This is to certify that the

dissertation entitled

On Integrating Vision Modules

presented by

Sharathchandra Pankanti

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in Computer Science

Anil K. Jain
Major professor

Date July 31, 1995

On Integrating Vision Modules

By

Sharathchandra Pankanti

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Computer Science Department

1995

Abstract

On Integrating Vision Modules

By

Sharathchandra Pankanti

Individual visual cues are often unreliable and ambiguous. Therefore, integrated vision systems are necessary to obtain a reliable interpretation of complex scenes. Design of such systems is challenging since each vision module works under a different and possibly conflicting sets of assumptions; an effective integration scheme must not only deal with noisy input images but must also overcome the artifacts and restrictive assumptions of the individual modules.

We propose a unified Bayesian integration framework for interaction among the vision modules to obtain a complete 3D reconstruction from a pair of intensity (stereo) images. The proposed integration architecture allows a parsimonious modeling of various interactions. Novel features of the proposed scheme include, (i) interaction of each module with *intrinsic map*, (ii) multi-resolution representations and hierarchical coarse-to-fine control, (iii) fine–grained feedback mechanisms, and (iv) robust estimation procedures based on the principle of *coherence.*

We have integrated perceptual grouping, stereo, shape from shading, and shape

from texture modules under the proposed Bayesian framework. We demonstrate the efficacy of our approach using real images of several different scenes and observe improvements in the quality of recovered 3D structure as a result of integration. The output of the integrated system is shown to be insensitive to violations of individual module assumptions. The numerical accuracy of the recovered depth is assessed for photo-realistically rendered images from several scenes containing a variety of generic surfaces. Average improvement due to integration in depths estimated from the synthetic textured images of surface primitives was about 20%. The average improvement in the shape estimates due to integration was 16%. For non-textured synthetic images, the corresponding improvements in depth and shape estimates were 25% and 23%.

Integrating vision modules is a difficult problem primarily due to our lack of understanding of two underlying issues: (i) an accurate assessment of the strengths and limitations of individual modules; (ii) the representations and control structures which can exploit complementary constraints provided by the imperfect modules to recover the true structure in the data. We have attempted to systematize the design procedure for an integrated system which takes into account these research issues and demonstrated that an integrated system thus designed leads to improved results in a limited scene domain. Much more research is needed to obtain a definitive solution to the integration problem.

To my family

Acknowledgments

Obtaining Ph.D. was a pleasant journey and I had plenty of opportunities to learn from many wonderful people throughout the course of this journey.

My thesis director, Dr. Jain has remained a constant source of inspiration. His insights in the areas of pattern recognition, image processing, and computer vision have a strong influence on my way analyzing and solving problems. Dr. Jain's gentle but firm reminders about the schedule have helped finish this work in a reasonable amount of time. I also greatly appreciate his taking interest in students' overall growth and I am grateful for his advise and encouragement.

I am thankful to the members Ph.D. Committee for agreeing to serve on my Ph.D. committee. Dr. Ruud Bolle has been very supportive and spent considerable time with me (in Cafe Venezia, for instance) in discussing crucial integration issues as well as some of the formulations. Dr. John Weng's critical as well as constructive comments have helped improve the quality of my work. Dr. William Punch's expertise in Artificial Intelligence and Dr. James Zack's expertise in the human visual perception have also enriched my work considerably.

Pattern Recognition and Image Processing Laboratory (PRIP) has been a won-

derful place to work, sleep, and live. Dr. Richard Dubes, Dr. George Stockman, and Dr. Mihran Tuceryan and PRIPies have played a vital role in formation of my ideas. They have been always helpful and without them working on Ph.D. would have been much less interesting. I was supported by National Science Foundation research grants IRI-9103143 and CDA-8806599 during my Ph.D. research during 1992-1995. Special thanks to Lisa Lees for promptly responding to my system needs and last minute panic. She also has been (lately) little kind to my disk extravaganza (impending 4GB disk?).

Integrating vision modules would have been even more difficult if implementations of some of the modules were not made available to us. Dr. John Oliensis provided us with his shape from shading software; Drs. Deb Trytten and Mihran Tuceryan provided the initial version of the line labeling module; Drs. Mubarak Shah and P. S. Tsai shared their shape from shading algorithm as well as some of the images; Dr. John Weng provided the stereo module; and Dr. Super provided assistance in re-implementing his shape from texture algorithm. I am grateful to all of them for patiently listening to my gripes and providing timely help in porting/reimplementing their software.

I need to thank my roomie Satya Kudapa for running countless errands for me (especially during the completion of my dissertation work) and for overall support. Last but not least, support, love, and patience of my family (including the latest addition, Praneetha) are also gratefully acknowledged.

TABLE OF CONTENTS

List of Tables

List of Figures

Chapter 1

Introduction

One of the central objectives in computer vision is to understand the input scene and objects therein from the image(s) of the scene for the purpose of recognition, manipulation, navigation, or other visual tasks. Since the initial efforts of Roberts [163], the field of computer vision has developed both in rigor as well as in the repertoire of the methodologies. Yet, the goal of building vision systems capable of undertaking visual tasks in a human-like way has remained elusive. Most computer vision systems have remained restrictive in their domain assumptions and brittle in their performance. A system making use of limited information will be fallible; robust vision systems would need to necessarily take into account many independent visual cues. How should one develop vision systems that are capable of effectively representing the world knowledge and information in the visual cues? What are the control structures which allow efficient use of these representations? How do these representations and control structures result in robust systems? The research reported here is an attempt to understand these issues by developing a system which integrates

visual cues to achieve a 3D reconstruction of the input scene based on sensed (stereo) intensity images.

This chapter is organized as follows. Section 1.1 discusses the motivation behind the integrated systems for visual reconstruction in some detail. Sections 1.2-1.7 discuss literature (mostly from psychology and physiology) which provides some clues on how the integration is actually effected in the human visual system. Section 1.8 reviews a computational basis for the classification of integration methodologies. Section 1.9 discusses the difficulties in integration of visual cues, both methodological as well as implementational. Section 1.10 describes critical research issues in the design of an integrated system. We also present a statement of the problem that we have solved (Section 1.11), a list of our contributions (Section 1.12), and organization of this thesis (Section 1.13). The chapter concludes with a summary.

1.1 Motivation

Many vision related tasks like grasping, navigation, and exploration require extraction of 3D information (*e.g.,* depth, surface normals) about the input scene using a variety of sensors. The generic 3D reconstruction problem in itself is an important research problem since its study in the past has increased our understanding of the systematic constraints and limitations in designing a vision system [7]. These constraints and limitations could then be gainfully employed to disambiguate 3D structure of a scene or to help design more robust vision systems.

A number of approaches have been used for recovering 3D characteristics of a

given scene. A comprehensive summary of these techniques could be found in Jain and Flynn [88]. One of the approaches for obtaining 3D information has been through direct depth measurement using time-of-flight sensors or structured light. Given precise imaging conditions and favorable surface characteristics of all the sensed objects, a very accurate 3D reconstruction of the scene could be obtained with a high accuracy. However, imaging conditions required by the direct range sensing strategies can not always be improvised; large outdoor structures and distant objects can not be effectively scanned by these special-purpose sensors (*e.g.*, laser, radar); and the object surfaces may not be suitable (*e.g.*, specular/shining surfaces) for the sensors used in such systems.

An important research issue is whether the 3D structure of a scene could be reliably recovered in less assuming and more realistic situations? For instance, can we precisely recover the 2.5D sketch[1] from a pair of intensity images of a scene? To obtain a reasonable depth map, it would seem that with a careful management and integration of the information available in the intensity images, we need not have to resort to precisely engineered solutions. Besides, in many computer vision applications, the imaging conditions and the surface characteristics of the objects to be imaged are well beyond our control. In such situations, a pair of stereo images of a scene illuminated with a finite number of point sources with known positions and possibly with (unknown) ambient light may be the only reasonable assumptions which can be made about the input. The primary objective of this thesis is to

[1] A 2.5D sketch represents depth and surface orientation at each *visible* point in the scene. According to Marr[116], it is an explicit representation of depth, surface orientation, discontinuities in depths, and discontinuities in surface orientations in a scene.

explore and evaluate methods for a complete and accurate 3D reconstruction under these conditions.

The projections of the 3D world onto 2D image planes suffer from a loss of explicit depth information. The resulting intensity images convey 3D information in several different and indirect ways. In the last couple of decades, a number of shape-from-X modules have been identified and shown to be capable of conveying shape of the object in constrained environments. Several other modules such as perceptual grouping have been demonstrated to be helpful in the depth recovery process [72]. Problems posed by many of these modules are *ill-posed* [2] and all these modules make assumptions restricting the scope of their application. Tables 1.1 and 1.2 summarizes various assumptions made by the (computer) vision modules. The primary considerations in making these assumptions might be classified into three categories:

- Ambiguity: A vision module (inherently) can not determine certain component of the 3D structure of the scene.

- Complexity: Although a vision module could potentially extract all the required 3D information, it might be easier to recover the information under simplifying assumptions.

- Reliability: In order for the recovered 3D structure of an input scene to be stable, it might be necessary to impose certain assumptions.

Violations of these assumptions result in making the module output less reliable. Only

[2] A problem is a *well-posed* problem when its solution exists, is unique, and continuously depends on the data. A problem which is not well-posed is an ill-posed problem [187].

Table 1.1: Natural Assumptions in Computer Vision.

Module	**Assumption**	**Example**
Grouping	General viewpoint	Lowe [109]
Stereo	Coherence	Prazdny [154]
Stereo	Cohesiveness	Marr & Poggio [119]
Shading	Smoothness	Horn & Brooks [30]
Stereo	Continuity	Mayhew & Frisby [123]
Stereo, texture	Sufficient context (aperture problem)	Malik & Rosenholtz [114]
Structure from motion	Smooth intensities	Nagel & Enkelmann [137]
Line labeling	Piecewise continuity	Malik [112]

Table 1.2: Synthetic Assumptions in Computer Vision.

Module	Assumption	Example
Edge detection	Local spatial interaction	Geman & Geman [63]
Stereo	Opacity	Grimson [68]
Shading, stereo	Lambertian surfaces	Ikeuchi & Horn [85]
Stereo	Gaussian Noise	Belhumeur & Mumford [15]
Texture	Homogeneity	Super & Bovik [178]
Texture	Isotropy	Garding [62]
Texture	Perfect segmentation	Ohta *et al.* [143]
Shading	Single source of illumination	Lee & Rosenfeld [104]
Structure from motion	Small motion	Horn & Schunck [81]
Line labeling	Origami world	Kanade [92]
Line labeling	Polyhedral world	Sugihara [177]
Shape from Symmetry	Symmetry	Gross [71]

Table 1.2: Synthetic Assumptions in Computer Vision (contd).

Module	Assumption	Example
Line Labeling	Orthographic Projection	Malik [112]
Texture	Para-perspective projection	Aloimonos [4]
Structure from motion	Rigidity	Weng *et al.* [197]
Shading	Constant albedo	Tsai & Shah [192]
Shading	Locally spherical surface	Pentland [152]
Focus	Gaussian blurring	Ens & Brown [55]
Contour	Isotropy	Horaud & Brady [78]
Shading	No interreflection	Horn [79]
Specularity	Dichromatic reflectance	Healey & Binford [74]
Shading & Texture	Fractal surface model	Pentland [151]
Shape using color	Separable colors	Christensen & Shapiro [40]

in an integrated environment could one (i) avoid making very restrictive assumptions, (ii) verify the accuracy of a module output using another independent output, and (iii) assess applicability of the assumptions from the scene properties.

For instance, shape from shading module can not disambiguate "concave or convex" ambiguity (Fig. 1.1). A line labeling module can not definitively resolve whether the top surface of the block shown in Fig. 1.2(b) is curved or planar. Conventionally, such ambiguities have been resolved either (i) by imposing arbitrary assumptions, *e.g.,* many shape from shading algorithms are biased towards 'convex surface', or (ii) by invoking *synthetic* constraints (see Section 1.8), *e.g.,* arguments based on symmetry to decide the curvature of the top surface of the block in Fig. 1.2(b). The imposition of arbitrary assumptions as well as a premature invocation of synthetic assumptions can be avoided by making use of the other visual cues in the input image(s). For instance, a stereo module could resolve the ambiguity in Fig. 1.1, or the shading information (Fig. 1.2(d)) could facilitate a more reliable interpretation of line diagram in Fig. 1.2(b).

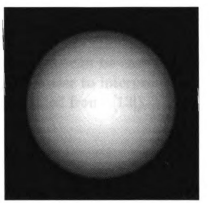


Figure 1.1: Ambiguity in shading: concave or convex?

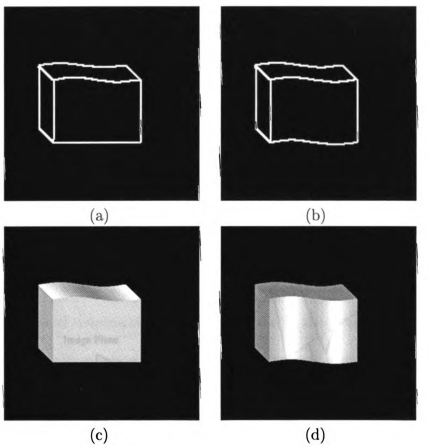


Figure 1.2: Ambiguous line diagrams can be easily resolved by shading information. Line diagrams in (a) and (b) are easier to interpret in conjunction with their shaded counterparts in (c) and (d) (Adapted from [13]).

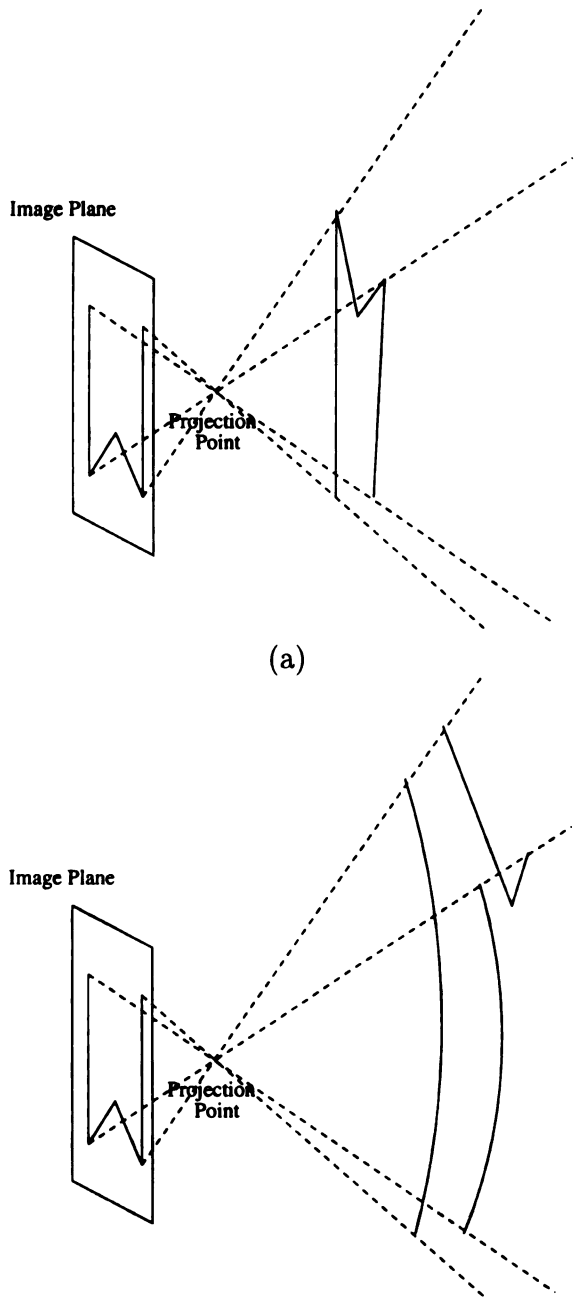


Figure 1.3: Imaging is a many–to–one mapping. The *general viewpoint* assumption states that the interpretation (a) is more likely than (b) [109].

In an integrated environment, it is also possible to assess the violation of a weak assumption[3]. General viewpoint assumption is a case in point. Due to a loss of the explicit depth information, the input images can not be uniquely interpreted based on the projective geometry alone (Fig. 1.3); the assumption of general viewpoint posits that significant structural relationships in an image are unlikely to have resulted due to accidental location or orientation of the observer. However, in a large image (especially, of a man-made environment), a few violations of general viewpoint assumption might have occurred. In an integrated system, the individual modules can afford to operate outside the scope of their nominal assumptions since other independent sources of information can correct its mistakes. For instance, a violation of general viewpoint assumption can be ascertained in an integrated environment using several independent sources of information.

The central role of constraints exerted by the world knowledge and visual cues in image interpretation is widely agreed upon. The details of how these constraints are effectively embedded into a working system are, for the most part, not known. In the hope of finding answers to these problems, we often turn to the human vision literature. Several psychophysical and neurophysiological experiments have been carried out to find the general structure of the perceptual processing in the last few decades. The following sections summarize integration research in human visual processing.

[3]A weak assumption (also referred to as natural constraint) is an assumption about the world which is *frequently* true. See Section 1.8.

1.2 Bottom-up View

This view of solving vision problem holds that different sources of information disambiguate image interpretation by building increasingly abstract representations in a domain-independent fashion. The system obtains higher-level descriptions from image- (lower-) level descriptions by recursively grouping the features at each level of description. This model enjoys a good support from conventional vision researchers. Marr's work is an excellent example of this approach [116].

Marr Paradigm

Marr suggested that it is useful to pose vision problems at a computational level before dealing with representations, algorithms, or implementations. What is being computed? Why is it being computed? While he was clearly aware of the top-down influences, he felt that their role in early vision processes was marginal. He proposed building the following three levels of representations during the course of 3D reconstruction:

Raw Primal Sketch: This representation is responsible for making the local structure in the *image* explicit. It consists of descriptions of physically meaningful events (changes in albedo, illumination, depth, or surface orientation) by a combination of spatially coincident zero-crossings extracted from consecutive levels of representations (*Spatial coincidence*)[4]. Spatially localized configurations of (zero-crossings) are identified into units of representation: *tokens.* Tokens are categorized into edges, bars,

[4]These principles of combination of information across different spatial channels can be quite involved and constitute the first systematic efforts in *integration* of information across the spatial channels [117].

blobs, terminations, *etc.* and have typically half a dozen features like orientation, contrast, *etc.* Each token is a very localized descriptor. For instance, a long thin line would be described by several small oriented segments. Spatial relations among the tokens are also made explicit in the raw primal sketch.

Full Primal Sketch: The tokens in the raw primal sketch are recursively grouped based on their similarity. Such grouping procedures are local mechanisms for organizing perceptually significant events such as texture boundaries and subjective contours. The full primal sketch makes organization of the perceptually significant 2D structures explicit.

2.5D sketch: The features in this representation are surface orientation, depth discontinuities, orientation discontinuities, and coarse depth estimates for *visible* surfaces in the scene. This representation is viewer-centered and is essentially the first description to make the 3D structure of the scene explicit. This representation is the precursor to the final object-centered, segmented, and volumetric representation [118]. Marr argued that a 2.5D sketch description is a convenient common substrate for the visual modules to combine their individual outputs.

Marr's proposal has a strong argument in favor of the modularity of the human vision system. Several psychophysical experiments have demonstrated that the human visual system processes different cues like shape from shading, shape from texture, stereo, shape from motion in a relatively independent way (Figure 1.4). For instance, subjects could fuse random dot stereograms which did not have any other cues [91]. Neurophysiological data available at that time demonstrated the existence of different channels of processing for different types of visual data [108]. Marr coined the term

vision module for these autonomous processes, each handling one type of visual cue. While there was no definitive evidence for the existence of vision modules, he argued that the modular development of vision system would simplify the complexity in its design.

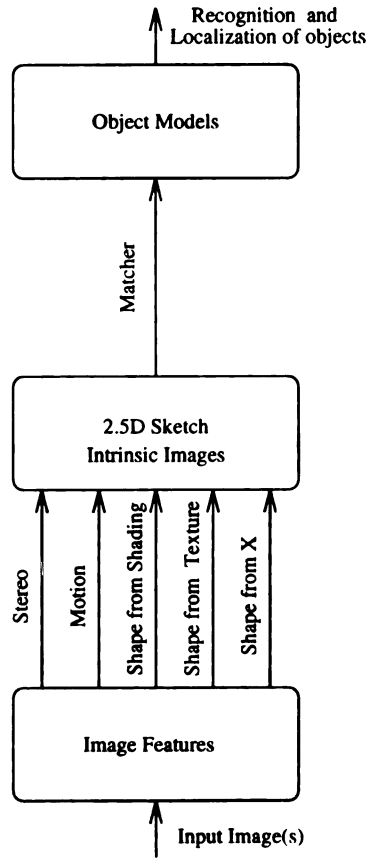


Figure 1.4: Marr paradigm [109].

One of the advantages of the bottom-up approach is its generality. It is usually characterized by lack of any restrictive domain-specific assumptions. Hence the use of the bottom-up, data-driven approach has been recommended in exploratory and generic computer vision tasks, where the details of the environment are not known *a priori*.

1.3 Top-down View

The world we live in is not an abstract space, but a three-dimensional world obeying laws of physics. Most of the objects we perceive in our everyday life are opaque. Object surfaces are smooth almost everywhere. The evolutionary process has built mechanisms into our perceptual apparatus to exploit these features for easier interpretation of our sensory stimuli. These *default* assumptions serve to disambiguate and stabilize our perception in situations where visual stimuli in themselves are not sufficient.

A bottom-up, hierarchical model of perception has influenced much of both classical philosophical thought and psychological theories, thus top-down forces were initially ignored. Marr's paradigm described in Section 1.2 is a case in point. It undermines our ability to infer 3D structures and recognize them in the absence of explicit and conclusive 3D cues.

The human visual system assumes that its viewpoint precludes any accidental alignment of non-causal object features. This assumption is called *general viewpoint assumption.* Several researchers [199, 109, 13, 18] have emphasized that perceptual organization imposes significant 3D constraints in building the 2.5D sketch as a result of general viewpoint assumption; the presence of significant structures in the full primal sketch can be related to the presence of relevant causal 3D structures, making our perception and recognition of objects from line drawings and from single 2D images effortless. Nakayama and Shimojo [138] have demonstrated several situations where interpretation of images even in the presence of conflicting explicit 3D cues

is influenced by the general viewpoint assumption. Although the general viewpoint assumption influences 3D organization of image features in the early human visual processing (and without domain-specific information), most of these inferences are respected in the final image interpretation [199]. The significance of our ability to infer 3D structure directly from a 2D organization of image features has led to the inclusion of a *3D inference* module into Marr's bottom-up paradigm [109]. One interesting example of the top-down influence is shown in Figure 1.5, where the Dalmatian dog is not 'visible' until we are told to look for it. Here, the model-driven processes appear to help the integration of the noisy visual cues, resulting in a relatively vivid perception of the Dalmatian dog [116]. These and several other pieces of evidence have prompted an augmentation of 'classical' bottom-up model of perception. Such cognition-based theories involve interaction of knowledge and expectations with the perceptual process in a more top-down manner [141].

Figure 1.5: Dalmatian dog [116].

The top-down information integration approach is often related to the goal-

directed processing. Goals are decomposed into subgoals till each subgoal is sufficiently simple to be solved directly. A common top-down technique is "hypothesize-and-test" paradigm; here an internal modeling process makes predictions about the way information from each visual module is being combined. Perception becomes an act of verifying such predictions or hypotheses that flow from the model [25, 69].

It is generally agreed that our (human) low-level vision system processes prodigious amounts of information in several cascaded parallel layers [108, 204]. With serial computational hardware, it is very expensive to duplicate the power of our low-level visual system. The desire to circumvent unnecessary low-level processing to reconstruct a huge amount of 3D data is understandable and has given rise to the purposive vision paradigm which emphasizes goal-oriented visual processing [8].

How could we incorporate top-down processes in a vision system that reconstructs the input scene? Traditionally, the top-down constraints have been (i) explicitly invoked by restricting the object domain (*e.g.*, polyhedral objects, smooth objects) or (ii) implicitly embedded in the representational framework (*e.g.*, polynomials, Fourier descriptors).

1.4 Interaction between Top-down and Bottom-up Processes

In computer vision practice, a judicious mixture of the data-driven analysis and model-driven prediction often seems to perform better than either process in iso-

lation. This hybrid control is often implemented using hierarchical representations with a simple pass-oriented control structure [10, 195]. The uncertainties and ambiguities in the sensed data make it difficult for the model-driven processes to effectively hypothesize locations of the features. In order to obtain a salient, noise-insensitive, and useful description over which a model-driven process can 'hypothesize-and-test' efficiently, a few bottom-up passes are usually deemed to be necessary for deriving a better representation. The low-level processes also seem to offer a certain degree of reliability in the performance of the overall system. A number of purely top-down approaches had to be eventually complemented by bottom-up groupings to offer an overall robust performance. A recent example of such an augmentation is provided by Jacob [86, 87].

Among others, Lowe [109] felt that Marr's principle of *least commitment* was too conservative in terms of making the information explicit. For instance, it had no expressive mechanism for exploiting probabilistic information. Further, it deemphasized the role of perceptual organization in a top-down recognition and reconstruction of the scene. Lowe's proposal for an object recognition system is shown in Figure 1.6. The human object recognition remains easy even in situations where the explicit 3D information contained in the image is minimal. He also argued that, in many cases, it may not be expedient/necessary to reconstruct the 3D structure of the scene for recognition of the objects therein.

