22 June 1984, Volume 224, Number 4655 SCIENCE

Computer Vision and Natural Constraints

C. M. Brown

Summary. Computer vision, the automatic construction of scene descriptions from image input data, has just entered its second decade. Approaches have varied widely, especially in the amounts of symbolic, domain-dependent knowledge and inference that are incorporated into the vision process. Much current research addresses the extraction of physical properties of the scene (depth, surface orientation, reflectance) from images by using only a few general assumptions about the scene domain. Extraction of physical parameters is part of a hierarchy of operations needed to transform image input data to symbolic descriptions. Two other processes that serve as examples are stereo fusion and the partitioning of image phenomena into related groups. Computer vision research is influencing theories of animal perception as well as the design of computing architectures for artificial intelligence.

Computer vision is a general term that embraces most aspects of the analysis of "visual" input by computer; it includes reliable industrial systems, academic research systems, and theoretical studies. One major goal of much computer vision research is to shed light on animal vision systems through computer models. Computers have provided fresh metaphors and models that are of increasing influence in the cognitive sciences (such as psychology, neuroscience, philosophy, and linguistics). In addition, computers lend themselves to a certain disciplined approach and to the use of computational (or information-processing) models to complement descriptive models. An important aspect of a computational model is the precise specification of the form and content of inputs and ancillary information available on the process under investigation. Likewise, the form and content of outputs must be precisely specified. Finally, the logical, mathematical, symbolic, or other transformations that are to produce the output from the input must be explicitly specified, as must the intermediate data representations in the process. If the model is indeed precise enough to be programmed, the resulting program is a finite, formal, experimental artifact whose performance may be thoroughly and quantitatively evaluated. This article is meant to be a brief tutorial incorporating three case studies in computer vision. It discusses some major directions of current computer vision research and indicates how such research is related to psychology and neuroscience in the investigation of general vision systems.

The technical scope of this article is narrow; attention is focused on a few results and many more are not mentioned. The three case studies (stereopsis, understanding image shading, and a general grouping process) were chosen from many that could have served equally well. The references are representative, not complete, and are mostly to accessible books and survey articles from which original sources may be traced. They include a text that emphasizes cognitive techniques in computer vision (*1*), a book that describes the state of computer vision systems (*2*), two position papers (*3*, *4*) and a collection of recent contributions (*5*) which give a good overview of the current state of computer vision, and Marr's book (*6*) which is an ambitious and exciting attempt to develop an intellectual basis for vision research and to integrate ideas and results from the neurosciences and computer vision into a coherent theory of animal vision.

Computer vision began in the 1950's with statistical pattern recognition (*7*), whose goal is to assign an input image into one of a small number of classes (optical character recognition is a representative application). Digital image processing technology for the enhancement, restoration, coding, and transmission of images began to appear at about the same time, and is now a large and sophisticated field that incorporates many recent computer vision techniques (*8*). True computer vision, with the goal of "understanding" images of complex three-dimensional scenes, was first attempted in the early 1960's (*9*). The immense computational complexity of vision began to become apparent; intuitively appealing detectors for visual features (such as object boundaries) and schemes to control processing proved unreliable and inadequate. Devoting massive amounts of processing at the early stages of vision was economically impossible, so in the 1970's a cognitive approach to computer vision arose that conveniently minimized image-level computation and emphasized the symbolic manipulations to which computers are well adapted. In such "knowledge-directed" vision, computational effort is directed by processes that use facts about such phenomena as gravity, support, occlusion, or the likely spatial relations between objects in the scene. Research turned toward representing and manipulating facts about particular domains (such as polyhedral blocks or office scenes) and exploiting the domain-specific knowledge in vision. The representation and application of knowledge is itself, however, a very difficult branch of artificial intelligence, and the available techniques proved inadequate to bridge

The author is an associate professor in the Computer Science Department, University of Rochester, Rochester, New York 14627.

the gap between the input image and the desired symbolic descriptions of it. In the 1980's the consensus of the computer vision community is that the gap is bridged by a varied and redundant set of visual data representations arranged in a hierarchy of increasing abstraction. Production of many intermediate representations requires a huge amount of computation, but animal vision systems indeed seem to do it, albeit with neural structures that operate differently from today's digital computers.

Much of current research centers around the production of physical property images, which are intermediate representations formed before object recognition is attempted. These image-like representations are registered with the input image and contain values of physical parameters of scene points such as the distance from a sensor to the point, the albedo of surfaces, the direction of motion of objects, the location of shadows and light sources, and so forth. It is usual to assume that the processes that produce physical property images are part of "early vision." That is, they do not require domain-dependent facts, much less conscious reasoning, but are robust general processes whose outputs are reliably correct in a broad range of natural circumstances. In fact, these processes cannot be completely general and reliable, since so much information is projected away in the two-dimensional input image. The fact that they so often work correctly in animal vision seems to imply that they rely on natural constraints or assumptions about the world to derive unambiguous output. A goal of modern computer vision research is the identification and use of such constraints. This, in turn, calls for seeking out properties of the physical world that could help a visual process do useful work, making mathematical models of their interaction with visual phenomena, and implementing the mathematics in computer programs. Currently, attention is centered on the design of processes that can operate in parallel computational architectures, since only through the cooperative, simultaneous activity of many processes is the speed and reliability of animal vision explicable. With this background, let us consider our first case study, stereopsis.

