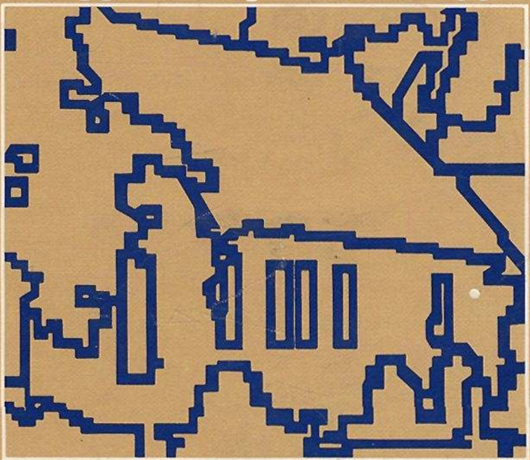
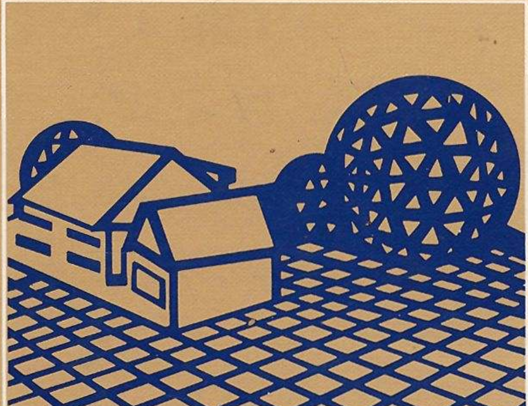
COMPUTER VISION

DANA H. BALLARD · CHRISTOPHER M. BROWN





COMPUTER VISION

Dana H. Ballard
Christopher M. Brown

Department of Computer Science
University of Rochester
Rochester, New York

PRENTICE-HALL, INC., Englewood Cliffs, New Jersey 07632

Library of Congress Cataloging in Publication Data

BALLARD, DANA HARRY.
Computer vision.

Bibliography: p.
Includes index.

1. Image processing. I. Brown, Christopher M.
II. Title.

TA1632.B34 621.38'0414 81-20974
ISBN 0-13-165316-4 AACR2

Cover design by Robin Breite

Englewood Cliffs, New Jersey 07632

Printed in the United States of America

10 9 8 7 6 5 4 3 2

ISBN 0-13-165316-4

PRENTICE-HALL INTERNATIONAL, INC., *London*
PRENTICE-HALL OF AUSTRALIA PTY. LIMITED, *Sydney*
PRENTICE-HALL OF CANADA, LTD., *Toronto*
PRENTICE-HALL OF INDIA PRIVATE LIMITED, *New Delhi*
PRENTICE-HALL OF JAPAN, INC., *Tokyo*
PRENTICE-HALL OF SOUTHEAST ASIA PTE. LTD., *Singapore*
WHITEHALL BOOKS LIMITED, *Wellington, New Zealand*

Contents

Part I
GENERALIZED IMAGES
13

PART II
SEGMENTED IMAGES
115

Preface

The dream of intelligent automata goes back to antiquity; its first major articulation in the context of digital computers was by Turing around 1950. Since then, this dream has been pursued primarily by workers in the field of *artificial intelligence,* whose goal is to endow computers with information-processing capabilities comparable to those of biological organisms. From the outset, one of the goals of artificial intelligence has been to equip machines with the capability of dealing with sensory inputs.

Computer vision is the construction of explicit, meaningful descriptions of physical objects from images. Image understanding is very different from image processing, which studies image-to-image transformations, not explicit description building. Descriptions are a prerequisite for recognizing, manipulating, and thinking about objects.

We perceive a world of coherent three-dimensional objects with many invariant properties. Objectively, the incoming visual data do not exhibit corresponding coherence or invariance; they contain much irrelevant or even misleading variation. Somehow our visual system, from the retinal to cognitive levels, understands, or imposes order on, chaotic visual input. It does so by using *intrinsic information* that may reliably be extracted from the input, and also through assumptions and *knowledge* that are applied at various levels in visual processing.

The challenge of computer vision is one of *explicitness.* Exactly what information about scenes can be extracted from an image using only very basic assumptions about physics and optics? Explicitly, what computations must be performed? Then, at what stage must domain-dependent, prior knowledge about the world be incorporated into the understanding process? How are world models and knowledge represented and used? This book is about the representations and mechanisms that allow image information and prior knowledge to interact in image understanding.

Computer vision is a relatively new and fast-growing field. The first experiments were conducted in the late 1950s, and many of the essential concepts have been developed during the last five years. With this rapid growth, crucial ideas have arisen in disparate areas such as artificial intelligence, psychology, computer graphics, and image processing. Our intent is to assemble a selection of this material in a form that will serve both as a senior/graduate-level academic text and as a useful reference to those building vision systems. This book has a strong artificial intelligence flavor, and we hope this will provoke thought. We believe that both the intrinsic image information and the internal model of the world are important in successful vision systems.

The book is organized into four parts, based on descriptions of objects at four different levels of abstraction.

1. Generalized images—images and image-like entities.
2. Segmented images—images organized into subimages that are likely to correspond to "interesting objects."
3. Geometric structures—quantitative models of image and world structures.
4. Relational structures—complex symbolic descriptions of image and world structures.

The parts follow a progression of increasing abstractness. Although the four parts are most naturally studied in succession, they are not tightly interdependent. Part I is a prerequisite for Part II, but Parts III and IV can be read independently.

Parts of the book assume some mathematical and computing background (calculus, linear algebra, data structures, numerical methods). However, throughout the book mathematical rigor takes a backseat to concepts. Our intent is to transmit a set of ideas about a new field to the widest possible audience.

In one book it is impossible to do justice to the scope and depth of prior work in computer vision. Further, we realize that in a fast-developing field, the rapid influx of new ideas will continue. We hope that our readers will be challenged to think, criticize, read further, and quickly go beyond the confines of this volume.

D. H. Ballard
C. M. Brown

Acknowledgments

Jerry Feldman and Herb Voelcker (and through them the University of Rochester) provided many resources for this work. One of the most important was a capable and forgiving staff (secretarial, technical, and administrative). For massive text editing, valuable advice, and good humor we are especially grateful to Rose Peet. Peggy Meeker, Jill Orioli, and Beth Zimmerman all helped at various stages.

Several colleagues made suggestions on early drafts: thanks to James Allen, Norm Badler, Larry Davis, Takeo Kanade, John Kender, Daryl Lawton, Joseph O'Rourke, Ari Requicha, Ed Riseman, Azriel Rosenfeld, Mike Schneier, Ken Sloan, Steve Tanimoto, Marty Tenenbaum, and Steve Zucker.

Graduate students helped in many different ways: thanks especially to Michel Denber, Alan Frisch, Lydia Hrechanyk, Mark Kahrs, Keith Lantz, Joe Maleson, Lee Moore, Mark Peairs, Don Perlis, Rick Rashid, Dan Russell, Dan Sabbah, Bob Schudy, Peter Selfridge, Uri Shani, and Bob Tilove. Bernhard Stuth deserves special mention for much careful and critical reading.

Finally, thanks go to Jane Ballard, mostly for standing steadfast through the cycles of elation and depression and for numerous engineering-to-English translations.

As Pat Winston put it: "A willingness to help is not an implied endorsement." The aid of others was invaluable, but we alone are responsible for the opinions, technical details, and faults of this book.

Funding assistance was provided by the Sloan Foundation under Grant 78-4-15, by the National Institutes of Health under Grant HL21253, and by the Defense Advanced Research Projects Agency under Grant N00014-78-C-0164.

The authors wish to credit the following sources for figures and tables. For complete citations given here in abbreviated form (as "from . . ." or "after . . ."), refer to the appropriate chapter-end references.

Fig. 1.2 from Shani, U., "A 3-D model-driven system for the recognition of abdominal anatomy from CT scans," TR77, Dept. of Computer Science, University of Rochester, May 1980.

Fig. 1.4 courtesy of Allen Hanson and Ed Riseman, COINS Research Project, University of Massachusetts, Amherst, MA.

Fig. 2.4 after Horn and Sjoberg, 1978.

Figs. 2.5, 2.9, 2.10, 3.2, 3.6, and 3.7 courtesy of Bill Lampeter.

Fig. 2.7a painting by Louis Condax; courtesy of Eastman Kodak Company and the Optical Society of America.

Fig. 2.8a courtesy of D. Greenberg and G. Joblove, Cornell Program of Computer Graphics.

Fig. 2.8b courtesy of Tom Check.

Table 2.3 after Gonzalez and Wintz, 1977.

Fig. 2.18 courtesy of EROS Data Center, Sioux Falls, SD.

Figs. 2.19 and 2.20 from Herrick, C.N., *Television Theory and Servicing: Black/White and Color,* 2nd Ed. Reston, VA: Reston, 1976.

Figs. 2.21, 2.22, 2.23, and 2.24 courtesy of Michel Denber.

Fig. 2.25 from Popplestone et al., 1975.

Fig. 2.26 courtesy of Production Automation Project, University of Rochester.

Fig. 2.27 from Waag and Gramiak, 1976.

Fig. 3.1 courtesy of Marty Tenenbaum.

Fig. 3.8 after Horn, 1974.

Figs. 3.14 and 3.15 after Frei and Chen, 1977.

Figs. 3.17 and 3.18 from Zucker, S.W. and R.A. Hummel, "An optimal 3-D edge operator," *IEEE Trans. PAMI 3,* May 1981, pp. 324-331.

Fig. 3.19 curves are based on data in Abdou, 1978.

Figs. 3.20, 3.21, and 3.22 from Prager, J.M., "Extracting and labeling boundary segments in natural scenes," *IEEE Tans. PAMI 12,* 1, January 1980. © 1980 IEEE.

Figs. 3.23, 3.28, 3.29, and 3.30 courtesy of Berthold Horn.

Figs. 3.24 and 3.26 from Marr, D. and T. Poggio, "Cooperative computation of stereo disparity," *Science,* Vol. 194, 1976, pp. 283-287. © 1976 by the American Association for the Advancement of Science.

Fig. 3.31 from Woodham, R.J., "Photometric stereo: A reflectance map technique for determining surface orientation from image intensity," *Proc. SPIE,* Vol. 155, August 1978.

Figs. 3.33 and 3.34 after Horn and Schunck, 1980.

Fig. 3.37 from Tanimoto, S. and T. Pavlidis, "A hierarchical data structure for picture processing," *CGIP 4,* 2, June 1975, pp. 104–119.

Fig. 4.6 from Kimme et al., 1975.

Figs. 4.7 and 4.16 from Ballard and Sklansky, 1976.

Fig. 4.9 courtesy of Dana Ballard and Ken Sloan.

Figs. 4.12 and 4.13 from Ramer, U., "Extraction of line structures from photgraphs of curved objects," *CGIP 4,* 2, June 1975, pp. 81–103.

Fig. 4.14 courtesy of Jim Lester, Tufts/New England Medical Center.

Fig. 4.17 from Chien, Y.P. and K.S. Fu, "A decision function method for boundary detection," *CGIP 3,* 2, June 1974, pp. 125–140.

Fig. 5.3 from Ohlander, R., K. Price, and D.R. Reddy, "Picture segmentation using a recursive region splitting method," *CGIP 8,* 3, December 1979.

Fig. 5.4 courtesy of Sam Kapilivsky.

Figs. 6.1, 11.16, and A1.13 courtesy of Chris Brown.

Fig. 6.3 courtesy of Joe Maleson and John Kender.

Fig. 6.4 from Connors, 1979. Texture images by Phil Brodatz, in Brodatz, *Textures.* New York: Dover, 1966.

Fig. 6.9 texture image by Phil Brodatz, in Brodatz, *Textures.* New York: Dover, 1966.

Figs. 6.11, 6.12, and 6.13 from Lu, S.Y. and K.S. Fu, "A syntactic approach to texture analysis," *CGIP 7,* 3, June 1978, pp. 303-330.

Fig. 6.14 from Jayaramamurthy, S.N., "Multilevel array grammars for generating texture scenes," *Proc. PRIP,* August 1979, pp. 391-398. © 1979 IEEE.

Fig. 6.20 from Laws, 1980.

Figs. 6.21 and 6.22 from Maleson et al., 1977.

Fig. 6.23 courtesy of Joe Maleson.

Figs. 7.1 and 7.3 courtesy of Daryl Lawton.

Fig. 7.2 after Prager, 1979.

Figs. 7.4 and 7.5 from Clocksin, W.F., "Computer prediction of visual thresholds for surface slant and edge detection from optical flow fields," Ph.D. dissertation, University of Edinburgh, 1980.

Fig. 7.7 courtesy of Steve Barnard and Bill Thompson.

Figs. 7.8 and 7.9 from Rashid, 1980.

Fig. 7.10 courtesy of Joseph O'Rourke.

Figs. 7.11 and 7.12 after Aggarwal and Duda, 1975.

Fig. 7.13 courtesy of Hans-Hellmut Nagel.

Fig. 8.1d after Requicha, 1977.

Figs. 8.2, 8.3, 8.21a, 8.22, and 8.26 after Pavlidis, 1977.

Figs. 8.10, 8.11, 9.6, and 9.16 courtesy of Uri Shani.

Figs. 8.12, 8.13, 8.14, 8.15, and 8.16 from Ballard, 1981.

Fig. 8.21 b from Preston, K., Jr., M.J.B. Duff; S. Levialdi, P.E. Norgren, and J-i. Toriwaki, "Basics of cellular logic with some applications in medical image processing," *Proc. IEEE,* Vol. 67, No. 5, May 1979, pp. 826–856.

Figs. 8.25, 9.8, 9.9, 9.10, and 11.3 courtesy of Robert Schudy.

Fig. 8.29 after Bribiesca and Guzman, 1979.

Figs. 9.1, 9.18, 9.19, and 9.27 courtesy of Ari Requicha.

Fig. 9.2 from Requicha, A.A.G., "Representations for rigid solids: theory, methods, systems," *Computer Surveys 12,* 4, December 1980.

Fig. 9.3 courtesy of Lydia Hrechanyk.

Figs. 9.4 and 9.5 after Baumgart, 1972.

Fig. 9.7 courtesy of Peter Selfridge.

Fig. 9.11 after Requicha, 1980.

Figs. 9.14 and 9.15b from Agin, G.J. and T.O. Binford, "Computer description of curved objects," *IEEE Trans. on Computers 25,* 1, April 1976.

Fig. 9.15a courtesy of Gerald Agin.

Fig. 9.17 courtesy of A. Christensen; published as frontispiece of *ACM SIGGRAPH 80 Proceedings.*

Fig. 9.20 from Marr and Nishihara, 1978.

Fig. 9.21 after Tilove, 1980.

Fig. 9.22b courtesy of Gene Hartquist.

Figs. 9.24, 9.25, and 9.26 from Lee and Requicha, 1980.

Figs. 9.28a, 9.29, 9.30, 9.31, 9.32, 9.35, and 9.37 and Table 9.1 from Brown, C. and R. Popplestone, "Cases in scene analysis," in *Pattern Recognition,* ed. B.G. Batchelor. New York: Plenum, 1978.

Fig. 9.28b from Guzman, A., "Decomposition of a visual scene into three-dimensional bodies," in *Automatic Interpretation and Classification of Images,* A. Grasseli, ed., New York: Academic Press, 1969.

Fig. 9.28c from Waltz, D., "Understanding line drawing of scenes with shadows," in *The Psychology of Computer Vision,* ed. P.H. Winston. New York: McGraw-Hill, 1975.

Fig. 9.28d after Turner, 1974.

Figs. 9.33, 9.38, 9.40, 9.42, 9.43, and 9.44 after Mackworth, 1973.

Figs. 9.39, 9.45, 9.46, and 9.47 and Table 9.2 after Kanade, 1978.
Figs. 10.2 and A2.1 courtesy of Dana Ballard.
Figs. 10.16, 10.17, and 10.18 after Russell, 1979.
Fig. 11.5 after Fischler and Elschlager, 1973.
Fig. 11.8 after Ambler et al., 1975.
Fig. 11.10 from Winston, P.H., "Learning structural descriptions from examples," in *The Psychology of Computer Vision,* ed. P.H. Winston. New York: McGraw-Hill, 1975.
Fig. 11.11 from Nevatia, 1974.
Fig. 11.12 after Nevatia, 1974.
Fig. 11.17 after Barrow and Popplestone, 1971.
Fig. 11.18 from Davis, L.S., "Shape matching using relaxation techniques," *IEEE Trans. PAMI 1,* 4, January 1979, pp. 60–72.
Figs. 12.4 and 12.5 from Sloan and Bajcsy, 1979.
Fig. 12.6 after Barrow and Tenenbaum, 1976.
Fig. 12.8 after Freuder, 1978.
Fig. 12.10 from Rosenfeld, A.R., A. Hummel, and S.W. Zucker, "Scene labeling by relaxation operations," *IEEE Trans. SMC 6,* 6, June 1976, p. 420.
Figs. 12.11, 12.12, 12.13, 12.14, and 12.15 after Hinton, 1979.
Fig. 13.3 courtesy of Aaron Sloman.
Figs. 13.6, 13.7, and 13.8 from Garvey, 1976.
Fig. A1.11 after Duda and Hart, 1973.
Figs. A2.2 and A2.3 from Hanson, A.R. and E.M. Riseman, "VISIONS: A computer system for interpreting scenes," in *Computer Vision Systems,* ed. A.R. Hanson and E.M. Riseman. New York: Academic Press, 1978.

Mnemonics for Proceedings and Special Collections Cited in the References

CGIP

Computer Graphics and Image Processing

COMPSAC

IEEE Computer Society's 3rd International Computer Software and Applications Conference, Chicago, November 1979.

CVS

Hanson, A. R. and E. M. Riseman (Eds.). *Computer Vision Systems.* New York: Academic Press, 1978.

DARPA IU

Defense Advanced Research Projects Agency Image Understanding Workshop, Minneapolis, MN, April 1977.

Defense Advanced Research Projects Agency Image Understanding Workshop, Palo Alto, CA, October 1977.

Defense Advanced Research Projects Agency Image Understanding Workshop, Cambridge, MA, May 1978.

Defense Advanced Research Projects Agency Image Understanding Workshop, Carnegie-Mellon University, Pittsburgh, PA, November 1978.

Defense Advanced Research Projects Agency Image Understanding Workshop, University of Maryland, College Park, MD, April 1980.

IJCAI

2nd International Joint Conference on Artificial Intelligence, Imperial College, London, September 1971.

4th International Joint Conference on Artificial Intelligence, Tbilisi, Georgia, USSR, September 1975.

5th International Joint Conference on Artificial Intelligence, MIT, Cambridge, MA, August 1977.

6th International Joint Conference on Artificial Intelligence, Tokyo, August 1979.

IJCPR

2nd International Joint Conference on Pattern Recognition, Copenhagen, August 1974.

