CHAPTER

Vision, the challenge

1

1.1 INTRODUCTION—MAN AND HIS SENSES

Of the five senses—vision, hearing, smell, taste, and touch—vision is undoubtedly the one that man has come to depend upon above all others, and indeed the one that provides most of the data he receives. Not only do the input pathways from the eyes provide megabits of information at each glance but also the data rates for continuous viewing probably exceed 10 Mbps. However, much of this information is redundant and is compressed by the various layers of the visual cortex, so that the higher centers of the brain have to interpret abstractly only a small fraction of the data. Nonetheless, the amount of information the higher centers receive from the eyes must be at least two orders of magnitude greater than all the information they obtain from the other senses.

Another feature of the human visual system is the ease with which interpretation is carried out. We see a scene as it is—trees in a landscape, books on a desk, widgets in a factory. No obvious deductions are needed and no overt effort is required to interpret each scene; in addition, answers are effectively immediate and are normally available within a tenth of a second. Just now and again some doubt arises—e.g., a wire cube might be "seen" correctly or inside out. This and a host of other optical illusions are well known, although for the most part we can regard them as curiosities—irrelevant freaks of nature. Somewhat surprisingly, illusions are quite important, since they reflect hidden assumptions that the brain is making in its struggle with the huge amounts of complex visual data it is receiving. We have to pass by this story here (although it resurfaces now and again in various parts of this book). However, the important point is that we are for the most part unaware of the complexities of vision. Seeing is not a simple process: it is just that vision has evolved over millions of years, and there was no particular advantage in evolution giving us any indication of the difficulties of the task (if anything, to have done so would have cluttered our minds with irrelevant information and slowed our reaction times).

In the present-day and age, man is trying to get machines to do much of his work for him. For simple mechanistic tasks this is not particularly difficult, but

Computer Vision. DOI: http://dx.doi.org/10.1016/B978-0-12-809284-2.00001-0

for more complex tasks the machine must be given the sense of vision. Efforts have been made to achieve this, sometimes in modest ways, for well over 40 years. At first, schemes were devised for reading, for interpreting chromosome images, and so on; but when such schemes were confronted with rigorous practical tests, the problems often turned out to be more difficult. Generally, researchers react to finding that apparent "trivia" are getting in the way by intensifying their efforts and applying great ingenuity, and this was certainly so with early efforts at vision algorithm design. However, it soon became plain that the task really is a complex one, in which numerous fundamental problems confront the researcher, and the ease with which the eye can interpret scenes turned out to be highly deceptive.

Of course, one of the ways in which the human visual system gains over the machine is that the brain possesses more than 10^{10} cells (or neurons), some of which have well over 10,000 contacts (or synapses) with other neurons. If each neuron acts as a type of microprocessor, then we have an immense computer in which all the processing elements can operate concurrently. Taking the largest single man-made computer to contain several hundred million rather modest processing elements, the majority of the visual and mental processing tasks that the eye−brain system can perform in a flash have no chance of being performed by present-day man-made systems. Added to these problems of scale, there is the problem of how to organize such a large processing system and also how to program it. Clearly, the eye−brain system is partly hard-wired by evolution but there is also an interesting capability to program it dynamically by training during active use. This need for a large parallel processing system with the attendant complex control problems shows that computer vision must indeed be one of the most difficult intellectual problems to tackle.

So what are the problems involved in vision that make it apparently so easy for the eye, yet so difficult for the machine? In the next few sections an attempt is made to answer this question.

1.2 THE NATURE OF VISION

1.2.1 THE PROCESS OF RECOGNITION

This section illustrates the intrinsic difficulties of implementing computer vision, starting with an extremely simple example—that of character recognition. Consider the set of patterns shown in Fig. 1.1A. Each pattern can be considered as a set of 25 bits of information, together with an associated class indicating its interpretation. In each case imagine a computer learning the patterns and their classes by rote. Then any new pattern may be classified (or "recognized") by comparing it with this previously learnt "training set," and assigning it to the class of the nearest pattern in the training set. Clearly, test pattern (1) (Fig. 1.1B) will be allotted to class U on this basis. Chapter 13, Basic Classification Concepts,