Stereopsis and Natural Constraints

Stereopsis is an example of an important visual ability, producing a physical property image of relative distance or depth. It illustrates several points about modern computer vision and its relation to neuroanatomical and psychophysical studies. Stereopsis is a typical vision process in that its neural implementation is not known, nor even is the true form of its input (it is not known how the visual system processes incoming light in stages before stereopsis). In stereo vision, relative depth is computed by triangulation, given the disparity of points from two images that are known to correspond to the same point in a scene. As a physical property of the real world unaffected by lighting changes, depth is more suited than image intensity data for higher-level visual tasks such as object recognition or description. Depth information is thus a step along the way from highly variable input data to the perception and recognition of stable objects. The most interesting and difficult operation in stereopsis is the matching between the two images that identifies corresponding points and hence yields disparity.

Several fairly effective computer algorithms have been developed to calculate disparities based on a correlation operation between image intensities. These algorithms do not purport to have any relation to human stereopsis. As understanding of human stereopsis grew, so did the desire to construct a computational model of it. Constructing and testing such an algorithm and explicitly confronting its technical issues is salutary, because the algorithm is a description of the process at a level between those provided by psychology and neuroscience. Such algorithms should both explain behavior and suggest useful forms of input, thus furnishing a framework for understanding both psychological and neurophysiological data.

A well-founded computational model makes explicit the input and output, the computational processes, and the underlying natural constraints upon which the computation rests. One influential stereopsis algorithm was proposed in (*10*). Although it has mainly been tested on random-dot stereograms, it is based on several natural constraints that guide the algorithm. For instance, the assumption that the world is made of smooth solid objects with opaque surfaces dictates that only one disparity will be sensed at each image point, and that disparity usually varies slowly, with neighboring image points likely to have similar disparities. To these physical contraints was added the constraint that the stereopsis algorithm should be implementable with an array of simple independent computing elements, connected only to their close neighbors. Such a model was mentioned in (*11*), although there the analogy was to an array of coupled magnets, not computing elements.

The cooperative algorithm for disparity calculation uses a three-dimensional array of simple computing units. Two of the dimensions correspond to the image dimensions, and the third to disparity values. Each unit is activated by a possible match between image elements at its disparity (that is, by the presence of identical image features in the two images offset horizontally by the unit's disparity). Each unit is connected to its neighbors in three dimensions. In a manner analogous to neuronal connections, some connections are inhibitory (activity in one unit reduces the activity of the connected neighbor) and other connections are excitatory. Each unit is connected with inhibition to all the others (each for a different disparity) at the same image location. This implements the natural constraint that a scene is usually made up of opaque surfaces and that each point on an opaque surface has a unique disparity. Each unit is connected with excitation to its spatially neighboring units at the same disparity value. This implements the constraint that a scene is mostly made up of smooth surfaces, and thus disparity will not often vary rapidly between neighboring image points. The network of units operates in parallel, achieving a stable state through a process that minimizes the constraint violations by communicating excitation and inhibition through the local connections. The algorithm is effective and, with its cooperation between units, exhibits the hysteresis (perseverance of fusion despite disparity increases) and interpolation (filling in areas) capability of human stereopsis (*11*). This sort of spatially indexed array of processing elements, connected to neighbors, calculating in parallel and collectively minimizing some set of constraint violations, is a constraint relaxation network. The algorithm it computes can be calculated by iterating the parallel computations until the network converges. Parallel iterative schemes and locally connected computing networks are quite popular today in computer vision because they accord with the basically two-dimensional structure of the retina and cortex, and their parallelism and simple units are biologically plausible and computationally fast. As a practical matter, they also may be adaptable to current fabrication technologies for integrated circuits.

The parallel algorithm is not a complete description of human stereopsis. It does not incorporate eye movements or vergence, which seem important for hu-

man stereo vision. Humans can overcome global differences between images that would defeat the algorithm, such as a 15 percent size difference or a defocused image. A second algorithm (*12, 13*) introduces several new ideas. First, the matching is not between simple image intensities but between edge elements developed by earlier processes. Edge information can be more robust than intensities when developed at several spatial resolutions. The algorithm uses four resolutions, or spatial frequency channels, suggested by psychophysical research (*14*). Vergence is set at some arbitrary value, to be modified later by the algorithm. At the current vergence setting, edge elements are matched in the four paired edge arrays. Edges with strictly horizontal offsets match if their angles are roughly equal (within 60°) and their contrasts are of the same polarity. Each match creates a positive, negative, or zero horizontal disparity. Edges that are not matched are marked, and if more than 30 percent of the edges in a region are unmarked, all matches in the region are deleted on grounds of inadequate evidence. Vergence is modified by using low-resolution edge matches to improve matching in high-resolution edges, and the process iterates. Finally, interpolation between points where disparity is computed produces a smooth surface in depth.