3rd International Joint Conference on Pattern Recognition, Coronado, CA, November 1976.

4th International Joint Conference on Pattern Recognition, Kyoto, November 1978.

5th International Joint Conference on Pattern Recognition, Miami Beach, FL, December 1980.

MI4

Meltzer, B. and D. Michie (Eds.). *Machine Intelligence 4*. Edinburgh: Edinburgh University Press, 1969.

MI5

Meltzer, B. and D. Michie (Eds.). *Machine Intelligence 5*. Edinburgh: Edinburgh University Press, 1970.

MI6

Meltzer, B. and D. Michie (Eds.). *Machine Intelligence 6*. Edinburgh: Edinburgh University Press, 1971.

M17

Meltzer, B. and D. Michie (Eds.). *Machine Intelligence 7*. Edinburgh: Edinburgh University Press, 1972.

PCV

Winston, P. H. (Ed.). *The Psychology of Computer Vision*. New York: McGraw-Hill, 1975.

PRIP

IEEE Computer Society Conference on Pattern Recognition and Image Processing, Chicago, August 1979.

Computer Vision 1

Computer Vision Issues

1.1 ACHIEVING SIMPLE VISION GOALS

Suppose that you are given an aerial photo such as that of Fig. 1.1a and asked to locate ships in it. You may never have seen a naval vessel in an aerial photograph before, but you will have no trouble predicting generally how ships will appear. You might reason that you will find no ships inland, and so turn your attention to ocean areas. You might be momentarily distracted by the glare on the water, but realizing that it comes from reflected sunlight, you perceive the ocean as continuous and flat. Ships on the open ocean stand out easily (if you have seen ships from the air, you know to look for their wakes). Near the shore the image is more confusing, but you know that ships close to shore are either moored or docked. If you have a map (Fig. 1.1b), it can help locate the docks (Fig. 1.1c); in a low-quality photograph it can help you identify the shoreline. Thus it might be a good investment of your time to establish the correspondence between the map and the image. A search parallel to the shore in the dock areas reveals several ships (Fig. 1.1d).

Again, suppose that you are presented with a set of computer-aided tomographic (CAT) scans showing "slices" of the human abdomen (Fig. 1.2a). These images are products of high technology, and give us views not normally available even with x-rays. Your job is to reconstruct from these cross sections the three-dimensional shape of the kidneys. This job may well seem harder than finding ships. You first need to know what to look for (Fig. 1.2b), where to find it in CAT scans, and how it looks in such scans. You need to be able to "stack up" the scans mentally and form an internal model of the shape of the kidney as revealed by its slices (Fig. 1.2c and 1.2d).

This book is about *computer vision.* These two example tasks are typical computer vision tasks; both were solved by computers using the sorts of knowledge and techniques alluded to in the descriptive paragraphs. Computer vision is the enterprise of automating and integrating a wide range of processes and representations used for vision perception. It includes as parts many techniques that are useful by themselves, such as *image processing* (transforming, encoding, and transmitting images) and *statistical pattern classification* (statistical decision theory applied to general patterns, visual or otherwise). More importantly for us, it includes techniques for geometric modeling and cognitive processing.

1.2 HIGH-LEVEL AND LOW-LEVEL CAPABILITIES

The examples of Section 1.1 illustrate vision that uses *cognitive processes*, *geometric models*, *goals*, and *plans*. These *high-level* processes are very important; our examples only weakly illustrate their power and scope. There surely would be some overall purpose to finding ships; there might be collateral information that there were submarines, barges, or small craft in the harbor, and so forth. CAT scans would be used with several diagnostic goals in mind and an associated medical history available. Goals and knowledge are high-level capabilities that can guide visual activities, and a visual system should be able to take advantage of them.

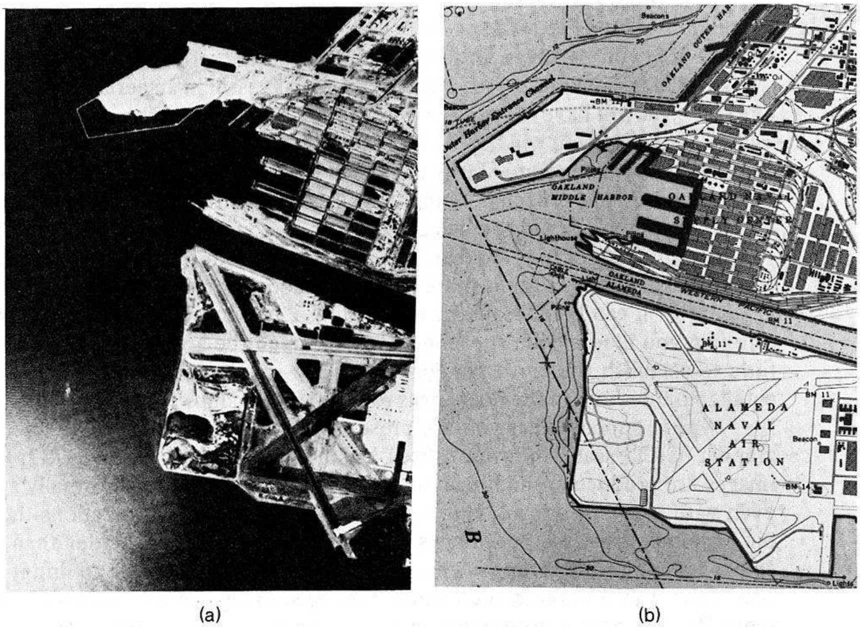


Fig. 1.1 Finding ships in an aerial photograph. (a) The photograph; (b) a corresponding map; (c) the dock area of the photograph; (d) registered map and image, with ship location.

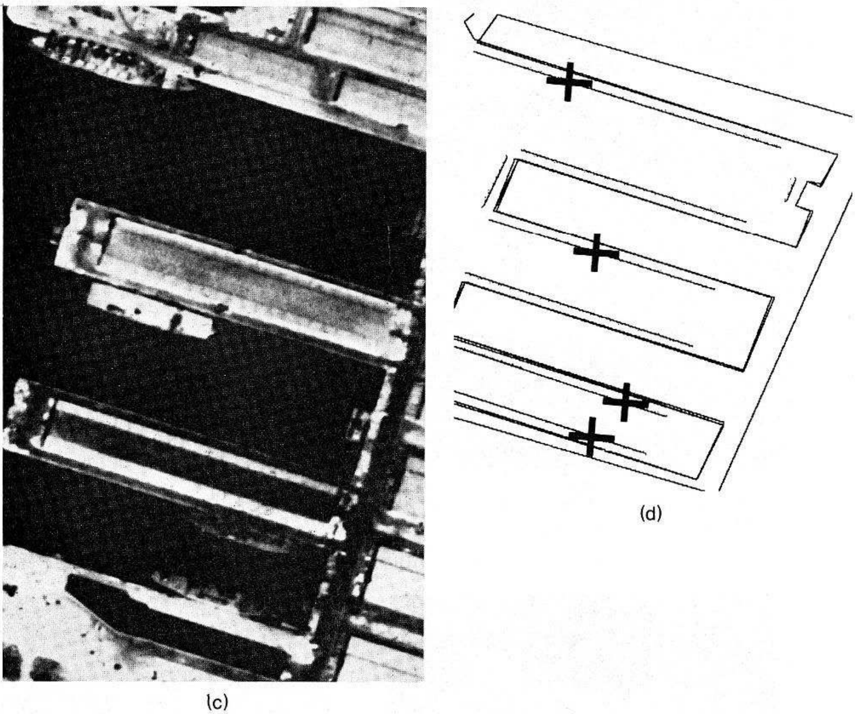


Fig. 1.1 (cont.)

Even such elaborated tasks are very special ones and in their way easier to think about than the commonplace visual perceptions needed to pick up a baby, cross a busy street, or arrive at a party and quickly "see" who you know, your host's taste in decor, and how long the festivities have been going on. All these tasks require judgment and large amounts of knowledge of objects in the world, how they look, and how they behave. Such high-level powers are so well integrated into "vision" as to be effectively inseparable.

Knowledge and goals are only part of the vision story. Vision requires many *low-level* capabilities we often take for granted; for example, our ability to extract *intrinsic images* of "lightness," "color," and "range." We perceive black as black in a complex scene even when the lighting is such that some black patches are reflecting more light than some white patches. Similarly, perceived colors are not related simply to the wavelengths of reflected light; if they were, we would consciously see colors changing with illumination. Stereo fusion (stereopsis) is a low-level facility basic to short-range three-dimensional perception.

An important low-level capability is *object perception*: for our purposes it does not really matter if this talent is innate, ("hard-wired"), or if it is developmental or even learned ("compiled-in"). The fact remains that mature biological vision systems are specialized and tuned to deal with the relevant objects in their environ-

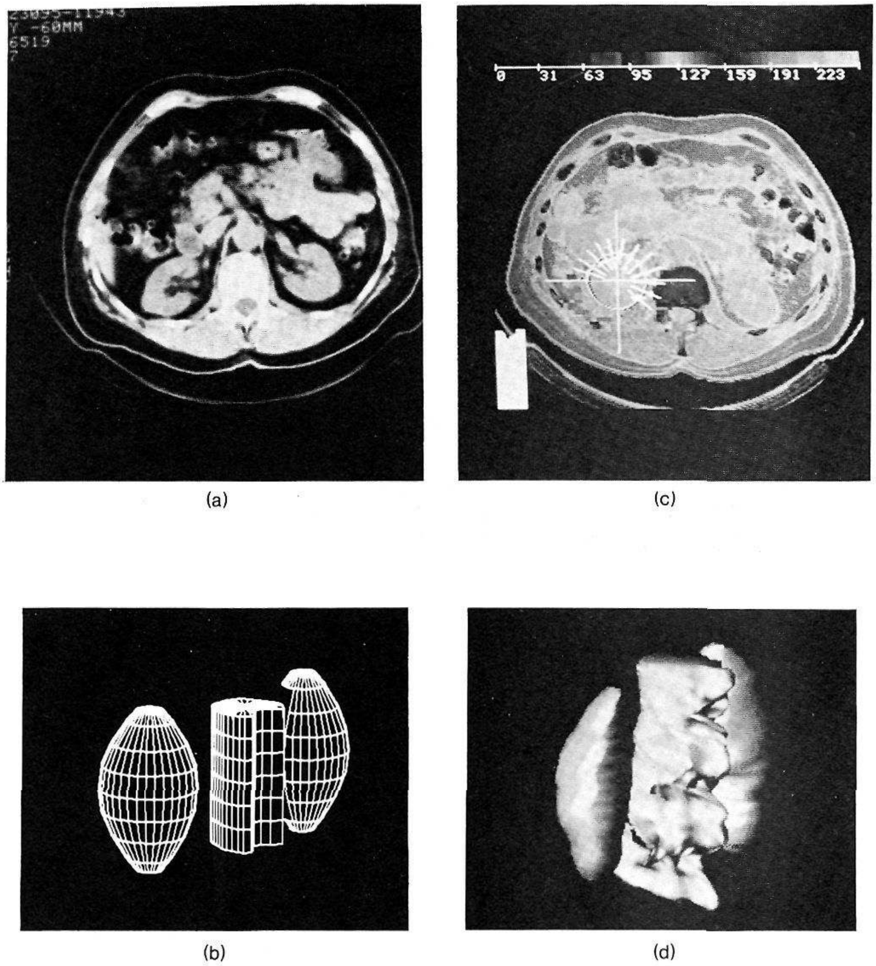


Fig. 1.2 Finding a kidney in a computer-aided tomographic scan. (a) One slice of scan data; (b) prototype kidney model; (c) model fitting; (d) resulting kidney and spinal cord instances.

ments. Further specialization can often be learned, but it is built on basic immutable assumptıons about the world which underlie the vision system.

A basic sort of object recognition capability is the "figure/ground" discrimination that separates objects from the "background." Other basic organizational predispositions are revealed by the "Gestalt laws" of clustering, which demonstrate rules our vision systems use to form simple arrays of stimuli into more coherent spatial groups. A dramatic example of specialized object perception for human beings is revealed in our "face recognition" capability, which seems to occupy a large volume of brain matter. Geometric visual illusions are more surprising symptoms of nonintuitive processing that is performed by our vision systems, either for some direct purpose or as a side effect of its specialized architecture. Some other illusions clearly reflect the intervention of high-level knowledge. For instance, the familiar "Necker cube reversal" is grounded in our three-dimensional models for cubes.

Low-level processing capabilities are elusive; they are unconscious, and they are not well connected to other systems that allow direct introspection. For instance, our visual memory for images is quite impressive, yet our quantitative verbal descriptions of images are relatively primitive. The biological visual "hardware" has been developed, honed, and specialized over a very long period. However, its organization and functionality is not well understood except at extreme levels of detail and generality—the behavior of small sets of cat or monkey cortical cells and the behavior of human beings in psychophysical experiments.

Computer vision is thus immediately faced with a very difficult problem; it must reinvent, with general digital hardware, the most basic and yet inaccessible talents of specialized, parallel, and partly analog biological visual systems. Figure 1.3 may give a feeling for the problem; it shows two visual renditions of a familiar subject. The inset is a normal image, the rest is a plot of the intensities (gray levels) in the image against the image coordinates. In other words, it displays information

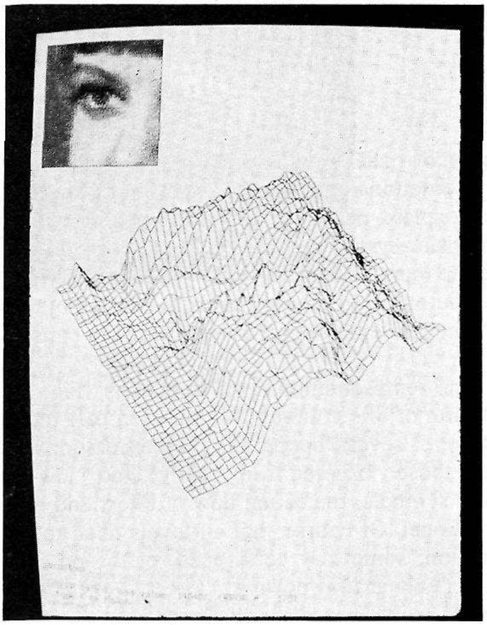


Fig. 1.3 Two representations of an image. One is directly accessible to our low-level processes; the other is not.

with "height" instead of "light." No information is lost, and the display is an image-like object, but we do not immediately see a face in it. The initial representation the computer has to work with is no better; it is typically just an array of numbers from which human beings could extract visual information only very painfully. Skipping the low-level processing we take for granted turns normally effortless perception into a very difficult puzzle.

Computer vision is vitally concerned with both low-level or "early processing" issues and with the high-level and "cognitive" use of knowledge. Where does vision leave off and reasoning and motivation begin? We do not know precisely, but we firmly believe (and hope to show) that powerful, cooperating, rich representations of the world are needed for any advanced vision system. Without them, no system can derive relevant and invariant information from input that is beset with ever-changing lighting and viewpoint, unimportant shape differences, noise, and other large but irrelevant variations. These representations can remove some computational load by predicting or assuming structure for the visual world.

Finally, if a system is to be successful in a variety of tasks, it needs some "meta-level" capabilities: it must be able to model and reason about its own goals and capabilities, and the success of its approaches. These complex and related models must be manipulated by cognitive-like techniques, even though introspectively the perceptual process does not always "feel" to us like cognition.

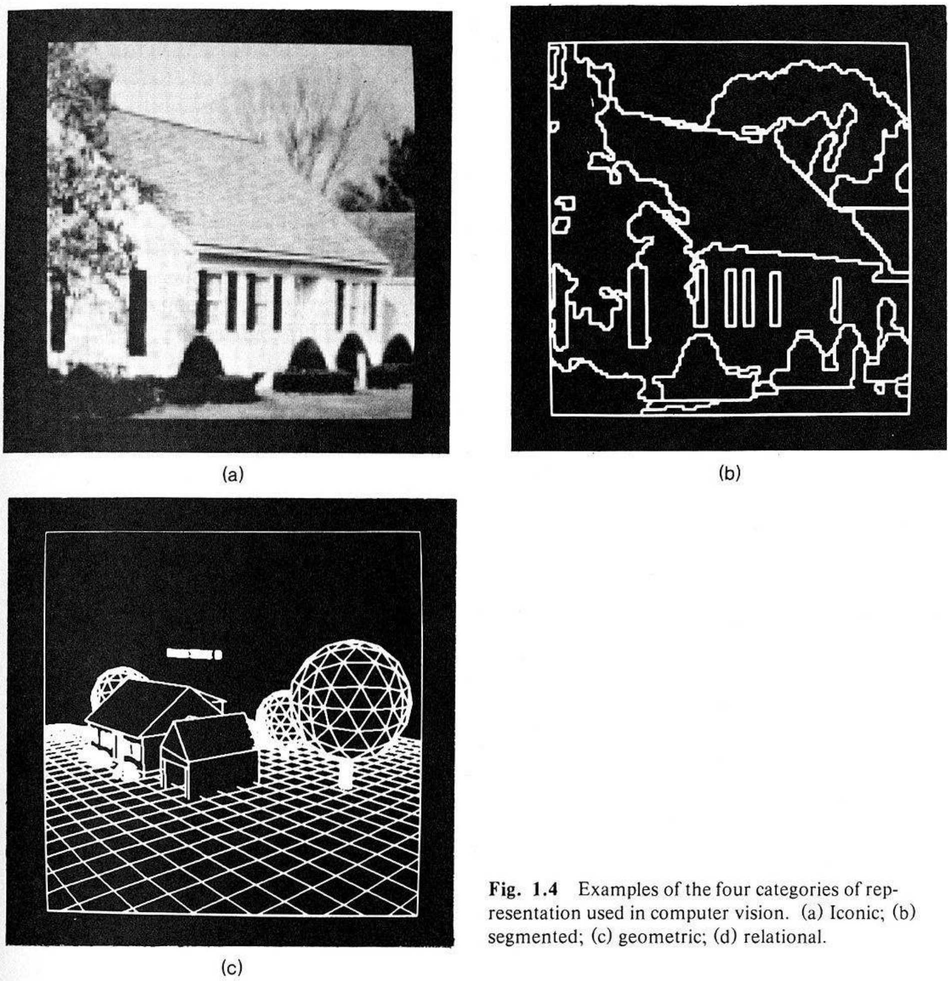
Computer Vision Systems

1.3 A RANGE OF REPRESENTATIONS

Visual perception is the relation of visual input to previously existing *models* of the world. There is a large representational gap between the image and the models ("ideas," "concepts") which explain, describe, or abstract the image information. To bridge that gap, computer vision systems usually have a (loosely ordered) *range of representations* connecting the input and the "output" (a final description, decision, or interpretation). Computer vision then involves the design of these intermediate representations and the implementation of algorithms to construct them and relate them to one another.