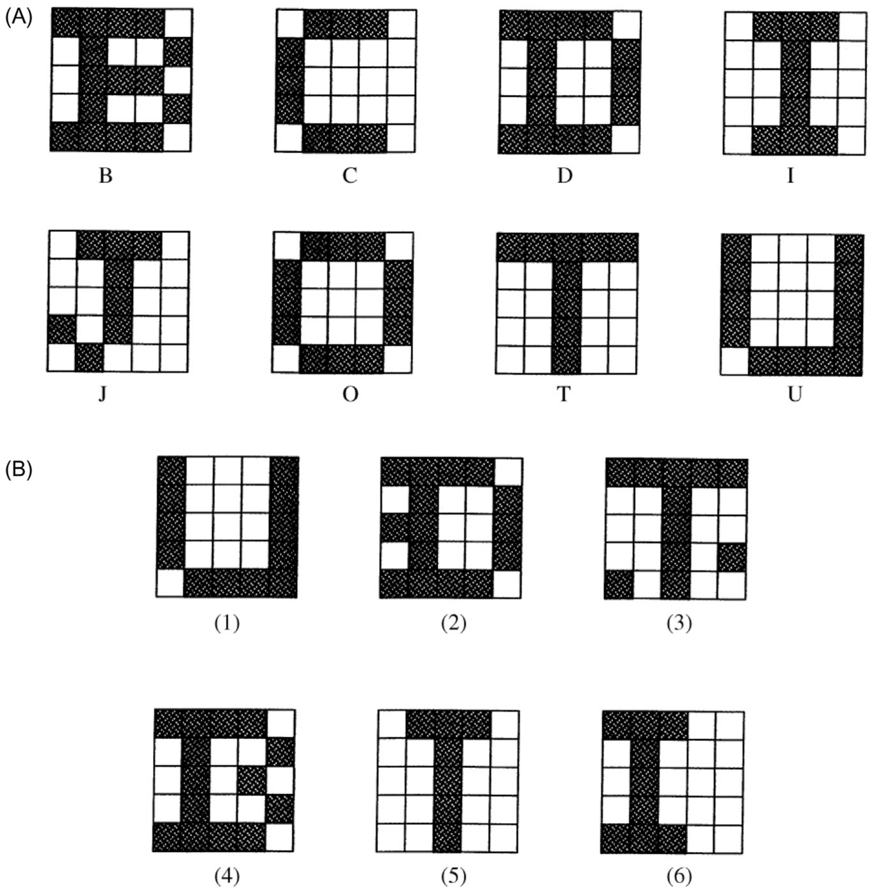

FIGURE 1.1

Some simple 25-bit patterns and their recognition classes used to illustrate some of the basic problems of recognition: (A) training set patterns (for which the known classes are indicated); (B) test patterns.

shows that this method is a simple form of the nearest neighbor approach to pattern recognition.

The scheme outlined above seems straightforward and is indeed highly effective, even being able to cope with situations where distortions of the test patterns occur or where noise is present: this is illustrated by test patterns (2) and (3). However, this approach is not always foolproof. First, there are situations where distortions or noise is excessive, so errors of interpretation arise. Second, there are situations where patterns are not badly distorted or subject to obvious noise, yet are misinterpreted: this seems much more serious, since it indicates an unexpected limitation of the technique rather than a reasonable result of noise or distortion. In particular, these problems arise where the test pattern is displaced or misorientated relative to the appropriate training set pattern, as with test pattern (6).

As will be seen in Chapter 13, Basic Classification Concepts, there is a powerful principle that indicates why the unlikely limitation given above can arise: it is simply that there are *insufficient training set patterns*, and that those that are present are *insufficiently representative* of what will arise in practical situations. Unfortunately, this presents a major difficulty, since providing enough training set patterns incurs a serious storage problem and an even more serious search problem when patterns are tested. Furthermore, it is easy to see that these problems are exacerbated as patterns become larger and more real (obviously, the examples of Fig. 1.1 are far from having enough resolution even to display normal typefonts). In fact, a "combinatorial explosion" takes place: this is normally taken to mean that one or more parameters produce fast-varying (often exponential) effects, which "explode" as the parameters increase by modest amounts. Forgetting for the moment that the patterns of Fig. 1.1 have familiar shapes, let us temporarily regard them as random bit patterns. Now the number of bits in these $N \times N$ patterns is N^2, and the number of possible patterns of this size is 2^{N^2}: even in a case where $N = 20$, remembering all these patterns and their interpretations would be impossible on any practical machine, and searching systematically through them would take impractically long (involving times of the order of the age of the universe). Thus it is not only impracticable to consider such brute force means of solving the recognition problem, but is also effectively impossible theoretically. These considerations show that other means are required to tackle the problem.