This algorithm uses nondirectional smoothing and feature detectors thought to be consistent with retinal anatomy, but the method of achieving variable resolution and edge detection is not central to the stereopsis algorithm. Grimson's book (*13*) is devoted to the implementation of the algorithm and its application to many natural and synthetic images (Fig. 1). The algorithm is partially implementable in parallel hardware, but the control structure governing the vergence and matching is not cooperative; hysteresis must be explained by another mechanism. Only edges are used in matching, while monocular cues and other local and global image features (texture, regions of similar intensity or color) are not. Vertical disparity is not acknowledged. Interpolation poses a difficult problem, since it should be interrupted across object boundaries. A brief critique of the theory and implementation of the algorithm appears in (*15*), and complementary studies on the subject have been published (*16, 17*).

Before proceeding to another case study, let us see how component computational processes and representations like stereopsis might fit into a general vision system.

A Hierarchy of Representations

The high-level tasks of a biological vision system involve recognition, description, manipulation, and locomotion in a world of moving solid objects, some rigid but many not, with complex surface composition and under complex and varying illumination. The Gestalt psychologists (*18*), and to a greater extent Gibson (*19*), often were concerned with high-level visual tasks and how they could be met with available data. Both schools were handicapped by inadequate appreciation of the power or necessity of computation. The Gestaltists wondered how we isolate (group) visual phenomena into objects. Gibson wondered how we extract invariant and unambiguous perceptions from continually changing input that theoretically could arise from many physical situations. Computer vision researchers believe these to be relevant, important questions about system goals, available input, and natural constraints. Gestaltist rules and Gibson's invariance calculations were inadequate to formalize and describe the necessary computations; computer science provides much more powerful techniques.

Modern computer vision spans the gap between input image and object perception with a hierarchy of representations, operated on by powerful computational processes (Table 1). These processes create representations that pass from image-like representations of physical parameters to symbolic descriptions of entities. The last levels of vision involve cognition, and here computer vision research overlaps the areas of artificial intelligence concerned with symbolic representations, problem-solving, and inference.

At the earliest level, the goal of a general vision system is to derive a representation of image brightness changes that can be used for stereo disparity calculations, detecting changes in surface composition, orientation, distance, reflectance, and so forth. Perceptual phenomena (for instance, subjective contours, the ability to discern colinearity of dissimilar shapes) suggest components for the earliest image representations (such as locations, orientations, and end points of features). Feature detectors that derive these components may then be designed. The next stage, intrinsic image computation, is one of

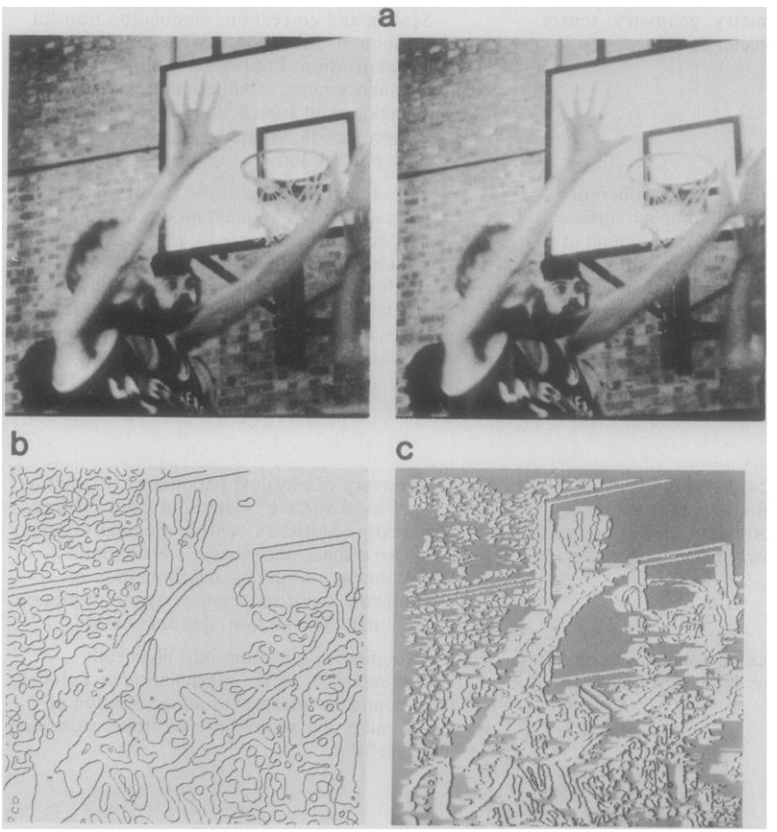

Fig. 1. (a) Two stereo images. (b) Loci of large intensity gradient in a restricted spatial frequency channel, computed as zero crossings of a Laplacian operator smoothed with a Gaussian filter. (c) A representation of the disparity image resulting from Marr's disparity algorithm as implemented by Grimson. [Photographs used with the permission of W. E. L. Grimson]