What mechanisms does the human visual system use for combining top-down and bottom-up processes? Guided search theory [200] hypothesizes an objective function which linearly combines bottom-up and top-down influences for directing visual

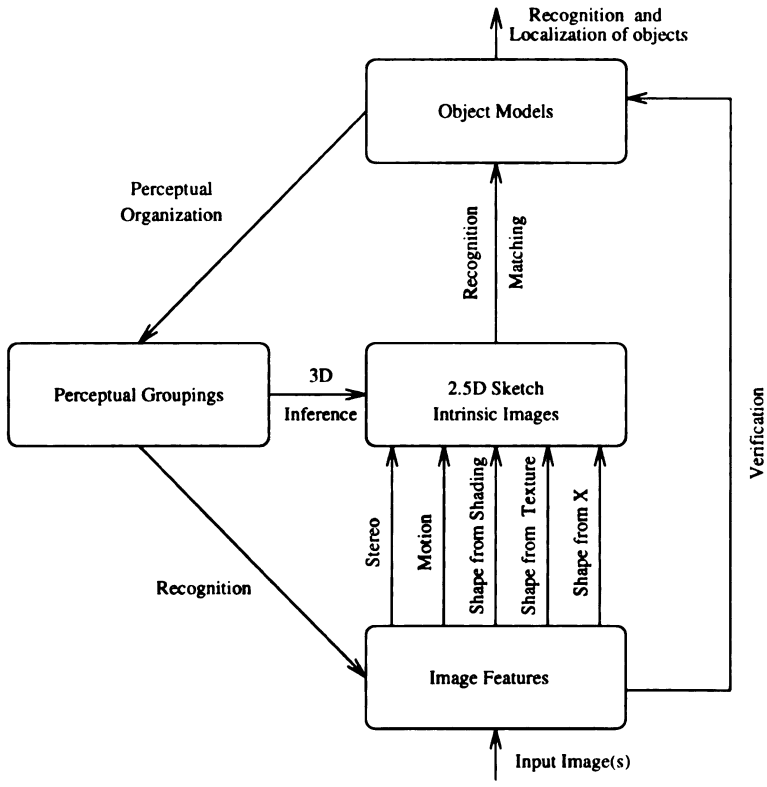


Figure 1.6: Lowe's paradigm [109].

attention. The bottom-up mechanism is represented by a metric measuring the similarity between the given object and neighboring objects (grouping) and the top-down mechanism is represented by a metric measuring the similarity between an 'adaptive' template (*expectations*) and the given object. Duncan and Humphrey [53] propose a theory which is similar in spirit except that they conjecture a non-linear two-way interaction between the top-down and bottom-up processes. Grossberg's adaptive resonance theory also contains similar mechanisms in the connectionist paradigm (Section 1.6).

In most of the computer vision literature, the interaction of top-down and bottom-up influences is implicit and is usually posed in the form of a constrained optimization involving parametric [174] or non-parametric representations [94, 186][5]. These methods are sensitive to the initial conditions and the parameter values. For instance, given undesirable initial conditions, the optimization scheme used by Solina and Bajcsy [174] can produce strange and unintuitive superquadric fits to a given set of 3D points.

1.5 Horizontal Interaction

Not only do the top-down and bottom-up processes interact, but the processes operating at the same level of the representation can also laterally exchange information to disambiguate image interpretation. One common situation in computer vision is

[5] A notable exception to the implicit top-down bottom-up interaction model is the distributed scheme developed by Bozma and Duncan [27]. However, the modeling assumptions have not been fully exploited in these studies due to the inability of the objective function to represent all the physical constraints.

the interaction among the processes in a spatial neighborhood. The importance of this spatial interaction becomes obvious from the fact that human observers have difficulty in interpreting isolated subimages (small portions of a scene) which can otherwise be effortlessly interpreted in their proper spatial context. There is also an evidence for lateral inhibitory influence of neurons on their spatial neighbors [168].

Another type of horizontal interaction is the exchange of information across the vision modules. These interactions have been studied by Bulthoff and Mallot [113] using psychophysical experiments. They hypothesize the following four types of such interactions:

1. Accumulation: The evidence produced by each module could be accumulated as probabilities or confidence indices. More generally, individual support for hypotheses could be linearly combined. Many of the cost functions in optimization formulations represent this kind of interaction.

2. Cooperation: This non-linear interaction could be used for synergistic interpretation of noisy data. Competition between the modules can be considered as an opposite of cooperation. Markov Random Fields (MRF) and game-theoretic models can be considered as examples of this interaction.

3. Disambiguation: One module can help obtain a unique solution from an underconstrained problem posed by another module. Many of the algebraic approaches proposed by Aloimonos [8] can be considered as examples of this type of interaction.

4. Veto: This interaction provides for an overriding role played by a module which

can not be challenged by the other modules. For instance, in several conflicting situations, stereo cue is considered more reliable than shape from shading and the stereo module vetoes alternative interpretations provided by the shading module.

Bulthoff and Mallot [34] studied the integration of stereo disparities, edge information, and shading in the 3D perception of synthetically generated images. Subjects were shown images of end-on views of flat and smoothly-shaded ellipsoids and they were asked to judge the perceived depth. Based on the experimental data, they concluded that

1. The human visual system underestimates depth if the stimulus consists of shading cues alone;

2. When both shading and stereo cues are present, the information provided by the stereo module dominates the final interpretation of the scene;

3. Disparate shading[6] yields a vivid stereoscopic depth (even in the absence of disparate edges); and

4. The human visual system interpolates the depth provided by the stereo system using the information provided by the shading cues.

A linear combination of individual module outputs is meaningful when the individual module outputs agree [113]. Maloney and Landy [113] have described the following results about the interaction between the cues based on their linear combination.

[6]Shading differences in the left and right stereo images.

1. The weight assigned to each cue depends upon the reliability of the cue. The measure of reliability in itself might be computed from some "ancillary" (not necessarily a depth) cue.

2. The concept of *promotion* states that information provided by one cue makes up for the deficiency in another cue.

3. The consistency among the cues can be used to adaptively weight the cues. If the majority of the cues concur on a depth value, say, then the weights associated with the dissenting cues are automatically reduced.

Studies related to the interaction between stereo and texture cues have been conducted by Johnston *et al.* [90] and Buckley *et al.* [32, 33, 31].

Johnston *et al.* [90] studied the interaction among stereo and texture cues. Their experimental results suggest that stereopsis and shape from texture are independent processes in their early stages. They conclude that stereo and texture cues interact simply by means of a weighted linear combination, irrespective of whether the perceptions resulting from the individual cues were in conflict or in agreement. Further, the information from stereopsis was weighted much more heavily than that provided by the texture cues. This indicates that stereopsis is considered to be a more reliable source of information for the human visual system.

Buckley *et al.* [32, 33] studied interactions between stereo and shape from texture. Their results can be summarized as follows:

1. Stereo and texture cues are *pooled* about the 3D surfaces only if evidence (information) provided by the individual cues is similar. The threshold on similarity

depends upon the expected noise in each channel [32].

2. Buckley *et al.* [33] studied the perception of discontinuities in a region of an image devoid of any binocular cues (monocular region). The test images consisted of a textured monocular region between the regions displaying strong binocular depth cues (binocular regions). The perceived location of a 3D edge in a monocular region was found to be consistent with the obvious texture boundaries in that region and the explicit 3D edges perceived in the neighboring binocular regions.

More recently, Buckley *et al.* [31] have studied interaction among stereo, texture, and outline cues[7]. The experimental designs are based on the *cue conflict* paradigm: experiments are designed such that each cue might produce conflicting evidence about the underlying 3D surface. Their experimental data consists of synthetic stereo pairs of either horizontally or vertically oriented ridges. Their findings suggest that:

1. If the synthetic ridge is horizontal, then stereo cues strongly dominate the final interpretation of the input scene.

2. If the synthetic ridge is vertical, then texture/outline cues dominate.

3. Stereo cues dominated in all real ridge stimuli. This result is in agreement with that of Johnston *et al.* [90].

[7] Authors refer to contours as outlines.

1.6 Connectionist Approaches

The most general approach of interaction would be where every intrinsic variable (*e.g.*, reflectance, depth, surface orientation) in some way interacts with every other intrinsic variable as well as with the observations (pixel values). These types of interactions can naturally be accommodated into a connectionist framework. Connectionist approaches are known for their learning capacity, graceful degradation in performance, and spontaneous generalization [125]. On the other hand, these approaches have a tendency to be prodigal in terms of the computational and communication resources. Due to the large number of parameters involved in the connectionist approaches, a good insight into the solution space is needed before a connectionist approach can be used to solve a non-trivial integration problem. Many studies have utilized connectionist models for integration of image features for recognition and reconstruction tasks [72, 203, 125, 115, 119].

1.7 Utilitarian Theory

The most iconoclastic theory explaining how human visual system integrates all pieces of information is proposed by Ramachandran [156]. He argues that the current theories of the human visual perception are "overarching" and impose unnecessary complex mechanisms. According to him, the human visual mechanism was molded into its present form by the evolutionary processes which had to exploit the available neural hardware to effect an opportunistic perception. He cites several examples to support his case. For instance, the human visual system has an "in-built" assumption

of an overhead source of illumination. This can be directly attributed to the upright posture of the human beings and single illumination source in nature (sun). He has devised a series of elegant and novel experiments to prove his case. But, how could one believe that the entire reconstruction problem can be solved by a collection of *ad hoc* heuristics? He counters that, in most cases, the information required by a visual task is qualitative in nature and could easily be extracted with the help of a few heuristics. Some of these arguments can be traced back to Gibson's theory of affordances [64]. The utilitarian theory is similar in spirit to the purposive vision paradigm [5] which has resulted in a few practical navigation and tracking systems [19] but not in any recognition systems [180].

1.8 Interactions between Constraints and Data

A popular computational framework for studying various integration frameworks has been adopted from Clark and Yuille [42]. They argue that all the integration methods can be best understood in terms of the types of constraints they employ and the methods of embedding constraints to achieve stable and unique solutions.

Clark and Yuille define the following three types of constraints:

1. Physical constraints: The constraints that can not be violated by our physical 3D world. These constraints often include the image formation models and other laws of physics.

2. Natural constraints: The constraints that hold good in most of the situations but fail occasionally. General viewpoint constraint or smoothness constraint are

examples of this type of constraint.

3. Synthetic constraints: These are the constraints imposed by the designer of a vision system by restricting the domain of the application of the vision system.

One classification scheme of integration methods is based on how the individual modules interact with the data. Accordingly, Clark and Yuille classify an integration system as either *weakly coupled* or *strongly coupled.*

In weakly coupled systems, each module independently processes the sensed data. In strongly coupled systems, the operation of a module may be affected by the operation of another module.

The weakly coupled systems are further classified into the following three subcategories:

1. **Class I:** Each module can produce a unique and stable solution. These individual outcomes are combined to reduce the uncertainties in the final result (Figure 1.7). The weight assigned to the information provided by each module is dependent on the reliability of that module. Accumulation can be considered as a Class I type of interaction [90, 42].

2. **Class II:** In order to obtain a unique solution from the outputs of the individual modules, it is necessary to apply a set of *a priori* constraints (Figure 1.8). These methods rely on the algebraic solutions of the analytically modeled problems. When the given situation can not be modeled exactly, this method can not be applied. These methods are sensitive to the modeling assumption and the

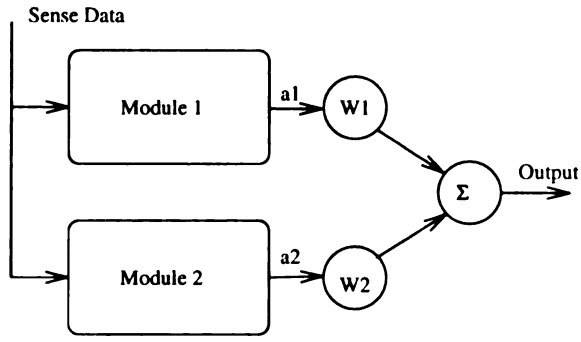


Figure 1.7: Class I: weakly coupled system.

resulting solutions are brittle. Disambiguation [34] or promotion [90] can be thought of as Class II type of interactions.

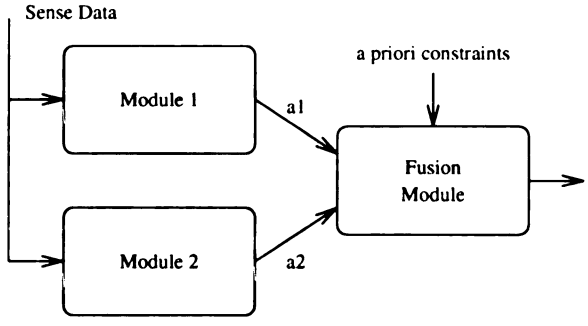


Figure 1.8: Class II: weakly coupled system.

3. **Class III:** This method can be considered as a combination of the Class I and Class II methods. An overdetermined system is achieved by a set of modules and constraints (Figure 1.9).

The strongly coupled systems are further classified into feedforward and recurrent systems (Figures 1.10 and 1.11). In recurrent systems, feedback paths are allowed. In feedforward systems, the feedback paths are not allowed. The way in which a module can affect another module is either by controlling its *a priori* constraints or

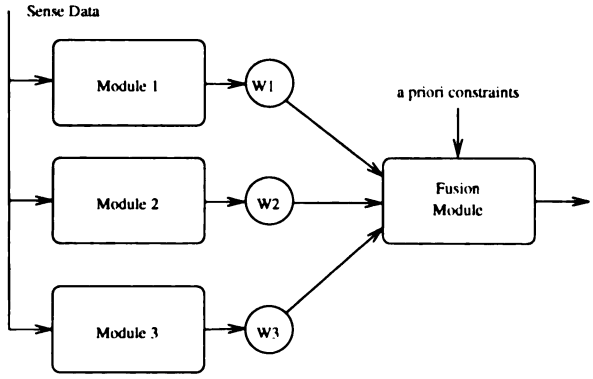


Figure 1.9: Class III: weakly coupled system.

by directly modifying their inputs. These models of interaction are the most general. Cooperation [34] and competition can be thought of as examples of strongly coupled systems.

1.9 Difficulties in Integration

It is generally agreed that vision researchers have not devoted sufficient efforts in designing and building complete vision systems [8]. Consequently, there is a dearth of expertise in the context of building *complete* vision systems.

The sources of complexity for the integration problem can be broadly classified into three categories: (i) the inherent difficulty in designing a complete vision system, (ii) theoretical issues related to integration, and (iii) the implementational problems. First, we list the difficulties arising due to the characteristics of the domain [164, 51]:

- Sheer quantity of data: The volume of input data to a computer vision system is overwhelming. An image with a reasonable resolution (512×512) and frame rate (20 per sec.) would involve about 20 Mb data per second. A simple

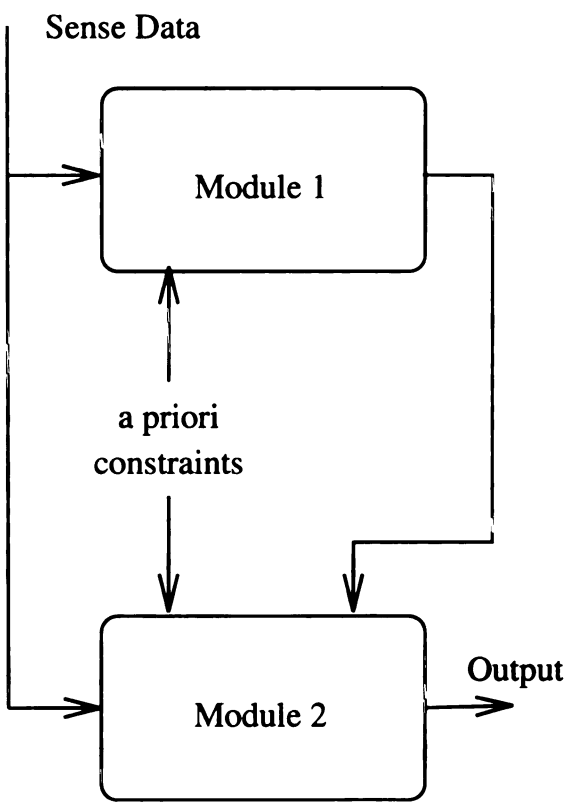


Figure 1.10: A feedforward system.

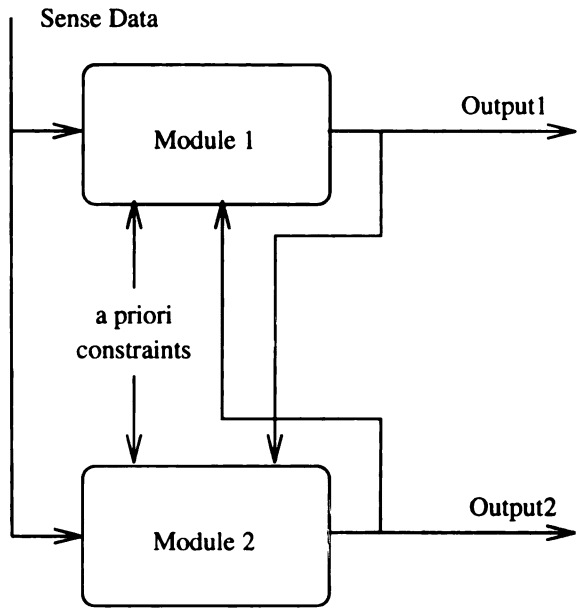


Figure 1.11: A recurrent system.

convolution operator over this data would require almost half a billion arithmetic instructions per second. Even massively parallel architectures will need to be very selective in their choice of processing strategies.

- Uncertainty in data: As mentioned earlier, visual data are often locally ambiguous. In addition, image noise further deteriorates the information content.

- Lack of introspection: Although every human being is an 'expert' in *performing* visual tasks, we do not know how we acquire such a capability. What constraints actually work? Why do the constraints that actually work are so unintuitive? The only process of introspection of our visual processing is through data on visual illusions and on psychophysical experiments.

We now discuss inherent theoretical and implementational difficulties in integration.

- One of the most difficult problems in integration is specification of the constraints and the role played by each of the constraints. The last few decades have identified several cues that are useful in 3D reconstruction. However, the exact role played by each of these cues in the context of a vision system recovering the 3D structure of an input scene and its relative significance is still an active research issue.

- Incorporation of these constraints into an integrated system appears to be the most difficult problem. Quite often, the constraints exerted by the cues are not obviously commensurable. For instance, the geometric constraints imposed by line labelling algorithm are qualitatively different from the information offered by stereo or perceptual organization module. Is it possible to establish a common framework/representation for interaction? The most general representation would be the multivariate joint probability distributions involving the 3D properties of the world and the response (features) of each module. How can we obtain this distribution? Even if such a distribution could be estimated, we feel that it would be too general to constrain the solution to the 3D geometry of the world. Is there a better choice? Marr proposed that we could use 2.5D sketch as a common interaction ground for all the vision modules. This appears to be a reasonable solution for all the modules providing depth information. What about the cues which are not directly involved in estimating depth? Even when we accept the depth map as an interaction ground for all the modules, it is not clear as to what representations and what levels of representations will be most

suitable. One could argue that coarse qualitative features would form a more stable basis for interaction among the modules. But, what is a desirable choice of qualitative features? How can these qualitative representations be made to work in a domain where constraints of many vision modules are expressed in quantitative terms?

- A simultaneous recovery of a large number of intrinsic variables (*e.g.*, reflectance and surface orientation at each pixel) is often intractable. If a limited autonomy is to be granted to each visual module, then such recovery of intrinsic variables and reconstruction inherently needs simplifying initial conditions to be incorporated into each module. Sometimes, these assumptions are necessary because 3D structure of the scene is not known *a priori*. For instance, if we desire to include a mutual illumination model into a shape from shading module, it needs an *a priori* knowledge of the geometry of the scene to be recovered! How do these assumptions affect the performance of the other modules? What approaches permit a gradual refinement of the initial assumptions made by each module to obtain the optimal reconstruction?

- Traditionally, researchers have used a linear combination of the individual objective functions to derive a global object function on the basis of simplicity arguments. Often, these cost functions do not represent the dynamics of the interaction among the constraints. The resultant systems often require a 'good' set of parameter values to deliver a reasonable performance.

- Another difficulty in the integration problem is due to the diverse sets of as-

sumptions made by the individual modules. Often the domains of applicability of the individual modules do not overlap or overlap only slightly.

- When should a module actually be employed? As simple as this sounds, often this issue can not be dealt with in a theoretical fashion. For instance, take the case of a shape-from-texture module. What is texture? Is texture isotropic? Are the surfaces of objects in the scene piecewise continuous? These and other questions need to be answered before we know when and where to 'apply' the module.

- Often, algorithms involved in each module are chosen off-the-shelf for the visual integration. The designer needs to take into account the artifacts and idiosyncrasies of the individual modules over and above the theoretical difficulties.

1.10 Design Issues

Design issues involved in developing a machine vision system are not very different from those involved in developing any other complex system. Any non-trivial system design will involve conflicting requirements and good judgment will be needed in making the correct trade-offs. We discuss here some of the important issues.

- *Correctness and Reliability:* The purpose of integration is to increase the reliability of the final output. Hence, in a formal sense, the system should *correctly* use all the available information. Unreasonable assumptions about the data and the constraints often introduce artifacts in the final solution [20]. The overall

solution should continuously depend on the data [187].

- *Completeness and Generality:* The integrated system should include a reasonable set of modules and constraints to provide an accurate depth information in commonly occurring situations where the individual modules fail. Yet, the proposed framework should be able to accommodate any additional information and cues.

- *Optimality and Tractability:* Systems employing a minimal number of constraints tend to be brittle; overdetermined systems deliver stable performance [42]. Although, computationally effective models would be attractive, the issues of optimality and tractability are currently of secondary interest.

- *Modularity and Extensibility:* System design, development, and maintenance are considerably streamlined if the system has a modular structure. It is also easy to extend such a system. However, modularity may often result in suboptimal solutions.

- *Defaults, Partial Information, and Ambiguity:* Perhaps, this is one of the most important issues related to integration. The individual modules need reasonable mechanisms to represent their intermediate inferences which can be used by other modules. In the absence of any external information, the modules need to be aware of the context-dependent defaults.

1.11 Problem Statement

We are trying to solve the following problem:

Given a pair of stereo images of a scene containing objects illuminated with a finite number of point sources with known positions and possibly with ambient light, our goals are (i) to recover a complete and accurate 3D structure (depth and surface normals) of the input scene, and (ii) to evaluate the efficacy of the proposed integrated reconstruction results.

A generic block diagram of modular integration is shown in Fig. 1.12. Given a stereo pair of intensity images, the integrated system recovers the 3D structure in the scene with the help of a given set of vision modules. Sample input images of a synthetic scene along with the desired outputs from the integration system are shown in Fig. 1.13.

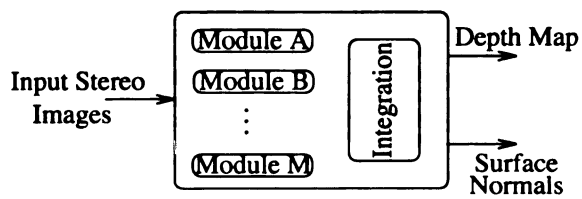


Figure 1.12: A generic integration scheme.

The choice of which modules to incorporate into our system was not easy, given the long (and extending) list of vision modules which have been used in the vision literature (Table 1.3). Stereo, shading, texture, grouping, and line labeling information are among the most extensively researched cues and form a representative sample from the entire gamut of information that could be extracted from the intensity image(s).

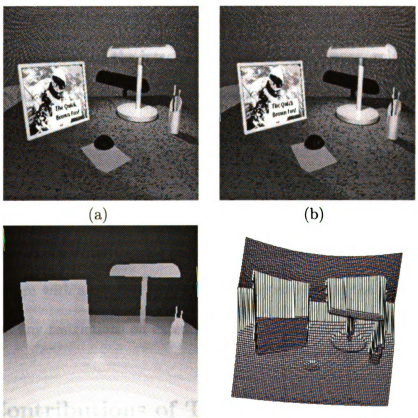


Figure 1.13: System input/output: (a) and (b) synthetic left and right input intensity images; (c) ground truth depth (pixels closer to us are brighter), (d) a wireframe representation of the depths.

Table 1.3: A list of vision modules.

Shading [80]	Texture [61]	Stereo [68]
Contour [78]	Symmetry [186]	Grouping [199]
Photometric stereo [202]	Focus [55]	Line labeling [139]
Color [40]	Motion [197]	Specularity [74]
Occlusion [97]	Vergence [1]	Perspective [131]
Darkness and Shadows [95]		

Further, principles of 3D reconstruction from several other cues closely resemble those of the modules constituting the integrated system. For instance, solutions to structure from motion and stereo problems are based on establishing correspondence and shape from structured light, shape from contour, and shape from texture modules attempt at inverting the projective imaging geometry. Consequently, the integration of the other modules into the existing systems should be fairly similar. The proposed unified Bayesian framework (Chapter 5) facilitates inclusion of a new module into the integrated system with a relative ease. It should be noted that the proposed systems do not impose any restrictions on extending them to include new modules.

1.12 Contributions of This Thesis

The following is a list of the contributions of our work.