1.2.2 TACKLING THE RECOGNITION PROBLEM

An obvious means of tackling the recognition problem is to standardize the images in some way. Clearly, normalizing the position and orientation of any 2D picture object would help considerably: indeed this would reduce the number of degrees of freedom by three. Methods for achieving this involve centralizing the objects—arranging that their centroids are at the center of the normalized image—and making their major axes (e.g., deduced by moment calculations) vertical or horizontal. Next, we can make use of the order that is known to be present in the image—and here it may be noted that very few patterns of real interest are indistinguishable from random dot patterns. This approach can be taken further: if patterns are to be nonrandom, isolated noise points may be eliminated. Ultimately, all these methods help by making the test pattern closer to a restricted set of training set patterns (although care must also be taken to process the training set patterns initially so that they are representative of the processed test patterns).

It is useful to consider character recognition further. Here we can make additional use of what is known about the structure of characters—namely, that they consist of limbs of roughly constant width. In that case the width carries no useful information, so the patterns can be thinned to stick figures (called skeletons—see Chapter 8: Binary Shape Analysis); then, hopefully, there is an even greater chance that the test patterns will be similar to appropriate training set patterns (Fig. 1.2). This process can be regarded as another instance of reducing the number of degrees of freedom in the image, and hence of helping to minimize the combinatorial explosion—or, from a practical point of view, to minimize the size of the training set necessary for effective recognition.

Next, consider a rather different way of looking at the problem. Recognition is necessarily a problem of discrimination—i.e., of discriminating between patterns of different classes. However, in practice, considering the natural variation of patterns, including the effects of noise and distortions (or even the effects of breakages or occlusions), there is also a problem of generalizing over patterns of the same class. In practical problems there is a tension between the need to discriminate and the need to generalize. Nor is this a fixed situation. Even for the character recognition task, some classes are so close to others (*n*'s and *h*'s will be similar) that less generalization is possible than in other cases. On the other hand, extreme forms of generalization arise when, for example, an *A* is to be recognized as an *A* whether it is a capital or small letter, or in italic, bold, suffix, or other form of font—even if it is handwritten. The variability is determined largely by the training set initially provided. What we emphasize here, however, is that generalization is as necessary a prerequisite to successful recognition as is discrimination.

At this point it is worth considering more carefully the means whereby generalization was achieved in the examples cited above. First, objects were positioned and orientated appropriately; second, they were cleaned of noise spots; and third, they were thinned to skeleton figures (although the latter process is relevant only for certain tasks such as character recognition). In the last case, we are generalizing over characters drawn with all possible limb widths, width being an irrelevant degree of freedom for this type of recognition task. Note that we could have

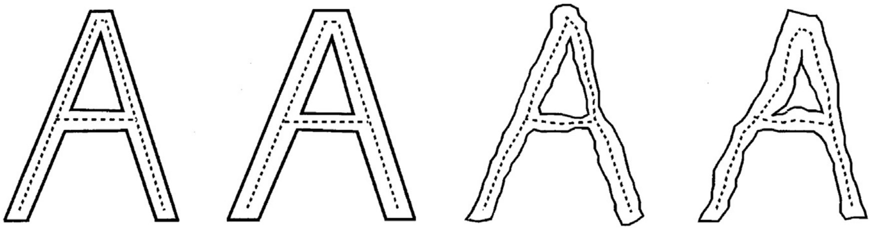

FIGURE 1.2

Use of thinning to regularize character shapes. Here character shapes of different limb widths—or even varying limb widths—are reduced to stick figures or skeletons. Thus irrelevant information is removed and at the same time recognition is facilitated.