the major foci of current computer vision research. Surfaces in three dimensions are of vital concern to behaving animals, and it seems likely that their visual systems deal with surfaces at a basic level. Computer vision algorithms and representations for surfaces are designed by taking into account the content of available input representations, natural constraints such as smoothness and homogeneity, known mathematical techniques such as interpolation theory, and implementational constraints such as preference for parallel algorithms. The step beyond intrinsic images is a large one; although they contain physical information they are still image-like entities, not yet described in terms of objects. The generation and use of symbolic descriptions is a large topic beyond the scope of this article.

Two of the most important visual phenomena involve motion and texture, which each transmit much information about the objects and surfaces in a scene. Extracting information from motion, or from the optic flow of the visual field on our retina as objects or viewer move, is at this writing one of the most active areas of research in computer vision, and is a particularly good illustration of the symbiosis that can occur between psychology and computer vision. Our second case study is a more subtle phenomenon and involves the information yielded by shading variations in a static image. The approach shows again the interaction of physical constraints in a parallel computation to derive a physical property image, in this case not depth but surface orientation.

Shape from Shading: A Surface Recovery Algorithm

The variation of an object's perceived brightness, its shading, is a strong clue to its shape. An egg usually appears to be rounded because of shading variations; scanning electron microscope images

Table 1. A general vision system is thought to develop a hierarchy of multiple redundant descriptions (*37*). Information flows in both directions, and processes and representations can be skipped in the vision process. For practical reasons, processes are not completely general but are tuned to a particular visual domain. Sources of constraints that operate in a general computer vision system are shown here opposite the affected processes. Early processes isolate and describe information-containing phenomena (discontinuities). Later processes extract physical characteristics of the scene, using input from earlier processes and relying on constraints operating in nature to recover information that has been confounded and projected away in imaging. The processes extracting surface orientation from shading and range from stereopsis (see text) are two examples. Processes for collecting elements into related groups are widespread. The Hough transform is a technique that is useful throughout the hierarchy. Computer vision research indicates that construction of the many intermediate descriptions is feasible and probably necessary but involves complex computations.

Source of constraints	Process	Representation
	Sensing: television input; digitization of photographs; satellite remote sensors; computer-aided tomography	
Photometry; geometry; sensor characteristics	Scaling and correction: modulation transfer function correction; gray-level histogram equalization; Fourier filtering; contrast enhancement; satellite image destriping; background subtraction; image warping and reprojection; computer-aided tomography reconstruction	Image: intensity at a point $I(x,y)$; multispectral (color) intensity; computer-aided tomography number; stereo pair; image sequence
Smooth, spatially coherent surfaces; complex surface reflectance phenomena at several scales; smooth loci of discontinuities denoting boundaries; continuity of motion	Two-dimensional analysis: feature finding and grouping; directional and circularly symmetric differential "edge detectors" (sometimes in hardware running at television rates); some aggregation of related features in image; operations at several levels of resolution; computation of spatial relations and feature statistics	Generalized images: multiple spatially organized and symbolic descriptions of image information, more robust for calculations than image; representations of edge elements (width, orientation, contrast), blobs, regions, feature terminations, groups, boundaries, virtual lines, interframe intensity changes, texture descriptions, fused stereo image, light source and transparency, average local intensity, average size, local density, local orientation, local distances of features at several levels of grouping
Physics; psychophysics; photometry; geometry; smoothness; object symmetries; known imaging geometry	Recovery of physical parameters: extraction of local surface orientation from shading, texture statistics, texture element or object two-dimensional shape, stereo, optic flow, boundary contours; determination of albedo, color reflectance, apparent motion, illuminant direction, depth contours	Physical property images: more robust for matching to three-dimensional world than image description; representations of local surface orientation, depth (distance), contours of light, shadow, object, background, color and reflectance, three-dimensional geometry of edges
Surface homogeneity, continuity, coherence; psychophysics	Grouping: aggregation into physically meaningful parts; interpolation techniques, boundary interpretation, association of similar characteristics; multiple resolutions useful	Surfaces, volumes, spatial relations: representations of surface patches (simple analytic surfaces, splined patches), volumes (combinations of simple volumes, complex volumes swept out by varying cross section along curve in three-dimensional space), and spatial relations (symbolic propositions or data structures like "semantic nets"); multiple resolutions useful
Physics; epistemology; domain-dependent knowledge; causality; intention; convention	Object, scene, event recognition: matching abstract relational structures; deriving invariants; problem-solving; knowledge representation; inference; planning	

give three-dimensional perceptions through shading that gives a "backlit velvet" effect; and the moon looks flat because its perceived brightness does not vary with surface orientation. Shape-from-shading algorithms (*20*) derive physical property images of local surface orientation from single intensity images. Like disparity, local orientation can be used to obtain relative depth, or shape. Symmetric objects in the world can, through their projected shapes in the image, provide information about surface orientation (*21*), but shape-from-shading algorithms use local clues rather than more global two-dimensional shape information [similar approaches can derive shape from other types of input such as texture or optic flow (*22*)].