We broadly categorize the representations into four parts (Fig. 1.4) which correspond with the organization of this volume. Within each part there may be several layers of representation, or several cooperating representations. Although the sets of representations are loosely ordered from "early" and "low-level" *signals* to "late" and "*cognitive*" symbols, the actual flow of effort and information between them is not unidirectional. Of course, not all levels need to be used in each computer vision application; some may be skipped, or the processing may start partway up the hierarchy or end partway down it.

Generalized images (Part I) are *iconic* (image-like) and *analogical* representations of the input data. Images may initially arise from several technologies.


(a)

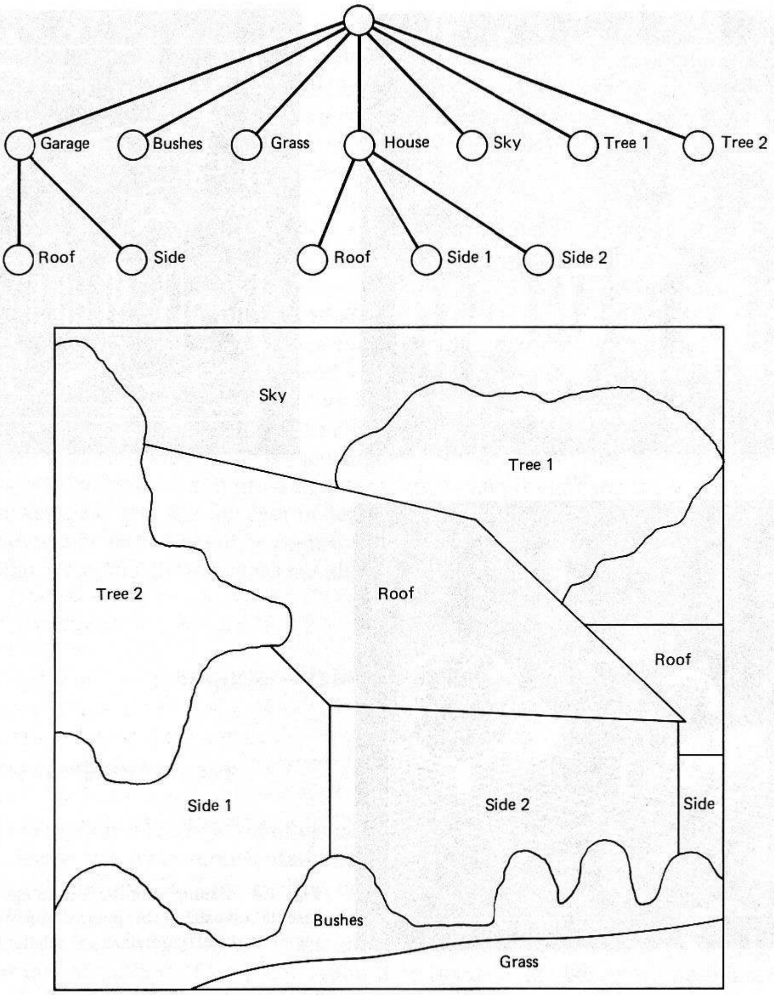
(b)

(c)

Fig. 1.4 Examples of the four categories of representation used in computer vision. (a) Iconic; (b) segmented; (c) geometric; (d) relational.

Domain-independent processing can produce other iconic representations more directly useful to later processing, such as arrays of *edge elements* (gray-level discontinuities). *Intrinsic images* can sometimes be produced at this level—they reveal physical properties of the imaged scene (such as surface orientations, range, or surface reflectance). Often *parallel processing* can produce generalized images. More generally, most "low-level" processes can be implemented with parallel computation.

Segmented images (Part II) are formed from the generalized image by gathering its elements into sets likely to be associated with meaningful *objects* in the scene. For instance, segmenting a scene of planar polyhedra (blocks) might result in a set of *edge segments* corresponding to polyhedral edges, or a set of two-



(d)

Fig. 1.4 (cont.)

dimensional *regions* in the image corresponding to polyhedral faces. In producing the segmented image, knowledge about the particular domain at issue begins to be important both to save computation and to overcome problems of *noise* and inadequate data. In the planar polyhedral example, it helps to know beforehand that the line segments must be straight. *Texture* and *motion* are known to be very important in segmentation, and are currently topics of active research; knowledge in these areas is developing very fast.

Geometric representations (Part III) are used to capture the all-important idea of two-dimensional and three-dimensional *shape*. Quantifying shape is as important as it is difficult. These geometric representations must be powerful enough to support complex and general processing, such as "simulation" of the effects of lighting and motion. Geometric structures are as useful for encoding previously acquired knowledge as they are for re-representing current visual input. Computer vision requires some basic mathematics; Appendix 1 has a brief selection of useful techniques.

Relational models (Part IV) are complex assemblages of representations used to support sophisticated high-level processing. An important tool in *knowledge representation* is *semantic nets*, which can be used simply as an organizational convenience or as a formalism in their own right. High-level processing often uses prior knowledge and models acquired prior to a perceptual experience. The basic mode of processing turns from *constructing* representations to *matching* them. At high levels, *propositional* representations become more important. They are made up of assertions that are true or false with respect to a model, and are manipulated by rules of *inference*. Inference-like techniques can also be used for *planning*, which models situations and actions through time, and thus must reason about temporally varying and hypothetical worlds. The higher the level of representation, the more marked is the flow of *control* (direction of attention, allocation of effort) downward to lower levels, and the greater the tendency of algorithms to exhibit *serial processing*. These issues of control are basic to complex information processing in general and computer vision in particular; Appendix 2 outlines some specific control mechanisms.

Figure 1.5 illustrates the loose classification of the four categories into analogical and propositional representations. We consider generalized and segmented images as well as geometric structures to be analogical models. Analogical models capture directly the relevant characteristics of the represented objects, and are manipulated and interrogated by simulation-like processes. Relational models are generally a mix of analogical and propositional representations. We develop this distinction in more detail in Chapter 10.

1.4 THE ROLE OF COMPUTERS

The computer is a congenial tool for research into visual perception.

- Computers are versatile and forgiving experimental subjects. They are easily and ethically reconfigurable, not messy, and their workings can be scrutinized in the finest detail.
- Computers are demanding critics. Imprecision, vagueness, and oversights are not tolerated in the computer implementation of a theory.
- Computers offer new metaphors for perceptual psychology (also neurology, linguistics, and philosophy). Processes and entities from computer science provide powerful and influential conceptual tools for thinking about perception and cognition.
- Computers can give precise measurements of the amount of processing they

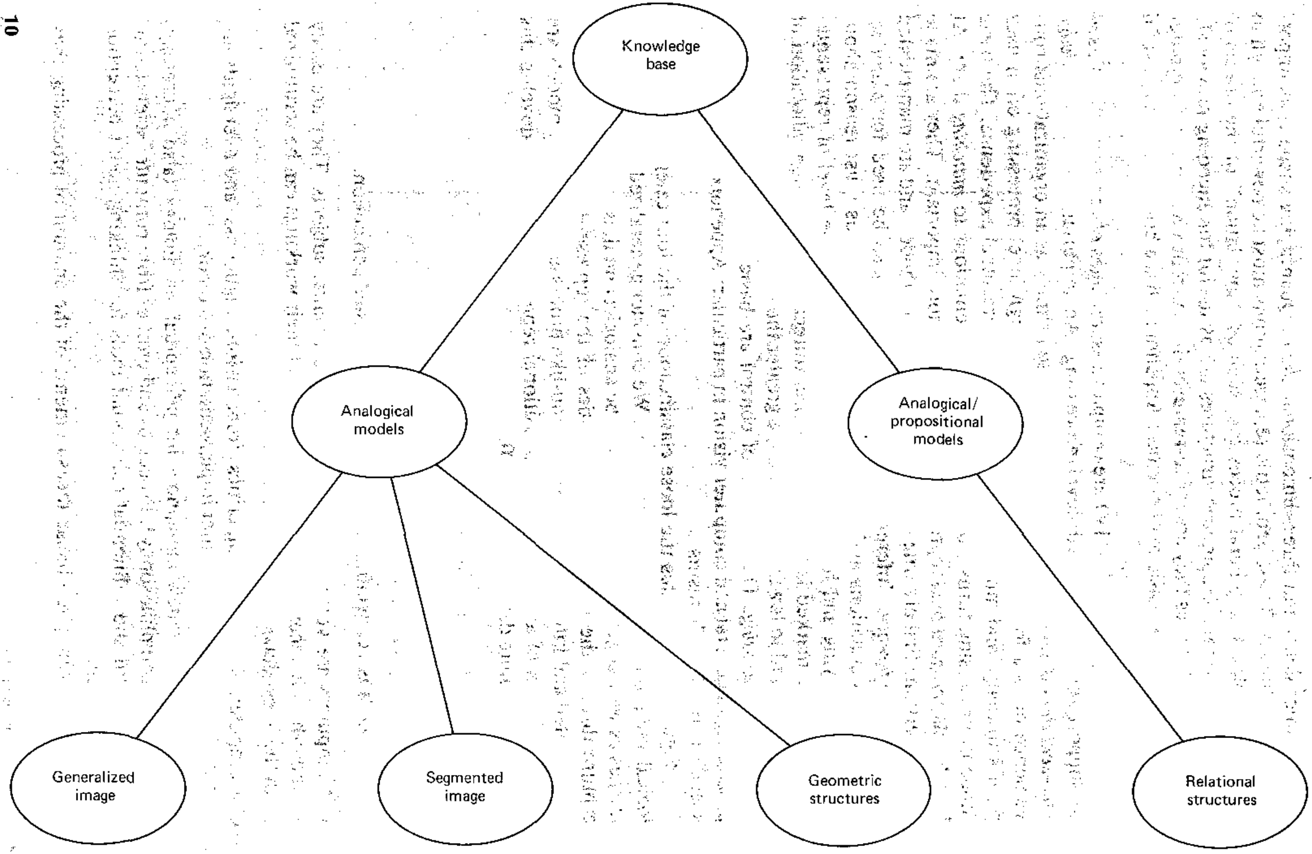


Fig. 1.5 The knowledge base of a complex computer vision system, showing four basic representational categories.

Table 1.1

EXAMPLES OF IMAGE ANALYSIS TASKS

Domain	*Objects*	*Modality*	*Tasks*	*Knowledge Sources*
Robotics	Three-dimensional outdoor scenes indoor scenes	Light X-rays	Identify or describe objects in scene Industrial tasks	Models of objects Models of the reflection of light from objects
	Mechanical parts	Light Structured light		
Aerial images	Terrain Buildings, etc.	Light Infrared Radar	Improved images Resource analyses Weather prediction Spying Missile guidance Tactical analysis	Maps Geometrical models of shapes Models of image formation
Astronomy	Stars Planets	Light	Chemical composition Improved images	Geometrical models of shapes
Medical Macro	Body organs	X-rays Ultrasound Isotopes Heat	Diagnosis of abnormalities Operative and treatment planning	Anatomical models Models of image formation
	Cells	Electronmicroscopy	Pathology, cytology	
Micro	Protein chains Chromosomes	Light	Karyotyping	Models of shape
Chemistry	Molecules	Electron densities	Analysis of molecular compositions	Chemical models Structured models
Neuroanatomy	Neurons	Light Electronmicroscopy	Determination of spatial orientation	Neural connectivity
Physics	Particle tracks	Light	Find new particles Identify tracks	Atomic physics

do. A computer implementation places an upper limit on the amount of computation necessary for a task.

- Computers may be used either to mimic what we understand about human perceptual architecture and processes, or to strike out in different directions to try to achieve similar ends by different means.
- Computer models may be judged either by their efficacy for applications and on-the-job performance or by their internal organization, processes, and structures—the theory they embody.

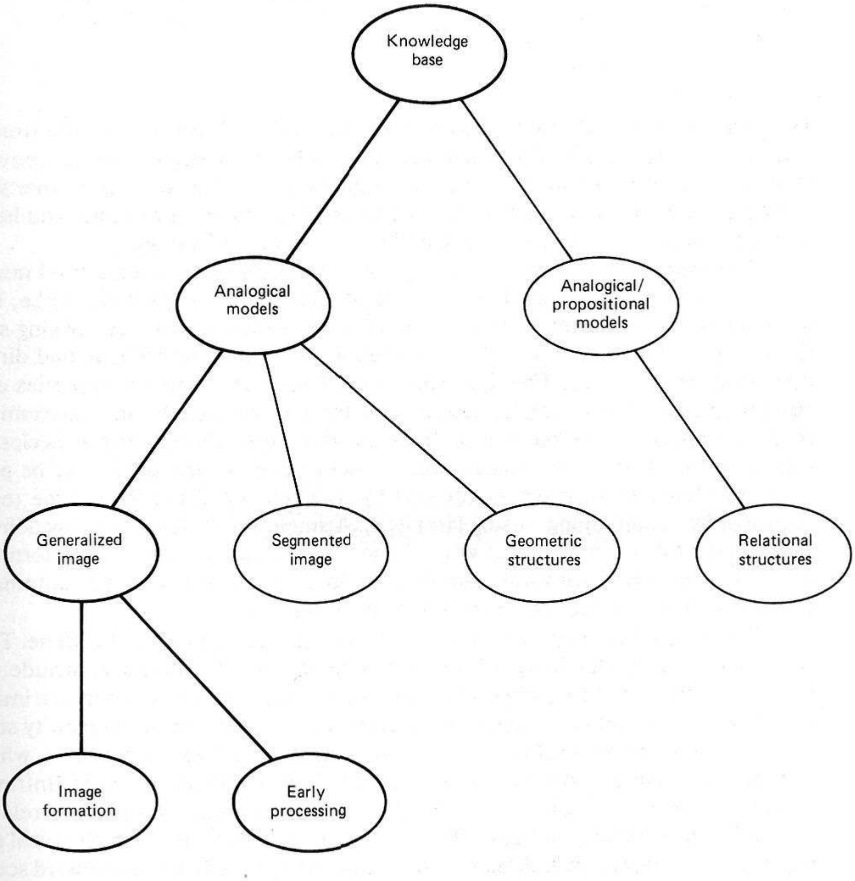
1.5 COMPUTER VISION RESEARCH AND APPLICATIONS

"Pure" computer vision research often deals with relatively domain-independent considerations. The results are useful in a broad range of contexts. Almost always such work is demonstrated in one or more applications areas, and more often than not an initial application problem motivates consideration of the general problem. Applications of computer vision are exciting, and their number is growing as computer vision becomes better understood. Table 1.1 gives a partial list of "classical" and current applications areas.

Within the organization outlined above, this book presents many specific ideas and techniques with general applicability. It is meant to provide enough basic knowledge and tools to support attacks on both applications and research topics.

GENERALIZED IMAGES

I



The first step in the vision process is image formation. Images may arise from a variety of technologies. For example, most television-based systems convert reflected light intensity into an electronic signal which is then digitized; other systems use more exotic radiations, such as x-rays, laser light, ultrasound, and heat. The net result is usually an array of samples of some kind of energy.

The vision system may be entirely passive, taking as input a digitized image from a microwave or infrared sensor, satellite scanner, or a planetary probe, but more likely involves some kind of *active imaging.* Automated active imaging systems may control the direction and resolution of sensors, or regulate and direct their own light sources. The light source itself may have special properties and structure designed to reveal the nature of the three-dimensional world; an example is to use a plane of light that falls on the scene in a stripe whose structure is closely related to the structure of opaque objects. Range data for the scene may be provided by stereo (two images), but also by triangulation using light-stripe techniques or by ''spotranging'' using laser light. A single hardware device may deliver range and multispectral reflectivity (''color'') information. The image-forming device may also perform various other operations. For example, it may automatically smooth or enhance the image or vary its resolution.

The *generalized image* is a set of related image-like entities for the scene. This set may include related images from several modalities, but may also include the results of significant processing that can extract *intrinsic images.* An intrinsic image is an ''image,'' or array, of representations of an important physical quantity such as surface orientation, occluding contours, velocity, or range. Object color, which is a different entity from sensed red–green–blue wavelengths, is an intrinsic quality. These intrinsic physical qualities are extremely useful; they can be related to physical objects far more easily than the original input values, which reveal the physical parameters only indirectly. An intrinsic image is a major step toward scene understanding and usually represents significant and interesting computations.

The information necessary to compute an intrinsic image is contained in the input image itself, and is extracted by "inverting" the transformation wrought by the imaging process, the reflection of radiation from the scene, and other physical processes. An example is the fusion of two stereo images to yield an intrinsic range image. Many algorithms to recover intrinsic images can be realized with *parallel* implementations, mirroring computations that may take place in the lower neurological levels of biological image processing.

All of the computations listed above benefit from the idea of *resolution pyramids.* A pyramid is a generalized image data structure consisting of the same image at several successively increasing levels of resolution. As the resolution increases, more samples are required to represent the increased information and hence the successive levels are larger, making the entire structure look like a pyramid. Pyramids allow the introduction of many different coarse-to-fine image-resolution algorithms which are vastly more efficient than their single-level, high-resolution-only counterparts.

Image Formation 2

2.1 IMAGES

Image formation occurs when a *sensor* registers *radiation* that has interacted with *physical objects.* Section 2.2 deals with mathematical models of images and image formation. Section 2.3 describes several specific image formation technologies.

The mathematical model of imaging has several different components.

1. An *image function* is the fundamental abstraction of an image.
2. A *geometrical model* describes how three dimensions are projected into two.
3. A *radiometrical model* shows how the imaging geometry, light sources, and reflectance properties of objects affect the light measurement at the sensor.
4. A *spatial* frequency model describes how spatial variations of the image may be characterized in a transform domain.
5. A *color model* describes how different spectral measurements are related to image colors.
6. A *digitizing model* describes the process of obtaining discrete samples.

This material forms the basis of much image-processing work and is developed in much more detail elsewhere, e.g., [Rosenfeld and Kak 1976; Pratt 1978]. Our goals are not those of image processing, so we limit our discussion to a summary of the essentials.

The wide range of possible sources of samples and the resulting different implications for later processing motivate our overview of specific imaging techniques. Our goal is not to provide an exhaustive catalog, but rather to give an idea of the range of techniques available. Very different analysis techniques may be needed depending on how the image was formed. Two examples illustrate this point. If the image is formed by reflected light intensity, as in a photograph, the image records both light from primary light sources and (more usually) the light reflected off physical surfaces. We show in Chapter 3 that in certain cases we can use these kinds of images together with knowledge about physics to derive the orientation of the surfaces. If, on the other hand, the image is a computed tomogram of the human body (discussed in Section 2.3.4), the image represents tissue density of internal organs. Here orientation calculations are irrelevant, but general segmentation techniques of Chapters 4 and 5 (the agglomeration of neighboring samples of similar density into units representing organs) are appropriate.

2.2 IMAGE MODEL

Sophisticated image models of a statistical flavor are useful in image processing [Jan 1981]. Here we are concerned with more geometrical considerations.

