over 70 of the algorithms discussed is available and can be downloaded in a separate document from the associated Web site (http://www.computervisionmodels.com). Throughout the text, the symbol [⚙] denotes that there is pseudocode associated with this portion of the text. This pseudocode uses the same notation as the book and will make it easy to implement many of the models. I chose not to include this in the main text because it would have decreased the readability. However, I encourage all readers of this book to implement as many of the models as possible. Computer vision is a practical engineering discipline, and you can learn a lot by experimenting with real code.

Other books

I am aware that most people will not learn computer vision from this book alone, so here is some advice about other books that complement this volume. To learn more about machine learning and graphical models, I recommend '*Pattern Recognition and Machine Learning*' by Bishop (2006) as a good starting point. There are many books on preprocessing, but my favorite is '*Feature Extraction and Image Processing*' by Nixon and Aguado (2008). The best source for information about geometrical computer vision is, without a doubt, '*Multiple View Geometry in Computer Vision*' by Hartley and Zisserman (2004). Finally, for a much more comprehensive overview of the state of the art of computer vision and its historical development, consider '*Computer Vision: Algorithms and Applications*' by Szeliski (2010).

Part I

Probability

The first part of this book (Chapters 2–5) is devoted to a brief review of probability and probability distributions. Almost all models for computer vision can be interpreted in a probabilistic context, and in this book we will present all the material in this light. The probabilistic interpretation may initially seem confusing, but it has a great advantage: it provides a common notation that will be used throughout the book and will elucidate relationships between different models that would otherwise remain opaque.

So why is probability a suitable language to describe computer vision problems? In a camera, the three-dimensional world is projected onto the optical surface to form the image: a two-dimensional set of measurements. Our goal is to take these measurements and use them to establish the properties of the world that created them. However, there are two problems. First, the measurement process is noisy; what we observe is not the amount of light that fell on the sensor, but a noisy estimate of this quantity. We must describe the noise in these data, and for this we use probability. Second, the relationship between world and measurements is generally many to one: there may be many real-world configurations that are compatible with the same measurements. The chance that each of these possible worlds is present can also be described using probability.

The structure of Part I is as follows: in Chapter 2, we introduce the basic rules for manipulating probability distributions including the ideas of conditional and marginal probability and Bayes' rule. We also introduce more advanced ideas such as independence and expectation.

In Chapter 3, we discuss the properties of eight specific probability distributions. We divide these into two sets of four distributions each. The first set will be used to describe either the observed data or the state of the world. The second set of distributions model the parameters of the first set. In combination, they allow us to fit a probability model and provide information about how certain we are about the fit.

In Chapter 4, we discuss methods for fitting probability distributions to observed data. We also discuss how to assess the probability of new data points under the fitted model and how to take account of uncertainty in the fitted model when we do this. Finally, in Chapter 5, we investigate the properties of the multivariate normal distribution in detail. This distribution is ubiquitous in vision applications and has a number of useful properties that are frequently exploited in machine vision.

Readers who are very familiar with probability models and the Bayesian philosophy may wish to skip this part and move directly to Part II.

Cambridge University Press
978-1-107-01179-3 - Computer Vision: Models, Learning, and Inference
Simon J. D. Prince
Excerpt
More information

Chapter 2

Introduction to probability

In this chapter, we provide a compact review of probability theory. There are very few ideas, and each is relatively simple when considered separately. However, they combine to form a powerful language for describing uncertainty.

2.1 Random variables

A random variable x denotes a quantity that is uncertain. The variable may denote the result of an experiment (e.g., flipping a coin) or a real-world measurement of a fluctuating property (e.g., measuring the temperature). If we observe several instances $\{x_i\}_{i=1}^{I}$ then it might take a different value on each occasion. However, some values may occur more often than others. This information is captured by the probability distribution $Pr(x)$ of the random variable.