The goal of the algorithm is to derive the local orientation of a surface from its image by using three constraints: (i) surfaces are smooth; (ii) the local orientation of the surface is fixed and known at some points, providing a boundary condition; and (iii) there is a known correspondence between image brightness and surface orientation at a point. The irradiance equation formalizes the third constraint:

$$I(\text{location}) = R(\text{orientation}) \qquad (1)$$

where I is the brightness at an image point and R is a reflectance map, which maps every possible surface orientation to a single image brightness. The unambiguous association of a single brightness to each orientation is a strong constraint, ruling out many phenomena such as shadows, mutual illumination, and "paint"—that is, a reflectance function or albedo that varies over the surface. One can derive R from the reflectance function of the surface and the imaging geometry (illumination, viewpoint) or from measurement. Boundary conditions giving local surface orientation can arise from several sources, but a powerful one is that the surface normal around the boundary of a smooth shape is fully determined, being orthogonal both to the grazing line of sight and to the local two-dimensional contour line in the image. Figure 2 illustrates the constraints in shape from shading.

The algorithm produces an intrinsic image of local surface orientation. This array can be integrated (since orientation determines differential depth changes) to provide an intrinsic image of relative surface depth (up to a constant of integration). The computational algorithm is an implementation of a mathematical optimization problem. The problem is to find the orientations that minimize violation of the smoothness constraint and of the irradiance constraint (Eq. 1). If the smoothness error term is the sum of squared differences of direction components of neighboring orientation vectors, the mathematics that emerges is a modified version of Gauss-Seidel iteration to solve a partial differential equation. In Eq. 2, $f(x,y)$ and $g(x,y)$ are the two components of surface orientation at location (x,y), $f^{n+1}(x,y)$ is the value of $f(x,y)$ at the $(n+1)$st iteration of the calculation; $f^{*n}(x,y)$ and $g^{*n}(x,y)$ are the average values of neighbors of $f^n(x,y)$ and $g^n(x,y)$; $I(x,y)$ is the image intensity function; $R(f,g)$ is the reflectance map that defines the reflectance of the surface at (viewer-centered) orientation (f,g); and w is a weight expressing the relative importance of the two terms in the calculation of the next value of f. A similar equation determines $g^{n+1}(x,y)$.

$$f^{n+1}(x,y) = f^{*n}(x,y) + w[I(x,y) - R(f^{*n}(x,y), g^{*n}(x,y))] \frac{\partial R}{\partial f} \qquad (2)$$

The algorithm has an easy geometric interpretation. The orientation at a point in the image is provisionally taken to be the average of its neighbors' orientations (to satisfy the smoothness constraint). This accounts for the first term in the right-hand side of Eq. 2. Any difference between the brightness this orientation would produce (calculated from the reflectance map) and the actual brightness at this point (taken from the image) is reduced by tilting the orientation vector. The second term of Eq. 2 embodies this irradiance constraint. The orientations around the boundary are usually assumed to be known, and by iteration of the basic step the algorithm works its way in from the boundary, adjusting the orientations to conform better to both constraints.

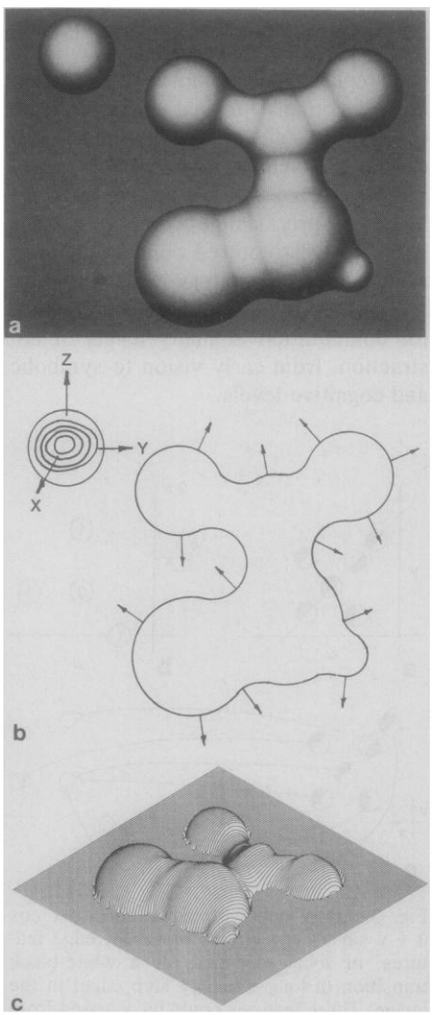

Fig. 2. (a) A synthesized image of a smooth object with a Lambertian (matte) reflectance function illuminated from the viewpoint. The sphere in the upper left is not part of the input image; it is a visualization of R, the reflectance map of Eq. 1. Knowing R is equivalent to having a spherical calibration object in the scene. (b) A small patch of given brightness on the object must have one of the orientations lying along the corresponding isobrightness contour on R. Thus the brightness constrains the orientation to a one-parameter family. The contour of a smooth object provides orientation boundary conditions. (c) The intrinsic image of local surface orientations yielded by the shape-from-shading algorithm may be integrated to derive depth.