1. We propose and implement a unified Bayesian framework for integrating vision modules.

2. We have built two working integrated vision systems: one integrates perceptual organization, stereo, shape from shading, line labeling and the other integrates perceptual organization, stereo, shape from shading, and shape from texture.

3. Design of a perceptual organization module which integrates region- and edge-based segmentations using *Gestalt* and intensity gradient cues.

4. A limb detection module for reliably detecting occluding limb boundaries.

5. An improved implementation of the line labeling module which exploits strong constraints exerted by limb boundaries.

1.13 Organization of the Thesis

Rest of this thesis is organized as follows. The details of computational mechanisms needed for the interactions described in this chapter are the topics of Chapter 2. The individual vision modules and their limitations are studied in Chapter 3. A non-uniform integration scheme is proposed and evaluated in Chapter 4. Chapter 5 provides a Bayesian integration framework for integrating vision modules. The concluding chapter describes the contributions of this thesis and directions for future research.

Chapter 2

Integration Methods

Computer vision researchers have studied several models of information integration. In this chapter, we will first describe different models of integration methods. This will be followed by a summary of the efforts on information fusion and integration available in the vision literature. Each section will briefly describe an integration model, followed by the relevant vision applications. We will conclude with a comparison of the strengths and shortcomings of various approaches.

2.1 Complexity of Integration

Complexity of integrating information can vary significantly. In this section we make a distinction between different types of integration based on their complexity and introduce a terminology for different information integration strategies in the literature.

- **Type A: Registration.** In this type of information integration, a common frame of reference is established between two (or more) representations to make

the spatio-temporal relationships between the components of each representation explicit with virtually no interpretation of the information (Fig. 2.1). Example: Ton and Jain [188].

- **Type B: Fusion.** The registered representations can locally be combined to obtain a more reliable information. Fusion usually takes into account a limited number of constraints like spatial dependency and continuity (Fig. 2.2). Example: Nadabar and Jain [133].

- **Type C: Collation.** Given a set of explicit constraints, two (or more) representations are related and combined to obtain a more complete and reliable information about the data (Fig. 2.3). Note that a collation problem might involve global constraints. Example: Chang and Aggarwal [38].

- **Type D: Integration (of cues).** In this scenario, the constraints are not explicit and often, the interactions among the constraints are not clearly understood. Further, the integration problem might often be underdetermined. (Fig. 2.4). Example: Moerdler and Boult [128].

The work described in this thesis primarily deals with the integration of cues.

2.2 Bayesian and Non-Bayesian Fusion

Consider a multisensor system with M sensors at each of the N locations producing a set of $M \times N$ observations, $\{X_{ij}\}$. Let $\{Y_{ij}\}$ denote the *true* (unknown) world

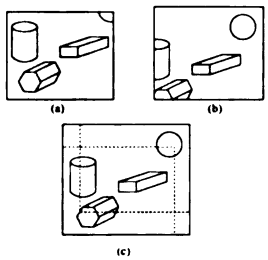


Figure 2.1: Registration: (a) and (b) are two images of a given scene; (c) shows an image which makes the spatial relationship between (a) and (b) explicit. The dotted lines demarcate positions of (a) and (b) with respect to each other.

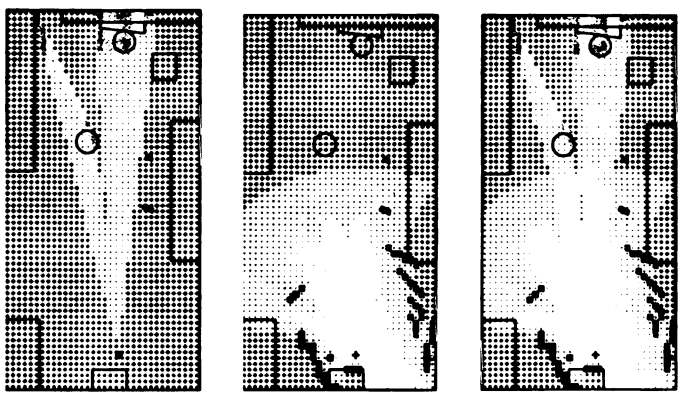


Figure 2.2: Fusion: (a) stereo, (b) sonar, and (c) fused data. The density of the dots represent the likelihood of occupancy of that space by a physical object [129].

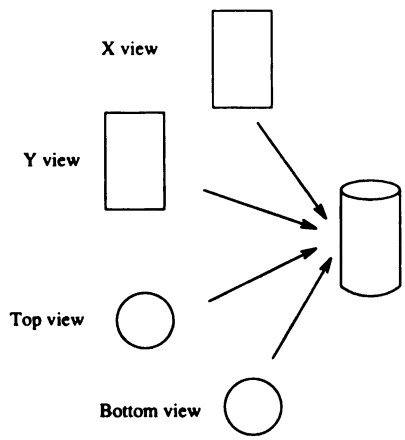


Figure 2.3: Collation.

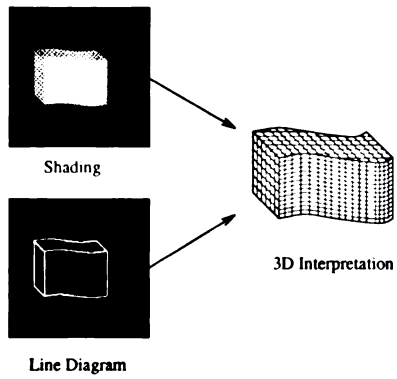


Figure 2.4: Integration of line diagram and shading data to obtain a correct 3D interpretation.

variables[1] denoting the state of nature at each of these locations. Given a set of possible labels of objects at these locations, the system attempts to determine which assignment of labels to the variables $\{Y_{ij}\}$ best describes the observations.

Bayes' formula states that the optimal prediction about the state of nature can be made according to

$$P(Y_{ij} = y_{ij}|\{X_{ij}\}) = \frac{P(\{X_{ij}\}|Y_{ij} = y)P(Y_{ij} = y_{ij})}{P(\{X_{ij}\})}, \tag{2.1}$$

where

- $P(Y_{ij} = y_{ij}|\{X_{ij}\})$ is the *posterior* probability of an object with label y_{ij}, given the set of observations, $\{X_{ij}\}$.
- $p(\{X_{ij}\}|Y_{ij} = y_{ij})$ is the *likelihood* of a set of observations $\{X_{ij}\}$, given $Y_{ij} = y_{ij}$.
- $P(Y_{ij} = y_{ij})$ is the *prior* probability of the label $Y_{ij} = y_{ij}$.

[1] Also referred to as real world variables [8].

- $P(\{X_{ij}\})$ is the normalizing factor.

Each sensor i will be able to supply its *opinion* of the observations representing a true label. Evaluation of $p(Y_{ij} = y_{ij}|\{X_{ij}\})$ by the system involves integration of the information received from all the sensors. Often, a simplifying assumption of the statistical independence of the likelihood function of each sensor is required in a Bayesian framework to yield tractable schemes of integration[2]. Such an integration results in an increased certainty of information [42].

Bayesian methods have been used by several researchers for information integration. Bayes' rule is the basis for determining 'occupancy' of the space in mobile robot work of Moravec [130] and of Mathies and Elfes [122]. They both fuse data originating from different sensors to reliably avoid obstacles in the robot's path. Chou and Brown [39] have used Bayesian methods for labelling image regions. Chu and Aggarwal [41] have used the *maximum likelihood estimate* for obtaining an edge-based segmentation from multiple sources of information. Bolle and Cooper [23, 24] have used Bayesian methods for integration of evidence offered by different parts of an object for recognition tasks. Rao and Whyte [159] used a decentralized hierarchical algorithm for integrated decision making in multisensor systems. Rigoutsos and Hummel [161] have used a distributed Bayesian formulation for objection recognition. One of the major difficulties in implementing a Bayesian integration is in specifying the *a priori* probabilities.

Several researchers have explored the possibility of making *robust* decisions instead of taking the Bayesian approach [99]. Robust statistics show a graceful degradation in

[2]More involved treatment of dependencies is deferred until Sections 2.4 and 2.5.

performance in the presence of noisy observations. For instance, the median of a data set is a more robust statistic than its arithmetic mean. Use of robust statistics usually results in more practical systems at the expense of tractability of rigorous analyses. Explicit voting schemes have also been used in the literature to reliably integrate different sources of evidence. A statistical multi-source classifier is a general method for classifying multispectral data [167]. A linear opinion pool classifier uses consensus-based classification [16] and is shown to perform better than the maximum likelihood classifier for certain simple data sets. Quite often, in the absence of quantitative data, subjective methods of fusion are pursued. Two such approaches which simplify the integration of evidence are: Dempster-Shafer theory [169] and Possibility theory [52]. They both are characterized by very simple rules of combination of evidence. The price one often pays for such simplicity is the erroneous results due to implicit independence assumptions underlying such subjective integration models. In order to overcome these limitations, researchers have attempted to use more sophisticated models [83]. However, it turns out that these models are not fool-proof, either [8].

2.3 Relaxation

A relaxation algorithm is an iterative method of constrained optimization. Given a set of labels, a set of objects[3], neighborhood relations among the objects, and constraints among the labels of the neighboring objects, a relaxation algorithm finds the most consistent configuration of the object labels. The idea is to initially assign all possible

[3]We have used the terms *objects*, *sites*, and *pixels* interchangeably in next few sections.

labels to all the objects and then remove a label from the label set of an object if it is found to be incompatible with *any* of the labels of the neighboring objects. This removal may, in turn, make some labels of some other objects inconsistent. The process continues until either (a) the label set of an object is empty indicating that no solution is possible; or (b) no label of any object can be removed indicating the desired configuration(s). In the latter case, we may need to impose more constraints to avoid ambiguity.

This concept can be extended to *probabilistic* relaxation. Here, instead of a definite association of certain labels with an object, the likelihoods of an object being associated with *each* label are computed. Note that the term *probabilistic* is merely a misnomer; the derivations for the expressions used for ranking the likelihoods are justified on the basis of subjective arguments. However, these (pseudo) probabilistic approaches have been found to be useful in building practical vision systems.

Let $P_t(y) \in [0,1]$ be the 'probability' or weight associated with site t (a pixel (i,j)) and label y. Constraints are provided in terms of compatibility functions, $r_{ts}(y,y')$, quantifying the compatibility between label y at site t and label y' at site s. The weights are updated in parallel according to the following iterative equation [107]:

$$P_t(y)^{(n+1)} = \frac{P_t(y)^{(n)}[1+\Delta P_t(y)^{(n)}]}{\sum_y P_t(y)^{(n)}[1+\Delta P_t(y)^{(n)}]}, \tag{2.2}$$

where

$$\Delta P_t(y)^{(n)} = \sum_s d_{ts}[\sum_{y'} r_{ts}(y,y')P_s(y')^{(n)}], \tag{2.3}$$

n is the number of iterations, and d_{ts} are the weights specifying the relative contributions from the neighbors of y. The algorithm terminates when weights for some label of *most* of the sites is close to 1. Given the values of visual cues at each pixel (say), a set of labels to be associated with each pixel, and the constraints among the labels of neighboring pixels, a relaxation algorithm is an iterative method of finding the most consistent set of (label, pixel) association. However, as the integration problem becomes more complex (more number of measurements, intrinsic variables) and the visual cues interact in a more complex way, it becomes harder to interpret the parameters (r_{ts}, for instance) of the relaxation process. Secondly, the true measurements are ignored after they have been used for constructing the initial estimates of the labels. Due to these two reasons, relaxation methods are considered to be weaker formulations for integration of evidence than the classical probabilistic models [132].

There is an extensive body of research on relaxation methods and their applications to vision problems [82, 165, 184, 67, 49, 96]. Marr and Poggio [119] were among the first to suggest the utility of relaxation techniques for cooperative computing in low-level vision modules. We will briefly mention a few studies that formulate visual integration in terms of relaxation.

Barrow and Tenenbaum [12] used relaxation for computing intrinsic properties like reflectance, surface orientation, and illumination of a given image. Given two images of a scene taken at different times and from different positions, a relaxation approach was used to match the images in [119, 11]. A multigrid technique in relaxation was introduced by Terzopoulos [185] to accelerate propagation of constraints in relaxation applications involving large high resolution images. Its efficacy was demonstrated in

various applications, including a) finding lightness (reflectance) of Mondrian surfaces; b) estimating surface orientation from shading cues; and c) computing optical flow from an image sequence.

2.4 Markov Random Fields

The Bayesian approach discussed in Section 2.2 gives us a method to find an 'optimum' estimate of the state of nature without exploiting any dependencies (*e.g.*, spatial dependency) among the neighboring pixels. In order to take advantage of the spatial constraints among the pixels (objects) in the neighborhood, we need a formulation which can couple probabilistic decision making with spatial constraints. The use of spatial constraints is often necessary to make the image interpretation problem well-posed [121].

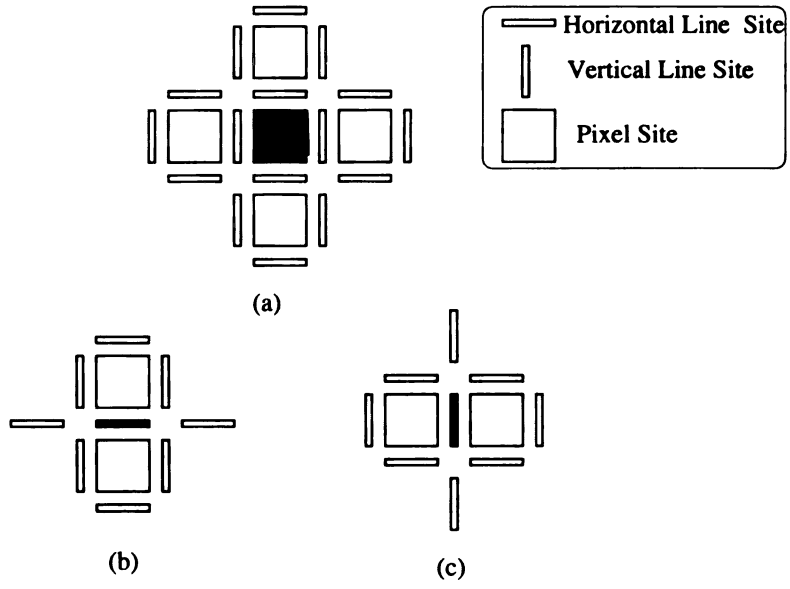


Figure 2.5: Example neighborhoods of pixel and line (edge) sites for (a) a pixel site; (b) a horizontal edge site; and (c) a vertical edge site (from [134]).

One method to realize this is to consider a multivariate probability distribution involving real world variables (intrinsic variables) at each pixel and observations of visual cues at each pixel. Given measurements about the observable visual cues, the prior constraints among the observed visual cues and the true label, one could formulate a Bayesian solution according to:

$$P(\mathbf{y}|\mathbf{x}) = \frac{f_{\mathbf{YX}}(\mathbf{x}|\mathbf{y})}{f_{\mathbf{X}}(\mathbf{x})},$$

where

$$\mathbf{x} = \{\ \mathbf{x}_{ij};\ \mathbf{x}_{ij}\ \textit{is the observation made by the } j^{th}\ \textit{visual cue at the } i^{th}\ \textit{site}\},$$

and

$$\mathbf{y} = \{\mathbf{y}_i;\ \mathbf{y}_i\ \textit{is the state of nature at the}\ \ i^{th}\ \textit{site}\}.$$

Note that $\mathbf{y}_i$ is a vector representing the intrinsic variables at site Q_i. This large system of variables makes the problem not only intractable but often underconstrained. We can, however, impose a reasonable assumption that only pixels within a local neighborhood can interact directly (*locality* assumption). This simplification is the essence of the Markov Random Fields (MRF) formulation. Let $\mathcal{N}(Q_i)$ denote the neighborhood of a site Q_i[4]. Let Y_{ik} represent one of the intrinsic[5] variables at site Q_i.

[4]For the sake of brevity, a linear ordering of the two-dimensional sites is assumed.
[5]Also referred to as real-world variables.

The Markovian property states that[6]

$$P(Y_{ik}|Z_{\setminus Y_{ik}}) = P(Y_{ik}|Z^{N_i}),$$

where $Z = \{Y_{ij}\} \cup \{X_{ij}\}$;

$Z^{N_i} = Y^{N_i} \cup X^{N_i}$;

$Y^{N_i} = \{Y_{nl}|Q_n \in \mathcal{N}(Q_i)\}$;

and

$X^{N_i} = \{X_{nl}|Q_n \in \mathcal{N}(Q_i)\}$;

The constraints between 'neighbors' are defined by assigning energies to all realizable configurations; lower energies indicate more likely configurations. These energies contribute to the 'internal' energies (*clique energies*) at each site. The external energy at each site depends upon $P(Y_{ij}|Z^{N_i})$. The energy of the entire system will be the sum of the internal and external energies of all the sites. Given $\mathbf{X}$, definition of a neighborhood $\mathcal{N}(.)$, and likelihoods of different configurations of neighbors (in terms of *clique* energies), the problem of most consistent interpretation of the observations is equated with the problem of finding a realization of Y which will result in a minimum energy of the system. Notice that MRF formulations not only permit modeling the constraints among the observed visual cues, X_{ij}, but they can also handle the constraints among Y_{ij}s and the constraints between Y_{ij}s and X_{ij}s.

Although MRF formulations can elegantly model many real-world constraints like transparency and continuity [111], there are certain difficulties in employing them for

[6]See appendix for notation.

practical applications. First of all, MRF models are computationally expensive. Many alternative formulations are being devised to alleviate this problem [17, 121, 120]. A second problem plaguing implementation of an MRF formulation is the difficulty in estimating the prior probabilities which represent the contextual information in the given application. A few techniques have been devised to partially alleviate this problem [134].

MRF models used for integration of different information sources require that all the information from individual sources be available simultaneously. Marroquin [120] modified the MRF methods to ease these restrictions by decomposing the overall processing into several stages. Each stage can be designed to handle partial data (estimates). Marroquin claims that these new models, random measure fields, are computationally less expensive than other regularization methods (see Section 2.7). He also argues that these models are more appropriate for visual reconstruction problems due to their ability to handle phenomenon like transparency, occlusion, etc.

The pioneering research in MRF has been the work of Geman and Geman [63] who formulated the problem of segmentation of intensity images using a tightly coupled system of 'line' and 'pixel' processes. The line processes prevent smoothing across a potential discontinuity. The *maximum a posteriori* (MAP) solution is achieved by the simulated annealing process.

Gamble *et al.* [60] have used MRF models to integrate the outputs of four vision modules: color, edge detection, motion, and stereo. The integration is based on the assumption that shape and color discontinuities are usually (spatially) related to the brightness (intensity) discontinuities. And that motion boundaries coincide

with the depth discontinuities. By selecting the coupling parameters empirically, they have obtained a reasonable improvement in the localization of boundaries over those estimated by the edge detection module alone. Modestino and Zhang [127] have used MRF models for consistent labeling of presegmented regions. Daily [48] has used MRF models to integrate information in color channels with the process of segmentation. Nadabar and Jain [133] used the MRF models for fusing range and intensity images.

2.5 Bayesian Networks

A more general modeling of the spatial and other constraints can be carried out by means of Bayesian networks. Bayesian networks, which are also called belief networks, or causal networks, are directed acyclic graphs with nodes representing propositions (or random variables) and arcs signifying direct dependencies quantified by the conditional probabilities. An example of the Bayesian network is shown in Figure 2.6. If the events X_1-X_9 were independent, their joint density would have been simply a product of the individual probabilities. The events in Figure 2.6 are not independent (perhaps, due to common underlying cause). For instance, the nodes X_8 and X_9 are related [7]. This dependency is represented by node X_7 and the joint density of events X_7, X_8, and X_9 is given by

$$P(X_7, X_8, X_9) = P(X_7)P(X_8|X_7)P(X_9|X_7).$$

[7] They represent two parallel boundaries, for instance.

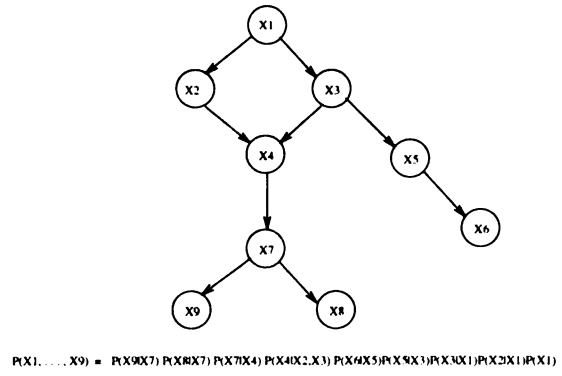


Figure 2.6: An example of a Bayesian network.

The Bayesian network formalism becomes intractable for any non-trivial application. Cooper has shown that the problem of constraint propagation in such networks is NP–hard [44]. Usually, there is a need to use additional domain-dependent heuristics to make these networks feasible for any practical application.

Such a domain-dependent extension of the Bayesian network called perceptual inference network was used by Sarkar and Boyer [166] to represent the statistical dependencies among various features (*e.g.*, curvilinearity, and symmetry) in images of man-made environments. They used this network to integrate the information about various spatial features to form composite hypotheses and a prediction of structures (*e.g.*, rectangles and ellipses).

2.6 Information-theoretic Methods

Gestalt laws of good form (*Prägnanz*) have been proposed as the key to understanding perceptual organization. This eventually led to the 'minimum' principle which states that "other things being equal, that perceptual response to a stimulus will be

obtained which requires the least amount of information to specify" [75]. The minimum description length (MDL) approach developed by Risannen [162] states that we should interpret the data in terms of minimal description with respect to a prespecified model of the domain (language). Given a reasonable description language, the simplicity criteria provides an accurate and *computable* determination of perceptions which are most likely. The Bayesian method of combining information is related to the MDL method with *a priori* probabilities $P(Y)$ corresponding to the length in the description and the $P(X|Y)$ term related to the error of description [42].

In the MDL approach, the integration problem can be stated as follows: Given a reasonable description language for the object domain, a bound on the acceptable error in describing the observations, and a set of observations X_{ij}, $i = 1, \ldots, N$, $j = 1, \ldots, M$, from N vision modules at M sites, determine the shortest description (most likely) of the observations within the prescribed error bound.

Leclerc [101] applied MDL approach for obtaining a reasonable segmentation of monocular intensity images using piecewise constant and piecewise continuous models for description of surfaces and boundaries, respectively.

MDL is an intuitive approach, but it has several implementational problems. First, what is the best language for a description of the domain? Researchers have used piecewise polynomial models (descriptions) in their MDL formulations without any objective justification. Since the power of MDL approach is derived from how well the language captures the object descriptions in the domain [8], it is necessary to match the constructs of the language with the peculiarities of the domain. But, how to design such a language in an objective manner is an open research issue. The second

issue regards the choice of the metric for the information content. The conventional approach has been to use the number of bits required to represent a description as a measure of the information content. While this is a good approximation of the information content, it is not obvious as to why all the descriptors of a language should be weighted equally. For instance, if we choose Fourier descriptors as our choice of the language, then not all descriptors are equally important. Given the domain of polyhedral objects, lower frequencies are more reliable sources of information than the higher frequencies. Some non-zero components may be more common (and, therefore, less meaningful) than the others. A particular combination of parameters may be more meaningful than the sum of the information content of the individual parameters. It is also not known how to relate the information content in one vision module with that of the others. Optimization formulations using the MDL approach require relating the cost of inaccuracy due to a description and the information content of the description into a single objective measure. Since the error in description and the information content are two conceptually different entities, it is often not possible to objectively relate the error in the approximation with the information measure of the description. Finally, the solution surface of the MDL formulations has been found to exhibit several local minima. Obtaining the correct solution usually requires computationally demanding techniques [101].

2.7 Mechanistic Models

Despite severe degradations and distortions undergone by sensed data, we are still able to perceive the world in a stable manner. For this reason, we restrict our solution space only to the *stable* solutions [20]: the solutions which continuously depend on the input data.

Most early vision problems are inherently *ill-posed.* Often, we introduce 'synthetic' constraints to obtain a unique and stable solution from an ill–posed problem. This method of converting an ill-posed problem to a well-posed problem by restricting the space of acceptable solutions is called *regularization.*

Regularization usually involves enforcing synthetic constraints like smoothness of the solution (first- and second- order differentiability). One approach to formulating regularization is based on physical modeling. Physical objects resist bending, stretching, twisting, and breaking under the influence of external force(s). Their tendency to maintain a characteristic *smooth* profile is used as the synthetic constraint in energy minimization approaches involving mechanistic modeling. In this formulation, the interpretation of the sensed observations is related to the behavior of a physical 'object' under the influence of a set of external forces and the sensed data are related to the set of external forces. The object is usually in the form of a thin membrane, a plate, or a rod. The stiffness, stretchability, and shear strength of the object are used to model the relative desirability of the fit of the object to the external forces (data) and the smoothness of its configuration. Each configuration of the object defines certain *energy* depending upon the physical parameters of the object and its conformation

to the external forces. It is hypothesized that the most likely solution will emerge as the minimum energy configuration of the equivalent physical system.

Regularization alone can not guarantee physically meaningful solutions. If all the 'natural' constraints are not well represented in the formulation, the 'synthetic' constraints may produce strange results. For instance, in the regularization formulation of shape from shading problem [30], the integrability constraint is not taken into consideration. As a result, the algorithm often produces abstract (as opposed to physical) surfaces. Subsequent enforcement of the integrability constraint corrected this problem [59]. Many approaches, therefore, couple the regularization techniques with the specific geometrical and physical constraints pertaining to 3D world.