A random variable may be *discrete* or *continuous*. A discrete variable takes values from a predefined set. This set may be ordered (the outcomes 1–6 of rolling a die) or unordered (the outcomes "sunny," "raining," "snowing" upon observing the weather). It may be finite (there are 52 possible outcomes of drawing a card randomly from a standard pack) or infinite (the number of people on the next train is theoretically unbounded). The probability distribution of a discrete variable can be visualized as a histogram or a Hinton diagram (Figure 2.1). Each outcome has a positive probability associated with it, and the sum of the probabilities for all outcomes is always one.

Continuous random variables take values that are real numbers. These may be finite (the time taken to finish a 2-hour exam is constrained to be greater than 0 hours and less than 2 hours) or infinite (the amount of time until the next bus arrives is unbounded above). Infinite continuous variables may be defined on the whole real range or may be bounded above or below (the 1D velocity of a vehicle may take any value, but the speed is bounded below by 0). The probability distribution of a continuous variable can be visualized by plotting the *probability density function* (pdf). The probability density for an outcome represents the relative propensity of the random variable to take that value (see Figure 2.2). It may take any positive value. However, the integral of the pdf always sums to one.

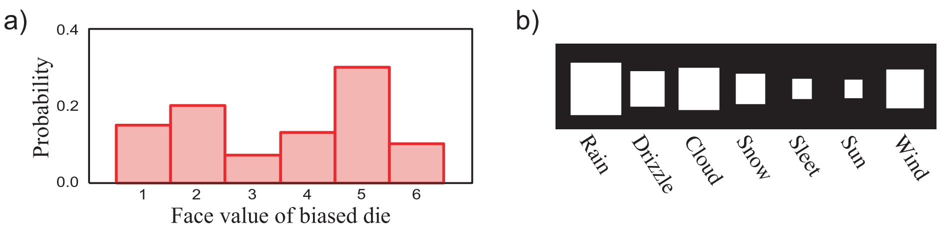

Figure 2.1 Two different representations for discrete probabilities a) A bar graph representing the probability that a biased six-sided die lands on each face. The height of the bar represents the probability, so the sum of all heights is one. b) A Hinton diagram illustrating the probability of observing different weather types in England. The area of the square represents the probability, so the sum of all areas is one.

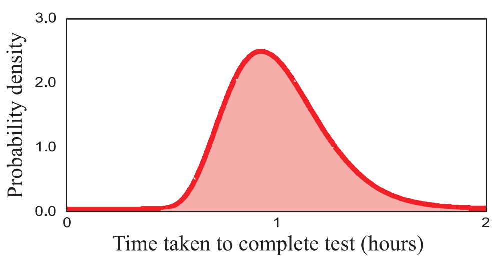

Figure 2.2 Continuous probability distribution (probability density function or pdf for short) for time taken to complete a test. Note that the probability density can exceed one, but the area under the curve must always have unit area.

2.2 Joint probability

Consider two random variables, x and y. If we observe multiple paired instances of x and y, then some combinations of the two outcomes occur more frequently than others. This information is encompassed in the *joint* probability distribution of x and y, which is written as $Pr(x,y)$. The comma in $Pr(x,y)$ can be read as the English word "and" so $Pr(x,y)$ is the probability of x *and* y. A joint probability distribution may relate variables that are all discrete or all continuous, or it may relate discrete variables to continuous ones (see Figure 2.3). Regardless, the total probability of all outcomes (summing over discrete variables and integrating over continuous ones) is always one.

In general, we will be interested in the joint probability distribution of more than two variables. We will write $Pr(x,y,z)$ to represent the joint probability distribution of scalar variables x, y, and z. We may also write $Pr(\mathbf{x})$ to represent the joint probability of all of the elements of the multidimensional variable $\mathbf{x} = [x_1, x_2, \ldots, x_K]^T$. Finally, we will write $Pr(\mathbf{x}, \mathbf{y})$ to represent the joint distribution of all of the elements from multidimensional variables $\mathbf{x}$ and $\mathbf{y}$.

2.3 Marginalization

We can recover the probability distribution of any single variable from a joint distribution by summing (discrete case) or integrating (continuous case) over all the other variables (Figure 2.4). For example, if x and y are both continuous and we know $Pr(x,y)$, then