Shape from shading is still an active area of research, with current effort bent on extending its power by reducing the restrictive constraints (nonvarying reflectance function) and necessary a priori knowledge (the precise imaging situation, or illuminant position). "Shape-from" algorithms have proliferated as ways were discovered to extract orientation from texture, flow, and even the seemingly sparse clue of an object's boundary contour (*23*).

These algorithms can be implemented in arrays of processors, one per spatial location, connected to neighbors and running in parallel. Compared to stereopsis, the computations are more complex, sometimes global information (such as the reflectance map) is necessary, and usually more precise information must be passed between units. The attractive and versatile computational properties of constraint relaxation algorithms make them useful throughout the vision hierarchy; for example, in symbolic processing they may be used to assign semantic categories or labels to scene elements. A different computational paradigm that is also useful throughout the hierarchy is employed in our third case, which exhibits noniterative but parallel computations that implement grouping processes.

Hough Transform: A Grouping Algorithm

Often a phenomenon of interest, such as a shape outline or a feature like a straight line, is represented in an image by partial and conflicting evidence mixed with confusing noise. The Hough transform has come to denote any of a wide variety of clustering, histogram analysis, and estimation strategies. The point of commonality is the transformation of data into a parameter space where phenomena of interest form clusters. The natural analysis strategies are based on these clusters, or modes, in the parameter space. Hough transformation is related to matched-filter detection strategies, and mode-based estimation makes Hough techniques highly resistant to the effects of outliers. In perceptual psychology, the basic idea has been articulated by Barlow (*24*), who speaks of accumulating linking features (local features to be grouped) in nontopographic maps (parameter spaces). In image analysis, the Hough transform was conceived as an operation like a Fourier transform for the detection of certain parameterizable curves in noisy data. Recently it has been realized that the Hough transform is useful throughout the hierarchy of visual processing as a generalized technique for grouping, representation transformation, and evidence weighing (*25*).

For example, consider the Hough transformation used for line detection. Accumulating local edge features into straight lines is a useful early processing step in computer vision and will serve as an example. The transform may be considered as a voting process, in which an image feature indicating a line produces a set of votes in a parameter space of lines, where votes are accumulated. Each feature votes for the lines that could have caused it and, after all the features are taken into account, the line with the most votes explains the most image evidence. Let a line be parameterized by (ρ,θ) in the line equation $\rho = x \cos\theta + y \sin\theta$ (Fig. 3a). Represent this parameter space by a two-dimensional array, named LineParams, whose two indices correspond to quantized values of ρ and θ. Last, suppose the feature detector applied to a point (x,y) of the image responds with a local edge orientation θ and a measure of edge contrast. Then one version of the algorithm is as follows.

For each point (x,y) in the image, do the following two steps:

1) Apply the detector to get θ at (x,y);

2) If edge contrast exceeds some threshold:

compute $\rho = x \cos\theta + y \sin\theta$;

increment LineParams[ρ,θ];

Several implementations of the Hough transform are possible. The straightforward sequential computer implementation just described represents parameter space in an array. This representation is costly for multiparameter transforms, since it demands space exponential in the number of parameters, but there has been progress in implementations of the accumulating parameter space that use less space. Finally, the Hough transform can be implemented in massively parallel computing networks in which prewiring accomplishes all the voting in one time step (Fig. 3c).

The neural net realization of the Hough transformation shows how complex wiring carrying simple excitation can replace the complex information flow of voting. Our final example of the cooperation of the brain sciences and computer sciences involves the recent renaissance of interest in neural nets. Nets of fairly simple computing units with highly structured connections carrying simple excitatory and inhibitory levels can provide a uniform architecture for computation at many levels of abstraction, from early vision to symbolic and cognitive levels.

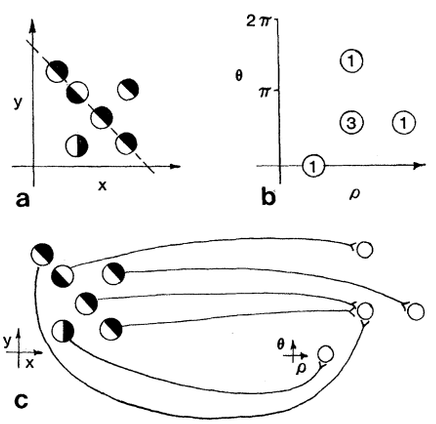

Fig. 3. (a) A line with equation $\rho = x \cos\theta + y \sin\theta$. The circles represent edge features, or local evidence for a white-black transition (a long intensity step edge) in the image. Three features could have arisen from one such straight long edge (dashed line); the other three are inconsistent with any single longer edge. In the Hough transform, a feature "votes" for parameters of phenomena that could have led to the feature. (b) The results of the voting algorithm (see text). This visualization of the line parameter space shows the votes cast for four different line parameters (ρ,θ). The three consistent features in (a) vote for the same line; the mode of the votes yields (without influence from other evidence) the parameters of the best line. (c) A prewired neural net implementation of the Hough transform with voting implemented by excitatory connections. Each edge feature excites a neural unit in the (nonretinotopic) line parameter space with which it is consistent. The best line unit receives the most excitation.