The integration problem can be formulated in the mechanistic setting in the following manner: Given the order of interaction among the objects, a set of parameters governing the relative strengths (energies) of interaction among the neighboring objects, and the sensed data from different modules, the problem of integration is to find the minimum energy (most likely) configuration as an interpretation of the sensed stimuli. Note that the power of the model lies in correctly choosing the parameters governing the interactions. These parameters assume the most likely "surfaces" in the given domain.

An example will clarify these concepts. Consider fitting an energy minimizing curve to the parameterized set of data points $\mathbf{x}(s) = (x(s), y(s))$. The energy of the fit $\mathbf{u}(s)$, E, consists of two components.

$$E = E_{ext} + E_{int}.$$

The external energy, E_{ext}, is defined as:

$$E_{ext} = \int \lambda_1 \|\mathbf{x}(s) - \mathbf{u}(s)\|^2$$

and the internal energy is defined as:

$$E_{int} = \int \lambda_2[(\mathbf{u}'(s))^2(1 - l(s)) + \alpha(s)l(s)] + \lambda_3[\mathbf{u}''(s)^2(1 - k(s)) + \beta(s)k(s)]ds,$$

where $l(s)$ and $k(s)$ are functions denoting the first-order and second-order derivative discontinuities, and $\alpha(s)$ and $\beta(s)$ are the penalties associated with these discontinuities. The parameters λ_1, λ_2, and λ_3 control the stretchability and stiffness of the curve. The external energy term enforces closeness to the data and the internal energy term enforces the smoothness properties of the curve. The resulting energy minimization could be solved using Euler's method [190].

We now mention a few examples of mechanistic and physical modeling formulation in computer vision literature. Depth from stereo and surface interpolation algorithms developed by Grimson [68] and Terzopoulos [183] are examples of mechanistic formulations. Blostein and Ahuja [22] integrate the extraction of texture elements (texels) with surface shape extraction by modeling the size change of the texel shape due to the projection process. Hoff and Ahuja [76] describe integration of depth from stereo and surface interpolation process by assuming that the surfaces in the real world are smooth. Their approach uses this constraint to mutually guide both processes in an integrated manner. Sugihara [177] describes the integration of boundary extraction

with 3D shape extraction for polyhedral objects. He uses various methods of shape extraction including shape from shading, shape from texture, and line labeling. The domain is sufficiently restricted to allow him to use surface models to complete the line drawings and subsequently use this information to construct the 3D shapes of objects. Unfortunately, his methods can not be easily generalized to more complex domains. Malik and Maydan [113] use their line labeling scheme to derive boundary conditions for object surfaces to compute shape from shading. Their method assumes perfect line drawing as an input to their line labeling module. Stockman *et al.* [176] integrate surface shape from range data and line labeling. Aloimonos and Shulman [8] describe several pairwise module integrations for obtaining depth, shape, or structure. They show that the direction of light and shape can be uniquely computed using motion and shading information. However, results are presented for synthetic images and the authors admit to stability problems related with their approach. They also describe a method of integrating texture and motion to derive shape if the motion parameters are known. Further, shape and 3D motion can be computed from contour and stereo information if the correspondences in the image pair are known.

It has been theoretically shown that under certain conditions both mechanistic and probabilistic models are equivalent [20, 179]. However, the efficacy in modeling the physical surfaces and the well-behaved resultant descriptions are the strengths of mechanistic modeling. Figure 2.7 is a case in point. Usually, first-order and second-order MRF models produce less intuitive results than the corresponding mechanistic models. However, mechanistic modeling has several shortcomings also. It is difficult to model phenomenon like transparency and statistical dependence using these

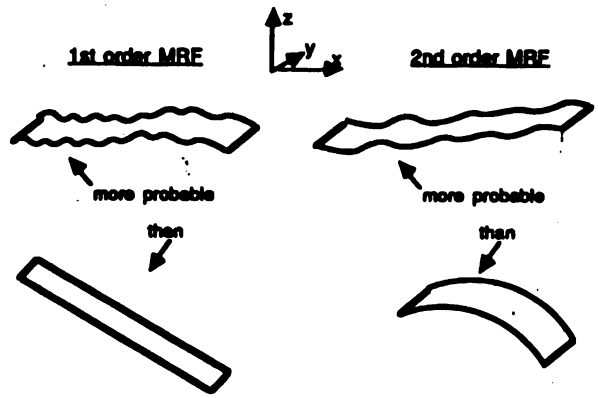


Figure 2.7: First-order and second-order MRF models produce unintuitive results (from [20]).

models. The strong assumptions underlying the mechanistic modeling produce perceptible artifacts in the resultant interpretation of the data. How will the models behave when the assumptions do not hold? Since statistical models can inherently model random variations, they are more robust than their mechanistic counterparts.

2.8 Game-theoretic Methods

Bozma and Duncan [28] have used a game-theoretic method to obtain reasonable parametric descriptions of objects in medical images. In this formulation, each player represents a vision module who can independently pursue its own objective. The interests of the players can potentially conflict (*non-cooperative game* [9]). We briefly describe some of the terminology below before presenting the formulation.

Formally, in a *game* with N players, each player is associated with a decision vector, $\mathbf{P_i}$, and a payoff function, F_i, as shown in Figure 2.8.

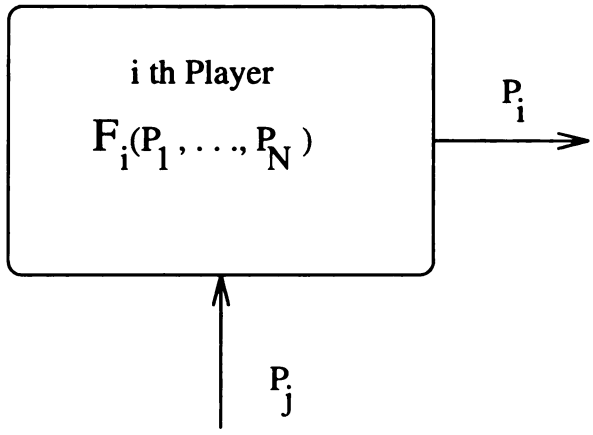


Figure 2.8: A single player.

1. At each decision epoch t of the game, player i chooses a specific decision vector $\mathbf{p_i^t}$. This is referred to as the move of the player i. The initial decision vector is assumed to be $\mathbf{p_i^0}$.

2. A *payoff function,* $F_i(\mathbf{p_1}, \ldots, \mathbf{p_{i-1}}, \mathbf{p_i}, \mathbf{p_{i+1}}, \ldots, \mathbf{p_N})$, evaluates the performance of player i based on the set of decisions made by all the players. Player i strives to maximize its payoff by modifying the decision vector, $\mathbf{p_i}$:

$$\textit{Player } i : \mathbf{p_i^\star} = \arg\max_{\mathbf{p_i}} F_i(\mathbf{p_1}, \ldots, \mathbf{p_i}, \ldots, \mathbf{p_N}).$$

The equilibrium point, $\mathbf{p^\star}$, of the game is defined in terms of the equilibrium points $\mathbf{p_i^\star}$ of the individual players: $\mathbf{p^\star} = (\mathbf{p_1^\star}, \ldots, \mathbf{p_i^\star}, \ldots, \mathbf{p_N^\star})$.

Nash Equilibrium, $\mathbf{p^\star} = (\mathbf{p_1^\star}, \ldots, \mathbf{p_i^\star}, \ldots, \mathbf{p_N^\star})$, is a set of decisions such that for player i,

$$\mathbf{p_i^\star} = \max_{\mathbf{p_i}} \mathbf{F_i}(\mathbf{p_1^\star}, \ldots, \mathbf{p_{i-1}^\star}, \mathbf{p_i^\star}, \mathbf{p_{i+1}^\star}, \ldots, \mathbf{p_N^\star}).$$

From the viewpoint of the i^{th} player, Nash equilibrium point $\mathbf{P_i^\star}$ is locally optimum.

Decision Model: Given an operating point $\mathbf{p}$, player i modifies its decision vector to maximize its payoff. A reaction map of a player is the collection of all of its optimal responses for each operating point in $(\mathbf{p_1}, \ldots, \mathbf{p_N})$.

A simple example of a 2-player game will clarify these concepts [26]. The output of player 1 is $p_1 \in R$, and the output of player 2 is $p_2 \in R$. The objective[8] of player 1 ($F_1 \in R$) is

$$F_1(p_1, p_2) = (p_1 - 1)^2 + (p_2 - 3)^2 + (p_1 - 1)(p_2 - 3).$$

The objective of the player 2 ($F_2 \in R$) is

$$F_2(p_1, p_2) = 0.75(p_2 - 1)^2 + 1.5(p_1 - 3)^2 + 0.75(p_1 - 3)(p_2 - 1).$$

The operating points for which an objective function takes a given value can be seen as a level curve of the inverse objective function. The level curves of $F_1^{-1}(c)$ and $F_2^{-1}(c)$ are shown in Figure 2.10. The $\mathbf{X}$–axis represents the decision, p_1, of player 1 and the $\mathbf{Y}$–axis represents the decision, p_2, of player 2. Each closed contour represents a set of points for which the corresponding payoff function takes a constant value. In Figure 2.11, a superposition of these two level sets and the individual reaction maps are shown. In Figure 2.12, we have shown a set of successive moves made by each player to eventually reach Nash equilibrium. It has been shown that even if

[8]Note that the use of cost function instead of payoff function converts a maximization problem into the corresponding minimization problem.

the decision of each player was updated in a parallel fashion, the Nash equilibrium is reached. The success of this formulation depends on the validity of the following three critical assumptions:

- Each objective function is locally convex in the neighborhood of each of the operating points during the evolution of the solution.
- Each objective function is C^2 with respect to the local variables.
- Each objective function is bounded from below by a finite cost.

Bozma and Duncan [28] have used non-cooperative game-theoretic methods to integrate two vision modules. Their strategy of integration allows for distributed, modular implementation. However, designing appropriate (locally convex) objective cost functions for the individual modules appears to be difficult. The bottom-up module is responsible for detecting edges; the top-down module fits parametric shapes to the detected edges. The two modules independently optimize their individual objective functions. Efficacy of their approach was demonstrated for detecting organs in fairly complex medical images.

2.9 Lattice-theoretic Methods

Jepson and Richards [89] proposed a framework for assimilation of information presented by several modules. In many situations, use of knowledge about the world for passively regularizing the sensory data is not justified. For instance, a cost function is often obtained by superposing several individual functions – each favoring a certain

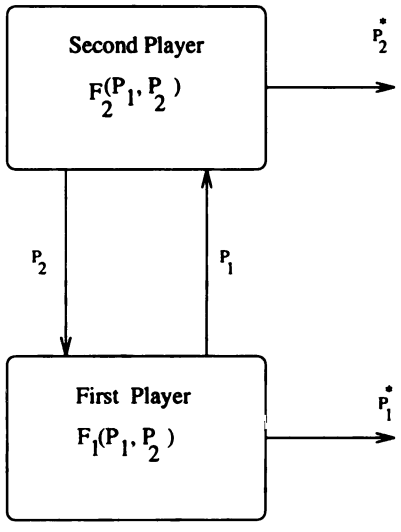


Figure 2.9: A 2–player game.

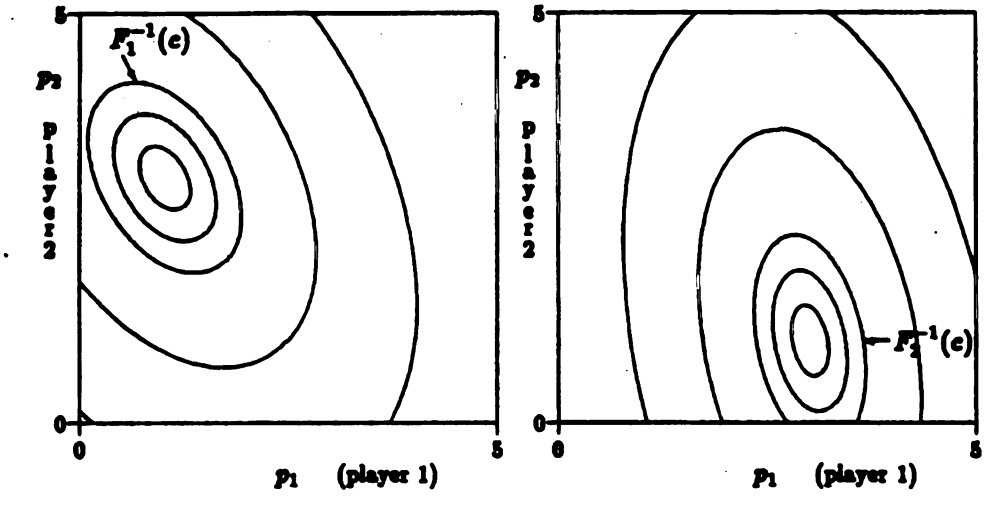


Figure 2.10: Level sets of the payoff functions for player 1 (left) and player 2 (right) [26].

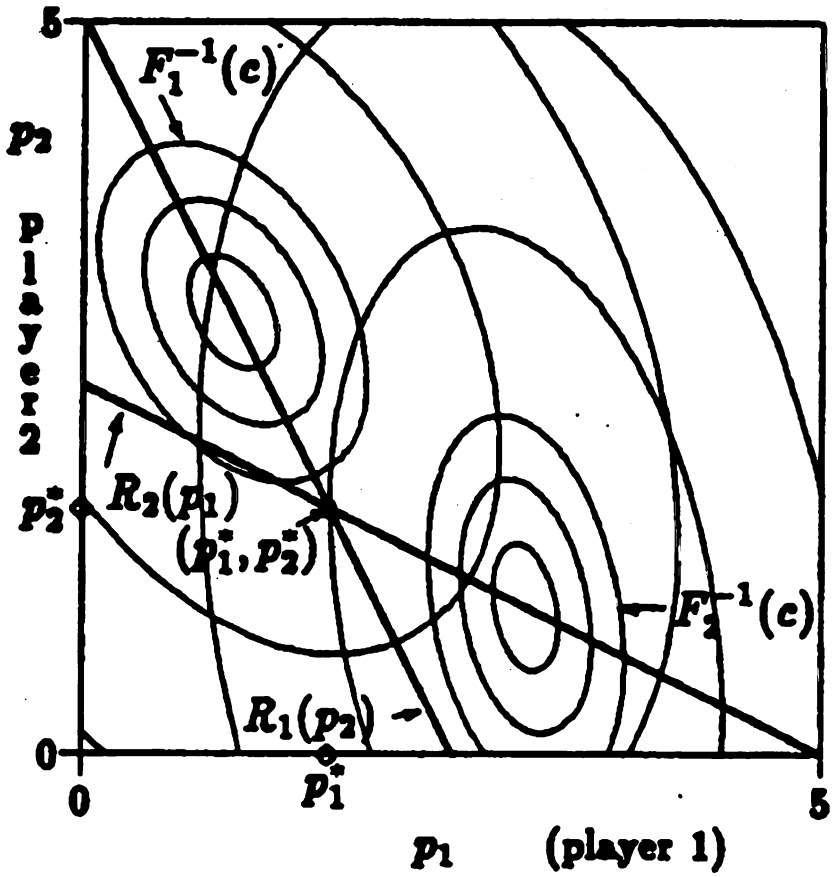


Figure 2.11: Payoff level sets for both the players are superimposed. Also shown are the reaction maps for both the players [26].

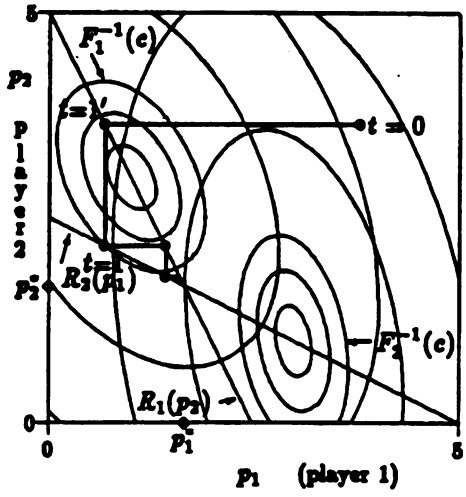


Figure 2.12: Successive moves made by the players to reach Nash equilibrium. R_1 and R_2 denote the reaction maps of player 1 and player 2. Note that player 1 always moves horizontally and player 2 always moves vertically in this graph. Also, note that in any given move, the payoff is maximized at the corresponding intercept on the reaction map [26].

desired property of the final solution. The solution obtained by minimizing such a cost function does not necessarily respect the physical reality of the world. In such situations, it is important to take into account consistency of an interpretation within the chosen world models. Use of vetoes, accumulation and votes for integration of the modular information without taking into account the underlying constraints of the modules could produce grossly erroneous results. They contend that the modules do not resolve their conflicts at the level of their outputs, but at the level of premises used to arrive at any conflicting individual interpretations.

Given a set of observations, a set of premises (assumptions made by each module), and a partial order of the assumptions, they represent the partial ordering of the premises in the form of a lattice (*fault–lattice*). Each vertex of the lattice represents a unique assertion about the validity (or the rejection) of *each* premise.

Their theory can be illustrated with the help of a concrete example [89]. Suppose an observer is watching a sequence of images of a moving 3D object being screened on a TV. The stereo module will conclude that the object is planar and the motion module will conclude that the object is three-dimensional. The resolution of this conflict can not be effected at the level of the outputs of the two modules.

Figure 2.13 shows a lattice formed by two premises D and G which are defined below.

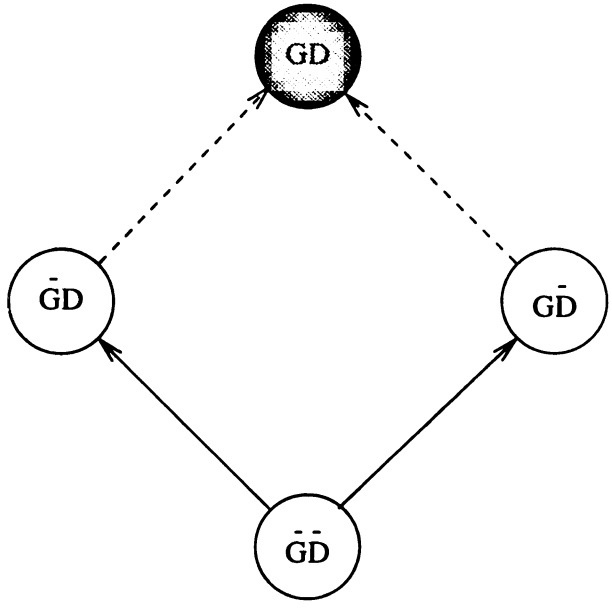


Figure 2.13: A simple fault–lattice based upon a rigid motion premise and a coplanarity premise (From [89]).

- **Coplanarity Premise (D)** is the assumption made by the stereo module that disparity represents the depth of the object. Since the television screen is flat, stereo module believes that objects depicted on a television screen are coplanar.

- **Rigidity Premise (G)** is the assumption made by the motion module that objects in the scene are rigid.

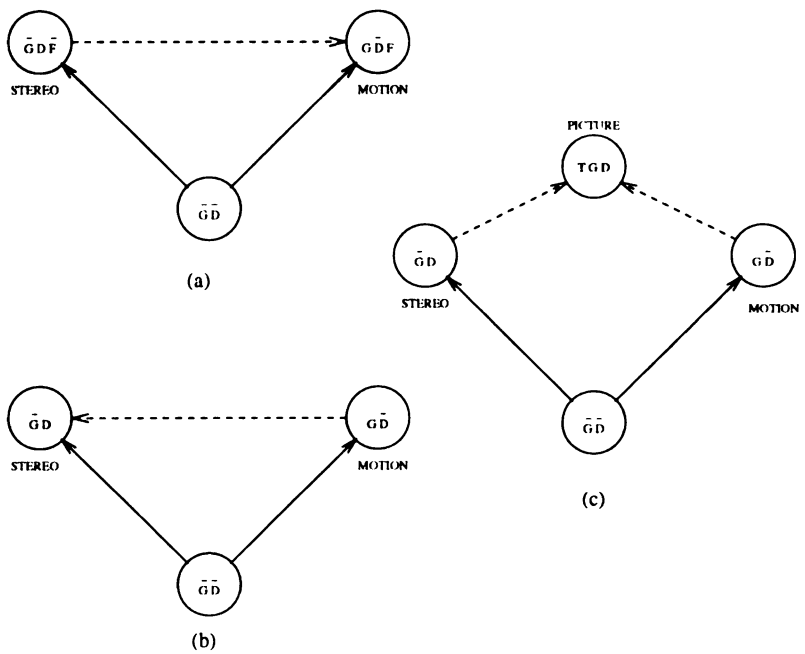


Figure 2.14: Three methods of enforcing a unique solution: (a) Voting; (b) Priors; and (c) Generation of a new hypothesis.

The vertex $\overline{G}D$ represents a situation when the premise G holds, but D does not hold. Similarly, $\overline{GD}$ means that both D and G do not hold for a given scene. Given the data and the two premises, there is no world structure which can be both rigid and coplanar. Hence the interpretation of an object being both rigid and planar is ruled out. This incompatibility of the node is depicted by cross-hatching it. At the opposite end is the node $\overline{GD}$. There are several possible non-coplanar and non-rigid 3D structures which can give rise to the given sensed data. Hence this node is valid. The remaining two nodes are the individual interpretations reached by the stereo and motion modules. Everything else being equal, we would prefer the interpretations for which the premises are satisfied over the interpretations for which they are violated. Hence, the nodes $\overline{G}D$ (stereo) and $G\overline{D}$ (motion) are the preferred interpretations

over the non-rigid and non-coplanar interpretations. This is pictorially depicted by drawing arcs from $\overline{GD}$ to $G\overline{D}$ and $\overline{G}D$[9]. Thus, two 'maximal' interpretations are possible. The issue of enforcing a unique interpretation can now be dealt in the following ways:

1. **Votes:** Here, the node which has defaulted the least number of premises is chosen as the most preferred perception. For instance, if there was a third premise F which was satisfied by motion module but violated by the stereo module, then the most preferred node would be $G\overline{D}F$ (motion) (Figure 2.14(a)).

2. **Priors:** In the absence of any hard evidence, we know that the stereo evidence is less infallible than the rigidity premise. In this case, $\overline{G}D$ (stereo) interpretation would be the most preferred interpretation (Figure 2.14(b)).

3. **Generation of a new premise:** We could generate a new premise corresponding to the situations in which conflicting evidences can be presented.

   Picture Premise (T): "given a stereo disparity consistent with coplanar surface and given other evidence about non-planar objects, the image depicts a *picture* of a 3D object. This would add a new node PGD, which is supported by both stereo and motion modules. This node will, then, become the most preferred interpretation (Figure 2.14(c)).

Jepson and Richards conclude that, under this framework, integration of vision modules emphasizes three main issues:

[9] An arc from node A to node B indicates that the interpretation offered by node B is the preferred interpretation.

1. What is the *general* set of constraints? How can we find a basis set for all the constraints/assumptions made by the vision modules?

2. What constitutes consistency?

3. What are the rules of processing which reason about consistency among the chosen premises?

Implicit in their work is the assumption that the outputs of individual modules are reliable and an accurate depiction of the real world. They assume that each module necessarily outputs 'correct' opinion about the world. This is often not true, especially in the computer implementations of vision modules.

2.10 Active Vision

Marr and Nishihara's thesis [118] hypothesized that the "higher level" processes do not themselves bear upon the formation of a *2.5D* sketch in the human visual system and that different low-level visual modules arrive at a "complete and accurate" reconstruction of the entire scene, independent of the task to be performed by the higher level processes. This hypothesis resulted in a clear dichotomy between the study of higher level processes (*recognition and navigation*) and lower level processes. A further side effect of this hypothesis has been the treatment of sensed data used for reconstruction (a low-level task) in isolation from the observers' actions and intentions (higher level tasks).

The proponents of the "active vision" school believe that both the complexity

and the brittleness of the solution to problems faced by the individual modules can be considerably reduced by making the observer sensing the world *active*. An active observer is "one who is capable of engaging in some kind of activity whose purpose is to control the parameters of sensors" [7]. By constantly interacting with the sensed data, the observer has more opportunities to dynamically adapt its sensing parameters to obtain favorable, significant, and unambiguous data. The activity of the observer may be used for (but not limited to) selecting the visual orientation and location of the sensor, and number and resolution of the frames of sensed data. Some results illustrate that the ill-posed and non-linear optimization problems can become well-posed and linear with the active observer [6]. An active observer also has the opportunity to refer to the *context* (of sensing) to further remove ambiguities.

The following example might illustrate the difference between the approaches adopted by the conventional and active vision groups. Many vision modules begin with the assumption of "general viewpoint". In any non-trivial scene, this assumption may be true most of the time but not *all* the time. Therefore, a conventional approach has to incorporate this assumption as a weak constraint and the resulting energy formulations (say) provide an energy profile with many local minima – each usually corresponding to the hypothesis of likely localizations of the violation of the 'general viewpoint' assumption. The exploration of this energy profile to detect the global minima is a difficult non-linear optimization problem. In the active vision paradigm, on the other hand, verification of the validity of the general viewpoint assumption and localization of its possible violations can be made possible by perturbations of the viewing positions or orientations. This strategy usually will result

in a considerable simplification of the subsequent processing.