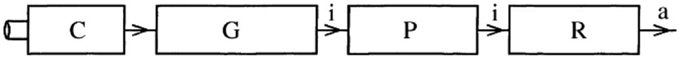

FIGURE 1.3

The two-stage recognition paradigm: C, input from camera; G, grab image (digitize and store); P, preprocess; R, recognize (i, image data; a, abstract data). The classical paradigm for object recognition is that of (1) preprocessing (image processing) to suppress noise or other artefacts and to regularize the image data and (2) applying a process of abstract (often statistical) pattern recognition to extract the very few bits required to classify the object.

generalized the characters further by normalizing their size and saving another degree of freedom. The common feature of all these processes is that they aim to give the characters a high level of standardization against known types of variability before finally attempting to recognize them.

The standardization (or generalization) processes outlined above are all realized by image processing, i.e., the conversion of one image into another by suitable means. The result is a two-stage recognition scheme: first, images are converted into more amenable forms containing the same numbers of bits of data; and second, they are classified with the result that their data content is reduced to very few bits (Fig. 1.3). In fact, recognition is a process of data abstraction, the final data being abstract and totally unlike the original data. Thus we must imagine a letter *A* starting as an array of perhaps 20 × 20 bits arranged in the form of an *A*, and then ending as the 7 bits in an ASCII representation of an *A*, namely 1000001 (which is essentially a random bit pattern bearing no resemblance to an *A*).

The last paragraph reflects to a large extent the history of image analysis. Early on, a good proportion of the image analysis problems being tackled were envisaged as consisting of an image "preprocessing" task carried out by image processing techniques, followed by a recognition task undertaken by pure pattern recognition methods (see Chapter 13: Basic Classification Concepts). These two topics—image processing and pattern recognition—consumed much research effort and effectively dominated the subject of image analysis, while "intermediate-level" approaches such as the Hough transform were, for a time, slower to develop. One of the aims of this book is to ensure that such intermediate-level processing techniques are given due emphasis, and indeed that the best range of techniques is applied to any computer vision task.

1.2.3 OBJECT LOCATION

The problem that was tackled above—that of character recognition—is a highly constrained one. In a great many practical applications it is necessary to search pictures for objects of various types, rather than just interpreting a small area of a picture.

Search is a task that can involve prodigious amounts of computation and is also subject to a combinatorial explosion. Imagine the task of searching for a letter *E* in a page of text. An obvious way of achieving this is to move a suitable "template" of size $n \times n$ over the whole image, of size $N \times N$, and to find where a match occurs (Fig. 1.4). A match can be defined as a position where there is exact agreement between the template and the local portion of the image but, in keeping with the ideas of Section 1.2.1, it will evidently be more relevant to look for a best local match (i.e., a position where the match is locally better than in adjacent regions) and where the match is also good in some more absolute sense, indicating that an *E* is present.

One of the most natural ways of checking for a match is to measure the Hamming distance between the template and the local $n \times n$ region of the image, i.e., to sum the number of differences between corresponding bits. This is essentially the process described in Section 1.2.1. Then places with a low Hamming distance are places where the match is good. These template-matching ideas can be extended to cases where the corresponding bit positions in the template and the image do not just have binary values but may have intensity values over a range 0–255. In that case the sums obtained are no longer Hamming distances but may be generalized to the form:

$$\mathcal{D} = \sum_{\mathrm{t}} |I_{\mathrm{i}} - I_{\mathrm{t}}| \tag{1.1}$$

I_{t} being the local template value, I_{i} being the local image value, and the sum being taken over the area of the template. This makes template matching practicable in many situations: the possibilities are examined in more detail in subsequent chapters.

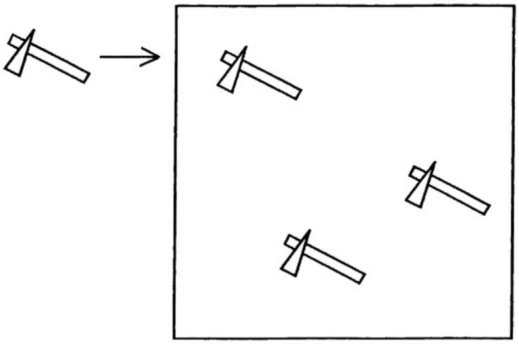

FIGURE 1.4

Template matching, the process of moving a suitable template over an image to determine the precise positions at which a match occurs, hence revealing the presence of objects of a particular type.