2.2.1 Image Functions

An *image function* is a mathematical representation of an image. Generally, an image function is a vector-valued function of a small number of arguments. A special case of the image function is the *digital (discrete) image function*, where the arguments to and value of the function are all integers. Different image functions may be used to represent the same image, depending on which of its characteristics are important. For instance, a camera produces an image on black-and-white film which is usually thought of as a real-valued function (whose value could be the density of the photographic negative) of two real-valued arguments, one for each of two spatial dimensions. However, at a very small scale (the order of the film grain) the negative basically has only two densities, "opaque" and "transparent."

Most images are presented by functions of two *spatial* variables $f(\mathbf{x}) = f(x, y)$, where $f(x, y)$ is the brightness of the gray level of the image at a spatial coordinate (x, y). A multispectral image $\mathbf{f}$ is a vector-valued function with components $(f_1 \ldots f_n)$. One special multispectral image is a color image in which, for example, the components measure the brightness values of each of three wavelengths, that is,

$$\boldsymbol{f}(\mathbf{x}) = \left\{ f_{\text{red}}(\mathbf{x}), f_{\text{blue}}(\mathbf{x}), f_{\text{green}}(\mathbf{x}) \right\}$$

Time-varying images $\boldsymbol{f}(\mathbf{x}, t)$ have an added temporal argument. For special three-dimensional images, $\mathbf{x} = (x, y, z)$. Usually, both the domain and range of $\boldsymbol{f}$ are bounded.

An important part of the formation process is the conversion of the image representation from a continuous function to a discrete function; we need some way of describing the images as samples at discrete points. The mathematical tool we shall use is the *delta function*.

Formally, the delta function may be defined by

$$\delta(x) = \begin{cases} 0 \text{ when } x \neq 0 \\ \infty \text{ when } x = 0 \end{cases} \tag{2.1}$$

$$\int_{-\infty}^{\infty} \delta(x)\, dx = 1$$

If some care is exercised, the delta function may be interpreted as the limit of a set of functions:

$$\delta(x) = \lim_{n \to \infty} \delta_n(x)$$

wh^re

$$\delta_n(x) = \begin{cases} n & \text{if } |x| < \dfrac{1}{2n} \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

A useful property of the delta function is the *sifting property:*

$$\int_{-\infty}^{\infty} f(x)\, \delta(x-a)\, dx = f(a) \tag{2.3}$$

A continuous image may be multipled by a two-dimensional "comb," or array of delta functions, to extract a finite number of discrete *samples* (one for each delta function). This mathematical model of the sampling process will be useful later.

2.2.2 Imaging Geometry

Monocular Imaging

Point projection is the fundamental model for the transformation wrought by our eye, by cameras, or by numerous other imaging devices. To a first-order approximation, these devices act like a pinhole camera in that the image results from projecting scene points through a single point onto an *image plane* (see Fig. 2.1). In Fig. 2.1, the image plane is behind the point of projection, and the image is reversed. However, it is more intuitive to recompose the geometry so that the point of projection corresponds to a *viewpoint* behind the image plane, and the image occurs right side up (Fig. 2.2). The mathematics is the same, but now the viewpoint is $+f$ on the z axis, with $z = 0$ plane being the image plane upon which the image is projected. (f is sometimes called the *focal length* in this context. The use of f in this section should not be confused with the use of f for image function.) As the imaged object approaches the viewpoint, its projection gets bigger (try moving your hand toward your eye). To specify how its imaged size changes, one needs only the geometry of similar triangles. In Fig. 2.2b y', the projected height of the object, is related to its real height y, its position z, and the focal length f by

$$\frac{y}{f - z} = \frac{y'}{f} \tag{2.4}$$

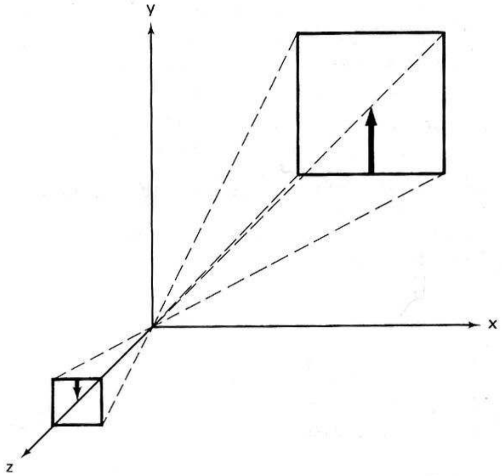


Fig. 2.1 A geometric camera model.

The case for x' is treated similarly:

$$\frac{x}{f - z} = \frac{x'}{f} \tag{2.5}$$

The projected image has $z = 0$ everywhere. However, projecting away the z component is best considered a separate transformation; the projective transform is usually thought to distort the z component just as it does the x and y. *Perspective distortion* thus maps (x, y, z) to

$$(x', y', z') = \left(\frac{fx}{f - z}, \frac{fy}{f - z}, \frac{fz}{f - z}\right) \tag{2.6}$$

The perspective transformation yields *orthographic projection* as a special case when the viewpoint is the *point at infinity* in the z direction. Then all objects are projected onto the viewing plane with no distortion of their x and y coordinates.

The perspective distortion yields a three-dimensional object that has been "pushed out of shape"; it is more shrunken the farther it is from the viewpoint. The z component is not available directly from a two-dimensional image, being identically equal to zero. In our model, however, the distorted z component has information about the distance of imaged points from the viewpoint. When this distorted object is projected orthographically onto the image plane, the result is a perspective picture. Thus, to achieve the effect of railroad tracks appearing to come together in the distance, the perspective distortion transforms the tracks so that they *do* come together (at a point at infinity)! The simple orthographic projection that projects away the z component unsurprisingly preserves this distortion. Several properties of the perspective transform are of interest and are investigated further in Appendix 1.

Binocular Imaging

Basic binocular imaging geometry is shown in Fig. 2.3a. For simplicity, we

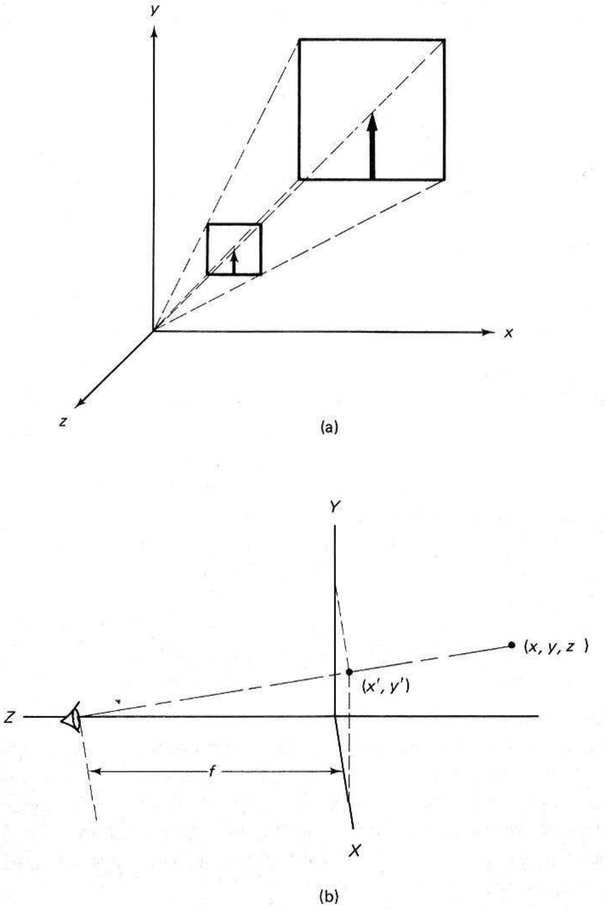


Fig. 2.2 (a) Camera model equivalent to that of Fig. 2.1; (b) definition of terms.

use a system with two viewpoints. In this model the eyes do not *converge*; they are aimed in parallel at the point at infinity in the $-z$ direction. The depth information about a point is then encoded only by its different positions (*disparity*) in the two image planes.

With the stereo arrangement of Fig. 2.3,

$$x' = \frac{(x - d)f}{f - z}$$

$$x'' = \frac{(x + d)f}{f - z}$$

where (x', y') and (x'', y'') are the retinal coordinates for the world point imaged

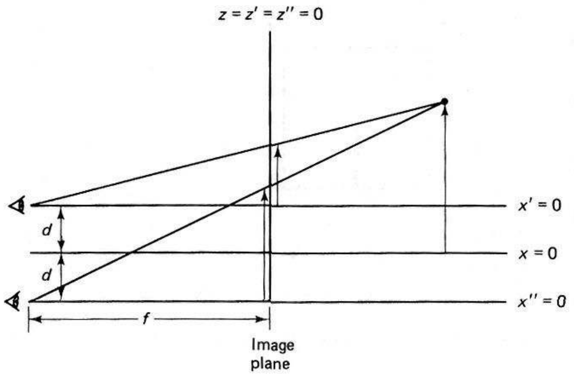


Fig. 2.3 A nonconvergent binocular imaging system.

through each eye. The *baseline* of the binocular system is $2d$. Thus

$$(f - z)x' = (x - d)f \tag{2.7}$$

$$(f - z)x'' = (x + d)f \tag{2.8}$$

Subtracting (2.7) from (2.8) gives

$$(f - z)(x'' - x') = 2df$$

or

$$z = f - \frac{2df}{x'' - x'} \tag{2.9}$$

Thus if points can be matched to determine the disparity $(x'' - x')$ and the baseline and focal length are known, the z coordinate is simple to calculate.

If the system can converge its directions of view to a finite distance, convergence angle may also be used to compute depth. The hardest part of extracting depth information from stereo is the *matching* of points for disparity calculations. "Light striping" is a way to maintain geometric simplicity and also simplify matching (Section 2.3.3).

2.2.3 Reflectance

Terminology

A basic aspect of the imaging process is the physics of the reflectance of objects, which determines how their "brightness" in an image depends on their inherent characteristics and the geometry of the imaging situation. A clear presentation of the mathematics of reflectance is given in [Horn and Sjoberg 1978; Horn 1977]. Light *energy flux* Φ is measured in watts; "brightness" is measured with respect to area and solid angle. The *radiant intensity* I of a source is the exitant flux per unit solid angle:

$$I = \frac{d\Phi}{d\omega} \quad \text{watts/steradian} \tag{2.10}$$

Here $d\omega$ is an incremental solid angle. The solid angle of a small area dA measured perpendicular to a radius r is given by

$$d\omega = \frac{dA}{r^2} \tag{2.11}$$

in units of steradians. (The total solid angle of a sphere is 4π.)

The *irradiance* is flux incident on a surface element dA:

$$E = \frac{d\Phi}{dA} \qquad \text{watts/meter}^2 \tag{2.12}$$

and the flux exitant from the surface is defined in terms of the *radiance* L, which is the flux emitted per unit foreshortened surface area per unit solid angle:

$$L = \frac{d^2\Phi}{dA\ \cos\theta\, d\omega} \qquad \text{watts/(meter}^2\text{ steradian)} \tag{2.13}$$

where θ is the angle between the surface normal and the direction of emission.

Image irradiance f is the "brightness" of the image at a point, and is proportional to scene radiance. A "gray-level" is a quantized measurement of image irradiance. Image irradiance depends on the reflective properties of the imaged surfaces as well as on the illumination characteristics. How a surface reflects light depends on its micro-structure and physical properties. Surfaces may be *matte* (dull, flat), *specular* (mirrorlike), or have more complicated reflectivity characteristics (Section 3.5.1). The *reflectance* r of a surface is given quite generally by its Bidirectional Reflectance Distribution Function (BRDF) [Nicodemus et al. 1977]. The BRDF is the ratio of reflected radiance in the direction towards the viewer to the irradiance in the direction towards a small area of the source.

Effects of Geometry on an Imaging System

Let us now analyze a simple image-forming system shown in Fig. 2.4 with the objective of showing how the gray levels are related to the radiance of imaged objects. Following [Horn and Sjoberg 1978], assume that the imaging device is properly focused; rays originating in the infinitesimal area dA_o on the object's surface are projected into some area dA_p in the image plane and no rays from other portions of the object's surface reach this area of the image. The system is assumed to be an ideal one, obeying the laws of simple geometrical optics.

The energy flux/unit area that impinges on the sensor is defined to be E_p. To show how E_p is related to the scene radiance L, first consider the flux arriving at the lens from a small surface area dA_o. From (2.13) this is given as

$$d\Phi = dA_o \int L \cos\theta\, d\omega \tag{2.14}$$

This flux is assumed to arrive at an area dA_p in the imaging plane. Hence the irradiance is given by [using Eq. (2.12)]

$$E_p = \frac{d\Phi}{dA_p} \tag{2.15}$$

Now relate dA_o to dA_p by equating the respective solid angles as seen from the lens; that is [making use of Eq. (2.12)],

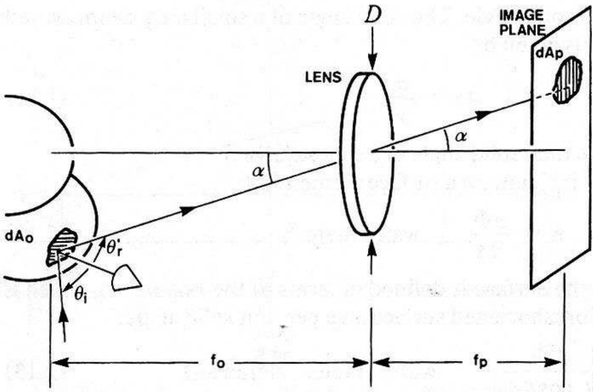


Fig. 2.4 Geometry of an image forming system.

$$dA_o \frac{\cos\theta}{f_o^2} = dA_p \frac{\cos\alpha}{f_p^2} \tag{2.16}$$

Substituting Eqs. (2.16) and (2.14) into (2.15) gives

$$E = \cos\alpha \left(\frac{f_o}{f_p} \right)^2 \int L d\omega \tag{2.17}$$

The integral is over the solid angle seen by the lens. In most instances we can assume that L is constant over this angle and hence can be removed from the integral. Finally, approximate $d\omega$ by the area of the lens foreshortened by $\cos\alpha$, that is, $(\pi/4)D^2 \cos\alpha$ divided by the distance $f_o/\cos\alpha$ squared:

$$d\omega = \frac{\pi}{4} D^2 \frac{\cos^3\alpha}{f_o^2} \tag{2.18}$$

so that finally

$$E = \frac{1}{4} \left(\frac{D}{f_p} \right)^2 \cos^4\alpha \, \pi L \tag{2.19}$$

The interesting results here are that (1) the image irradiance is proportional to the scene radiance L, and (2) the factor of proportionality includes the fourth power of the off-axis angle α. Ideally, an imaging device should be calibrated so that the variation in sensitivity as a function of α is removed.

2.2.4 Spatial Properties

The Fourier Transform

An image is a spatially varying function. One way to analyze spatial variations is the decomposition of an image function into a set of orthogonal functions, one such set being the Fourier (sinusoidal) functions. The Fourier transform may be used to transform the intensity image into the domain of *spatial frequency*. For notational convenience and intuition, we shall generally use as an example the continuous one-dimensional Fourier transform. The results can readily be extended to the discrete case and also to higher dimensions [Rosenfeld and Kak 1976]. In two dimensions we shall denote transform domain coordinates by (u, v). The one-dimensional Fourier transform, denoted $\mathcal{F}$, is defined by

$$\mathcal{F}[f(x)] = F(u)$$

where

$$F(u) = \int_{-\infty}^{+\infty} f(x)\exp(-j2\pi ux)\,dx \tag{2.20}$$

where $j = \sqrt{(-1)}$. Intuitively, Fourier analysis expresses a function as a sum of sine waves of different frequency and phase. The Fourier transform has an *inverse* $^{-1}[F(u)] = f(x)$. This inverse is given by

$$f(x) = \int_{-\infty}^{\infty} F(u)\exp(j2\pi ux)\,du \tag{2.21}$$

The transform has many useful properties, some of which are summarized in Table 2.1. Common one-dimensional Fourier transform pairs are shown in Table 2.2.

The transform F(u) is simply another representation of the image function. Its meaning can be understood by interpreting Eq. (2.21) for a specific value of x, say x_0:

$$f(x_0) = \int F(u)\exp(j2\pi ux_0)\,du \tag{2.22}$$

This equation states that a particular point in the image can be represented by a weighted sum of complex exponentials (sinusoidal patterns) at different spatial frequencies u. $F(u)$ is thus a *weighting function* for the different frequencies. Low-spatial frequencies account for the "slowly" varying gray levels in an image, such as the variation of intensity over a continuous surface. High-frequency components are associated with "quickly varying" information, such as edges. Figure 2.5 shows the Fourier transform of an image of rectangles, together with the effects of removing low- and high-frequency components.

The Fourier transform is defined above to be a continuous transform. Although it may be performed instantly by optics, a discrete version of it, the "fast Fourier transform," is almost universally used in image processing and computer vision. This is because of the relative versatility of manipulating the transform in the digital domain as compared to the optical domain. Image-processing texts, e.g., [Pratt 1978; Gonzalez and Wintz 1977] discuss the FFT in some detail; we content ourselves with an algorithm for it (Appendix 1).