Thus, the paradigm of active vision emphasizes dynamic integration of the visual cues. The information that will most effectively eliminate the ambiguity in the data will be sought and dynamically integrated. An active vision system needs a meta-level knowledge about the set of constraints, the utility of each constraint, a selection method, and an ability to manipulate the sensor parameters.

2.11 Knowledge-based Methods

It is commonly agreed that there is a *power-generality*[10] trade-off in any automation task [57, 37]. This issue concerns types of knowledge utilized in the system (including control knowledge). *Strong* methods apply task-specific inference mechanisms using domain-specific knowledge. They allow easy and feasible solutions for the problems that are too difficult to be solved by less specific approaches. *Weak* methods are general in nature and can be applied across many domains. The price paid for the generality is the cost in terms of (search) time and increased computational complexity.

Special-purpose vision systems using strong methods have shown considerable success within their limited task domains [136, 51, 84, 56]. To date, however, there have been no general-purpose vision systems that work across a variety of vision domains. Thus, the strong methods, being better able to define, structure, and apply the knowledge, realize effective and practical systems. Visual knowledge includes

[10] *Powerful* methods are less general and *vice versa*.

domain-independent knowledge about occlusion, perspective, physical support, etc., as well as the domain-specific knowledge about the objects the system is expected to encounter, their 3D structure and appearance, their expected appearance in the 2D image, and their relationship to the other objects. Control knowledge addresses how the information in the image can be efficiently extracted, organized, and matched against stored models, and when the object knowledge can be used to prune the goal-directed search tree.

Vision systems working in restricted domains can bring very specific recognition and control knowledge to bear on their assigned tasks. This allows them to perform sophisticated inferencing with relatively little computational effort.

Several knowledge-based architectures have been proposed for the integration problem, including a logical framework [160], production systems [135] and blackboard architectures [54, 189].

Several researchers have used knowledge-based methods for solving image understanding problems. Lakin *et al.* have used a blackboard architecture for data fusion in strategic naval control problems [98]. Nagao *et al.* analyzed aerial photographs using this approach [136]. Shafer *et al.* have used a distributed architecture for real-time navigation [170]. For a broader overview of the blackboard applications is vision, the reader is referred to Engelmore and Morgan [54], for instance.

Although knowledge-based approaches work effectively for high-level vision, it is our opinion that low-level visual integration tasks are less intensive in knowledge requirements. The success of the low-level integration primarily depends upon obtaining general consensus (and compatibility) among a fine-grained ensemble of unre-

liable information pivoted on a few important high-level cues. The knowledge-based approaches also need an enumeration of all the possible situations and a meticulous programming for each such anticipated situation. The integration of huge amounts of uncertain data (with their spatial dependencies) with a few high-level knowledge sources is an open issue in vision research. However, the blackboard systems have remained an attractive choice of integration architecture owing to the ease of system development and flexibility of control.

2.12 Summary

Table 2.1 presents a summary of various integration frameworks. We will now compare two of the most influential models for integration in computer vision: probabilistic and mechanistic models. There is a continuing debate on the suitability of probabilistic and mechanistic models for different computer vision applications. Here, we will discuss only a few significant issues:

1. Which approach is more suitable for modeling the world constraints? There is no definitive answer to this question. The appropriate approach to model constraints depends upon how naturally a significant set of constraints offered by the given object domain can be captured. For instance, transparency and statistical dependence of the data can be captured quite well by the probabilistic models. Probabilistic models also appear to be better equipped to handle imaging noise. The constraints of smoothness and continuity can be well captured by the mechanistic models. Nadabar and Jain [133] have found an elegant

way of combining both the approaches: they use MRF models for representing the prior constraints and mechanistic models to generate a large number of synthetic images to estimate these constraints[11].

2. How are the model-specific representations useful for further processing? For instance, the statistical models offer an unintuitive and diffuse final representations. The mechanical models offer more intuitive and compact final representations which make useful information explicit for further processing, e.g., object recognition.

3. The statistical models are inherently better suited for handling noise and outliers in the data. Hence they offer stability to the final representations. Both the models appear to be deficient in handling geometric and qualitative constraints.

4. Both the approaches entail an *ad hoc* selection of parameters. The parameter values represent relative signficance assigned to each constraint. Such a prior weighting of different information sources precludes the possibility of handling all the possible scene configurations equally well. In some applications, it is not possible to 'optimally' rank the information sources. For instance, McCafferty [124] found that different perceptual organization cues can not be ranked based on different *Gestalt* cues.

5. These models are computationally demanding, both in terms of convergence as well as in terms of propagation of the non-local constraints. Many statistical

[11]They observe that it is easier to generate images with edge continuity and smoothness using mechanistic models.

models are inherently parallelizable.

The other integration models discussed in this chapter are relatively new. While they introduce novel ways of embedding certain specific constraints, we feel that they are not sufficiently general to offer a central role in the integration architecture.

In summary, the existing models of integration appear to be incapable of extracting stable, compact, and intuitive representations from the large quantities of visual data. The limited implementations and evaluations of the existing integration frameworks appear to be lacking in their ability to capture the essence of the sensed data. The existing frameworks are brittle. We believe that effective vision systems need to incorporate hierarchical *feedback* control structures to design robust systems integrating vision modules. This issue is further motivated in Chapters 4 and 5.

Table 2.1: Integration Methodologies.

Method	Strengths	Limitations	Examples
Bayesian Fusion	Universal applicability and optimality	Difficulty in specifying priors	Sarkar and Boyer [166]
Relaxation	Simplicity	Data is less significant than the constraints	Zucker [206], Terzopoulos [183]
MRF	Captures locality and continuity	Local interactions and slow convergence	Gamble *et al.* [60], Nadabar and Jain [133]
Regularization	Captures continuity and smoothness	Where and when to suspend smoothing?	Kass *et al.* [94]
Game-theoretic	Distributed architecture	Limited applicability	Bozma and Duncan [28]
Lattice-theoretic	Intelligent selection of assumptions	Assumes accurate module behavior	Jepson and Richards [89]
Connectionist	Robust performance and learning capability	Slow convergence	Grossberg [72]
Information-theoretic	Requires fewer number of free parameters	Difficulty in selection of descriptive language	Leclerc [101]
Estimation-theoretic	Optimality	Computational complexity	Singh and Allen [173]
Active Vision	Dynamic and directed integration	Additional complexity in control	Aloimonos [7]
Knowledge-based	Tractable	Domain-dependent and noise sensitive	Brolio *et al.* [30], Pankanti *et al.* [148]

Chapter 3

Vision Modules

In Chapter 1 we described the origins of the modular processing in computer vision and illustrations about the nature of limitations of a vision system which relies solely on the output produced by a single module. In this chapter we describe in detail several modules that will be incorporated into our integrated system. Sections 3.1-3.5, describe the individual modules, their objectives, limitations, and the underlying assumptions.

3.1 Perceptual Organization (grouping)

Segmentation is one of the central problems in computer vision. Even defining the problem of segmentation is difficult since it is tightly coupled with the semantics of the image content and the visual task under consideration. It is, therefore, expedient to set up a *working* definition for the segmentation based on the photometric attributes of the intensity image [2]. By this "segmentation" we mean a grouping of the pixels

based on homogeneity of their photometric attributes [2]. Given this definition, a region-based or an edge-based operator would be sufficient to effect a perfect segmentation in an idealized situation. However, in practice, poor imaging conditions,

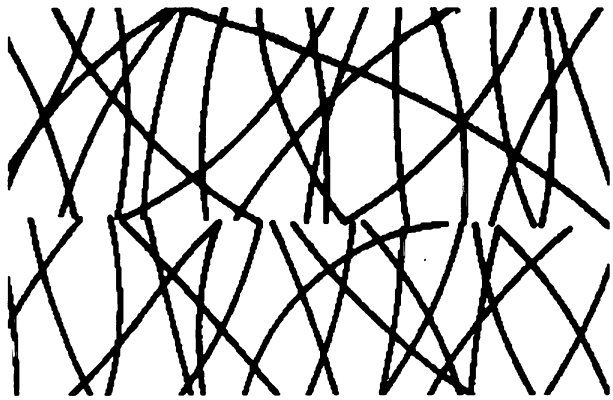


Figure 3.1: Perceptual organization helps complete the obscure boundaries [116].

insufficient contrast, noise sensitivity of the selected attributes, and artifacts inherent in segmentation operators all conspire to produce an imperfect segmentation (See Fig. 3.12). Image properties alone are not adequate in segmenting the input scene. For instance, two perceptually significant regions cannot be identified in Fig. 3.1 using intensity gradients. In such situations, the principles of perceptual organization could be invoked to obtain a reasonable segmentation.

The human visual system recognizes statistically significant relationships in a given image and uses them to infer causal structures in the scene without using any higher level domain-dependent knowledge [199]. These perceptual phenomenon were closely studied by psychologists and were accounted for in very subjective terms such as *Gestalt* and *Prägnanz*. Some of these statistically significant relations (Fig. 3.2) are collinearity, parallelism, symmetry, and connectivity. Human perceptual mechanisms

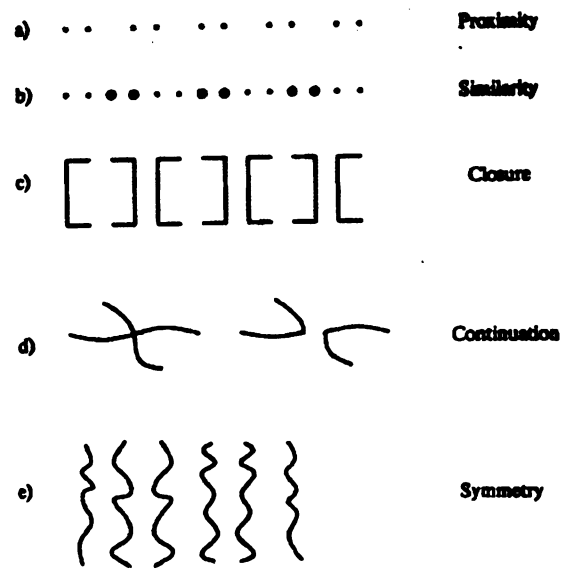


Figure 3.2: Relations significant in grouping [109].

help us organize the image features, complete the features obliterated by noise or occlusion, permit 3D inferences from the 2D image relations, and facilitate efficient indexing of the world knowledge [109]. This phenomenon of grouping is referred to as perceptual organization. Many believe that perceptual organization plays an important role in simplifying the computer vision problems [72, 155, 109].

The potential of the perceptual grouping module in computer vision systems was relatively unexplored until recently. Witkin and Tenenbaum [199] were among the first to recognize the importance of this module. Lowe [109] established a computational framework for selecting perceptually significant relations. Several other researchers have used perceptual organization, both, for 2D and 3D image features. These efforts can be broadly classified into region-based and contour-based methods. The region-based methods have used the similarity of attributes of spatially adjacent pixels to

categorize them into meaningful regions. Often, relaxation mechanisms are then used to overcome noise and to impose a reasonable consistent relationship among the neighboring pixels [207]. Hoffman and Jain's region segmentation algorithm [77] and 2D region segmentation proposed by Zucker *et al.* [207] are typical examples of this strategy. In the contour-based algorithms, edge-like features are grouped into boundaries. These boundaries are segmented into perceptually significant segments [109]. The segments may be finally grouped into objects and assemblies [166]. Kass *et al.*'s [94] 2D snakes, and Lowe's [109] methods of boundary segmentation can be considered as typical examples of contour-based groupings. Ferrie and Whaite [58] and Pankanti *et al.* [146] have used principles of Gestalt for grouping 3D boundaries. Only a few researchers have attempted to integrate the region-based and contour-based groupings [150, 124, 41].

We will now describe the perceptual organization module (See Fig. 3.3) that will be used in the integrated system (Chapters 4 and 5). The novelty of this module is that it not only takes into account the statistically significant image relationships but also the intensity gradient across a potential boundary.

Recent research in the human vision has shown that both region-based and edge-based mechanisms are disjoint at a very low-level and serve complementary functions [108, 157, 72]. These observations can be empirically corroborated by our experience with several edge-based and region-based operators. The segmentation produced by each operator has certain desirable properties, but the integration of these segmentations often leads to better results than provided by either one of them [150]. Our approach to integration (of region-based and edge-based segmentation) is carried out

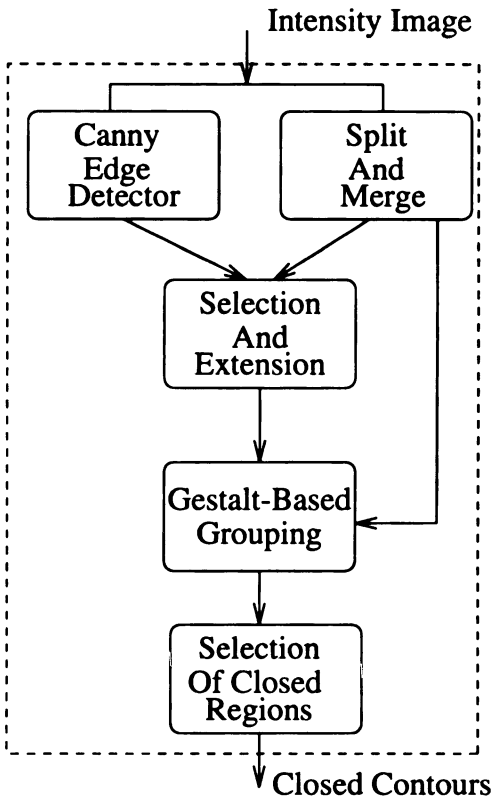


Figure 3.3: Perceptual Organization Module.

in two steps as described below. We have used outputs of Canny edge operator [35] and split and merge segmentation [149] as our edge-based and region-based segmentations, respectively. But, in general, other edge-based or region-based operators can also be used.

- Selection and Extension: In the first phase, we select a subset of boundaries which are supported by *both* the segmentations. This step is performed by searching for the boundaries of the region segmentation (RB) in a fixed rectangular neighborhood of the edges detected by the edge-based segmentation (EB). The presence of RBs in a given neighborhood of the EBs is taken as a strong evidence for the presence of a boundary (as opposed to a spurious edge) and we

select all such boundaries. These boundaries are localized at the *EB* positions[1]. The boundaries thus obtained are tangentially extended based on the presence of boundaries in region-based *or* edge-based segmentations.

- Voronoi-based Grouping: In the second phase, the boundary terminations and corners in the representation obtained from the previous stage are linked based on a set of *Gestalt* criteria (proximity, collinearity, cotermination) and the presence of a significant gradient across the linking edge in the region-based segmentation. Instead of considering all possible linking edges, only the edges connecting Voronoi neighbors are considered [3, 147]. Ideally, it would be desirable to consider the smooth snake-like extensions between the edge terminations selected for linking, but the present implementation links these terminations using a straight line segment. Following components are superimposed to obtain an objective function estimating the significance of a Voronoi edge:

  1. **Proximity.** Voronoi edges which are short will be considered perceptually more significant than those which are longer. The contribution of a Voronoi edge of length d connecting the edges of lengths D_1 and D_2 is defined as $E_p = w_p d^2 / D_1 D_2$, where w_p is the relative significance of proximity attribute.

  2. **Curvilinearity.** Voronoi edges which are in the tangential direction of the edge terminations are more significant. A Voronoi edge which subtends angles θ_1 and θ_2 with the terminations of the edges at its either end will

[1] We have found that our edge-based detector has a better localization than the region-based segmentation. Some researchers have used the maximum likelihood estimate provided by both the segmentations for this integration [41].

contribute $E_l = w_s(\theta_1 + \theta_2)/4\pi$ to the cost function, where w_s is the relative significance of the curvilinearity.

3. **Cotermination.** The coterminant is the common point shared by terminations of two (or more) smooth boundaries. Coterminations are perceptually significant. We estimate the cost of a cotermination by $E_c = w_c 2/(n_1 + n_2)$, where n_1 and n_2 are the numbers of Voronoi neighbors of each termination and w_c is the weight indicating the perceptual significance of cotermination.

4. **Gradient.** Significance of a Voronoi edge is proportional to the average intensity difference of the regions it abuts. Instead of computing a raw intensity gradient, we consider difference in means of intensity values of the adjoining regions as a reliable indicator of this criteria. The cost of the gradient is measured by $E_g = M/G$, where M is the maximum number of gray levels in the image and G is the average difference of intensity across the length of the edge.

Among the several Voronoi edge alternatives, only the minimum cost edge is chosen, provided that its total cost is below a threshold T. Finally, we choose closed boundaries to obtain regions with reasonably uniform photometric attributes. There is no systematic way of estimating the "correct" weights of each *Gestalt* component [124]. At present, we set the values of relative significance of each component of the cost function and the threshold, T, empirically. Fig. 3.4 shows results of our grouping algorithm for Mushroom and Vase image.

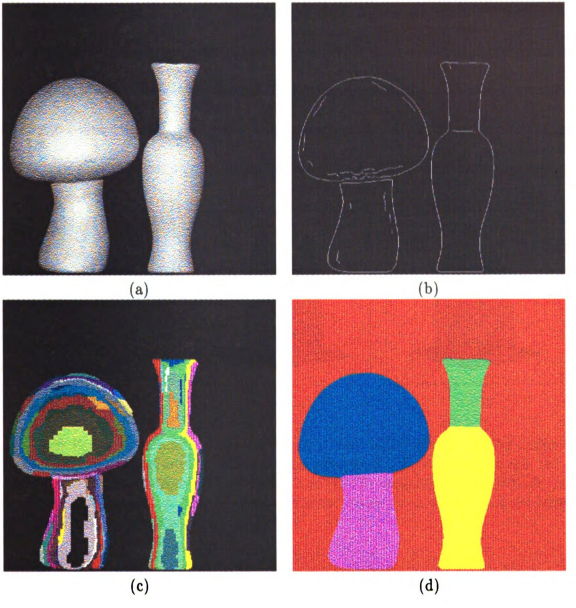


Figure 3.4: A grouping example for Mushroom and Vase image (512×512): (a) input intensity image; (b) output from Canny edge detector; (c) output of region-based segmentation; (d) significant closed regions after integrating segmentations in (b) and (c) using *Gestalt* rules.

3.2 Shape From Shading

Shading is an important source of information about the 3D structure of the input scene, especially when binocular disparity and motion cues are absent. These situations commonly occur in pictures and scenes consisting of smoothly sculpted surfaces. Artists have known the effectiveness of shading in visualization and have been using this cue to convey the 3D structure in their paintings. However, inferring shape from shading using a computer algorithm has proved to be a difficult and underdetermined

Figure 3.5: Recovery of saddle–shaped surfaces is especially difficult: (a) An image of Lambertian hyperboloid surface with constant albedo; (b) True surface shape; (c) A convex shape recovered from (a) by Oliensis and Dupuis' algorithm using *default* convex surface assumption. A default concave surface assumption would have recovered an entirely concave surface.

problem (see Fig. 3.5) and it has been generally regarded as an unreliable source of information [205].

We will now briefly describe the physics of imaging. This description will be interspersed with the imaging assumptions made by a typical shape from shading algorithm. This will be followed by a summary of the common approaches to solve shape from shading problem.

Light reflected from the surface of an object depends on the following factors:

1. **Incident light:** The amount of incident light on an object depends upon the illumination source(s), its distance from the object surface, and geometry of the object. Light energy incident on an object surface is related to the source flux by the law of inverse squares. This often gives rise to an intensity gradient across the image illuminated by a single point source. However, all the previous researchers have made the simplifying assumption that the light source is located at a sufficiently large distance (*orthographic imaging geometry*) and incident light energy does not depend upon the distance from the source. The light reflecting from object surfaces often illuminates surrounding surfaces, a phenomenon called the *mutual illumination.* Mutual illumination depends upon the geometry of the object surfaces. Modeling mutual illumination is a difficult problem which requires information about the object shape – which is usually not known. Almost all the shape from shading algorithms hypothesize that the incident light energy mainly depends upon the primary illumination and the effects of secondary (mutual) illumination can be disregarded.

2. **Surface characteristics:** A simple scene geometry is depicted in Figure 3.6. When the light energy incidents on the surface of an opaque object, a part of it is absorbed. The reflected part consists of two components: specular and diffuse.

   The specular component models reflection from mirror-like surfaces. In the case of an ideal specular surface, light rays incident on the surface are reflected

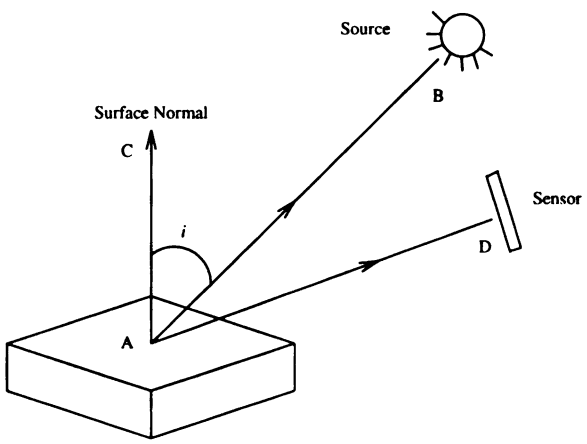


Figure 3.6: A simple image formation geometry. Rays AB, AC, and AD are in the direction of source (**s**), surface normal (**n**), and sensor (**k**), respectively. The angles BAC, BAD, CAD will be referred to as incidence angle (i), emittance angle (e), and phase angle (g), respectively [80].

such that the angle of reflection equals the angle of incidence. In a typical surface, however, specular reflections are restricted to a compact lobe around the ideal specular reflection direction. Thus, if the sensor direction changes, the amount of irradiance from a specular surface changes considerably and specular highlights shift their positions.

A matte surface, on the other hand, models Lambertian or diffuse reflections. Such reflections are a result of multiple bendings and bouncings undergone by the light rays *below* the surface microstructure. The reflected light from a perfect matte surface does not depend upon the sensor direction and depends only upon the source direction and the object surface orientation.

The specular and diffuse components of the surface reflectance can be modeled by the following expression relating image irradiance to the scene luminance

[42]:

$$E(x,y) = I(x,y)(R_s(\mathbf{n}) + R_l(x,y)),$$

where R_s is specular reflectance map, R_l is Lambertian reflectance map, $E(x,y)$ is image irradiance, and $I(x,y)$ is the incident irradiance.

The specular reflectance can be modeled as:

$$R_s(\mathbf{n}) = (\mathbf{k}.(2(\mathbf{n}(x,y).\mathbf{s})\mathbf{n}(x,y) - \mathbf{s}))^m,$$

where m is a parameter related to the sharpness of specularity of a given surface and '.' denotes the "dot" product.

The Lambertian component is modeled by:

$$R_l(x,y) = (\mathbf{n}(x,y).\mathbf{s}).$$

Typically, the shape from shading research has been restricted to the pure Lambertian surface models. Incorporating specular reflections in the shape from shading formulation is a relatively recent phenomenon [42]. Researchers have used *a priori* decomposition of the surface image irradiance into specular and diffuse reflectances using dichromatic reflectance models [73] or using empirical methods devised by Wolff and Boult [201]. Others have attempted to estimate these components from the images using optimization methods [42].

3. **Sensor characteristics:** It is generally assumed that the lens is isotropic and

responds equally well to the objects in peripheral as well as central regions in the image. These situations are, in most cases, exceptions rather than a rule. Everything else remaining the same, two locations in the image may have different intensities due to optical distortions of the lens system or non-uniform sensitivity of the sensor array. In addition, the inverse square law may be effective if the width of the sensor array is comparable to the average distance to the object surfaces being imaged (wide-angled lens).

Given an imaging model and a single intensity image, a shape from shading algorithm needs to reliably estimate the geometry of the sensed surfaces. First of all, due to the ubiquitous assumption of an orthographic projection, we can not determine absolute depths from shading cues alone (even when the reflectance map is completely specified). Hence, all the shape from shading problems should be interpreted in terms of recovery of surface normal information. The difficulty in recovering surface orientation from shading (intensity) cues is due to a lack of a sufficient number of constraints: we have only one (intensity) value at each location (pixel) while we are expected to recover two parameters (tilt and slant) per location. In principle, it is possible to obtain an infinite number of solutions if we hypothesize that the surface normals at the neighboring locations are independent. In order to recover physically meaningful and unique surfaces, additional constraints are necessary.

One approach to obtaining a unique solution to the shape from shading problem is by imposing a smoothness constraint and incorporating it into a regularization framework. If the boundary conditions are known, Blake *et al.* [21] have derived

some theoretical results stating uniqueness of the solution under restricted situations. In practice, these algorithms begin with known occluding boundaries and formulate shape from shading as a variational problem with known boundary conditions [30, 85]; the solution typically involves regularization. There are several difficulties with this formulation. For instance, correct recovery of spherical surfaces is difficult [205]. The commonly used smoothness constraint might result in unintuitive results. Frankot and Chellappa [59] have solved this problem by projecting each intermediate solution on to the frequency domain representation to enforce the integrability condition. The regularization approach is computationally intensive; it typically requires several thousand iterations before obtaining a reasonable solution.

Another approach to obtaining shape from shading restricts the surface geometry to provide local solutions. For instance, Pentland [152] assumes the surfaces to be locally spherical. This requires recovering only a single parameter from each surface location. Tsai and Shah [192] offer an iterative solution by assuming a linear approximation of the reflectance function. While these strategies are computationally adequate, the restrictive assumptions make it difficult to improve upon the initial results incrementally. Thus, this approach is a poor choice for an integration environment.