We referred above to a combinatorial explosion in this search problem too. The reason this arises is as follows. First, when a 5×5 template is moved over an $N \times N$ image in order to look for a match, the number of operations required is of the order of 5^2N^2, totaling some 1 million operations for a 256×256 image. The problem is that when larger objects are being sought in an image, the number of operations increases as the square of the size of the object, the total number of operations being N^2n^2 when an $n \times n$ template is used. For a 30×30 template and a 256×256 image, the number of operations required rises to ~ 60 million. Note that, in general, a template will be larger than the object it is used to search for, because some background will have to be included to help demarcate the object.

Next, recall that in general, objects may appear in many orientations in an image (*E*'s on a printed page are exceptional). If we imagine a possible 360 orientations (i.e., one per degree of rotation), then a corresponding number of templates will in principle have to be applied in order to locate the object. This additional degree of freedom pushes the search effort and time to enormous levels, so far away from the possibility of real-time implementation that new approaches must be found for tackling the task. ["Real-time" is a commonly used phrase meaning that the information has to be processed as it becomes available: this contrasts with the many situations (such as the processing of images from space probes) where the information may be stored and processed at leisure.] Fortunately, many researchers have applied their minds to this problem and there are a many good ideas for tackling it. Perhaps the most important general means for saving effort on this sort of scale is that of two-stage (or multistage) template matching. The principle is to search for objects via their features. For example, we might consider searching for *E*'s by looking for characters that have horizontal line segments within them. Similarly, we might search for hinges on a manufacturer's conveyor by looking first for the screw holes they possess. In general it is useful to look for small features, since they require smaller templates and hence involve significantly less computation, as demonstrated above. This means that it may be better to search for *E*'s by looking for corners instead of horizontal line segments.

Unfortunately, noise and distortions give rise to problems if we search for objects via small features—there is a risk of missing the object altogether. Hence it is necessary to collate the information from a number of such features. This is the point where the many available methods start to differ from each other. How many features should be collated? Is it better to take a few larger features than many smaller ones? And so on. Also, we have not answered in full the question of what types of feature are the best to employ. These and other questions are considered in the subsequent chapters.

Indeed, in a sense, these questions are the subject of this book. Search is one of the fundamental problems of vision, yet the details and the application of the basic idea of two-stage template matching give the subject much of its richness: to solve the recognition problem, the data set needs to be explored carefully. Clearly, any answers will tend to be data-dependent but it is worth exploring to what extent there are generalized solutions to the problem.

1.2.4 SCENE ANALYSIS

The last subsection considered what is involved in searching an image for objects of a certain type: the result of such a search is likely to be a list of centroid coordinates for these objects, although an accompanying list of orientations might also be obtained. This subsection considers what is involved in scene analysis—the activity we are continually engaged in as we walk around, negotiating obstacles, finding food, and so on. Scenes contain a multitude of objects, and it is their interrelationships and relative positions that matter as much as identifying what they are. It may seem that there is no need for a search *per se* and that we could passively take in what is in the scene. However, there is much evidence (e.g., from analysis of eye movements) that the eye−brain system interprets scenes by continually asking questions about what is there. For example, we might ask the following questions: Is this a lamppost? How far away is it? Do I know this person? Is it safe to cross the road? And so on. It is not the purpose here to dwell on these human activities or introspection about them but merely to observe that scene analysis involves enormous amounts of input data, complex relationships between objects within scenes and, ultimately, descriptions of these complex relationships. The latter no longer take the form of simple classification labels, or lists of object coordinates, but have a much richer information content: indeed, a scene will, to a first approximation, be better described in English than as a list of numbers. It seems likely that a much greater combinatorial explosion is involved in determining relationships between objects than in merely identifying and locating them. Hence, all sorts of props must be used to aid visual interpretation: there is considerable evidence of this in the human visual system, where contextual information and the availability of immense databases of possibilities clearly help the eye to a considerable degree.

Note also that scene descriptions may initially be at the level of factual content but will eventually be at a deeper level—that of meaning, significance, and relevance. However, we shall not be able to delve further into these areas in this book.

1.2.5 VISION AS INVERSE GRAPHICS

It has often been said that vision is "merely" inverse graphics. There is a certain amount of truth in this. Computer graphics is the generation of images by computer, starting from abstract descriptions of scenes and knowledge of the laws of image formation. Also, it is difficult to quarrel with the idea that vision is the process of obtaining descriptions of sets of objects, starting from sets of images and a knowledge of the laws of image formation (indeed, it is good to see a definition that explicitly brings in the need to know the laws of image formation, since it is all too easy to forget that this is a prerequisite when building descriptions incorporating heuristics that aid interpretation).