Computing Architecture and Implementational Constraints

Digital computers of the usual (von Neumann) architecture have internal information that is represented in a binary code and interpreted as program or data. The program is a sequence of individual instructions that usually specify operations (arithmetic, logical, copying, tests), but it also allows the sequence of instructions to be altered as a result of tests. Instructions are executed one at a time at extremely high rates (on the order of 10 million a second). Computer vision algorithms have been implemented almost exclusively on such computers, insulated from the hardware by several layers of abstraction culminating in a high-level programming language. Recently, several forms of alternative computing architecture have emerged. Some operate on vectors of data as well as individual data items; many have multiple processors, either identical or specialized, working together. Computing architectures are being designed and built for image processing and management (*26*); these and new technologies such as very large scale integrated (VLSI) circuits can be used to implement more directly the locally connected, parallel computations that are common in intrinsic image computation.

Neurons are simple, relatively slow computing units that are highly interconnected (often on the order of 10,000 connections) into complex structures operating in parallel. Such massively parallel, structured computational architectures will be very fast if the semantics of the information passed between units resides largely in the wiring connecting them. The wired-in semantics of these connections substitutes for the time-consuming interpretation process needed in systems that pass symbolic information. This is important because complete, complex animal behaviors can occur in less than a hundred neural firing times, which are in the millisecond range, where existing sequential artificial intelligence programs require millions of steps. Differences of this magnitude may indicate that the qualitatively different computation methods of animal brains may have to be taken into serious consideration in a viable computational model. There was considerable research activity in the 1960's to model the behavior of random nets of simple computing units and to obtain visual discrimination from such networks by self-organizing changes in the weights units that gave their excitatory and inhibitory inputs [for example, see (*27*)]. As the limitations and difficulties of such approaches were

better understood (*28*) they gave way to those based on processes and, more recently, on highly structured nets (*29*).

There are many reasons for the renewal of interest in nonsequential models of computation. Of course, the ultimate goal of linking perception to brain theory and thus reducing behavior to structure is always in the background. The neurosciences are elucidating the structure and physiology of brain regions and are beginning to formulate theories of function. Many of their theories involve neural nets [for example, (*30*)]. Psychologists have long used models of spreading activation (*31, 32*) and are now beginning to explore questions involving errors, deficits, reaction times, perceptual rivalry, associative memory, and so on for which conventional computer models are not well suited. Computer architectures for vision and parallel cognitive algorithms are being designed. Prototype systems that compute with connections at several levels of abstraction have been built [for example (*33*)]. Other processes being investigated for neural net implementation are motor control in the vestibulo-ocular reflex system (*34*), spreading activation models applied to the disambiguation of word senses in natural language sentences (*35*), recognizing hierarchically stored geometric models, learning, and change. These efforts promise to extend significantly our ability to conceptualize and reason about the powerful, parallel, and distributed computations needed in a general vision system (*36*).

Concluding Remarks

Considered as an information-processing task, vision can be usefully described in three ways, all of which are needed for a complete description.

1) Visual tasks, inputs, strategies, and assumptions (described by psychologists and physicists).

2) Visual processes and representations (described by computer vision).

3) Visual "hardware" (described by neurosciences).

All these descriptions influence and constrain each other. The gap between the concerns of psychology and of brain science is often wide, and their descriptions of the vision system have been developing separately for good scientific reasons. Computer vision may provide an intermediate set of descriptions that can help bridge the gap.

The evolution of a theory of general vision can be guided by formulations of system goals and examination of natural constraints. The former suggest what the system may be computing and the latter suggest how the computation may be possible. The theory should incorporate algorithms and representations for important visual subtasks and should be constrained by facts about the implementation of vision in animals. As the theory evolves, it will provide structure for the choice, evaluation, and understanding of experiments in psychophysics and the neurosciences. It will incorporate smoothly and in a well-founded way many phenomena not currently addressed by vision theory, such as perception of figure and ground, color constancy, and surface orientation in natural scenes. Computer vision represents a relatively holistic stance in the investigation of seeing systems, but a larger synthesis is possible and may ultimately be necessary. For example, consider the following three issues.

1) How are physical property images linked with the symbolic and cognitive models we seem to have that let us reason about visual scenes rather than just react to them?

2) Seeing animals develop with motor systems, yet current research in computer vision supposes that the vision system can be (logically) dissociated from motor capabilities. How are the two systems linked, and can one really be constructed without the other?