The Convolution Theorem

Convolution is a very important image-processing operation, and is a basic operation of linear systems theory. The convolution of two functions f and g is a function h of a displacement y defined as

$$h(y) = f*g = \int_{-\infty}^{\infty} f(x)g(y-x)\,dx \tag{2.23}$$

Table 2.1

PROPERTIES OF THE FOURIER TRANSFORM

		Spatial Domain	*Frequency Domain*
		$f(x)$ $g(x)$	$F(u) = \mathfrak{F}[f(x)]$ $G(u) = \mathfrak{F}[g(x)]$
(1)	Linearity $c_1 f(x) + c_2 g(x)$ c_1, c_2 scalars		$c_1 F(u) + c_2 G(u)$
(2)	Scaling $f(ax)$		$\frac{1}{\lvert a \rvert} F\left[\frac{u}{a}\right]$
(3)	Shifting $f(x - x_0)$		$e^{-2\pi j x_0} F(u)$
(4)	Symmetry $F(x)$		$f(-u)$
(5)	Conjugation $f^*(x)$		$F^*(-u)$
(6)	Convolution $h(x) = f * g = \int_{-\infty}^{\infty} f(x') g(x - x')\, dx'$		$F(u) G(u)$
(7)	Differentiation $\frac{d^n f(x)}{dx^n}$		$(2\pi j u)^n F(u)$

Parseval's theorem:

$$\int_{-\infty}^{\infty} |f(x)|^2 dx = \int_{-\infty}^{\infty} |F(\xi)|^2 d\xi$$

$$\int_{-\infty}^{\infty} f(x) g^*(x)\, dx = \int_{-\infty}^{\infty} F(\xi) G^*(\xi)\, d\xi$$

$f(x)$	$F(\xi)$
Real(R)	Real part even (RE) Imaginary part odd (IO)
Imaginary (I)	RO,IE
RE,IO	R
RE,IE	I
RE	RE
RO	IO
IE	IE
IO	RO
Complex even (CE)	CE
CO	CO

Table 2.2

FOURIER TRANSFORM PAIRS

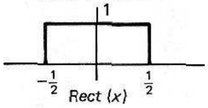
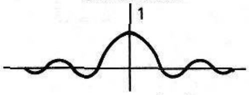
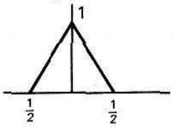
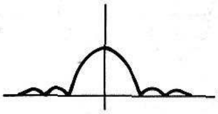
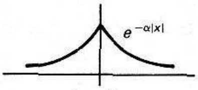
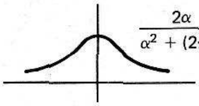
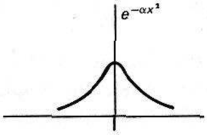
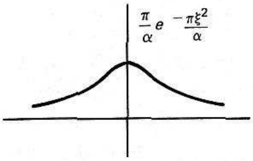
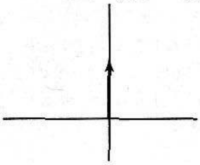
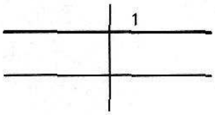
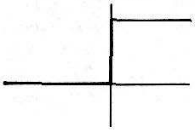
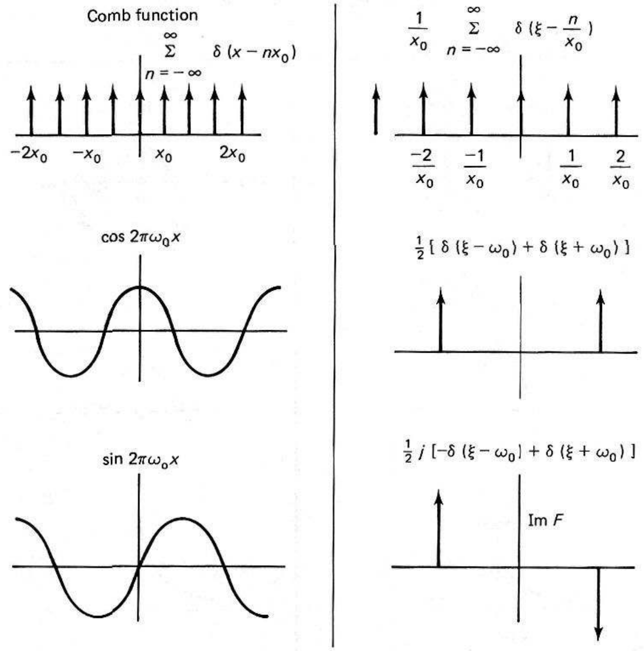
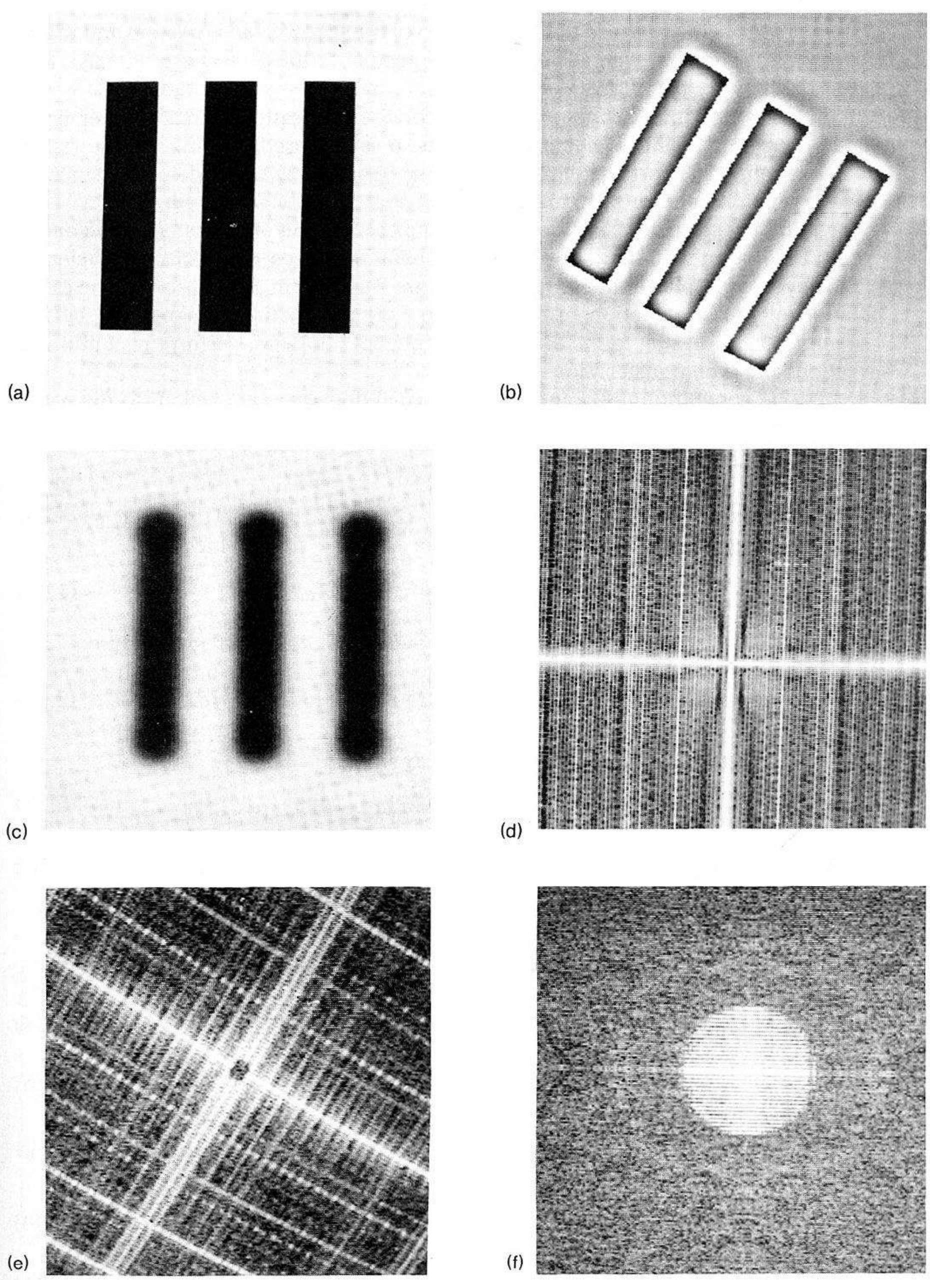
$f(x)$	$F(\xi)$
Rectangle function 	Sinc function  $\text{Sinc}(\xi) = \frac{\sin \pi\xi}{\pi\xi}$
Triangle function 	 — $\text{Sinc}^2\ (\xi)$
Exponential 	 $\frac{2\alpha}{\alpha^2 + (2\pi\xi)^2}$
Gaussian 	 $\frac{\pi}{\alpha} e^{-\frac{\pi\xi^2}{\alpha}}$
Unit impulse $\delta(x)$ 	 1
Unit step 	$\frac{1}{2}\ \delta\ (\xi) + \frac{1}{2\pi j\xi}$

Table 2.2 (cont.)



Intuitively, one function is "swept past" (in one dimension) or "rubbed over" (in two dimensions) the other. The value of the convolution at any displacement is the integral of the product of the (relatively displaced) function values. One common phenomenon that is well expressed by a convolution is the formation of an image by an optical system. The system (say a camera) has a "point-spread function," which is the image of a single point. (In linear systems theory, this is the "impulse response," or response to a delta-function input.) The ideal point-spread function is, of course, a point. A typical point-spread function is a two-dimensional Gaussian spatial distribution of intensities, but may include such phenomena as diffraction rings. In any event, if the camera is modeled as a linear system (ignor-

Fig. 2.5 (on facing page) (a) An image, $f(x, y)$. (b) A rotated version of (a), filtered to enhance high spatial frequencies. (c) Similar to (b), but filtered to enhance low spatial frequencies. (d), (e), and (f) show the logarithm of the power spectrum of (a), (b), and (c). The power spectrum is the log square modulus of the Fourier transform $F(u, v)$. Considered in polar coordinates (ρ, θ), points of small ρ correspond to low spatial frequencies ("slowly-varying" intensities), large ρ to high spatial frequencies contributed by "fast" variations such as step edges. The power at (ρ, θ) is determined by the amount of intensity variation at the frequency ρ occurring at the angle θ.


(a)
(b)
(c)
(d)
(e)
(f)

ing the added complexity that the point-spread function usually varies over the field of view), the image is the convolution of the point-spread function and the input signal. The point-spread function is rubbed over the perfect input image, thus blurring it.

Convolution is also a good model for the application of many other linear operators, such as line-detecting templates. It can be used in another guise (called correlation) to perform matching operations (Chapter 3) which detect instances of subimages or features in an image.

In the spatial domain, the obvious implementation of the convolution operation involves a shift–multiply–integrate operation which is hard to do efficiently. However, multiplication and convolution are "transform pairs," so that the calculation of the convolution in one domain (say the spatial) is simplified by first Fourier transforming to the other (the frequency) domain, performing a multiplication, and then transforming back.

The convolution of f and g in the spatial domain is equivalent to the pointwise product of F and G in the frequency domain,

$$\mathcal{F}(f*g) = FG \tag{2.24}$$

We shall show this in a manner similar to [Duda and Hart 1973]. First we prove the *shift theorem*. If the Fourier transform of $f(x)$ is $F(u)$, defined as

$$F(u) = \int_x f(x) \exp[-j2\pi(ux)]dx \tag{2.25}$$

then

$$\mathcal{F}[f(x-a)] = \int_x f(x-a) \exp[-j2\pi(ux)]dx \tag{2.26}$$

changing variables so that $x' = x - a$ and $dx = dx'$

$$= \int_{x'} f(x') \exp\{-j2\pi[u(x'+a)]\}dx' \tag{2.27}$$

Now $\exp[-j2\pi u(x'+a)] = \exp(-j2\pi ua)\exp(-j2\pi ux')$, where the first term is a constant. This means that

$$\mathcal{F}[f(x-a)] = \exp(-j2\pi ua)F(u) \qquad \text{(shift theorem)}$$

Now we are ready to show that $\mathcal{F}[f(\mathrm{x})*g(\mathrm{x})] = F(u)G(u)$.

$$\mathcal{F}(f*g) = \int_y\{\int_x f(x)g(y-x)\} \exp(-j2\pi uy)\,dx\,dy \tag{2.28}$$

$$= \int_x f(x)\{\int_y g(y-x) \exp(-j2\pi uy)\,dy\}\,dx \tag{2.29}$$

Recognizing that the terms in braces represent $\mathcal{F}[g(y-x)]$ and applying the shift theorem, we obtain

$$\mathcal{F}(f*g) = \int_x f(x)\exp(-j2\pi ux)G(u)\,dx \tag{2.30}$$

$$= F(u)G(u) \tag{2.31}$$

2.2.5 Color

Not all images are monochromatic; in fact, applications using multispectral images are becoming increasingly common (Section 2.3.2). Further, human beings intuitively feel that color is an important part of their visual experience, and is useful or even necessary for powerful visual processing in the real world. Color vision provides a host of research issues, both for psychology and computer vision. We briefly discuss two aspects of color vision: color spaces and color perception. Several models of the human visual system not only include color but have proven useful in applications [Granrath 1981].

Color Spaces

Color spaces are a way of organizing the colors perceived by human beings. It happens that weighted combinations of stimuli at three principal wavelengths are sufficient to define almost all the colors we perceive. These wavelengths form a natural basis or coordinate system from which the color measurement process can be described. Color perception is not related in a simple way to color measurement, however.

Color is a perceptual phenomenon related to human response to different wavelengths in the visible *electromagnetic spectrum* [400 (blue) to 700 nanometers (red); a nanometer (nm) is 10^{-9} meter]. The sensation of color arises from the sensitivities of three types of neurochemical sensors in the retina to the visible spectrum. The relative response of these sensors is shown in Fig. 2.6. Note that each sensor responds to a range of wavelengths. The illumination source has its own spectral composition $f(\lambda)$ which is modified by the reflecting surface. Let $r(\lambda)$ be this reflectance function. Then the measurement R produced by the "red" sensor is given by

$$R = \int f(\lambda) r(\lambda) h_R(\lambda)\, d\lambda \tag{2.32}$$

So the sensor output is actually the integral of three different wavelength-dependent components: the source f, the surface reflectance r, and the sensor h_R.

Surprisingly, only weighted combinations of three delta-function approximations to the different $f(\lambda)h(\lambda)$, that is, $\delta(\lambda_R)$, $\delta(\lambda_G)$, and $\delta(\lambda_B)$, are necessary to

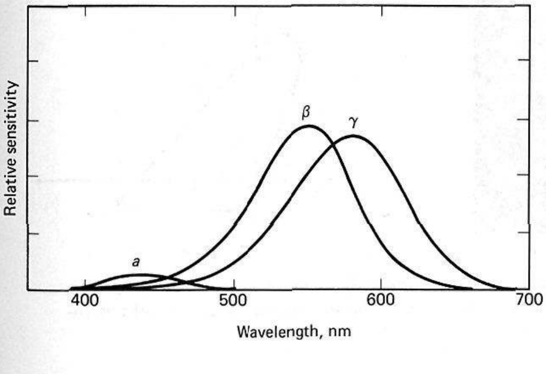


Fig. 2.6 Spectral response of human color sensors.

produce the sensation of nearly all the colors. This result is displayed on a *chromaticity diagram*. Such a diagram is obtained by first normalizing the three sensor measurements:

$$\begin{aligned} r &= \frac{R}{R+G+B} \\ g &= \frac{G}{R+G+B} \\ b &= \frac{B}{R+G+B} \end{aligned} \tag{2.33}$$

and then plotting perceived color as a function of any two (usually red and green). Chromaticity explicitly ignores intensity or brightness; it is a section through the three-dimensional color space (Fig. 2.7). The choice of $(\lambda_R, \lambda_G, \lambda_B) = (410, 530, 650)\,nm$ maximizes the realizable colors, but some colors still cannot be realized since they would require negative values for some of r, g, and b.

Another more intuitive way of visualizing the possible colors from the *RGB* space is to view these measurements as Euclidean coordinates. Here any color can be visualized as a point in the unit cube. Other coordinate systems are useful for different applications; computer graphics has proved a strong stimulus for investigation of different color space bases.

Color Perception

Color perception is complex, but the essential step is a transformation of three input intensity measurements into another basis. The coordinates of the new

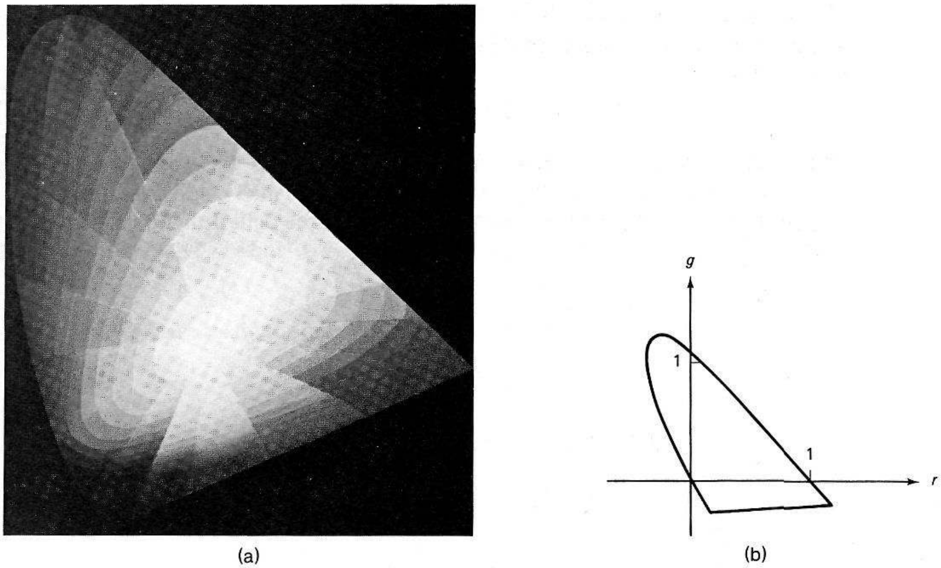


Fig. 2.7 (a) An artist's conception of the chromaticity diagram—*see color insert*; (b) a more useful depiction. Spectral colors range along the curved boundary; the straight boundary is the line of purples.

basis are more directly related to human color judgments.

Although the *RGB* basis is good for the acquisition or display of color information, it is not a particularly good basis to explain the perception of colors. Human vision systems can make good judgments about the relative surface reflectance $r(\lambda)$ despite different illuminating wavelengths; this reflectance seems to be what we mean by surface color.

Another important feature of the color basis is revealed by an ability to perceive in "black and white," effectively deriving intensity information from the color measurements. From an evolutionary point of view, we might expect that color perception in animals would be compatible with preexisting noncolor perceptual mechanisms.

These two needs—the need to make good color judgments and the need to retain and use intensity information—imply that we use a transformed, non-*RGB* basis for color space. Of the different bases in use for color vision, all are variations on this theme: Intensity forms one dimension and color is a two-dimensional subspace. The differences arise in how the color subspace is described. We categorize such bases into two groups.

1. *Intensity/Saturation/Hue (IHS).* In this basis, we compute intensity as

$$\text{intensity:} = R + G + B \tag{2.34}$$

The saturation measures the lack of whiteness in the color. Colors such as "fire engine" red and "grass" green are saturated; pastels (e.g., pinks and pale blues) are desaturated. Saturation can be computed from *RGB* coordinates by the formula [Tenenbaum and Weyl 1975]

$$\text{saturation:} = 1 - \frac{3 \min (R, G, B)}{\text{intensity}} \tag{2.35}$$

Hue is roughly proportional to the average wavelength of the color. It can be defined using *RGB* by the following program fragment:

$$\text{hue:} = \cos^{-1}\left\{\frac{\{\tfrac{1}{2}[(R - G) + (R - B)]\}}{\sqrt{(R - G)^2 + (R - B)(G - B)^{1/2}}}\right\} \tag{2.36}$$

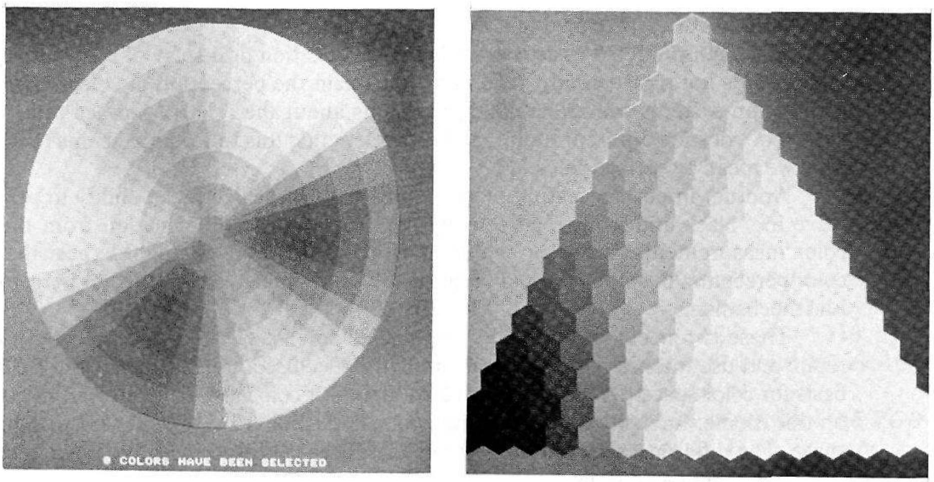
$$\text{If } B > G \text{ then hue:} = 2pi - \text{hue}$$

The IHS basis transforms the *RGB* basis in the following way. Thinking of the color cube, the diagonal from the origin to (1, 1, 1) becomes the intensity axis. Saturation is the distance of a point from that axis and hue is the angle with regard to the point about that axis from some reference (Fig. 2.8).