However, this similarity in formulation of the two processes hides some fundamental points. First, graphics is a "feedforward" activity, i.e., images can be produced straightforwardly once sufficient specification about the viewpoint and the objects, and knowledge of the laws of image formation, have been obtained. True, considerable computation may be required but the process is entirely determined and predictable. The situation is not so straightforward for vision because search is involved and there is an accompanying combinatorial explosion. Indeed, some vision packages incorporate graphics (or CAD) packages (Tabandeh and Fallside, 1986), which are inserted into feedback loops for interpretation: the graphics package is then guided iteratively until it produces an acceptable approximation to the input image, when its input parameters embody the correct interpretation (there is a close parallel here with the problem of designing analog-to-digital converters by making use of digital-to-analog converters). Hence, it seems inescapable that vision is intrinsically more complex than graphics.

We can clarify the situation somewhat by noting that, as a scene is observed, a 3-D environment is compressed into a 2-D image and a considerable amount of depth and other information is lost. This can lead to ambiguity of interpretation of the image (both a helix viewed end-on and a circle project into a circle), so the 3-D to 2-D transformation is many-to-one. Conversely, the interpretation must be one-to-many, meaning that there are many possible interpretations, yet we know that only one can be correct: vision involves not merely providing a list of all possible interpretations but providing the most likely one. Hence, some additional rules or constraints must be involved in order to determine the single most likely interpretation. Graphics, in contrast, does not have these problems, as the above ideas show it to be a many-to-one process.

1.3 FROM AUTOMATED VISUAL INSPECTION TO SURVEILLANCE

So far we have considered the nature of vision but not what man-made vision systems may be used for. There is in fact a great variety of applications for artificial vision systems—including, of course, all of those for which man employs his visual senses. Of particular interest in this book are surveillance, automated inspection, robot assembly, vehicle guidance, traffic monitoring and control, biometric measurement, and analysis of remotely sensed images. By way of example, fingerprint analysis and recognition have long been important applications of computer vision, as have the counting of red blood cells, signature verification and character recognition, and aeroplane identification (both from aerial silhouettes and from ground surveillance pictures taken from satellites). Face recognition and even iris recognition have become practical possibilities and vehicle guidance by vision will in principle soon be sufficiently reliable for urban use. Whether the public will accept this, with all its legal implications, is another matter, but note that radar blind-landing aids for aircraft have been in wide use for some years. In fact, last-minute automatic action to prevent accidents is a good compromise (see Chapter 24: Epilogue—Perspectives in Vision, for a related discussion on driver assistance schemes).

Among the applications of vision considered in this book are those of manufacturing industry—particularly, automated visual inspection and vision for automated assembly. In these cases, much of the same manufactured components are viewed by cameras: the difference lies in how the resulting information is used. In assembly, components must be located and orientated so that a robot can pick them up and assemble them. For example, the various parts of a motor or brake system need to be taken in turn and put into the correct positions, or a coil may have to be mounted on a television tube, an integrated circuit placed on a printed circuit board, or a chocolate placed into a box. In inspection, objects may pass the inspection station on a moving conveyor at rates typically between 10 and 30 items per second, and it has to be ascertained whether they have any defects. If any defects are detected, the offending parts will usually have to be rejected: that is the feedforward solution. In addition, a feedback solution may be instigated—i.e., some parameter may have to be adjusted to control plant further back down the production line (this is especially true for parameters that control dimensional characteristics such as product diameter). Inspection also has the potential for amassing a wealth of information that is useful for management, on the state of the parts coming down the line: the total number of products per day, the number of defective products per day, the distribution of sizes of products, and so on. The important feature of artificial vision is that it is tireless and that *all* products can be scrutinized and measured: thus quality control can be maintained to a very high standard. In automated assembly too, a considerable amount of on-the-spot inspection can be performed and this may help to avoid the problem of complex assemblies being rejected or having to be subjected to expensive repairs, because (for example) a proportion of screws were threadless and could not be inserted properly.