3) Learning is a basic ability of animals that is still so little understood that only a few artificial intelligence researchers have considered it seriously. Do general vision systems need learning, and if so can a usable theory of learning be developed?

Despite much progress in the cognitive sciences, there is not yet a complete description (the tasks, processes, and hardware) of any interesting visual process. In computer vision, very difficult technical problems remain at all levels of the vision hierarchy, from feature detection to description of shapes. More fundamentally, abstract definitions of information processing have been dominated until recently by a model of computation that may be inadequate for tasks such as general vision. However, computer vision has made much progress in a few years. Building on a base of mathematics, engineering, technology, computer science, psychology, and neuroscience, it will continue to develop powerful hardware, software, and conceptual systems with which to explore theories of vision.

References and Notes

1. D. H. Ballard and C. M. Brown, *Computer Vision* (Prentice-Hall, Englewood Cliffs, N.J., 1982).
2. A. R. Hanson and E. M. Riseman, Eds., *Computer Vision Systems* (Academic Press, New York, 1977).
3. M. Brady, *Comput. Surv.* **14**, 3 (1982).
4. H. G. Barrow and J. M. Tenenbaum, *Proc. IEEE* **69**, 572 (1981).
5. M. Brady, Ed., *Artif. Intell.* **17** (1981) (special issue).
6. D. Marr, *Vision* (Freeman, San Francisco, 1982).
7. K. Fukunaga, *Introduction to Statistical Pattern Recognition* (Academic Press, New York, 1972).
8. A. Rosenfeld and A. C. Kak, *Digital Picture Processing* (Academic Press, New York, 1982).
9. L. G. Roberts, in *Optical and Electro-Optical Interaction Processing*, J. P. Tippett *et al.*, Eds. (MIT Press, Cambridge, Mass., 1965).
10. D. Marr and T. Poggio, *Science* **194**, 283 (1976).
11. B. Julesz, *Foundations of Cyclopean Perception* (Univ. of Chicago Press, Chicago, 1971).
12. D. Marr and T. Poggio, *Proc. R. Soc. London Ser. B.* **211**, 151 (1979).
13. W. E. L. Grimson, *From Images to Surfaces* (MIT Press, Cambridge, Mass., 1981).
14. H. R. Wilson and J. R. Bergen, *Vision Res.* **19**, 19 (1979).
15. H. H. Baker and P. Blicher, *Sigart Newslett.* **82**, 12 (1982).
16. J. E. W. Mayhew and J. P. Frisby, *Artif. Intell.* **17**, 349 (1981).
17. H. H. Baker and T. O. Binford, *Proc. 7th Int. Joint Conf. Artif. Intell.* (1981), pp. 631–636.
18. K. Koffka, *Principles of Gestalt Psychology* (Harcourt, New York, 1935).
19. J. J. Gibson, *The Perception of the Visual World* (Riverside Press, Cambridge, Mass., 1950).
20. K. Ikeuchi and B. K. P. Horn, *Artif. Intell.* **17**, 141 (1981).
21. T. Kanade, *ibid.*, p. 409.
22. Examples may be found in (*3–6*).
23. H. G. Barrow and J. M. Tenenbaum, *Artif. Intell.* **17**, 75 (1981).
24. H. B. Barlow, *Proc. R. Soc. London Ser. B* **212**, 1 (1981).
25. D. H. Ballard, *Proc. 7th Int. Joint Conf. Artif. Intell.* (1981), pp. 1068–1078.
26. Y. T. Chien and E. A. Parrish, Eds., *IEEE Trans. Comput.* **31** (October 1982) (special issue).
27. F. Rosenblatt, *Principles of Neurodynamics* (Spartan, New York, 1962).
28. M. Minsky and S. Papert, *Perceptrons* (MIT Press, Cambridge, Mass., 1969).
29. J. A. Feldman and D. H. Ballard, *Cogn. Sci.* **6**, 205 (1982).
30. P. Dev, *Int. J. Man-Machine Stud.* **7**, 511 (1975).
31. G. E. Hinton and J. A. Anderson, Eds., *Parallel Models of Associative Memory* (Erlbaum, Hillsdale, N.J., 1981).
32. J. L. McClelland and D. E. Rumelhart, *Psychol. Rev.* **88**, 375 (1981).
33. D. Sabbah, thesis, University of Rochester, Rochester, N.Y. (1981).
34. S. Addanki, thesis, University of Rochester, Rochester, N.Y. (1983).
35. G. W. Cottrell and S. L. Small, *Cogn. Brain Theory* **6**, 89 (1983).
36. D. H. Ballard, G. E. Hinton, T. J. Sejnowski, *Nature (London)* **306**, 21 (1983).
37. This table is adapted from H. G. Barrow and J. M. Tenenbaum (*4*, p. 581).

38. Preparation of this article was supported by the Defense Advanced Research Projects Agency under grant N00014-82-K-0193 and by the National Science Foundation under grant MCS-8203028.