This basis is essentially that used by artists [Munsell 1939], who term saturation *chroma.* Also, this basis has been used in graphics [Smith 1978; Joblove and Greenberg 1978].

One problem with the IHS basis, particularly as defined by (2.34) through (2.36), is that it contains essential singularities where it is impossible to define the color in a consistent manner [Kender 1976]. For example, hue has an essential singularity for all values of (R, G, B), where $R = G = B$. This means that special care must be taken in algorithms that use hue.

2. *Opponent processes.* The opponent process basis uses Cartesian rather than



(a) (b)

Fig. 2.8 An IHS Color Space. (a) Cross section at one intensity; (b) cross section at one hue—*see color inserts.*

cylindrical coordinates for the color subspace, and was first proposed by Hering [Teevan and Birney 1961]. The simplest form of basis is a linear transformation from R, G, B coordinates. The new coordinates are termed "$R - G$", "$Bl - Y$", and "$W - Bk$":

$$\begin{bmatrix} R - G \\ Bl - Y \\ W - Bk \end{bmatrix} = \begin{bmatrix} 1 & -2 & 1 \\ -1 & -1 & 2 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

The advocates of this representation, such as [Hurvich and Jameson 1957], theorize that this basis has neurological correlates and is in fact the way human beings represent ("name") colors. For example, in this basis it makes sense to talk about a "reddish blue" but not a "reddish green." Practical opponent process models usually have more complex weights in the transform matrix to account for psychophysical data. Some startling experiments [Land 1977] show our ability to make correct color judgments even when the illumination consists of only two principal wavelengths. The opponent process, at the level at which we have developed it, does not demonstrate how such judgments are made, but does show how stimulus at only two wavelengths will project into the color subspace. Readers interested in the details of the theory should consult the references.

Commercial television transmission needs an intensity, or "$W - Bk$" component for black-and-white television sets while still spanning the color space. The National Television Systems Committee (NTSC) uses a "YIQ" basis extracted from RGB via

$$\begin{bmatrix} I \\ Q \\ Y \end{bmatrix} = \begin{bmatrix} 0.60 & -0.28 & -0.32 \\ 0.21 & -0.52 & 0.31 \\ 0.30 & 0.59 & 0.11 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

This basis is a weighted form of

$$(I,\ Q,\ Y) = (\text{“}R\text{–cyan,” “magenta–green,” “}W\text{–}Bk\text{”})$$

2.2.6 Digital Images

The *digital images* with which computer vision deals are represented by m-vector discrete-valued image functions $\boldsymbol{f}(\mathbf{x})$, usually of one, two, three, or four dimensions.

Usually $m = 1$, and both the domain and range of $f(\mathbf{x})$ are discrete. The domain of f is finite, usually a rectangle, and the range of f is positive and bounded: $0 \leqslant f(\mathbf{x}) \leqslant M$ for some integer M. For all practical purposes, the image is a continuous function which is represented by measurements or *samples* at regularly spaced intervals. At the time the image is sampled, the intensity is usually *quantized* into a number of different *gray levels.* For a discrete image, $f(\mathbf{x})$ is an integer gray level, and $\mathbf{x} = (\mathrm{x}, \mathrm{y})$ is a pair of *integer* coordinates representing a sample point in a two-dimensional image plane. Sampling involves two important choices: (1) the *sampling interval,* which determines in a basic way whether all the information in the image is represented, and (2) the *tesselation* or spatial pattern of sample points, which affects important notions of connectivity and distance. In our presentation, we first show qualitatively the effects of sampling and gray-level quantization. Second, we discuss the simplest kinds of tesselations of the plane. Finally, and most important, we describe the sampling theorem, which specifies how close the image samples must be to represent the image unambiguously.

The choice of integers to represent the gray levels and coordinates is dictated by limitations in sensing. Also, of course, there are hardware limitations in representing images arising from their sheer size. Table 2.3 shows the storage required for an image in 8-bit bytes as a function of m, the number of bits per sample, and N, the linear dimension of a square image.

For reasons of economy (and others discussed in Chapter 3) we often use images of considerably less spatial resolution than that required to preserve fidelity to the human viewer. Figure 2.9 provides a qualitative idea of image degradation with decreasing spatial resolution.

As shown in Table 2.3, another way to save space besides using less spatial resolution is to use fewer bits per gray level sample. Figure 2.10 shows an image represented with different numbers of bits per sample. One striking effect is the "contouring" introduced with small numbers of gray levels. This is, in general, a problem for computer vision algorithms, which cannot easily discount the false contours. The choice of spatial and gray-level resolution for any particular computer vision task is an important one which depends on many factors. It is typical in

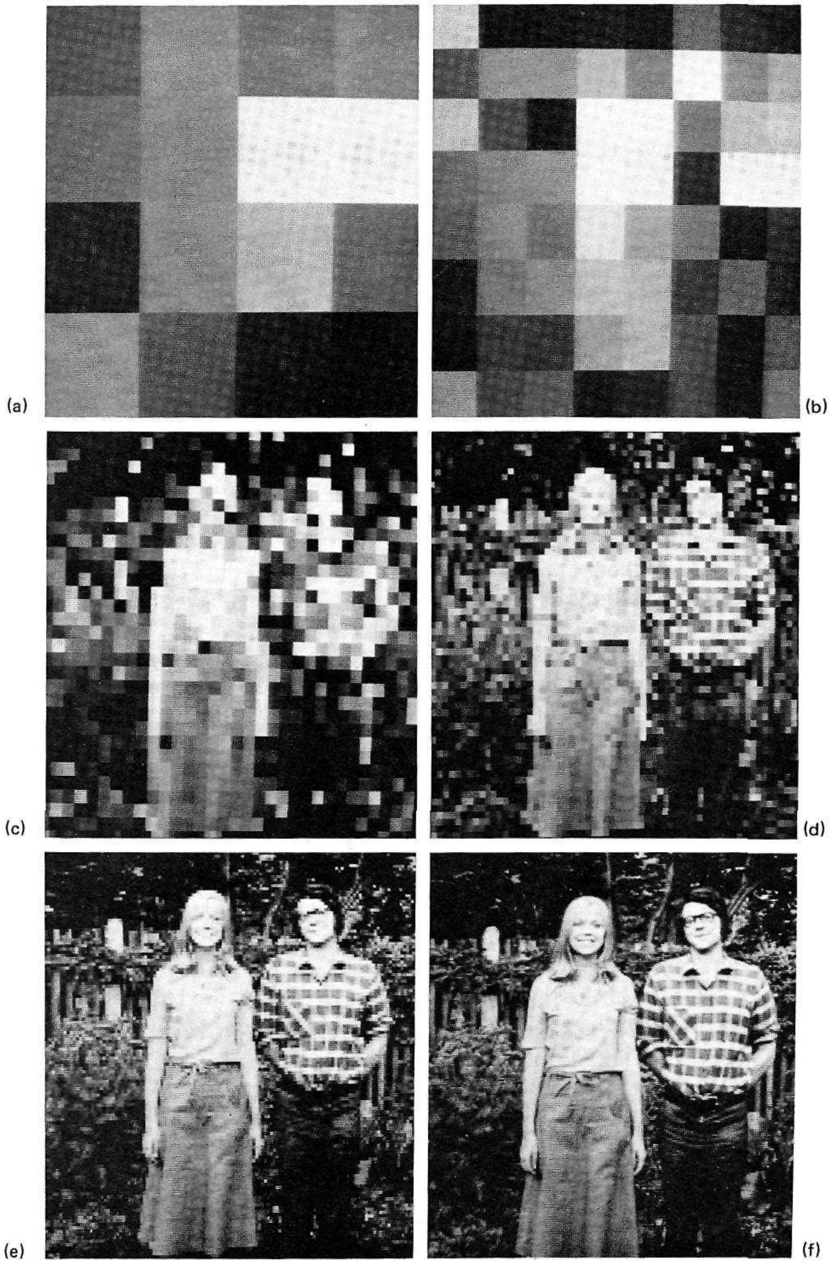


Fig. 2.9 Using different numbers of samples. (a) $N = 16$; (b) $N = 32$; (c) $N = 64$; (d) $N = 128$; (e) $N = 256$; (f) $N = 512$.

Table 2.3

NUMBER OF 8-BIT BYTES OF STORAGE FOR VARIOUS VALUES OF N AND M

N	32	64	128	256	512
m					
1	128	512	2,048	8,192	32,768
2	256	1,024	4,096	16,384	65,536
3	512	2,048	8,192	32,768	131,072
4	512	2,048	8,192	32,768	131,072
5	1,024	4,096	16,384	65,536	262,144
6	1,024	4,096	16,384	65,536	262,144
7	1,024	4,096	16,384	65,536	262,144
8	1,024	4,096	16,384	65,536	262,144

computer vision to have to balance the desire for increased resolution (both gray scale and spatial) against its cost. Better data can often make algorithms easier to write, but a small amount of data can make processing more efficient. Of course, the image domain, choice of algorithms, and image characteristics all heavily influence the choice of resolutions.

Tesselations and Distance Metrics

Although the spatial samples for $f(\mathbf{x})$ can be represented as points, it is more satisfying to the intuition and a closer approximation to the acquisition process to think of these samples as finite-sized cells of constant gray-level partitioning the image. These cells are termed *pixels*, an acronym for *picture elements*. The pattern into which the plane is divided is called its *tesselation*. The most common regular tesselations of the plane are shown in Fig. 2.11.

Although rectangular tesselations are almost universally used in computer vision, they have a structural problem known as the "connectivity paradox." Given a pixel in a rectangular tesselation, how should we define the pixels to which it is connected? Two common ways are *four-connectivity* and *eight-connectivity*, shown in Fig. 2.12.

However, each of these schemes has complications. Consider Fig. 2.12c, consisting of a black object with a hole on a white background. If we use four-connectedness, the figure consists of four disconnected pieces, yet the hole is separated from the "outside" background. Alternatively, if we use eight-connectedness, the figure is one connected piece, yet the hole is now connected to the outside. This paradox poses complications for many geometric algorithms. Triangular and hexagonal tesselations do not suffer from connectivity difficulties (if we use three-connectedness for triangles); however, *distance* can be more difficult to compute on these arrays than for rectangular arrays.

The distance between two pixels in an image is an important measure that is fundamental to many algorithms. In general, a distance *d* is a *metric*. That is,

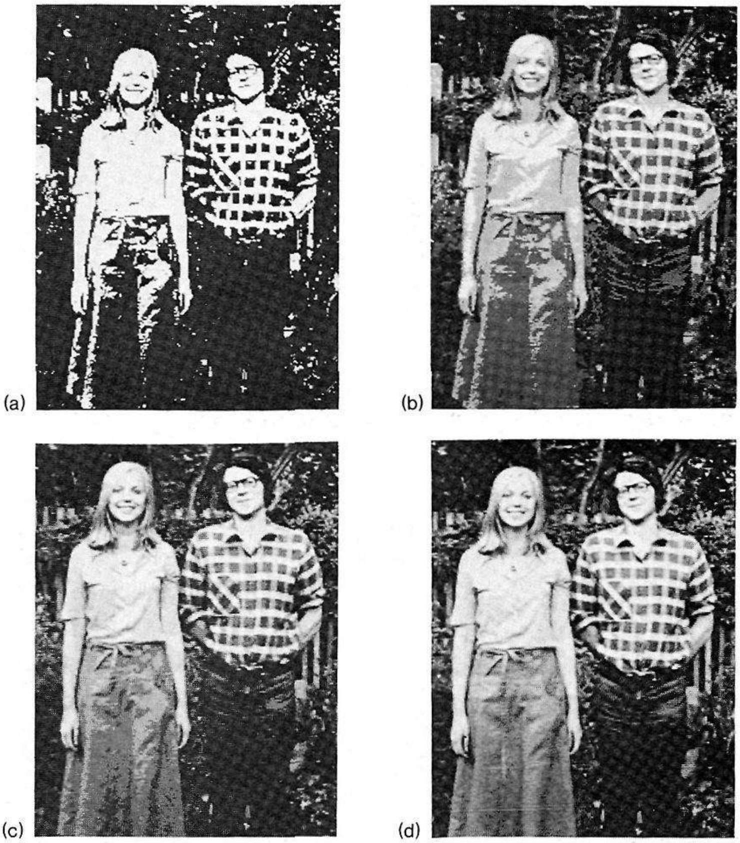


Fig. 2.10 Using different numbers of bits per sample. (a) $m = 1$; (b) $m = 2$; (c) $m = 4$; (d) $m = 8$.

(1) $d(\mathbf{x}, \mathbf{y}) = 0$ iff $\mathbf{x} = \mathbf{y}$

(2) $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$

(3) $d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}) \geqslant d(\mathbf{x}, \mathbf{z})$

For square arrays with unit spacing between pixels, we can use any of the following common distance metrics (Fig. 2.13) for two pixels $\boldsymbol{x} = (x_1, y_1)$ and $\boldsymbol{y} = (x_2, y_2)$.

Euclidean:

$$d_e(\mathbf{x}, \mathbf{y}) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{2.37}$$

City block:

$$d_{cb}(\mathbf{x}, \mathbf{y}) = |x_1 - x_2| + |y_1 - y_2| \tag{2.38}$$

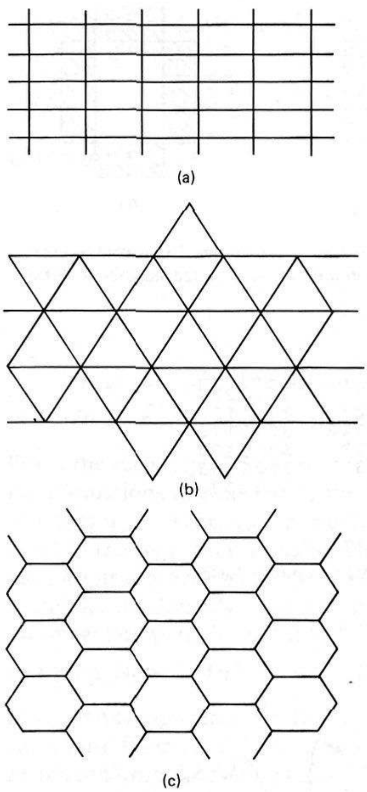


Fig. 2.11 Different tesselations of the image plane. (a) Rectangular; (b) triangular; (c) hexagonal.

Chessboard:

$$d_{ch}(\mathbf{x}, \mathbf{y}) = \max\left\{|x_1-x_2|, |y_1-y_2|\right\} \tag{2.39}$$

Other definitions are possible, and all such measures extend to multiple dimensions. The tesselation of higher-dimensional space into pixels usually is confined to (n-dimensional) cubical pixels.

The Sampling Theorem

Consider the one-dimensional "image" shown in Fig. 2.14. To digitize this image one must sample the image function. These samples will usually be separated at regular intervals as shown. How far apart should these samples be to allow reconstruction (to a given accuracy) of the underlying continuous image from its samples? This question is answered by the Shannon sampling theorem. An excellent rigorous presentation of the sampling theorem may be found in [Rosenfeld and Kak 1976]. Here we shall present a shorter graphical interpretation using the results of Table 2.2. For simplicity we consider the image to be periodic in order to avoid small edge effects introduced by the finite image domain. A more rigorous

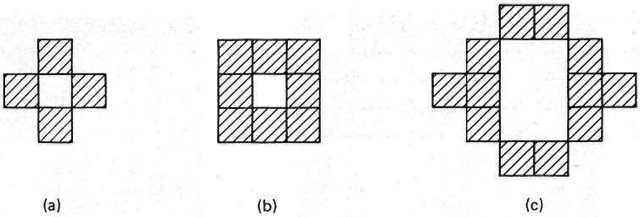


Fig. 2.12 Connectivity paradox for rectangular tesselations. (a) A central pixel and its 4-connected neighbors; (b) a pixel and its 8-connected neighbors; (c) a figure with ambiguous connectivity.

```
    2 3 2             3          3 3 3 3 3 3 3
  3 2 2 2 3         3 2 3        3 2 2 2 2 2 3
2 2 1 1 1 2 2     3 2 1 2 3      3 2 1 1 1 2 3
3 2 1 0 1 2 3   3 2 1 0 1 2 3    3 2 1 0 1 2 3
2 2 1 1 1 2 2     3 2 1 2 3      3 2 1 1 1 2 3
  3 2 2 2 3         3 2 3        3 2 2 2 2 2 3
    2 3 2             3          3 3 3 3 3 3 3

     (a)             (b)              (c)
```

Fig. 2.13 Equidistant contours for different metrics.

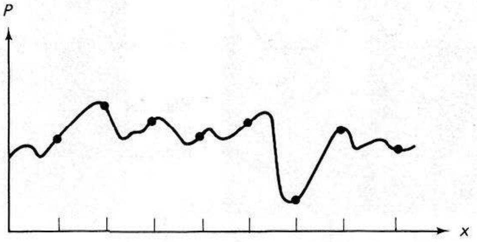


Fig. 2.14 One-dimensional image and its samples.

treatment, which considers these effects, is given in [Andrews and Hunt 1977].