An important feature of most industrial tasks is that they take place in real time: if it is used, vision must be able to keep up with the manufacturing process. For assembly, this may not be too exacting a problem, since a robot may not be able to pick up and place more than one item per second—leaving the vision system a similar time to do its processing. For inspection, this supposition is rarely valid: even a single automated line (e.g., one for stoppering bottles) is able to keep up a rate of 10 items per second (and, of course, parallel lines are able to keep up much higher rates). Hence, visual inspection tends to press computer hardware very hard, so care needs to be taken in the design of hardware accelerators for such applications.

Finally, we return to the starting discussion about the huge variety of applications of vision, and it is interesting to consider surveillance tasks as the outdoor analogs of automated inspection (indeed, it is amusing to imagine that cars speeding along a road are just as subject to inspection as products speeding along a product line!). In fact, they have recently been acquiring close to exponentially increasing application. Thus the techniques used for inspection have acquired an injection of vitality, and many more techniques have been developed. Naturally, this has meant the introduction of whole tranches of new subject matter, such as motion analysis and perspective invariants (see Part 4, 3-D Vision and Motion). It is also interesting that such techniques add richness to topics as face recognition (see Chapter 21: Face Detection and Recognition: the Impact of Deep Learning).

1.4 WHAT THIS BOOK IS ABOUT

The foregoing sections have examined something of the nature of computer vision and have briefly considered its applications and implementation. It is already clear that implementing computer vision involves considerable practical difficulties but, more important, these practical difficulties embody substantial fundamental problems: these include various factors giving rise to excessive processing load and time. Practical problems may be overcome by ingenuity and care: however, by definition, truly fundamental limitations cannot be overcome by *any* means—the best that we can hope for is that we will be able to minimize their effects following a complete understanding of their nature.

Understanding is thus a cornerstone for success in computer vision. It is often difficult to achieve, since the data set (i.e., all pictures that could reasonably be expected to arise) is highly variegated. Indeed, much investigation is required to determine the nature of a given data set, including not only the objects being observed but also the noise levels, degrees of occlusion, breakage, defect, and distortion that are to be expected, and the quality and nature of the lighting. Ultimately, sufficient knowledge might be obtained in a useful set of cases so that a good understanding of the milieu can be attained. Then it remains to compare and contrast the various methods of image analysis that are available. Some methods will turn out to be quite unsatisfactory for reasons of robustness, accuracy or cost of implementation, or other relevant variables: and who is to say in advance what a relevant set of variables is? This, too, needs to be ascertained and defined. Finally, among the methods that could reasonably be used, there will be competition: tradeoffs between parameters such as accuracy, speed, robustness, and cost will have to be worked out first theoretically and then in numerical detail to find an optimal solution. This is a complex and long process in a situation where workers have in the past aimed to find solutions for their own particular (often short-term) needs. Clearly, there is a need to ensure that practical computer vision advances from an art to a science. Fortunately this process has been developing for some years, and one of the aims of this book is to throw additional light on the problem.

Before proceeding further, there are one or two more pieces to fit into the jigsaw. First, there is an important guiding principle: *if the eye can do it, so can the machine*. Thus, if an object is fairly well hidden in an image, yet the eye can see it and track it, then it should be possible to devise a vision algorithm that can do the same. Next, although we can expect to meet this challenge, should we set our sights even higher and aim to devise algorithms that can beat the eye? There seems no reason to suppose that the eye is the ultimate vision machine: it has been built through the vagaries of evolution, so it may be well adapted for finding berries or nuts, or for recognizing faces, but ill-suited for certain other tasks. One such task is that of measurement. The eye probably does not need to measure the sizes of objects, at a glance, to better than a few percent accuracy. However, it could be distinctly useful if the robot eye could achieve remote size measurement, at a glance, and with an accuracy of say 0.001%. Clearly, the robot eye could acquire capabilities superior to those of biological systems. Again, this book aims to point out such possibilities where they exist.

Finally, it will be useful to clarify the terms *Machine Vision* and *Computer Vision*. In fact, these arose a good many years ago when the situation was quite different from what it is today. Over time, computer technology has advanced hugely and at the same time knowledge about the whole area of vision has been radically developed. In the early days, *Computer Vision* meant the study of the science of vision and the *possible* design of the software—and to a lesser extent with what goes into an integrated vision system, whereas *Machine Vision* meant the study not only of the software but also of the hardware environment and of the image acquisition techniques needed for real applications—so it was a much more engineering-orientated subject. At this point in time, computer technology has advanced so far that a sizeable proportion of real-world and real-time applications can be realized on unaided PCs. This and the many developments in knowledge in this area have led to significant convergence between the terms, with the result that they are often used more or less interchangeably, although in this book we aim to unify the subject under the name *Computer Vision*.