Oliensis and Dupuis [144] have shown several counterexamples to the widely believed claim that surfaces are uniquely constrained by limb (occluding contours) edges. They, instead, show that shape from shading problem is well-constrained only in the presence of singular points – pixels with *maximal* brightness. They propose a novel, computationally attractive algorithm based on the method of *characteristic strips*

[79]. We have adapted Oliensis and Dupuis' algorithm for our module integration research. The characteristic strips are the curves of the steepest ascent in the direction away from the light source. Given the depths at singular points, Oliensis and Dupuis [145, 144] developed a noise-resistant method for reconstructing the characteristic strips. This idea is based on an elegant method of propagating the depth values at singular points to the rest of the locations in the image. If relative albedo at each point on the image surface is known, then this concept can be easily extended to arbitrary images with Lambertian surfaces.

3.3 Stereo

In humans, binocular stereo is a robust estimator of depth at an acute visual angle, particularly in the image regions with a significant variation in intensity. A stereo module is known to produce an erroneous depth map in regions of the image with no texture or with very little texture [195]. It is also difficult to compute depth at those parts of the scene which are visible only from a single camera.

A simplifying assumption which is frequently made in the stereo matching algorithms is called the *parallel axis geometry* (Fig. 3.7). Consider a point P in 3D space (Figure 3.7). Let us image this point from two known camera positions defined by their optical centers, O_l and O_r. If we can identify the locations A and B, of the point P in the two images taken from two known positions, then the 3D location of P can be recovered from the *disparity* of the *corresponding* image points using the principle of triangulation. Establishing the correspondence of all the points in the

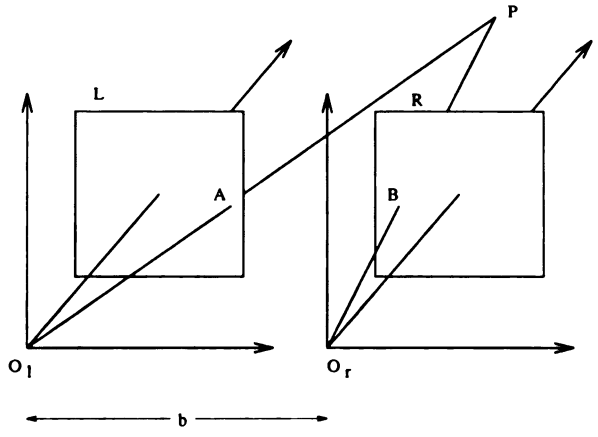


Figure 3.7: Parallel axis stereo geometry [50].

image pair is called the *correspondence* problem. Given the imaging geometry and the correspondence of physical points in the stereo image pair, computation of the depth from the disparity is relatively straightforward. There are two primary categories of stereo algorithms: (i) Pixel-based: algorithms which establish correspondence for every pixel in the stereo images; and (ii) Feature-based: algorithms which establish correspondence for only selective points, called *feature* points. The depth for the remaining points is determined by smooth interpolation.

Pixel-based Stereo:

There are two strong assumptions underlying the pixel-based stereo algorithms.

- Lambertian Surfaces: It is hypothesized that the intensities at the corresponding pixels are identical under the camera transformation. If we assume that the optical axes of the two cameras are parallel, then this assumption is equivalent to assuming that the object surfaces are Lambertian.

- Sufficient Saliency: It is also assumed that each pixel has sufficiently distinctive

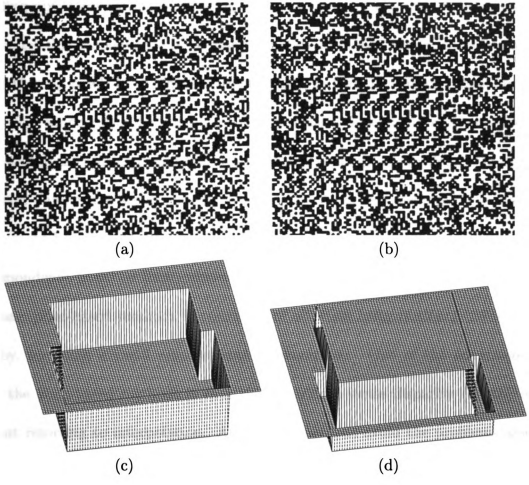


Figure 3.8: Stereogram of ambiguously perceivable center square flanked with unambiguous areas in front of and behind the surround [91]. (a),(b) are left and right random dot stereograms; (c), (d) are two perceived depth maps.

spatial context to eliminate potential false matches in establishing the correct correspondence.

In practice, due to noise in the imaging process, the gray level values at the corresponding pixels are not identical; such a property is valid only in a statistical sense. To counter this problem, the pixel-based methods often use correlation between sets of neighboring pixels in the left and right images to find the correct correspondence [129]. The window used for computing the correlation depends upon the noise characteristics of the sensing conditions. However, this strategy is not effective in dealing with non-Lambertian scene surfaces. The *sufficient saliency* assumption is violated if the image has only slowly varying intensities – a lack of distinctive features makes correspondence based on local, window-based properties difficult. The search region for these pixel-based methods is restricted to alleviate computational burden and, thereby, imposing a small relative motion assumption. One of the advantages of using the pixel-based methods is that they provide a dense disparity (depth) map without resorting to interpolation. Also, these methods avoid the problems associated with the feature-extractors and their artifacts.

Feature-based Stereo:

The feature-based algorithms require a selection of viewpoint invariant and noise-resistant feature points. Since feature points are sufficiently distinctive, finding correspondence in a feature-based stereo algorithm is relatively easy. However, these algorithms result in a sparse disparity map and estimating the entire depth map for sparse disparity map is a difficult problem (*interpolation problem*).

What feature points are desirable for establishing correspondence? Several candidates, including zero-crossings of LoG operator [117], oriented step-edges [126], peaks of LoG operator [123], and frequency domain-based descriptors [195] have been proposed in the literature. Many of them are justified on the basis of either psychological studies or computational arguments. Some less general features like line segments have also been used to simplify the correspondence problem [126].

Both pixel-based and feature-based approaches implicitly assume that the more similar the attributes of a given pair of pixels (features) are from the left and right stereo images, the more likely they represent the correct correspondence. At limb boundaries or due to specular highlights and occlusion, this hypothesis is not true (and can be misleading). Although pixels near occluding limb edges in stereo images have similar intensity profiles, they are images of different physical parts of the scene. The specular highlights depend upon the viewpoint and the "corresponding" highlights might mislead matching based on similarity measures. Finally, it is meaningless to obtain the correspondence based on a given similarity measure for the parts of a scene which are visible in only one of the stereo image pair. In addition, the stereo constraints alone are not sufficient in constraining the false correspondence problem. Use of relaxation or other optimization [195] schemes is often necessary to impose additional constraints to alleviate this problem. Marr's computational theory prescribed that the assumption of cohesive matter and unique correspondence should be used for disambiguation. If the objects in the scene are cohesive, then the resultant disparities should vary smoothly *almost* everywhere. The principle of unique correspondence states that each feature may have only one correspondence. Mayhew

and Frisby [123] observed that smoothing the disparities across the depth boundaries can be avoided by use of 'figural continuity'; the disparities *along* a boundary should vary smoothly. Prazdny argued that 'coherence' is a more general principle for disambiguating matches [154].

There are several difficulties in interpolating depth between feature points. Many interpolation methods use an implicit smoothing function for estimating the depth at pixels between feature points. The rationale of these methods is based on the argument that absence of feature points at these intermediate pixels indicates absence of sharp depth discontinuities (*no-news principle* [68]). Naive implementations of these methods might smooth the sharp depth discontinuities, or produce noisy depth map. Blake *et al.* [20] have suggested a discontinuity-preserving smoothing method based on regularization. Hoff and Ahuja [76] have proposed an integration scheme which combines matching and integration.

The local matching schemes can not take into account large disparity ranges. Hierarchical schemes have been suggested in the literature for handling large disparity range. Here, the disparities obtained from matching feature points at a coarse level guide the matching process at the next finer resolution [116, 195].

In our integrated systems (Chapters 4 and 5), we use the multi-resolution stereo matcher proposed by Weng *et al.* [195]. This algorithm matches four attributes of the intensity images which are reasonably insensitive to the relative camera motion. These attributes are the (smoothed) image intensity, magnitude of the image gradient, and "positive" and "negative" curvatures. These four features grossly correspond to functionals of zeroth-, first-, and second- order derivatives of the original image

function tailored for the stability in terms of relative camera motion. The matcher also imposes intra- and inter- regional smoothness constraints on the disparities. The main idea underlying the matching algorithm is as follows: If the attributes of input stereo images are insensitive to the relative motion of the camera, then the correct correspondence implies that the dissimilarity (*residuals*) in the corresponding attribute values should be minimum. Given an estimate of the correspondence vector field, the stereo matcher at a given level of resolution obtains a refinement of the correspondence vector field by minimizing a weighted sum of squared residuals using the gradient descent method.

$$\min_{\mathbf{d}} \sum_{\mathbf{u}} \sum_{i} \mathbf{w}_i [\mathbf{R}_i(\mathbf{u}, \mathbf{d})]^2, \tag{3.1}$$

where $\mathbf{d}$ is the correspondence vector field, $\mathbf{R_i(u, d)}$ is the residual contributed by the i^{th} attribute image at location $\mathbf{u}$ due to correspondence vector $\mathbf{d}$, and $\mathbf{w_i}$ is a prespecified weight associated with the residual $\mathbf{R_i}$.

The stereo system starts at the coarsest level of resolution and the final estimate of the disparity obtained at any level guides the matcher at the next finer level. In regions with small changes in intensity, there is not sufficient information available for a gradient-descent technique to drive the disparities in the correct direction. Fig. 3.9 illustrates this limitation of the stereo algorithm.

3.4 Line Labeling

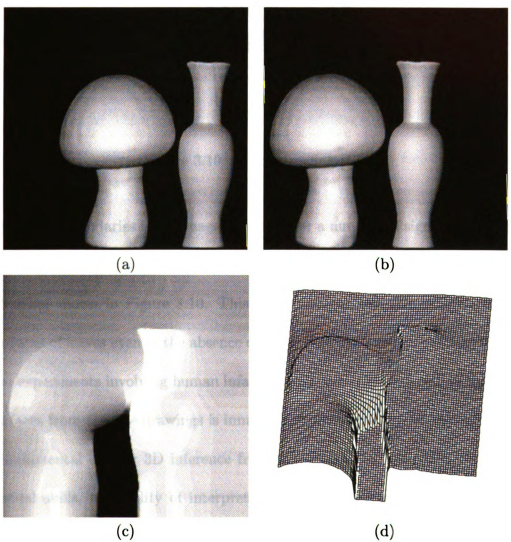


Figure 3.9: Mushroom and Vase image (size 512×512): (a), (b) Left and right stereo images; (c) Recovered depth from stereo alone; (d) A wire-frame representation of (c).

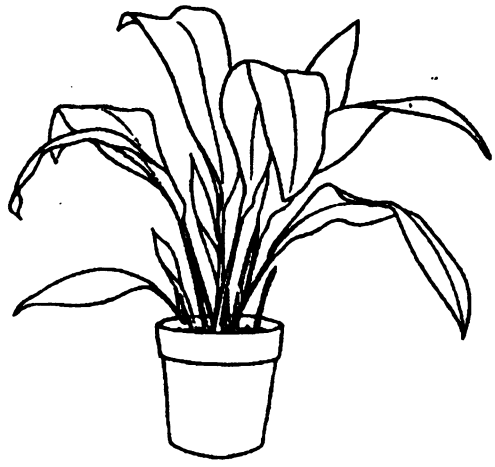


Figure 3.10: A line drawing [13].

Given the boundaries in an image, we can infer a number of significant properties of the surfaces giving rise to these boundaries. For example, take the case of the binary line drawing shown in Figure 3.10. This figure conveys valuable information about the surfaces of leaves even in the absence of other cues like stereo, shading, or texture. Several experiments involving human infants have shown that our skill in interpreting 3D surfaces from the line drawings is innate [109]. It is now agreed that line labeling is a fundamental cue for 3D inference from 2D images [13, 14]. Like other human perceptual skills, the ability of interpreting line drawings is based on assumptions of general viewpoint and detection of statistically significant properties, independent of any higher level and domain-dependent knowledge. The line labeling module is unique in that it exploits local geometric constraints and conveys its inferences in a qualitative way. The labeled line diagram can be used for scene interpretation, object recognition, and matching [142, 45, 13, 100, 171].

In a given image, a boundary detected by an edge detector could be a result of

a number of different physical events such as discontinuity in illumination (shadow), change in surface albedo, surface markings, discontinuity in surface orientation, discontinuity in depth, and self occlusion. Before a line drawing can be interpreted, it needs to be parsed into a graph representation. The vertices of this graph represent intersections (junctions) of the boundaries and arcs of this graph represent the individual boundary segments between two junctions. Given a graph representation of a correct line drawing, the objective of the line interpretation module is to obtain an approximate and qualitative 2.5D sketch of the input scene by appropriately categorizing each arc in the graph. The architecture of line drawing interpretation module consists of the following three components:

1. Line Labeling: The line labeling module labels the boundaries into several different categories. The correct line labels of only a few boundaries can be initially determined.

2. Junction Labeling: Boundary intersections in an image result in junctions which are 2D depictions of the 3D surface intersections. The junctions can often be categorized into a few qualitative categories based on the degree of the intersection and the angles between the boundaries involved. The correct labels of only a few junctions can be initially determined.

3. Relaxation: The 3D world exerts strong constraints on the compatibility of the neighboring junction labels and line labels. However, these constraints alone are typically inadequate to obtain a unique solution; a unique line interpretation is usually imposed on a line drawing using constraints based on perceptual

mechanisms [103, 191].

Several efforts in interpreting line drawings have been made. The initial line drawing interpretation for polyhedral objects was presented by Huffman and Clowes [43], which was further augmented by Waltz [194]. Sugihara [177] derived a linear programming type algorithm for labeling line diagrams of polyhedral objects. Kanade [92] extended the object domain to include *origami* objects – objects composed of very thin surfaces. Chakravarty [36] extended the original polyhedral object domain to include curved objects. A mathematically rigorous and minimal junction catalog was provided by Malik [112] for orthographic views of 3D scenes containing C^3 surfaces. This was later extended to perspective projection by Nalwa [139]. Leclerc and Fischler [103] have formulated the problem of line drawing interpretation of wireframe objects using minimum description length (MDL) approach.

Malik [112] classified lines into depth and orientation discontinuities. The depth discontinuities are further classified into limb and non-limb categories. The orientation discontinuities are further classified into convex and concave boundaries. Malik's algorithm does not allow surface markings, shadow edges, and albedo edges. His junction catalog is shown in Figure 3.11. When a unique solution is not possible, Malik's algorithm imposes a uniqueness constraint by preferring an interpretation with a minimum number of faces. Trytten's algorithm [189] is a refinement of the Malik's algorithm in that it imposes uniqueness by preferring "floating object interpretation". It also has a different control structure to facilitate integration with other modules. Since the assumptions of orthographic projection geometry and C^3 surfaces are not

too restrictive in practice, we prefer to use Trytten's refined method of line labeling for our non-uniform integration scheme.

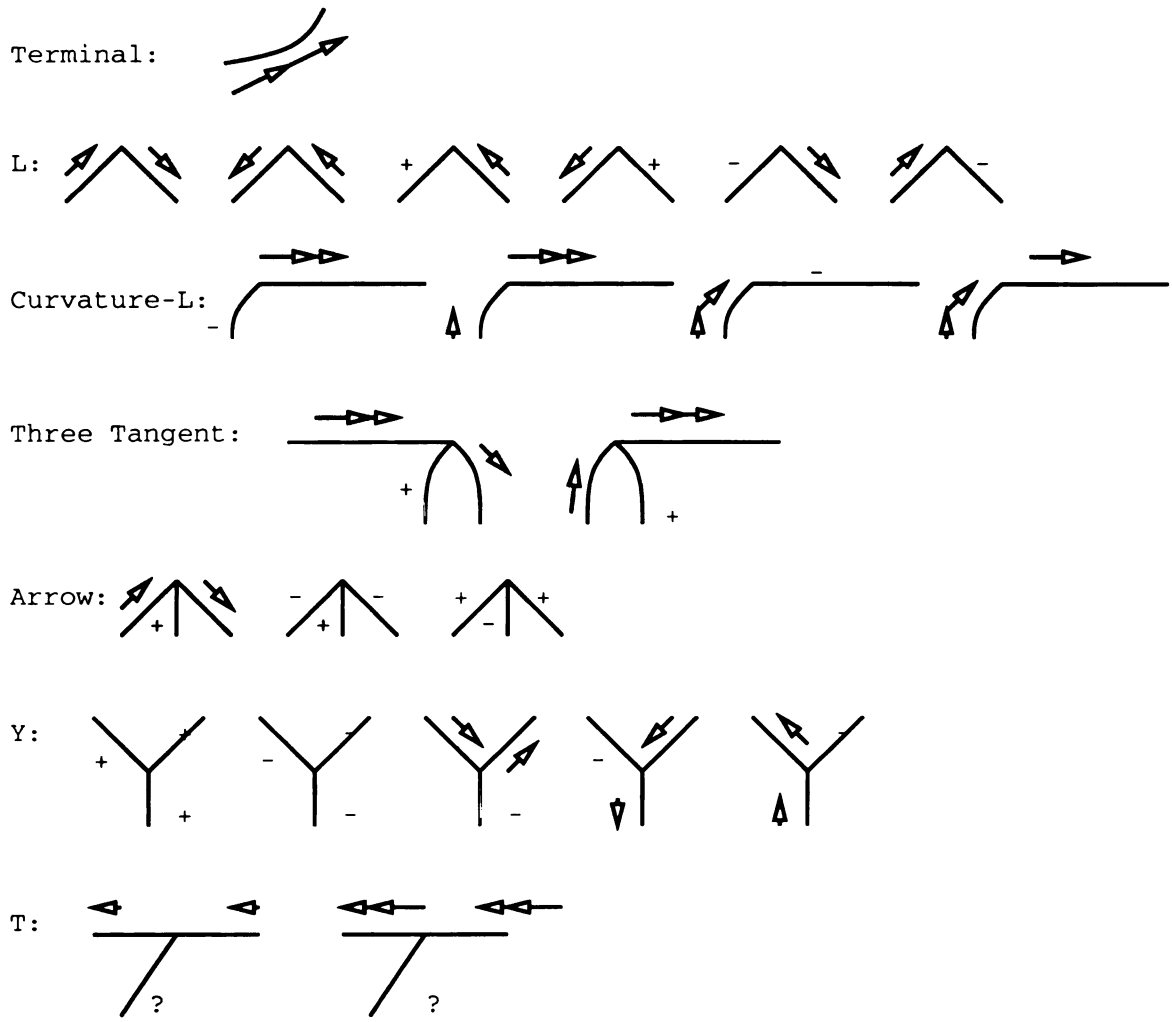
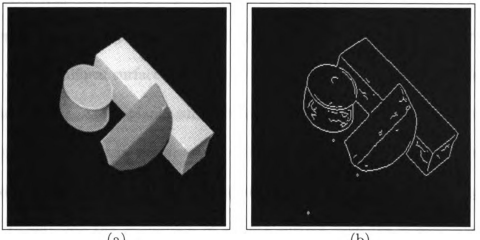


Figure 3.11: Malik's junction catalog. The double arrows indicate limb edges. The object boundaries are shown by single arrows. Symbols '+' and '−' denote convex and concave (internal) edges. Symbol '?' denotes *a don't care* line label [112].

Although the line labeling problem has been rigorously studied for a limited object domain, it is plagued with innumerable implementation problems. Most reported work on the line labeling problem assume availability of a perfect line diagram at the outset. Obtaining a good line diagram appears to be a difficult problem (Fig. 3.12).



(a) (b)

Figure 3.12: Obtaining line diagrams is a difficult problem even in case of simple scenes: (a) An image from the blocks world; (b) A typical edge map using Canny edge detector. Note that the edge detector failed to detect many junctions and introduced several extraneous edges.

Further, the parsing of a line diagram into vertices and arcs has been shown to be extraordinarily tricky [139]. Many researchers wonder whether an ideal line diagram and its accurate parsing could ever be produced in a practical situation. Even when the line diagram is correctly parsed, the line labeling module may label it erroneously. For instance, it often mistakes an L-junction for a curvature-L junction and T-junction for a 3-tangent junction. Fig. 3.13 illustrates the difficulty faced by a line labeling module in labeling a curvature-L junction.

3.5 Shape From Texture

The term texture defies a formal definition [106]. The simplest types of texture can be characterized by the 'regularity' in placement of a texture element (texel). In an image of a 3D surface, the spatial distribution of surface markings is distorted.

This distortion systematically depends upon the shape and orientation of the surface with respect to the imaging geometry (Figure 3.14). For instance, consider a plane covered with a uniform surface texture; the spatial relationship among texels on this surface is uniform. As the planar surface is tilted, the number of texels per unit image area increases the most in the direction of slant (texture gradient). The magnitude of the compression depends upon the amount of tilt. Shape from texture algorithms attempt to estimate the orientation of a scene surface from the measurement(s) on the spatial distribution of the surface markings.

Notation and Imaging Geometry: Following Garding [61], we will a assume perspective spherical projection (Fig. 3.15). The optical imaging projects the events present on the 3D surface $\mathbf{S}$ on to the unit viewsphere Σ around the focal point. The angle subtended by the surface normal $\mathbf{N}$ to the visual ray $\mathbf{p}$ will be called *slant* (σ) and the angle subtended by the projection of the surface normal (on to the tangent plane) to the reference X-axis is referred to as *tilt* direction, $\mathbf{t}$. Together, slant and tilt uniquely determine the orientation of the surface normal $\mathbf{n}$. The tilt direction $\mathbf{t}$ and direction orthogonal to tilt direction, $\mathbf{b}$ ($= \mathbf{n} \times \mathbf{t}$) in the tangent plane form a natural local coordinate frame for description of distortion due to the imaging geometry.

The linear backprojection map, $F_\star$, projecting from tangent plane of $\mathbf{\Sigma}$ to tangent plane of $\mathbf{S}$ is assumed to be sufficiently well-behaved. This map can be tersely expressed in the bases ($\mathbf{t}$, $\mathbf{b}$) and ($\mathbf{T}$, $\mathbf{B}$) [61] as an affine transform, A, composed of three generic components [61], scale, rotation, and shear:

$$A = S(s)R(\theta)D(\alpha, \mu),$$

where S specifies scaling parameterized by a scaling factor:

$$S(s) = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix};$$

$R(\theta)$ denotes rotation by an angle θ:

$$R(\theta) = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix};$$

and $D(\alpha, \mu)$ represents an area-preserving shear distortion whose axis and magnitude are determined by μ and α:

$$\begin{aligned} D(\alpha,\mu) &= R(\mu)M(\alpha)R^T(\mu) \\ &= \begin{bmatrix} cos\mu & -sin\mu \\ sin\mu & cos\mu \end{bmatrix} \begin{bmatrix} \alpha & 0 \\ 0 & \frac{1}{\alpha} \end{bmatrix} \begin{bmatrix} cos\mu & sin\mu \\ -sin\mu & cos\mu \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\alpha}sin^2\mu + \alpha cos^2\mu & (\alpha - \frac{1}{\alpha})sin\mu cos\mu \\ (\alpha - \frac{1}{\alpha})sin\mu cos\mu & \frac{1}{\alpha}sin^2\mu + \alpha cos^2\mu \end{bmatrix}. \end{aligned}$$

If we exclude isotropic rotation component R, then A can be expressed as

$$F_\star = \begin{pmatrix} r/\cos\sigma & 0 \\ 0 & r \end{pmatrix} \equiv \begin{pmatrix} 1/m & 0 \\ 0 & 1/M \end{pmatrix}, \tag{3.2}$$

where r is the radial distance of the point on the surface under consideration. M and m can be visualized as the half major and minor axes of the ellipse in Σ resulting

from imaging a unit circle in S.

Having realized that the absolute measurement of the distortion $F_\star$ is difficult in reality, vision researchers have focused on the measurement of spatial variation in the distortion map as a basis for recovering shape from texture. These different measures are collectively referred to as *texture gradients*. The following four texture gradients have been commonly used: density, area, perspective, and scale. The density and area gradients refer to the spatial variation in the texel densities and areas, respectively. Perspective gradient refers to the variation in the linear dimension of the texels in the direction of the surface tilt and scale gradient is the similar measurement taken in the direction orthogonal to the direction of the surface tilt.

The exact nature of the surface markings on the scene surfaces is not generally known. Therefore, constraints imposed by the projective geometry alone are not sufficient. Additional constraints in the form of assumptions about the texture are necessary. The most commonly used assumptions are: (i) uniform texture over all the surfaces in the input scene, and (ii) orthographic imaging geometry. If the texels can be reliably detected in the image, then it is possible to compute the gradient of texture density. However, the texel identification problem is difficult due to imaging noise, edge detection artifacts, and occlusion. Ahuja and Blostein [22] have integrated texel identification with the estimation of the surface orientation to alleviate the problem of erroneous preprocessing. Witkin [198] posed the shape from texture problem under *regular projective geometry*, isotropy, and independence assumptions. Under these assumptions, the maximal variation in the data is accounted for by the projective geometry. He plotted the histogram of the orientation of the edge elements and

showed that under orthographic imaging geometry, the amplitude and 'phase' of this histogram are related to the orientation of the 3D surface. One significant advantage of this scheme is that it does not require extraction and identification of the texels.