Suppose that the image is sampled with a "comb" function of spacing x_0 (see Table 2.2). Then the sampled image can be modeled by

$$f_s(x) = f(x)\sum_n \delta(x - nx_0) \tag{2.40}$$

where the image function modulates the comb function. Equivalently, this can be written as

$$f_s(x) = \sum_n f(nx_0)\,\delta(x - nx_0) \tag{2.41}$$

The right-hand side of Eq. (2.40) is the product of two functions, so that property

(6) in Table 2.1 is appropriate. The Fourier transform of $f_s(x)$ is equal to the convolution of the transforms of each of the two functions. Using this result yields

$$F_s(u) = F(u) * \frac{1}{x_0} \sum_n \delta(u - \frac{n}{x_0}) \tag{2.42}$$

But from Eq. (2.3),

$$F(u) * \delta(u - \frac{n}{x_0}) = F(u - \frac{n}{x_0}) \tag{2.43}$$

so that

$$F_s(u) = \frac{1}{x_0} \sum_n F(u - \frac{n}{x_0}) \tag{2.44}$$

Therefore, sampling the image function $f(x)$ at intervals of x_0 is equivalent in the frequency domain to replicating the transform of f at intervals of $\frac{1}{x_0}$. This limits the recovery of $f(x)$ from its sampled representation, $f_s(x)$. There are two basic situations to consider. If the transform of $f(x)$ is *bandlimited* such that $F(u) = 0$ for $|u| > 1/(2x_0)$, then there is no overlap between successive replications of $F(u)$ in the frequency domain. This is shown for the case of Fig. 2.15a, where we have arbitrarily used a triangular-shaped image transform to illustrate the effects of sampling. Incidentally, note that for this transform $F(u) = F(-u)$ and that it has no imaginary part; from Table 2.2, the one-dimensional image must also be real and even. Now if $F(u)$ is not bandlimited, i.e., there are $u > \frac{1}{2x_0}$ for which $F(u) \neq 0$, then components of different replications of $F(u)$ will interact to produce the composite function $F_s(u)$, as shown in Fig. 2.15b. In the first case $f(x)$ can be recovered from $F_s(u)$ by multiplying $F_s(u)$ by a suitable $G(u)$:

$$G(u) = \begin{cases} 1 & |u| < \frac{1}{2x_0} \\ 0 & \text{otherwise} \end{cases} \tag{2.45}$$

Then

$$f(x) = \mathfrak{F}^{-1}[F_s(u)G(u)] \tag{2.46}$$

However, in the second case, $F_s(u)G(u)$ is very different from the original $F(u)$. This is shown in Fig. 2.15c. Sampling a $F(u)$ that is not bandlimited allows information at high spatial frequencies to interfere with that at low frequencies, a phenomenon known as *aliasing.*

Thus the sampling theorem has this very important result: As long as the image contains no spatial frequencies greater than one-half the sampling frequency, the underlying continuous image is unambiguously represented by its samples. However, lest one be tempted to insist on images that have been so sampled, note that it may be useful to sample at lower frequencies than would be required for total reconstruction. Such sampling is usually preceded by some form of blurring of

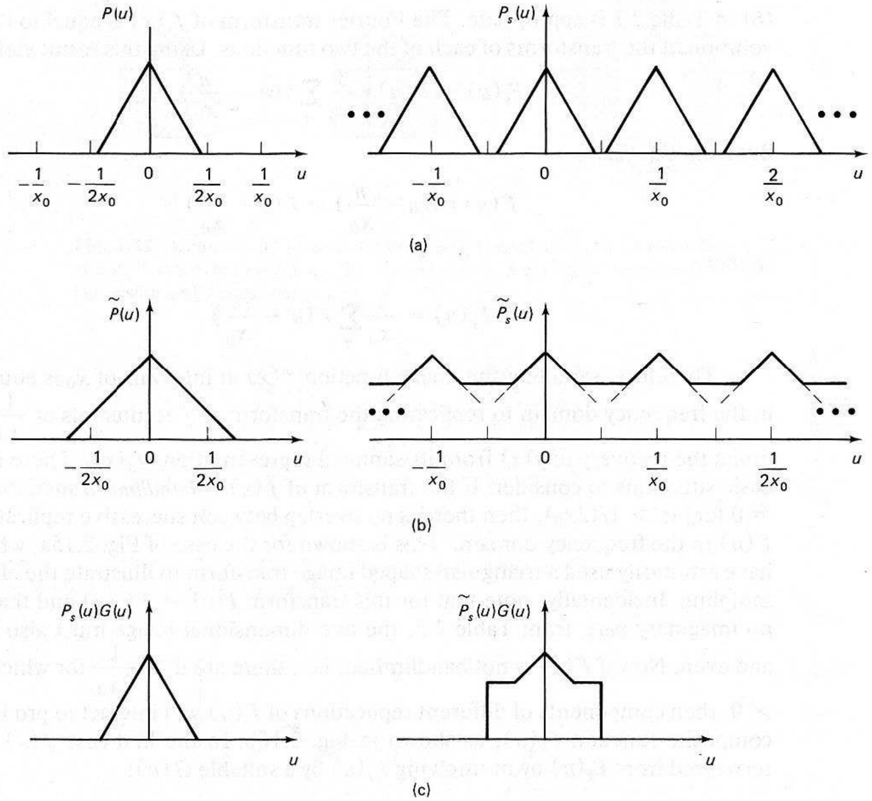


Fig. 2.15 (a) $F(u)$ bandlimited so that $F(u) = 0$ for $|u| > 1/2x_0$. (b) $F(u)$ not bandlimited as in (a). (c) reconstructed transform.

the image, or can be incorporated with such blurring (by integrating the image intensity over a finite area for each sample). Image blurring can bury irrelevant details, reduce certain forms of noise, and also reduce the effects of aliasing.

2.3 IMAGING DEVICES FOR COMPUTER VISION

There is a vast array of methods for obtaining a digital image in a computer. In this section we have in mind only "traditional" images produced by various forms of radiation impinging on a sensor after having been affected by physical objects.

Many sensors are best modeled as an *analog* device whose response must be *digitized* for computer representation. The types of imaging devices possible are limited only by the technical ingenuity of their developers; attempting a definitive

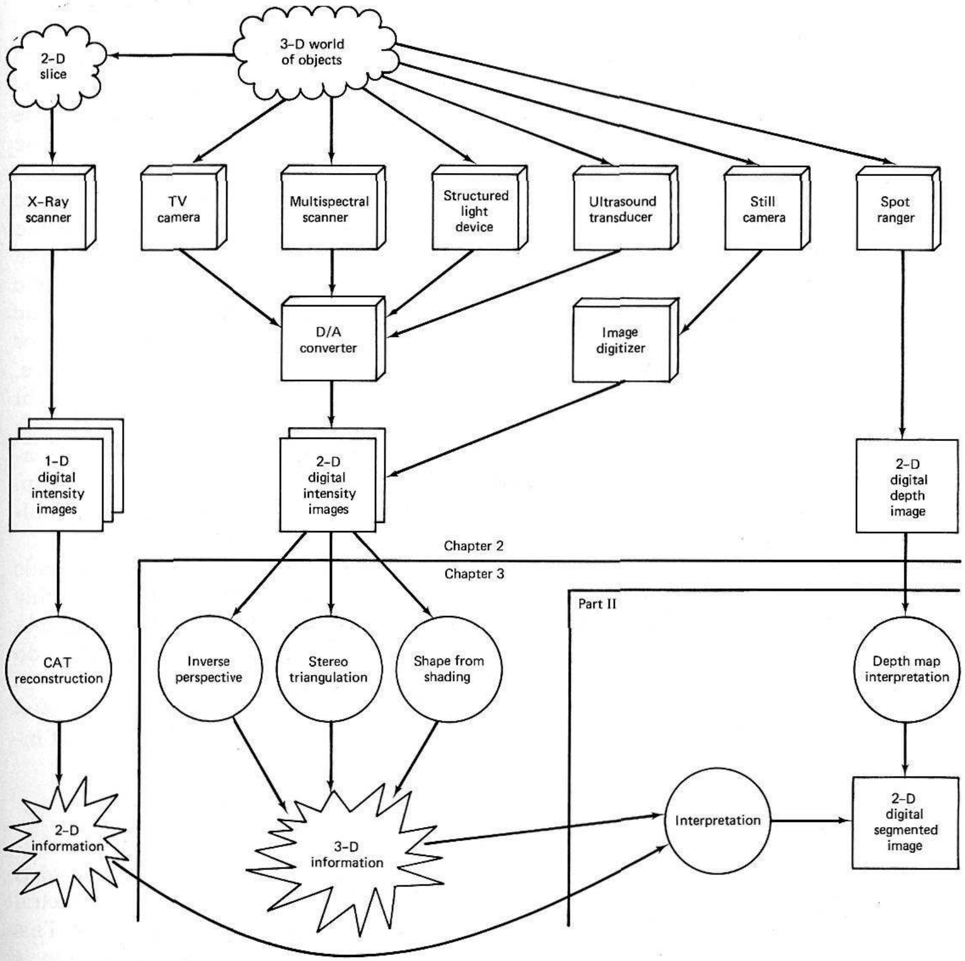


Fig. 2.16 Imaging devices (boxes), information structures (rectangles), and processes (circles).

taxonomy is probably unwise. Figure 2.16 is a flowchart of devices, information structures, and processes addressed in this and succeeding sections.

When the image already exists in some form, or physical considerations limit choice of imaging technology, the choice of digitizing technology may still be open. Most images are carried on a permanent medium, such as film, or at least are available in (essentially) analog form to a digitizing device. Generally, the relevant technical characteristics of imaging or digitizing devices should be foremost in mind when a technique is being selected. Such considerations as the signal-to-noise ratio of the device, its resolution, the speed at which it works, and its expense are important issues.

2.3.1 Photographic Imaging

The camera is the most familiar producer of optical images on a permanent medium. We shall not address here the multitudes of still- and movie-camera options; rather, we briefly treat the characteristics of the photographic film and of the digitizing devices that convert the image to machine-readable form. More on these topics is well presented in the References.

Photographic (black-and-white) film consists of an emulsion of silver halide crystals on a film base. (Several other layers are identifiable, but are not essential to an understanding of the relevant properties of film.) Upon exposure to light, the silver halide crystals form *development centers,* which are small grains of metallic silver. The photographic development process extends the formation of metallic silver to the entire silver halide crystal, which thus becomes a binary ("light" or "no light") detector. Subsequent processing removes undeveloped silver halide. The resulting film *negative* is dark where many crystals were developed and light where few were. The resolution of the film is determined by the *grain* size, which depends on the original halide crystals and on development techniques. Generally, the *faster* the film (the less light needed to expose it), the coarser the grain. Film exists that is sensitive to infrared radiation; x-ray film typically has two emulsion layers, giving it more gray-level range than that of normal film.

A repetition of the negative-forming process is used to obtain a photographic *print.* The negative is projected onto photographic paper, which responds roughly in the same way as the negative. Most photographic print paper cannot capture in one print the range of densities that can be present in a negative. Positive films do exist that do not require printing; the most common example is color slide film.

The response of film to light is not completely linear. The photographic *density* obtained by a negative is defined as the logarithm (base 10) of the ratio of incident light to transmitted light.

$$D = \log_{10}\left(\frac{I}{I_t}\right)$$

The *exposure* of a negative dictates (approximately) its response. Exposure is defined as the energy per unit area that exposed the film (in its sensitive spectral range). Thus exposure is the product of the *intensity* and the time of exposure. This mathematical model of the behavior of the photographic exposure process is correct for a wide operating range of the film, but *reciprocity failure* effects in the film keep one from being able always to trade light level for exposure time. At very low light levels, longer exposure times are needed than are predicted by the product rule.

The response of film to light is usually plotted in an "H&D curve" (named for Hurter and Driffield), which plots density versus exposure. The H&D curve of film displays many of its important characteristics. Figure 2.17 exhibits a typical H&D curve for a black and white film.

The *toe* of the curve is the lower region of low slope. It expresses reciprocity failure and the fact that the film has a certain bias, or *fog* response, which dominates its behavior at the lowest exposure levels. As one would expect, there is an upper limit to the density of the film, attained when a maximum number of silver

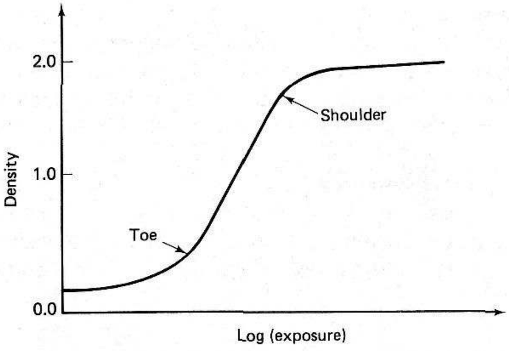


Fig. 2.17 Typical H & D curve.

halide crystals are rendered developable. Increasing exposure beyond this maximum level has little effect, accounting for the *shoulder* in the H&D curve, or its flattened upper end.

In between the toe and shoulder, there is typically a linear operating region of the curve. High-contrast films are those with high slope (traditionally called *gamma*); they respond dramatically to small changes in exposure. A high-contrast film may have a gamma between about 1.5 and 10. Films with gammas of approximately 10 are used in graphics arts to copy line drawings. General-purpose films have gammas of about 0.5 to 1.0.

The resolution of a general film is about 40 lines/mm, which means that a 1400 × 1400 image may be digitized from a 35mm slide. At any greater sampling frequency, the individual film grains will occupy more than a pixel, and the resolution will thus be grain-limited.

Image Digitizers (Scanners)

Accuracy and speed are the main considerations in converting an image on film into digital form. Accuracy has two aspects: spatial resolution, loosely the level of image spatial detail to which the digitizer can respond, and gray-level resolution, defined generally as the range of densities or reflectances to which the digitizer responds and how finely it divides the range. Speed is also important because usually many data are involved; images of 1 million samples are commonplace.

Digitizers broadly take two forms: mechanical and "flying spot." In a mechanical digitizer, the film and a sensing assembly are mechanically transported past one another while readings are made. In a flying-spot digitizer, the film and sensor are static. What moves is the "flying spot," which is a point of light on the face of a cathode-ray tube, or a laser beam directed by mirrors. In all digitizers a very narrow beam of light is directed through the film or onto the print at a known coordinate point. The light transmittance or reflectance is measured, transformed from analog to digital form, and made available to the computer through interfacing electronics. The location on the medium where density is being measured may also be transmitted with each reading, but it is usually determined by relative offset from positions transmitted less frequently. For example, a "new scan line" impulse is transmitted for TV output; the position along the current scan line yields an *x* position, and the number of scan lines yields a *y* position.

The mechanical scanners are mostly of two types, *flat-bed* and *drum*. In a flat-bed digitizer, the film is laid flat on a surface over which the light source and the sensor (usually a very accurate photoelectric cell) are transported in a raster fashion. In a drum digitizer, the film is fastened to a circular drum which revolves as the sensor and light source are transported down the drum parallel to its axis of rotation.

Color mechanical digitizers also exist; they work by using colored filters, effectively extracting in three scans three "color overlays" which when superimposed would yield the original color image. Extracting some "composite" color signal with one reading presents technical problems and would be difficult to do as accurately.

Satellite Imagery

LANDSAT and ERTS (Earth Resources Technology Satellites) have similar scanners which produce images of 2340 x 3380 7-bit pixels in four spectral bands, covering an area of 100 × 100 nautical miles. The scanner is mechanical, scanning six horizontal scan lines at a time; the rotation of the earth accounts for the advancement of the scan in the vertical direction.

A set of four images is shown in Fig. 2.18. The four spectral bands are numbered 4, 5, 6, and 7. Band 4 [0.5 to 0.6 μm (green)] accentuates sediment-laden water and shallow water, band 5 [0.6 to 0.7 μm (red)] emphasizes cultural features such as roads and cities, band 6 [0.7 to 0.8 μm (near infrared)] emphasizes vegetation and accentuates the contrast between land and water, band 7 [0.8 to 1.1 μm (near infrared)] is like band 6 except that it is better at penetrating atmospheric haze.

The LANDSAT images are available at nominal cost from the U.S. government (The EROS Data Center, Sioux Falls, South Dakota 57198). They are furnished on tape, and cover the entire surface of the earth (often the buyer has a choice of the amount of cloud cover). These images form a huge data base of multispectral imagery, useful for land-use and geological studies; they furnish something of an image analysis challenge, since one satellite can produce some 6 billion bits of image data per day.

Television Imaging

Television cameras are appealing devices for computer vision applications for several reasons. For one thing, the image is immediate; the camera can show events as they happen. For another, the image is already in electrical, if not digital form. "Television camera" is basically a nontechnical term, because many different technologies produce video signals conforming to the standards set by the FCC and NTSC. Cameras exist with a wide variety of technical specifications.

Usually, TV cameras have associated electronics which scan an entire "picture" at a time. This operation is closely related to broadcast and receiver standards, and is more oriented to human viewing than to computer vision. An entire image (of some 525 scan lines in the United States) is called a *frame*, and consists of two *fields*, each made up of alternate scan lines from the frame. These fields are generated and transmitted sequentially by the camera electronics. The transmitted image is thus *interlaced*, with all odd-numbered scan lines being "painted" on the

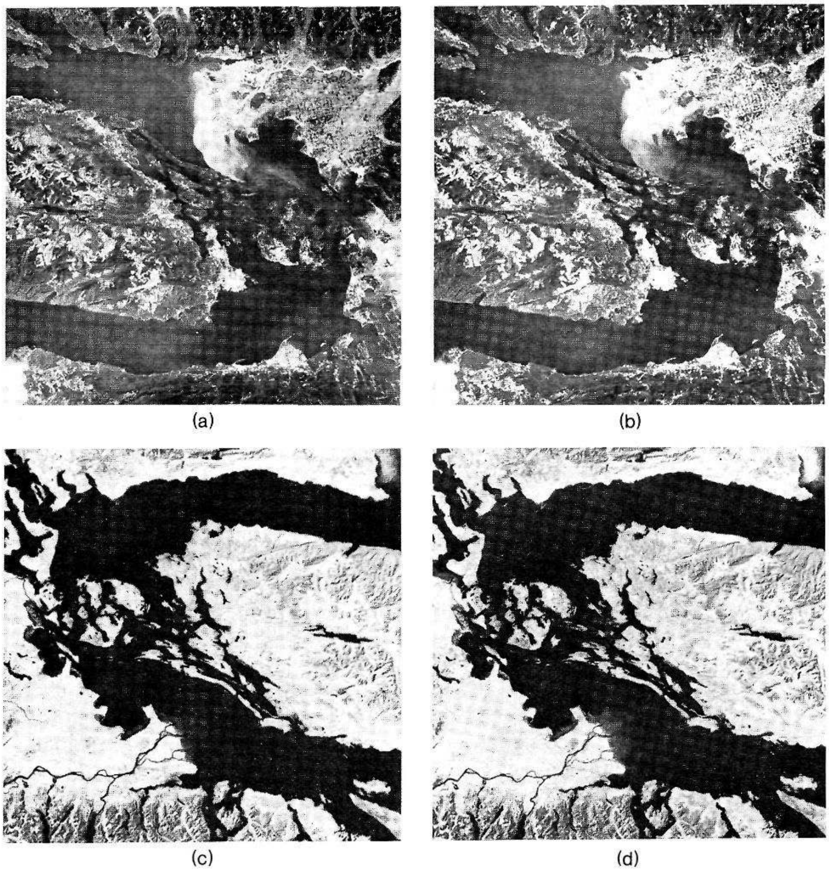


Fig. 2.18 The straits of Juan de Fuca as seen by the LANDSAT multispectral scanner. (a) Band 4; (b) band 5; (c) band 6; (d) band 7.

screen alternating with all even-numbered scan lines. In the United States, each field takes 1/60 sec to scan, so a whole frame is scanned every 1/30 sec. The interlacing is largely to prevent flickering of the image, which would become noticeable if the frame were painted from top to bottom only once in 1/30 sec. These automatic scanning electronics may be replaced or overridden in many cameras, allowing "random access" to the image. In some technologies, such as the image dissector, the longer the signal is collected from any location, the better the signal-to-noise performance.