1.5 THE PART PLAYED BY MACHINE LEARNING

During the whole period that computer vision was developing in the way described above, the subject of pattern recognition was also progressing. Basic ideas on pattern recognition that started with Bayes theory and the nearest neighbor approach gradually changed with the advent of artificial neural networks, which were designed to emulate the neuron networks known to exist in the human brain. In addition, other methods such as support vector machines and boosting arrived on the scene: then, during the last decade or so, "deep learning" came into prominence. All these techniques led to a new subject called Machine Learning, which embodies pure pattern recognition but emphasizes not only minimization of error rates but also systematic inclusion of probability and mathematical optimization. The impact of this subject on computer vision has been increasingly dramatic over the past decade and particularly during the past 4–5 years. This book aims to include this development as an integral part of its coverage: Chapters 2, Images and Imaging Operations and Chapter 13, Basic Classification Concepts, introduce in turn the imaging side of computer vision and the machine learning side, while Chapter 15, Deep Learning Networks, leads the reader into the newer area of deep learning.

Broadly, the modern subject of Computer Vision embodies both the earlier methods of computer and machine vision and also a range of machine learning techniques: the latter are based on earlier pattern recognition methods including standard artificial neural networks, the latest "deep learning" networks and a range of rigorous techniques involving probabilistic optimization.

1.6 THE FOLLOWING CHAPTERS

Chapters 2 and 13 form the introductions to the two main branches of the subject—image processing and machine learning. Chapters 2–7 follow the image-processing theme, covering low-level vision and various widely used segmentation techniques, ranging from thresholding, through edge and feature detection to texture analysis. Chapters 8–12 move on to intermediate-level processing, which has developed significantly in the past two decades and is important for the inference of complex objects: to this end, key model-based vision techniques such as the Hough transform and RANSAC are covered in detail (Chapters 10 and 11). Active shape models (Chapter 12) are also important for many practical applications. However, the latter require knowledge of PCA and other machine learning concepts: these are covered in Chapter 14. Chapters 16–19 develop the subject of 3-D vision, while Chapter 20 introduces motion. Chapters 21–23 attend to three key application areas—face detection and recognition; surveillance; and in-vehicle vision systems. Chapter 24 reiterates and highlights some of the lessons and topics dealt with in the book; Appendix A develops the subject of Robust Statistics, which relates to a large proportion of the methods that are covered here; and Appendix B covers a topic that is essential background to a subject such as vision—namely the sampling theorem. Appendix C discusses the representation of color, while Appendix D is relevant to machine learning and contains important material on sampling from distributions.

To help the reader by giving more perspective on the later chapters, the main text is divided into five parts:

Part 1 (Chapters 2–7)	Low-Level Vision
Part 2 (Chapters 8–12)	Intermediate-Level Vision
Part 3 (Chapters 13–15)	Machine Learning and Deep Learning Networks
Part 4 (Chapters 16–20)	3-D Vision and Motion
Part 5 (Chapters 21–23)	Putting Computer Vision to Work.

This last heading is used to emphasize real-world applications with high data flow rates and the need to integrate all the necessary recognition processes into reliable working systems.

Although the sequence of chapters follows the somewhat logical order just described, the ideas outlined in Section 1.4—understanding of the visual process, constraints imposed by realities such as noise and occlusion, tradeoffs between relevant parameters, and so on—are mixed into the text at relevant junctures, as they reflect all-pervasive issues.

Finally, there are many topics that could not be included in the book for reasons of space. The chapter bibliographies, the main list of references, and the indexes are intended to make good this limitation.

1.7 BIBLIOGRAPHICAL NOTES

The purpose of this chapter has been to introduce the reader to some of the problems of machine vision, showing the intrinsic difficulties but not at this stage getting into details. For detailed references the reader should consult the subsequent chapters. Meanwhile, further background on the world of machine learning can be obtained from Bishop (2006), Prince (2012), and Theodoridis (2015). In addition, some insight into human vision can be obtained from the fascinating monograph by Hubel (1995).