We use the shape from texture algorithm proposed by Super and Bovik [178]. Their algorithm consists of the following three steps:

1. **Estimation of instantaneous frequency:** In this step, a set of Gabor filters (and their derivatives) are used to estimate the local frequency content at each pixel.

2. **Computation of invariant moments:** A second-order moment matrix is computed at each pixel using the local spatial frequency spectrograms. A pair of coordinate frame invariant moments (M, m) can then be computed in the directions (θ, θ') of the first two eigenvectors of the second-order moment matrix.

3. **Estimation of surface normals:** Given invariant moments (M_p, m_p) at a pixel $\mathbf{p}(x_1, y_1)$ and invariant moments (M_q, m_q) and slant σ_q at pixel $\mathbf{q}(x_q, y_q)$, the slant σ_p at $\mathbf{p}$ can be computed using

$$\cos\sigma_p = \cos\sigma_q \sqrt{\frac{M_q m_q}{M_p m_p}}. \tag{3.3}$$

The tilt τ_p can be computed using the following equations:

$$\tau_p = \begin{cases} \theta_q - \theta_p \pm \frac{1}{2}\arccos\lambda_p, \\ \theta_q - \theta_p \pm \frac{1}{2}\arccos\lambda_p + \pi, \end{cases} \tag{3.4}$$

where θ_p, θ_q are the orientations associated with M_p, m_p, respectively, and λ_p is defined by:

$$\lambda_p = \frac{(\cos^2 \sigma_p + 1)(M_p + m_p) - 2(M_q + m_q)}{\sin^2 \sigma_p (M_p - m_p)}, \tag{3.5}$$

provided that $\sigma_p(M_p - m_p)$ is not zero.

Figure 3.16 illustrates a typical quantitative problems in estimating surface orientation near limb edges using texture information. Note that these systemic inaccuracies were observed even after the perfect texture segmentation and scale information was made available to the algorithm.

3.6 Summary

Vision researchers continue to propose, develop, and demonstrate novel algorithms for shape-from-X modules. There are several limitations in the performance of the individual modules:

1. **Parameters** Like most computer vision algorithms, a shape-from-X module involves various parameters and its performance is often critically sensitive to its *operating region* in the parameter space. The individual modules are unreliable since the appropriate values of the parameters (i) can not be easily selected; (ii) remain unstable over the domain of the images; and (iii) can not be, even qualitatively, related to the tangible characteristics of the image domain. Utility of most of the shape-from-X algorithms in practical situations is not obvious due

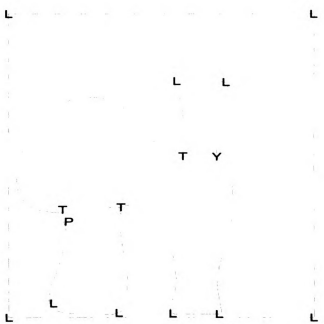
to a lack of studies on the relationship between performance of the algorithm and its operating region.

2. **Ambiguities** Often the individual modules can not completely constrain the object surfaces from the input image(s).

3. **Assumptions** Different assumptions are often built into vision modules. The synthetic constraints prove to be too restrictive for real images. The violations of natural constraints can not be reliably detected.

All these factors make it difficult to rely on the individual modules for obtaining robust performance. How these individually fragile modules can be integrated to obtain better performance is the topic of the next two chapters.

(a)



(b)

Figure 3.13: Junction labeling example for Mushroom and Vase image (Fig. 3.9(b)): (a) correct input line diagram; (b) labeling using line diagram alone; L, C, T, Y, A, and P denote L, curvature-L, tangent, Y, arrow, and phantom junctions, respectively.

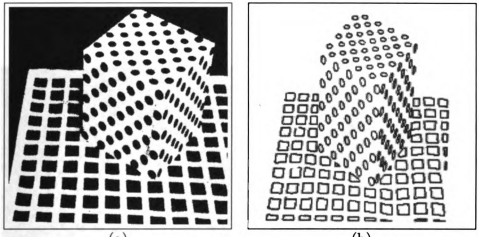


Figure 3.14: Shape from texture example: (a) image of an object with uniform surface texture; (b) Edge map (scaled) extracted from the image shown in (a). Both pictures adapted from [106].

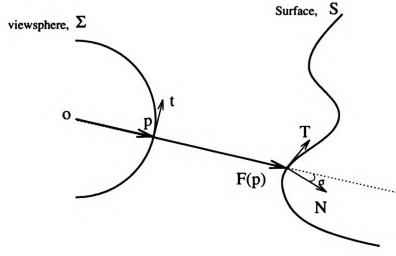


Figure 3.15: Imaging Geometry (adapted from [61]).

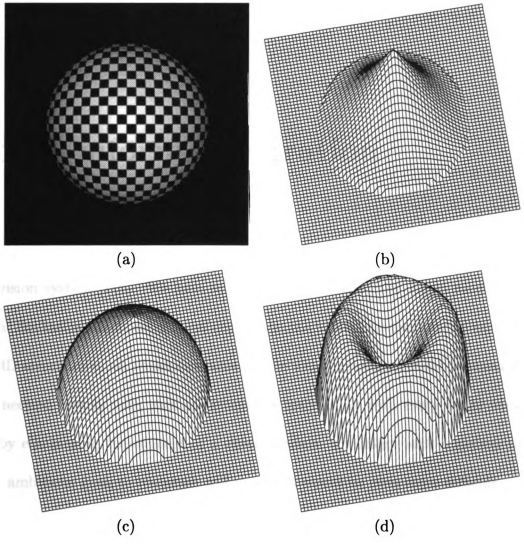


Figure 3.16: Shape from texture example: (a) a synthetic texture-mapped image of an object with uniform surface texture (10% *i.i.d.* noise and Lambertian shading); (b) recovered slant field (here we show 90° complement of slant for better visibility) from (a) using a shape from texture algorithm [178]; (c) ground truth slant field for the object in (a); and (d) normalized error in the recovered slant.

Chapter 4

Non-Uniform Integration

The shape-from-X modules are motivated by Marr's paradigm of modular design of a vision system [116]. We have seen in Chapter 1 that the vision modules have several limitations and one can not depend on the individual modules to extract the true 3D structure in all situations. Ideally, the vision processes could be modeled as a connection matrix with each state variable interacting with every other variable, thereby eliminating the concept of a vision module. To design such a system would be an ambitious goal; to maintain and extend it would be even more difficult.

A reasonable implementation of an integrated system would involve models of each vision module and models of their *interactions*. However, past computer vision research in integration of vision modules has not emphasized explicit information exchange among the modules in the context of a complete system. We believe that designing integration strategies based on explicit information exchange between the modules is a first step towards building more robust and tractable vision systems. This approach also emphasizes cooperation and *resonance* between the individual modules

which many researchers believe to be the key to the effectiveness of the human vision system [72, 46, 193, 172]. Earlier efforts in integration assumed that the estimates obtained by the individual modules are reasonably accurate and hence adopted a feedforward strategy. However, in reality, each module is imperfect and, therefore, those integrated systems based on feedforward strategy alone are *critically* dependent on the performance of individual modules. Our effort here is to demonstrate the efficacy of a feedback-feedforward strategy for integration.

Integration models proposed in the literature are based on Bayesian [166], MRF [133], lattice-theoretic [89], game-theoretic [29], regularization [94, 153], and energy minimization [20] formulations (Table 2.1). It is reasonable to assume that each strategy is best suited for a certain type of interaction and hence can not be used as the integration framework for the entire vision system without sacrificing either performance or efficiency. Thus, each interaction could be modeled by the strategy best suited for that particular interaction. The success of these individual interaction models depends on a number of user-specified parameters. It is hoped that the dynamics of the system with interacting modules obviates these elaborate models of interaction among the modules and replaces them with simpler interaction schemes, facilitating the implementation of large integrated systems.

The general integration problem is formidable. Illustration of the advantages of integration in a general setting is both abstract as well as ambitious. Instead, we integrate a few *specific* vision modules with an objective of obtaining accurate 3D reconstruction. We demonstrate that the integrated system can withstand the violation of the nominal assumptions underlying the design of the component modules. In this

Table 4.1: Vision modules in the proposed integrated system.

Module	Strengths	Problems
Stereo	Reliable short-range depth information	Correspondence and Occlusion
Shape from Shading	Orientation estimation irrespective of distance	Mutual illumination and fine textures
Line Labeling	Geometric constraints	Extraction of line diagram
Perceptual Organization	Boundary completion in noisy images	Oversegmentation

chapter, we describe an implementation of the integration of perceptual organization, stereo, shape from shading, and line labeling modules [148]. These modules were chosen primarily because of their importance in low-level vision. Also, they are known to interact with each other [72, 158] and are complementary in their strengths (Table 4.1). Our strategy for integration can be extended to include additional modules. The overall block diagram of the proposed system is shown in Figure 4.1.

Rest of this chapter is organized as follows. In the next section, we describe the interactions among the modules. Section 4.2 presents the integration algorithm. In Section 4.3 we will discuss our experiments and results. We will conclude with a summary of our experimental results and research issues that need to be addressed in the design of robust integrated vision systems.

4.1 Interactions

In this section we describe our formulation of interactions among the modules and their implementation. The relevant literature will also be mentioned.

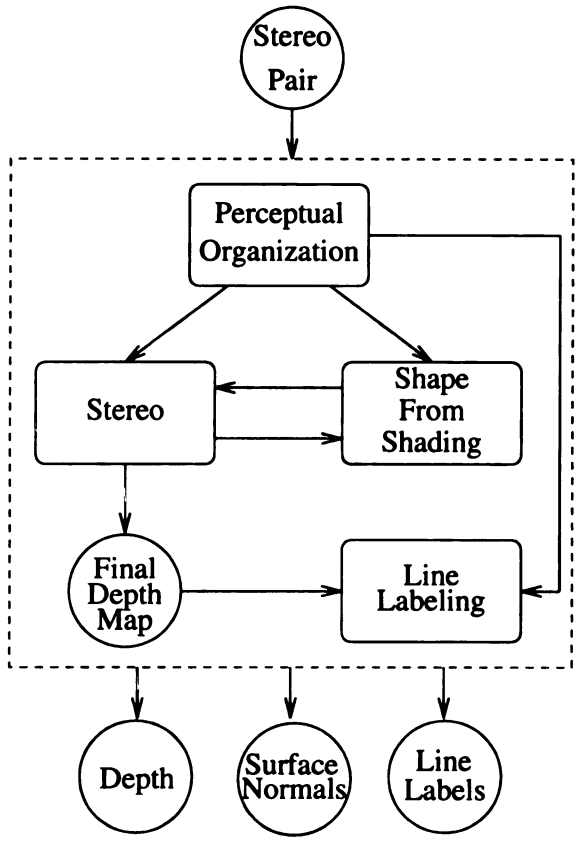


Figure 4.1: Overall Block Diagram.

4.1.1 Interaction between Shape from Shading and Stereo Modules

The algorithm proposed by Oliensis and Dupuis [145] has several shortcomings typical to the traditional shape from shading and early vision problems. It assumes Lambertian surfaces and a finite number of point sources. It is extremely difficult to take into account the effects of interreflections, specularity, and innumerable other nonlinear effects in a general situation – since that requires an *a priori* knowledge of the object geometry itself! Finally, every shape from shading approach is inher-

ently incapable of determining concavity/convexity of the surface. It is also not easy to model the photometric variations due to variable distance of scene surfaces from the illumination source. These shortcomings can be overcome with the help of the information provided by the other modules.

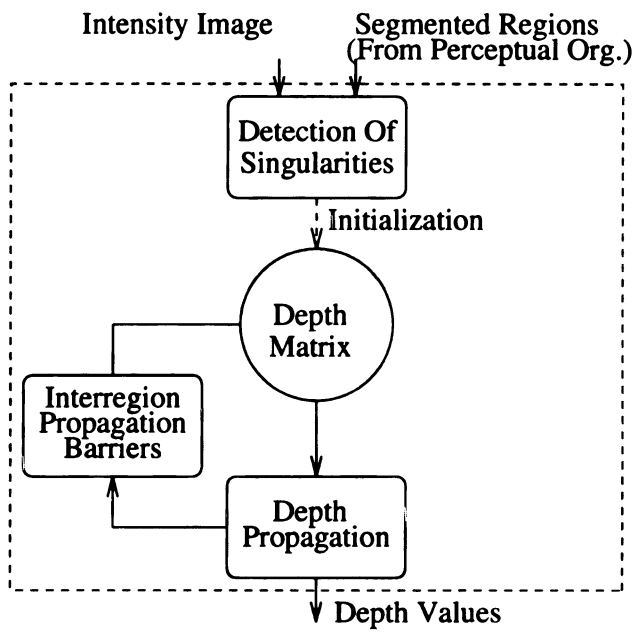


Figure 4.2: Shape From Shading Module.

We have extended the algorithm by Oliensis and Dupuis to work on piecewise constant albedo surfaces. The extension is based on segmenting the image into regions with constant albedo and treating them independently. The singular points are now detected in the individual regions and the propagation of the depth values is prevented across the region boundaries. Such a treatment, when implemented in an isolated module, is plagued with erroneous depth recovery. The resulting reconstruction often is only qualitatively reasonable and may not have desired numerical accuracy primarily due to the following reasons: (i) the stereo module, needed for

initial depth values at singular points, itself might provide erroneous depths; (ii) the Lambertian model may not be an accurate model for the image surfaces; and (iii) interreflections and specularities may also cause inaccurate reconstruction.

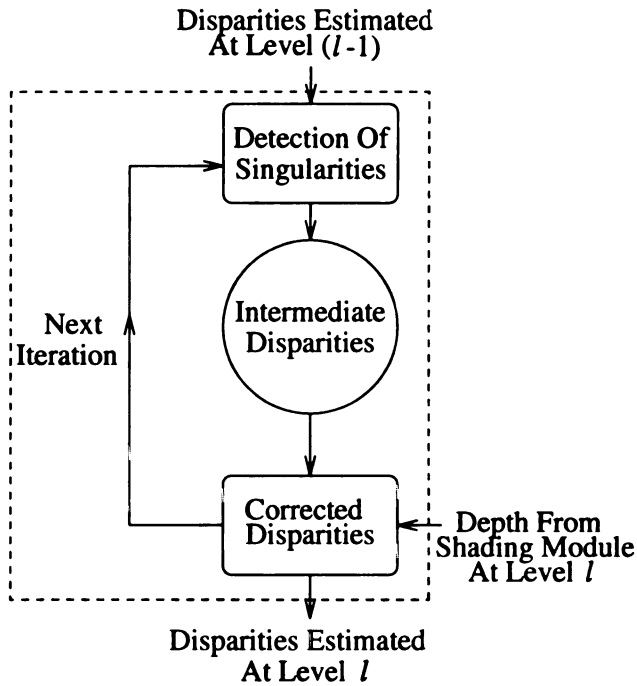


Figure 4.3: Stereo Module.

According to Bulthoff *et al.* [34], the depth conveyed by disparity overrides the information conveyed by shading. When both cues actually agree, the perception of the surface curvature is reinforced. They also have accumulated evidence that in scenes with scarce zero crossings, shape from shading interpolates depth in the area between two sparse zero crossings.

Some attempts have been made in the past to integrate the shape from shading and stereo modules. Grimson [70] has used shading to constrain the image irradiance equation to obtain the surface normal directions. Blake *et al.* [21] have established

that given the depth at the boundary conditions, the surface reconstruction by shape from shading is unique. Leclerc and Bobick [102] have used shading to interpolate the depth obtained from the stereo. Cryer *et al.* [47] have used frequency domain methods based on the psychophysical theories to integrate the shape from shading and stereo modules. Most of these earlier integration schemes assumed a single constant albedo over the surfaces comprising the entire scene. Further, they do not include any strategy for two-way interaction between the stereo and shape from shading nor does the shading guide the computation of stereo disparities.

In our approach, we have a more general model of surface reflectance of the scene (piecewise constant albedo) and a more reliable and robust strategy for treating scenes deviating from our model assumptions. Specifically, the feedback loop between the shape from shading and stereo modules is designed to counter the limitations of both the modules. When there is sparse texture in the scene, the shape from shading module will interpolate the surface depths between the (sparse) zero crossings. On the other hand, when the reconstruction offered by shape from shading is in error, the gradient descent iteration of the stereo module will attempt to force the resultant reconstruction towards the true depth value. This synergistic cooperation compensates for the errors in individual modules.

Usually, albedo of the scene surfaces is not known. In such a situation, the depth map obtained from the shape from shading module is only qualitatively correct and can not be directly related to the depth map obtained from stereo module. Then, how should the information obtained from the shape from shading module be meaningfully used to improve the depth map obtained from stereo? Given an initial estimate of

the depth from the stereo module, the proposed integration method uses a variational approach to make the surface orientation at each point on the depth map 'consistent' with the surface orientation at the corresponding point obtained from the shape from shading module. In addition, we impose a smoothness constraint in driving the final depth reconstruction. A detailed formulation of our approach is described below:

1. *Smoothness Constraint:* The surface depth varies smoothly over each segmented region. The measure of departure from smoothness of the surface can be expressed by $(D_x^2 + D_y^2)$, where D_x and D_y represent partial derivatives of depth map, D, in x and y direction, respectively. We minimize:

$$e_s = \int_x \int_y (D_x^2 + D_y^2)\, dx\, dy. \tag{4.1}$$

2. *Shading Constraint:* The resultant shape obtained by the integrated system should conform with the shape obtained by the shading module. More specifically, the surface orientation at each point (x, y) of the reconstructed surface, D, should be consistent with the orientation of the surface estimated at that point by the shading module alone. If D^{sh} is the estimate of the reconstructed surface by the shape from shading module, then the aforementioned consistency can be measured by the term:

$$e_c = \int_x \int_y \left[(D_x - D_x^{sh})^2 + (D_y - D_y^{sh})^2\right] dx\, dy. \tag{4.2}$$

Let $\tilde{e}_s$ and $\tilde{D}$ represent the discretized representations of e_s and D, respectively. Let $\tilde{e}_c$

and $\tilde{D}^{sh}$ represent the corresponding discretized versions of e_c and D^{sh}, respectively.

Given an initial estimate of the reconstructed (discretized) depths by the stereo module, $\tilde{D}^{st}$, and that by the shape from shading module, $\tilde{D}^{sh}$, the integrated output, $\tilde{D}$, is the surface that minimizes

$$\tilde{e} = \tilde{e}_c + \lambda \tilde{e}_s, \tag{4.3}$$

where λ is the *regularization* parameter; $\tilde{e}$, $\tilde{e}_c$, and $\tilde{e}_s$ are the discretized versions of e, e_c, and e_s, respectively.

Differentiating $\tilde{e}$ with respect to $\tilde{D}(i,j)$, we obtain

$$\frac{\partial \tilde{e}}{\partial \tilde{D}(i,j)} = \frac{\partial \tilde{e_c}}{\partial \tilde{D}(i,j)} + \lambda \frac{\partial \tilde{e_s}}{\partial \tilde{D}(i,j)}, \tag{4.4}$$

After differentiation[1] and a rearrangement of term, we have

$$\frac{\partial \tilde{e_s}}{\partial \tilde{D}(i,j)} = 2\left(\tilde{D}(i,j) - \overline{D}(i,j)\right), \tag{4.5}$$

where $\overline{D}(i,j)$ denotes the local 4-neighbor arithmetic mean of $D(i,j)$, and

$$\begin{aligned} \frac{\partial \tilde{e_c}}{\partial \tilde{D}(i,j)} &= 2\left(\tilde{D}(i,j) - \overline{D}(i,j)\right) \\ &\quad +2\left(\tilde{D}^{sh}(i,j) - \overline{D}^{sh}(i,j)\right). \end{aligned} \tag{4.6}$$

[1] We use finite differences (on the 4-neighborhood system) to approximate differentiation.

Setting $\frac{\partial \tilde{e}}{\partial \tilde{D}(i,j)}$ to zero gives us an iterative solution of the depth estimate:

$$\tilde{D}^{n+1}(i,j) = \overline{D}^{n}(i,j) - \frac{\lambda}{1+\lambda}\|\Delta\tilde{D}^{sh}(i,j)^{n}\|, \tag{4.7}$$

where

$$\begin{aligned}\|\Delta\tilde{D}^{sh}(i,j)^{n}\| = \ & \tfrac{1}{4}(\tilde{D}^{sh}(i+1,j) + \tilde{D}^{sh}(i,j+1) + \\ & \tilde{D}^{sh}(i-1,j) + \tilde{D}^{sh}(i,j-1)) \\ & -\tilde{D}^{sh}(i,j),\end{aligned} \tag{4.8}$$

and $\tilde{D}^{0} = \tilde{D}^{st}$. In order to avoid the instability in the reconstruction process, Eq. (4.7) is applied only when the sign of $\|\Delta\tilde{D}^{sh}(i,j)^{n}\|$ differs from the sign of $\|\Delta\tilde{D}(i,j)^{n}\|$, where

$$\begin{aligned}\|\Delta\tilde{D}(i,j)^{n}\| \quad = & \tfrac{1}{4}(\tilde{D}(i+1,j) + \tilde{D}(i,j+1) \\ & \tilde{D}(i-1,j) + \tilde{D}(i,j-1)) - \tilde{D}(i,j).\end{aligned} \tag{4.9}$$

Parameter α $(= \frac{\lambda}{1+\lambda})$ is the coupling coefficient between the shape from shading module and the stereo module. Notice that correction term in Eq. (4.7) is not based on any precise calibration, but is set to an arbitrary monotonic function of the depth depending on the value of α. In practice, we have seen that the performance of the integrated system does not critically depend on the value of α as long as it is sufficiently small ($\alpha \leq 0.05$).

The flow of information from the stereo module to the shading module is relatively

straightforward. The depth values at the singular points are initialized to the corresponding depth values predicted by the stereo module. The concavity or convexity of the surface at the singular points is also estimated from the depth map obtained from the stereo module.

4.1.2 Interactions between the depth modules and the Perceptual Organization Module

Both the general viewpoint and the relationship of significant 2D features (See Figure 3.2) with 3D causality are based on fallible assumptions. The advantage of an integrated environment is the capacity to oversegment the image using these fallible assumptions and then, at a later stage, recover the "correct" segmentation with the help of other information cues.

The human visual system interprets gradual changes in irradiance as a change in surface orientation of a single surface while abrupt changes in the irradiance are interpreted as the presence of boundaries between two distinct surfaces (or change in albedo). The process of demarcating the image plane into regions is shown to be a precursor to the 3D interpretation of the scenes. However, in the real world, this delineation is not decided by the image characteristics alone, but also by perceptually significant relations among the image features. Regions formed by perceptual boundaries are filled-in with the appropriate features (like color, brightness, texture, etc.) and the perceptual boundaries act as feature barriers [72]. Many vision researchers have been using intensity gradient as a deterrent to smoothing across uniform regions

[195], but perceptual boundaries have not been used frequently for this purpose.

In shape from shading module used in our integrated system we have prevented the propagation of depth values across the perceptual boundary. Similarly, in stereo module, the smoothness constraint is not enforced across perceptual boundaries. This has significantly reduced the blurring of the sharp depth boundaries in the recovered depth map. We also set the depth at perceptual boundaries to the values predicted by the stereo module since the reliability of the stereo module is at its best in these regions.

4.1.3 Interactions with Line Labeling Module

Although the line labeling problem has been rigorously studied for a limited object domain, it is plagued with innumerable implementation problems. Most reported work on the line labeling problem assumes availability of a perfect line diagram at the outset. Obtaining a good line diagram appears to be a difficult problem. Further, the parsing of a line diagram into vertices and arcs has been shown to be extraordinarily tricky [139]. Many researchers wonder whether ideal line diagram and its accurate parsing could ever be produced in a practical situation.

Integrating line labeling module with other modules is a challenging task since the constraints involved in it are quite different from those in stereo, shape from shading, or shape from texture modules. The line drawing interpretation module uses geometrical and qualitative constraints whereas the other modules rely on quantitative constraints.

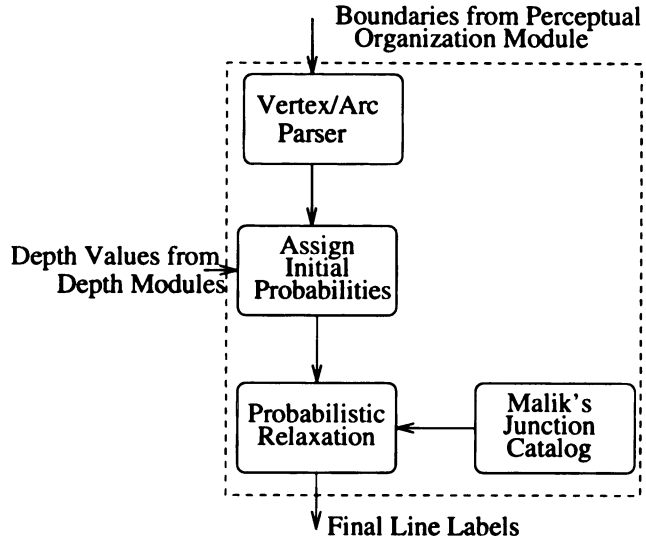


Figure 4.4: Line Labeling Module.