There are a number of different systems used to generate television images. We discuss five main methods below.

Image orthicon tube. This is one of the two main methods in use today (in addition to the vidicon). It offers very stable performance at all incident light levels and is widely used in commercial television. It is a storage-type tube, since it depends on the neutralization of positive charges by a scanning electron beam.

The image orthicon (Fig. 2.19) is divided into an imaging and readout section. In the imaging section, light from the scene is focused onto a semitransparent photocathode. This photocathode operates the same way as the cathode in a phototube. It emits electrons which are magnetically focused by a coil and are accelerated toward a positively charged target. The target is a thin glass disk with a fine-wire-mesh screen facing the photocathode. When electrons strike it, secondary emission from the glass takes place. As electrons are emitted from the photocathode side of the disk, positive charges build up on the scanning side. These charges correspond to the pattern of light intensity in the scene being viewed.

In the readout section, the back of the target is scanned by a low velocity electron beam from an electron gun at the rear of the tube. Electrons in this beam are absorbed by the target in varying amounts, depending on the charge on the target. The image is represented by the amplitude-modulated intensity of the returned beam.

Vidicon tube. The vidicon is smaller, lighter, and more rugged than the image orthicon, making it ideal for portable use. Here the target (the inner surface of the face plate) is coated with a transparent conducting film which forms a video signal electrode (Fig. 2.20). A thin photosensitive layer is deposited on the film, consisting of a large number of tiny resistive globules whose resistance decreases on illumination. This layer is scanned in raster fashion by a low velocity electron beam from the electron gun at the rear of the tube. The beam deposits electrons on the layer, thus reducing its surface potential. The two surfaces of the target essentially form a capacitor, and the scanning action of the beam produces a capacitive current at the video signal electrode which represents the video signal.

The plumbicon is essentially a vidicon with a lead oxide photosensitive layer. It offers the following advantages over the vidicon: higher sensitivity, lower dark current, and negligible persistence or lag.

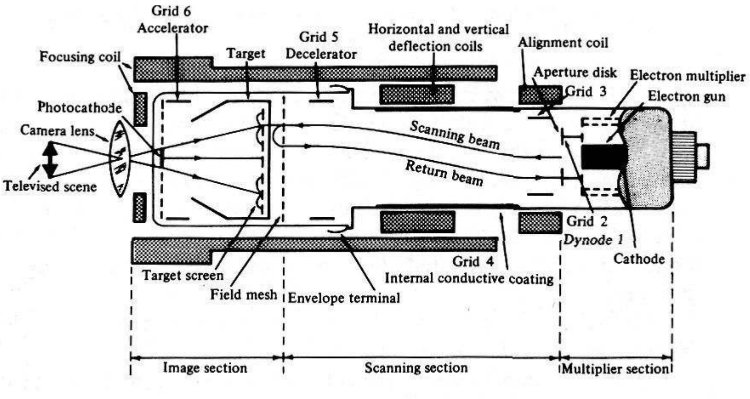


Fig. 2.19 The image orthicon.

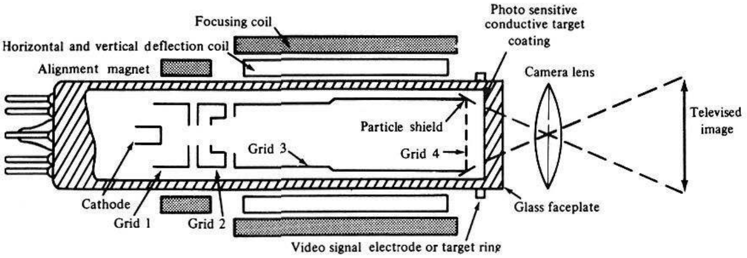

Focusing coil
Horizontal and vertical deflection coil
Alignment magnet
Photo sensitive conductive target coating
Camera lens
Televised image
Particle shield
Grid 3
Grid 4
Cathode
Grid 1
Grid 2
Glass faceplate
Video signal electrode or target ring

Fig. 2.20 The vidicon.

Iconoscope tube. The iconoscope is now largely of historical interest. In it, an electron beam scans a target consisting of a thin mica sheet or mosaic coated with a photosensitive layer. In contrast to the vidicon and orthicon, the electron beam and the light both strike the same side of the target surface. The back of the mosaic is covered with a conductive film connected to an output load. The arrangement is equivalent to a matrix of small capacitors which discharge through a common lead.

Image dissector tube. The image dissector tube operates on instantaneous scanning rather than by neutralizing positive charges. Light from the scene is focused on a cathode coated with a photosensitive layer (Fig. 2.21). The cathode emits electrons in proportion to the amount of light striking it. These electrons are accelerated toward a target by the anode. The target is an electron multiplier covered by a small aperture which allows only a small part of the "electron image" emitted by the cathode to reach the target. The electron image is focused by a focusing coil that produces an axial magnetic field. The deflection coils then scan the electron image past the target aperture, where the electron multiplier produces a varying voltage representing the video signal. The image is thus "dissected" as it is scanned past the target, in an electronic version of a flat-bed digitizing process.

Charge transfer devices. A more recent development in image formation is that of solid-state image sensors, known as charge transfer devices (CTDs). There are two main classes of CTDs: charge-coupled devices (CCDs) and charge-injection devices (CIDs).

CCDs resemble MOSFETs (metal-oxide semiconductor field-effect transistor) in that they contain a "source" region and a "drain" region coupled by a depletion-region channel (Fig. 2.22). For imaging purposes, they can be considered as a monolithic array of closely spaced MOS capacitors forming a shift register (Fig. 2.23). Charges in the depletion region are transferred to the output by applying a series of clocking pulses to a row of electrodes between the source and the drain.

Photons incident on the semiconductor generate a series of charges on the CCD array. They are transferred to an output register either directly one line at a time (line transfer) or via a temporary storage area (frame transfer). The storage

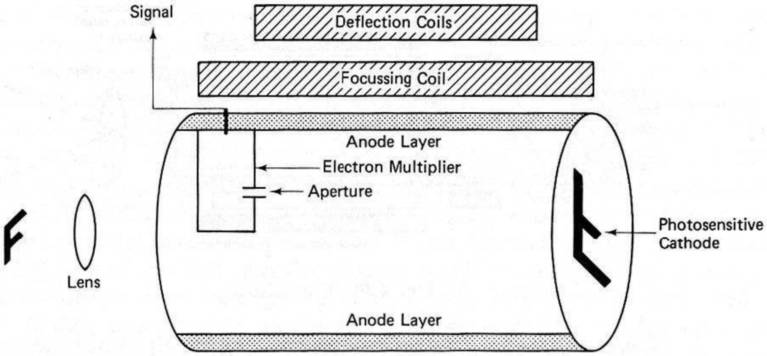


Fig. 2.21 Image dissector.

area is needed in frame transfer because the CCD array is scanned more rapidly than the output can be directly accommodated.

Charge injection devices (CIDs) resemble CCDs except that during sensing the charge is confined to the image site where it was generated (Fig. 2.24). The charges are read using an *X-Y* addressing technique similar to that used in computer memories. Basically, the stored charge is "injected" into the substrate and the resulting displacement current is detected to create the video signal.

CTD technology offers a number of advantages over conventional-tube-type cameras: light weight, small size, low power consumption, resistance to burn-in, low blooming, low dark current, high sensitivity, wide spectral and dynamic range, and lack of persistence. CIDs have the further advantages over CCDs of tolerance to processing defects, simple mechanization, avoidance of charge transfer losses, and minimized blooming. CTD cameras are now available commercially.

Analog-to-Digital Conversion

With current technology, the representation of an image as an analog electrical waveform is usually an unavoidable precursor to further processing. Thus the operation of deriving a digital representation of an analog voltage is basic to computer vision input devices.

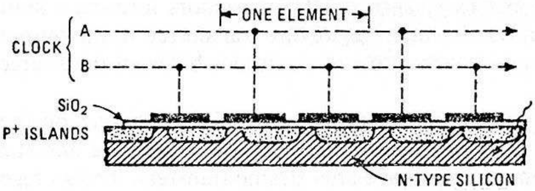


Fig. 2.22 Charge coupled device.

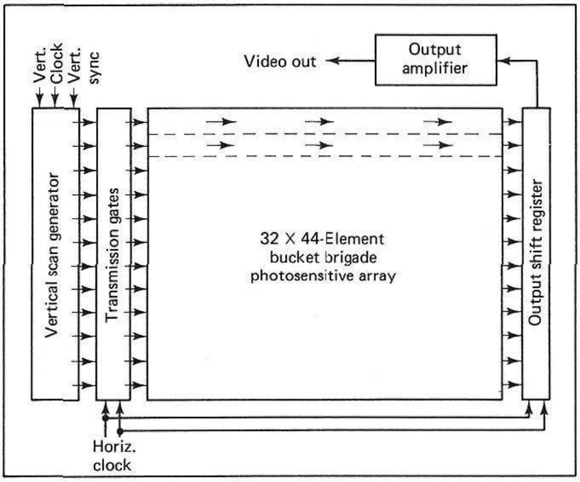


Fig. 2.23 A CCD array (line transfer).

The function of an analog-to-digital (A/D) converter is to take as input a voltage such as a video signal and to produce as output a representation of the voltage in digital memory, suitable for reading by an interface to a digital computer. The quality of an A/D converter is measured by its temporal resolution (the speed at which it can perform conversions) and the accuracy of its digital output. Analog-

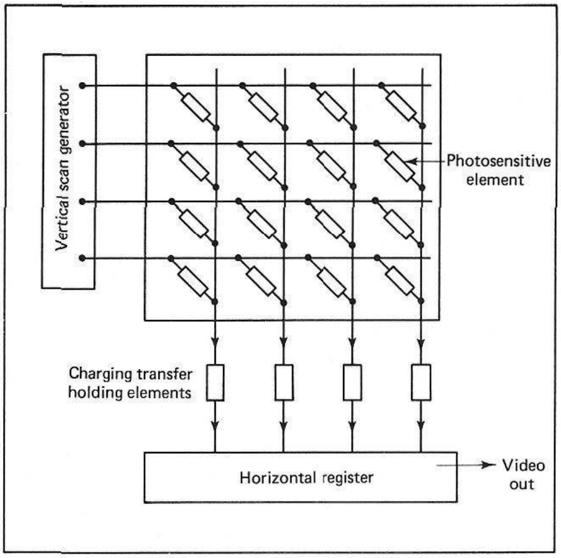


Fig. 2.24 A CID array.

to-digital converters are being produced as integrated circuit chips, but high-quality models are still expensive. The output precision is usually in the 8- to 12-bit range.

It is quite possible to digitize an entire frame of a TV camera (i.e., approximately 525 scan lines by 300 or so samples along a scan line) in a single frame time (1/30 sec in the United States). Several commercial systems can provide such fast digitization into a "frame buffer" memory, along with raster graphics display capabilities from the same frame buffer, and "video rate processing" of the digital data. The latter term refers to any of various low-level operations (such as averaging, convolution with small templates, image subtraction) which may be performed as fast as the images are acquired.

One inexpensive alternative to digitizing entire TV frames at once is to use an interface that acquires the TV signal for a particular point when the scan passes the requested location. With efficient programming, this point-by-point digitization can acquire an entire frame in a few seconds.

2.3.2 Sensing Range

The third dimension may be derived from binocular images by triangulation, as we saw earlier, or inferred from single monocular visual input by a variety of "depth cues," such as size and occlusion. Specialized technology exists to acquire "depth images" directly and reliably. Here we outline two such techniques: "light striping," which is based on triangulation, and "spot ranging," which is based on different principles.

Light Striping

Light striping is a particularly simple case of the use of *structured light* [Will and Pennington 1971]. The basic idea is to use geometric information in the illumination to help extract geometric information from the scene. The spatial frequencies and angles of bars of light falling on a scene may be clustered to find faces; randomly structured light may allow blank, featureless surfaces to be matched in stereo views; and so forth.

Many researchers [Popplestone et al. 1975; Agin 1972; Sugihara 1977] have used striping to derive three dimensions. In light striping, a single plane of light is projected onto a scene, which causes a stripe of light to appear on the scene (Fig. 2.25). Only the part of the scene illuminated by the plane is sensed by the vision system. This restricts the "image" to be an essentially one-dimensional entity, and simplifies matching corresponding points. The plane itself has a known position (equation in world coordinates), determinable by any number of methods involving either the measurement of the projecting device or the measurement of the final resulting plane of light. Every image point determines a single "line of sight" in three-space upon which the world point that produces the image point must lie. This line is determined by the focal point of the imaging system and the image point upon which the world point projects. In a light-striping system, any point that is sensed in the image is also guaranteed to lie on the light plane in three-space. But the light plane and the line of sight intersect in just one point (as long as

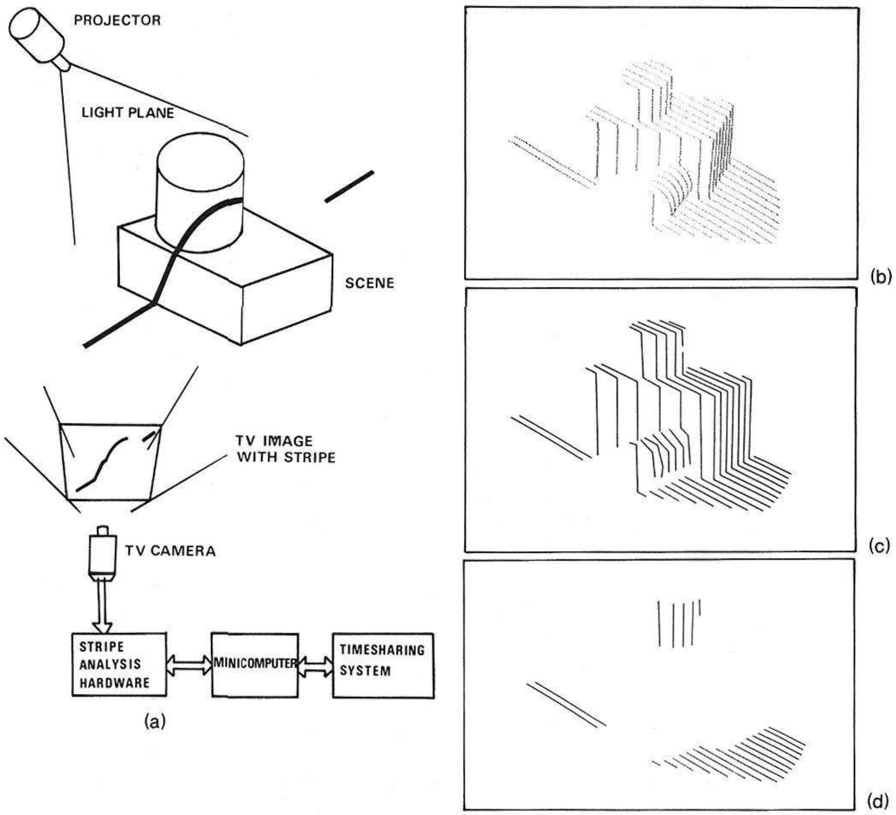


Fig. 2.25 Light striping. (a) A typical arrangement; (b) raw data; (c) data segmented into strips; (d) strips segmented into two surfaces.

the camera's focal point is not in the light plane). Thus by computation of the intersection of the line of sight with the plane of light, we derive the three-dimensional point that corresponds to any image point visible as part of a stripe.

The plane of light may result from a laser or from the projection of a slit. Only the light stripe should be visible to the imaging device; unless a laser is used, this implies a darkened room. If a camera is fitted with the proper filter, a laser-based system can be operated in normal light. Another advantage of the laser is that it can be focused into a narrower plane than can a slit image.

The only points whose three-dimensional coordinates can be computed are those that can be "seen" by both the light-stripe source and the camera at once. Since there must be a nonzero baseline if triangulation is to derive three-dimensional information, the camera cannot be too close to the projector, and thus concavities in the scene are potential trouble spots, since both the striper and the camera may not be able to "see" into them. Surfaces in the scene that are nearly parallel with the light plane will have a relatively small number of stripes projected onto them by any uniform stripe placement strategy. This problem is ameliorated by striping with two sets of parallel planes at right angles to each other [Agin 1972]. A major advantage of light striping over spot ranging is that (barring shadows) its continuity and discontinuity indicate similar conditions on the surface. It is easy to "segment" stripe images (Part II): Stripes falling on the same surface may easily be gathered together. This set of related stripes may be used in a number of ways to derive further information on the characteristics of the surface (Fig. 2.25b).

Spot Ranging

Civil engineers have used laser-based "spot range finders" for some time. In laboratory-size environments, they are a relatively new development. There are two basic techniques. First, one can emit a very sharp pulse and time its return ("lidar," the light equivalent of radar). This requires a sophisticated laser and electronics, since light moves 1 ft every billionth of a second, approximately. The second technique is to modulate the laser light in amplitude and upon its return compare the phase of the returning light with that of the modulator. The phase differences are related to the distance traveled [Nitzan et al. 1977]. A representative image is shown in Fig. 2.26.

Both these techniques produce results that are accurate to within about 1% of the range. Both of them allow the laser to be placed close to a camera, and thus "intensity maps" (images) and range maps may be produced from single viewpoints. The laser beam can easily poke into holes, and the return beam may be sensed close to the emitted one, so concavities do not present a serious problem. Since the laser beam is attenuated by absorption, it can yield intensity information as well. If the laser produces light of several wavelengths, it is possible to use filters and obtain multispectral reflectance information as well as depth information from the same device [Garvey 1976; Nitzan et al. 1977].

The usual mode of use of a spot ranging device is to produce a range map that corresponds to an intensity map. This has its advantages in that the correspondence may be close. The structural properties of light stripes are lost: It can be hard to "segment" the image into surfaces (to tell which "range pixels" are associated with the same surface). Range maps are amenable to the same sorts of segmentation techniques that are used for intensity images: Hough techniques, region growing, or differentiation-based methods of edge finding (Part II).

Ultrasonic Ranging

Just as light can be pulsed to determine range, so can sound and ultrasound (frequencies much higher than the audible range). Ultrasound has been used extensively in medicine to produce images of human organs (e.g., [Waag and Gramiak 1976]). The time between the transmitted and received signal determines range; the sound signal travels much slower than light, making the problem of timing the returning signal rather easier than it is in pulsed laser devices. However, the signal is severely attenuated as it travels through biological tissue, so that the detection apparatus must be very sensitive.