Attempts to integrate shape from shading and line labeling for curved objects have been reported by Malik and Maydan [113]. They formulated the integration problem as an optimization problem to simultaneously recover surface orientation and line labels. Their cost function also includes a regularization term. However, their problem has been formulated for scenes composed of the single constant albedo and demonstrated only for synthetic images. Malik's line labeling algorithm has been used by Stockman *et al.* [176] for constraining the fit of the object models to range data. Trytten has attempted to label the line drawings obtained from perceptually grouped edgels [191]. Our approach is an improvement over the strategy proposed by Trytten [191] and exploits the powerful constraints exerted by the limb boundaries to disambiguate line drawing interpretation.

The proposed integration mechanism described here investigates whether the errors in extraction of the line diagram and its parsing can be compensated for by the

information provided by the depth modules described earlier. We use the line labeling module for curved objects proposed by Malik [112] who provided an explicit catalog of legal line labels for objects with C^3 surfaces. Our method of extracting and parsing the line diagram and favors an interpretation of the scene in which the surfaces of one object do not touch surfaces of any other object (*floating object hypothesis* [189]).

Traditionally, deterministic discrete relaxation methods have been used for labeling line diagrams. However, the errors in the measurement of various features (*e.g.*, boundary curvatures, tangent directions) from the line diagram warrant a better modeling of the uncertainties involved in the process of extraction of the line diagram and its parsing. One obvious choice is probabilistic relaxation formulation.

Let $\mathcal{L} = \mathcal{L}_\mathcal{J} \cup \mathcal{L}_\mathcal{A}$ be the set of labels, where $\mathcal{L}_\mathcal{J}$ is the set of junction labels and $\mathcal{L}_\mathcal{A}$ is the set of arc labels. Define $\mathcal{L}_\mathcal{J} = \{L, C, T, Y, A, R, P\}$, where L, C, T, Y, A, R, and P represent L, curvature-L, T, Y, arrow, three-tangent, and phantom junctions, respectively. Define $\mathcal{L}_\mathcal{A} = \{x, +, -, \rightarrow, \rightarrow\!\!\rightarrow\}$, where $+$, $-$, $\rightarrow$, $\rightarrow\!\!\rightarrow$ denote convex, concave, (outside) boundary, and limb labels for arcs of the line diagram, respectively. Note that the arc label x could represent any non-depth arc including surface markings, albedo, or shadow edges. Function $P_k(l_p)$ denotes the 'probability' of vertex (arc) k having a label l_p.

Junction labels determined from tangent-based and curvature-based features of an arc are unreliable [189]. We, therefore, supplement this information by using three depth-based features extracted from the depth modules: dihedral angles, depth discontinuity, and limbness of an arc (See Table 4.2). A dihedral angle (ϕ_1) is estimated using the 3D information extracted from several patches from regions abutting an

arc. The dihedral angles are measured in such a way that for convex arcs the angle measurements are positive. The present implementation of estimating dihedral angles does not fully address the issues related to *invisible junction* [112]. We have grouped the invisible junctions and the spurious degree-2 junctions formed due to noise into *phantom* junctions. The presence of the depth discontinuity is measured in a similar manner. The measurement of limbness of an arc is described below.

Estimation of "Limbness" of a Boundary

To reliably determine "limbness" of a boundary, we use the information offered by an ensemble of surface normals in its vicinity. Let us call the angle defined by the surface normal ($\mathbf{n_i}$), the viewing direction, and the boundary normal ($\mathbf{n}$) as θ_i (see Fig. 4.5). Usually, these angles decrease monotonically as we move in the direction of boundary normal pointing inside the region enclosed by the limb edge. For non-limb edges, this trend should be considerably less significant.

Suppose we are investigating "limbness" of a point b on the boundary B and the two estimated normals of the boundary are in the directions $\mathbf{n}$ and $-\mathbf{n}$. Let $\mathbf{n_1}, \mathbf{n_2}, \mathbf{n_3}, \cdots, \mathbf{n_m}$ be the m surface normals sampled in the direction $\mathbf{n}$ at distances $d_1, d_2, d_3, \cdots, d_m$ from the boundary. We assume that the sampling is guided by segmentation to avoid observations coming from different surfaces of the scene. Construct a separate rank ordering of the angles θ_i and distances (breaking any existing ties randomly) to obtain two rank sequences r_i and s_i. We compute the Spearman's rank-order correlation statistic, S [105]. The significance of a negative value of S is tested by computing $R = S\sqrt{(m-2)/(1-S^2)}$ and comparing it to a threshold with

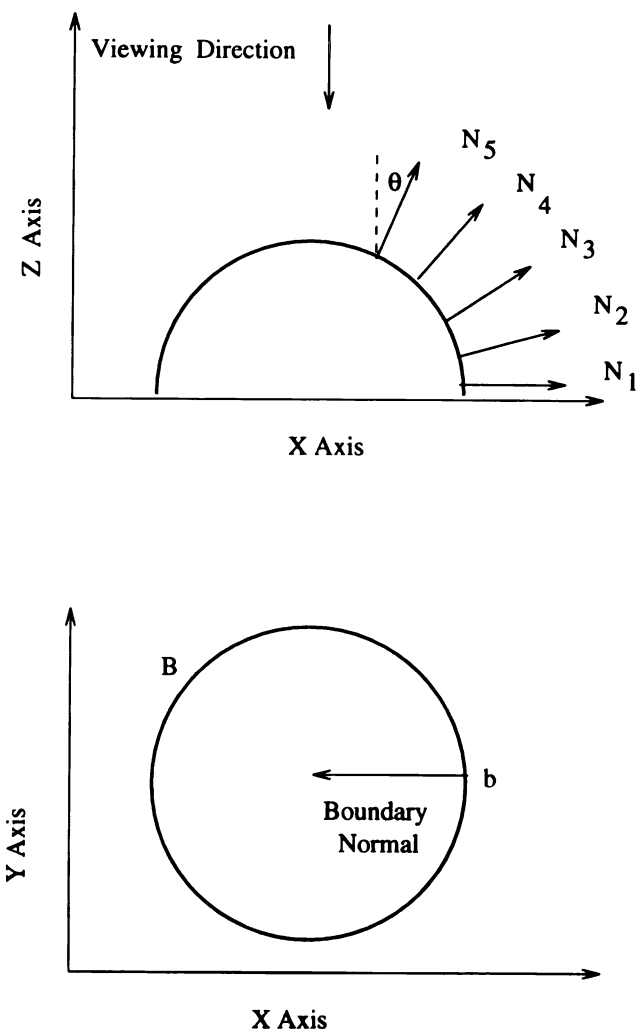


Figure 4.5: Limb Detection.

a size of 0.05. This test is supplemented by the average starting angle test. This test evaluates whether the arithmetic mean of the first few angles (θ_i) is larger than a threshold, t. Both of these tests are repeated for the samples of surface normals taken in the direction of the other boundary normal ($-\mathbf{n}$). A point on the boundary is assessed to be part of a limb edge if at least one set of surface normals passes both the trend test as well as the average starting angle test. The fraction of points belonging to an arc passing the limbness test determines the limbness of that arc.

Estimation of Relaxation Parameters

We compute the initial probabilities using Tables 4.3–4.4 ($(T_{\phi_1} = \pi/20$, $T_g = 4$ pixels, $\delta_1 = 0.5$, $\delta_2 = 0.7$, $T_{\phi_3} = \pi/3$, $T_d = 5)$. We have used the generic sigmoid function for specifying the a priori probabilities:

$$\sigma(x; \delta, \kappa) = 1/(1 + \exp(-\delta * (x - \kappa))).$$

For notational convenience, we denote $\sigma(-x; \delta, \kappa)$ by $\sigma'(x; \delta, \kappa)$. The compatibility functions, r_{ts}: $\mathcal{L}_t \times \mathcal{L}_s \rightarrow [-1, 1]$, model the 'likelihood' of labels t and s being labels of the neighboring arcs (or vertices). A compatibility function is negative, positive, or zero depending upon whether the labels are incompatible, compatible, or independent, respectively. The compatibility functions are directly determined from Malik's junction catalog.

The relaxation algorithm alternates between two epochs: vertex and arc. In the vertex epoch, it updates the probabilities of the vertex labels. At the end of the

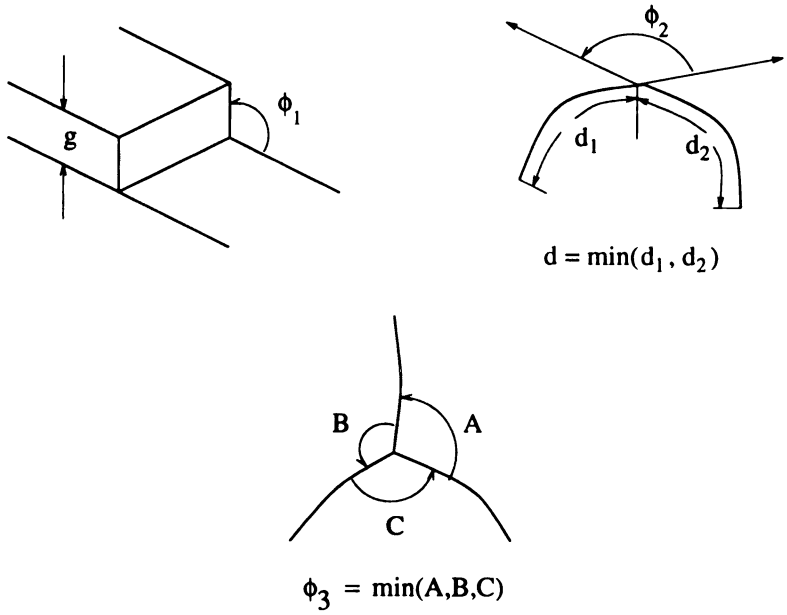


Figure 4.6: Line diagram features.

Table 4.2: Arc and vertex features. The features g, ϕ_1, l_1 relate to an arc; ϕ_2, d, and l_2 are degree-2 vertex features; ϕ_3 and l_3 are degree-3 vertex features (see Fig. 4.6).

Feature	Definition
g	Average depth gradient across the arc.
ϕ_1	Average dihedral angle between the surface normals across the arc.
l_1	Limbness of the arc.
ϕ_2	Angle between tangents to the two arcs.
d	Length of the shortest of the two arcs.
l_2	Maximum of limbness of the two arcs.
ϕ_3	Maximum of the angles between tangents of each pair of arcs.
l_3	Max. of the limbness of all arcs.

Table 4.3: Prior probabilities for arc labels. The term $Z = \sum_{i \in \mathcal{L}_A} \mathcal{I}(i)$ is the normalization factor. Note that T_k specifies threshold for parameter k.

Label	Indicator Function	Prior Probability
$+$	$\mathcal{I}(v) = \sigma(\phi_1; \delta_1, \lvert T_{\phi_1} \rvert)$	$\mathcal{I}(v)/Z$
$-$	$\mathcal{I}(c) = \sigma'(\phi_1; \delta_1, \lvert T_{\phi_1} \rvert)$	$\mathcal{I}(c)/Z$
$\rightarrow$	$\mathcal{I}(b) = \sigma(\lvert g \rvert; \delta_2, \lvert T_g \rvert) * (1 - l_1)$	$\mathcal{I}(b)/Z$
$\rightarrow\rightarrow$	$\mathcal{I}(l) = \sigma(\lvert g \rvert; \delta_2, \lvert T_g \rvert) * l_1$	$\mathcal{I}(l)/Z$
$\times$	$\mathcal{I}(x) = 2(1 - \max_{i \in \{v,c,b,l\}} \mathcal{I}(i))$	$\mathcal{I}(x)/Z$

epoch, it assigns a tentative label to each vertex according to the ranking offered by the computed probabilities. Thus, in each vertex epoch, the relative probability of label y assigned to vertex v is updated according to

$$P_v(y)^{(n+1)} = \frac{P_v(y)^{(n)}[1 + \Delta P_v(y)^{(n)}]}{\sum_{y \in \mathcal{L}_\mathcal{J}} P_v(y)^{(n)}\,[1 + \Delta P_v(y)^{(n)}]}, \tag{4.10}$$

Table 4.4: Prior probabilities for junctions. Note that T_k specifies threshold for parameter k and $Sgn(.)$ is the conventional *sign* function.

Label	Prior Probability
P	$\sigma(\phi_2; d/T_d, \frac{\pi}{2})$
L	$(1-\sigma(\phi_2; d/T_d, \frac{\pi}{2}) * (1-l_2)$
C	$(1-\sigma(\phi_2; d/T_d, \frac{\pi}{2}) * l_2$
A	$p(a) = \sigma(N_3 - \pi; \delta, \lvert T_{\phi_3}\rvert)$
Y	$p(y) = \sigma'(\phi_3 - \pi; \delta, \lvert T_{\phi_3}\rvert))$
T	$Sgn(\phi_3 - \pi) * (1-p(a))+$ $(1-Sgn(\phi_3-\pi)) * (1-p(y)) * (1-l_3)$
R	$Sgn(\phi_3 - \pi) * (1-p(a))+$ $(1-Sgn(\phi_3-\pi)) * (1-p(y)) * l_3$

where

$$\Delta P_v(y)^{(n)} = \sum_{u \in \mathcal{N}_a(v)} \sum_{y' \in \mathcal{L}_\mathcal{J}} r_{vu}(y, y') P_u(y')^{(n)}, \tag{4.11}$$

n is iteration number and $\mathcal{N}_a(v)$ is the set of all arcs incident on vertex v. Considering these tentative vertex labels, the label probabilities of the arcs are updated in the arc epoch using an updating strategy similar to that in Eq (4.10). Since the boundary arcs (vertices) exert stronger constraints, the order in which each arc (vertex) is visited favors peripheral arcs (vertices). The relative probabilities of each label of each arc (vertex) are restored to their original values after entering each epoch. The algorithm terminates when the weights for some label of *most* of the sites is close to 1 or there are no changes in the successive epochs.

4.2 Integration Algorithm

In this section we present a high-level description of the overall integration algorithm. Given a stereo pair of intensity images, I^1 and I^2, direction of the illumination source, **S**, weight vector **w** (Eq. (3.1)), a coupling coefficient (α) (Eq. (4.7)), various threshold parameters (Tables 4.3 and 4.4), the depth values and the line labels are reconstructed using the following steps.

1. Compute the four attribute (smoothed, gradient, positive curvature, and negative curvature) images. Initialize the disparity at each pixel in the coarsest level ($l = 6$) to zero.

2. Compute the closed regions of uniform intensity using perceptual organization module (Section 3.1). This will be referred to as the label image.

3. Starting with the coarsest level ($l = 6$), do at each level l:

   A. Obtain the four attribute images at level l by blurring the attribute images at level 0.

   B. Obtain label image at level l by median filtering the label image at level 0. This is essentially a map of the regions of uniform intensity in the image. Applying an edge detector to this label image will provide perceptual boundaries at level l (boundary image, $\mathcal{B}_l$).

   C. Identify singularities in each region from the intensity image at level l ($\mathcal{S}_l$).

      Do steps (i) to (v) for $N(= 20)$ times.

(i) Update the disparities at level l, ($\mathbf{d}_l$), by one iteration of Weng *et al.*'s algorithm (Eq. (3.1)). The resulting depth map computed from these disparities will be denoted by $\tilde{D}_l^{st}$.

(ii) Initialize the depth at each singularity position in $\mathcal{S}_l$ by appropriate depth values from $\tilde{D}_l^{st}$.

(iii) Normalize the intensities in each region with respect to the average intensities in that region. Assess from $\tilde{D}_l^{st}$ the convexity/concavity of each region.

(iv) Apply shape from shading algorithm to obtain a depth map, $\tilde{D}_l^{sh}$.

(v) Obtain the corrected depth map $\tilde{D}_l$ from $\tilde{D}_l^{sh}$ and $\tilde{D}_l^{st}$ using the shape from shading and stereo integration (Eq. (4.7)). Update the disparity map using $\tilde{D}_l$.

D. Parse the boundaries in $\mathcal{B}_l$ into a line diagram graph $\mathcal{G} = \{\mathcal{V}, \mathcal{A}\}$, where $\mathcal{V}$ is the set of vertices and $\mathcal{A}$ is the set of arcs.

E. Compute the line diagram features from the depth map $\tilde{D}_l$ and the boundary image $\mathcal{B}_l$ (Table 4.2).

F. Initialize the a *priori* probabilities of labels (Tables 4.3 and 4.4).

G. Iterate through the probabilistic relaxation until the label probabilities stabilize (Eq. (4.10) shows the updating scheme for vertex epoch).

H. Project the disparities to level $(l-1)$ by replicating disparity $2\mathbf{d}_l(i,j)$ to locations $(2i,\ 2j)$, $(2i+1,\ 2j)$, $(2i,\ 2j+1)$, and $(2i+1,\ 2j+1)$.

Table 4.5: CPU times for Mushroom and Vase Image on Sun Sparc 20.

Computation	Time
Perceptual Grouping	61s
Shape from shading	810s
Stereo	594s
Line labeling	357s
Integrated System	2284s

4. Output the final depth map $\tilde{D} = \tilde{D}_0$, and line labels. The surface orientation (normals) at each point can be computed from $\tilde{D}$.

The major components of the integration themselves require *local* computations (Eqs. (4.7) and (4.10)). The integration of the stereo and shape from shading modules needs one pass over the entire image and its computational complexity is $O(P^2)$, for a total of P^2 pixels. The computation of features from the line diagram is somewhat image dependent and its overall complexity is $O(|V|^2)$, where $|V|$ is the cardinality of $\mathcal{V}$. The probabilistic relaxation does not usually take more than a few iterations over the entire graph. Thus, the proposed integration mechanism is an efficient and reliable method of integrating vision modules. Table 4.2 presents actual timing statistics on a Sun Sparc 20 for Mushroom Vase image (Fig. 4.8).

4.3 Experiments

All the real images were captured by an inexpensive CCD camera (Panasonic GP-KR202, $f = 25$ cm, maximum aperture). The images were subsequently gamma corrected with $\gamma = 2.0$ and normalized to 256 gray levels. The stand off was

approximately 80 *cm* and to obtain a stereo pair of images, the camera was either translated or translated and rotated. The translation was in the direction of x-axis and rotation was about the y-axis (the z-axis being approximately aligned with the optical axis). The imaging setup was not calibrated; all alignments, translations, and rotations were approximate and were not precisely measured/verified. The rotation of the camera was effected to bring the disparity of the region of interest close to zero. The scene was illuminated with ambient light and a single incandescent light source was located (30 cm) behind the camera (approximately in x-z plane) pointing in the direction $(0, 0, 1)$. A polarizing setup similar to the one suggested in [140] was used (when necessary) to reduce the specular component of the reflection.

In most of the experiments, we have chosen to compare the reconstruction results obtained from the integrated system with corresponding results from stereo module alone; the reconstruction results from the other individual modules were not as reliable as those of the stereo module. All the reconstruction results are presented for the right image of the stereo pair and for all the experiments the parameter α was set to 0.001. Further, the shape from shading constraints were exploited only for four different spatial resolutions (64×64, 128×128, 256×256, and 512×512). Use of shape from shading module for lower resolution representations did not significantly improve the results. Figures 4.8(a) and 4.8(b) show a stereo pair of a scene consisting of two unglazed ceramic objects (image size 512×512) with near ideal Lambertian surfaces. Figures 4.8(c) and 4.8(d) show the (relative) depth reconstruction obtained by the stereo module and the integrated system, respectively. The diffusion of disparities across the perceptual boundaries significantly blurs the depth map by the

stereo module; this is prevented in the depth map obtained by the integrated system. Figures 4.8(e) and 4.8(f) show the orientation of the object surfaces for stereo and integrated system, respectively. In case of the stereo depth map, it can be observed that as the size of the untextured region increases, the quality of reconstruction deteriorates.

The objective of the second set of experiments is to quantitatively demonstrate that our integration mechanism can handle images of various types of surfaces, including concave, convex, and saddle shaped objects. In particular, this set of experiments is designed to evaluate our integrated system on a range of intensity images of synthetic scenes consisting of Lambertian algebraic surfaces. The shaded intensity images (Fig. (4.9)) were synthesized using ray tracing method [66] of photo-realistic rendering (with atmospheric turbulence modeling the imaging noise) with a parallel camera geometry (standoff = 50 units, cameras at $(-5, 0, 0)$ and $(5, 0, 0)$) and a distant point source illumination (at $(0, 0, -100)$) (see Fig. 4.7). We then matched these stereo pairs obtained from synthetic rendering using (a) stereo module alone and (b) the integrated system. The resultant reconstructions were compared with the depth data projected from the range images (ground truth) using a squared error function based on differences in true and estimated depth and shape (surface normals) measurements (D_{err} and S_{err}, respectively). Percentage reduction in this squared error is used for assessing the performance of the proposed integration strategy. Table 4.6 shows that the surface reconstruction using the integrated method is superior to that obtained by stereo module alone. Note that stereo module can almost always correctly find correspondence at the boundaries of the object. The accuracy of the disparity map in

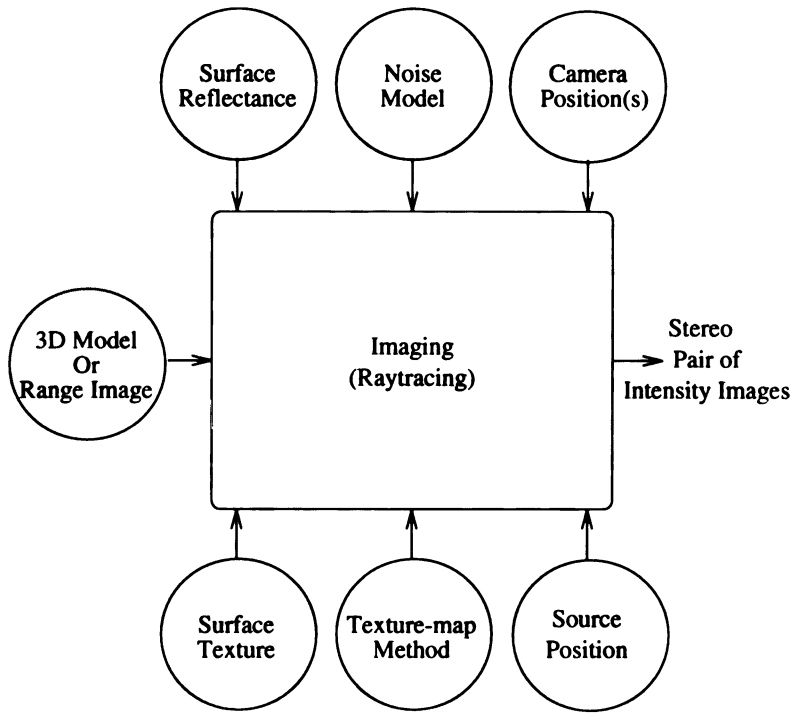


Figure 4.7: Image synthesis using ray tracing.

the regions between the boundaries is dependent on the changes in the corresponding brightness patterns in the image. The fewer the changes in the brightness in this region, the greater the scope of improvement in the performance due to integration. If the foreground *regions* in the image have smaller *diameter* (maximal width) then the stereo module alone can produce reasonably accurate correspondences in the regions between the boundaries. Consequently, it is harder for the integration to improve the performance in this situation.

Figure 4.10 shows the images synthesized using imaging conditions identical to those in Figure 4.9 except that Phong model of specularity was used ($\eta = 0.8$) instead of the Lambertian assumption. Table 4.7 shows that the reconstruction results are relatively stable when surfaces are non-Lambertian (specular) (Figure 4.10). The

improvement in the performance in the case of specular surfaces is smaller in magnitude than for the corresponding Lambertian surface (Table 4.6). This is largely due to the improvement in performance of stereo module in the case of specular surfaces – the sharper changes in intensity at specular highlights make the correspondence problem easier[2].

Figures 4.11(a) and 4.11(d) show the range images of two objects used for quantitatively evaluating the accuracy of the depth map obtained by our integrated system. Two pairs of synthetic stereo intensity images were generated from these depth maps using Lambertian surface assumption. An additive *i.i.d.* Gaussian noise (with standard deviation 2%) was then added to the left and right images separately. A sample pair of these synthetically generated stereo pairs is shown in Figures 4.11(b), 4.11(c), 4.11(e), and 4.11(f). A comparison of reconstructions is summarized in Table 4.8.

Table 4.6: Improvement in surface reconstruction due to integration: Lambertian surfaces.

Surface primitive	S_{err}(%)	D_{err} (%)
Parallelopiped	30.0	41.3
Sphere	23.4	35.7
Cylinder	22.0	26.5
Paraboloid	18.1	20.0
Hyperboloid	24.1	20.2
Torus	15.8	17.6

Figures 4.12(a) and 4.12(b) show stereo images of an unglazed ceramic object

[2]Location of specularities in an image depends upon the viewing direction and then the correspondence based on the features derived from specularity are unreliable. However, when an entire surface lacks any significant change in the albedo (or the irradiance) and change in the viewing direction is not very large, the correspondence is somewhat improved due to the presence of specular features.

Table 4.7: Improvement in surface reconstruction due to integration: Specular surfaces.

Surface primitive	$S_{err}(\%)$	D_{err} (%)
Parallelopiped	30.0	41.3
Ellipsoid	10.0	13.9
Cylinder	12.5	16.9
Paraboloid	13.4	17.6
Hyperboloid	15.3	13.6
Torus	6.8	8.1

Table 4.8: Improvement in surface reconstruction due to integration: intensity images synthesized from real range images.

Object	$S_{err}(\%)$	D_{err} (%)
Tomato	25.2	22.7
Pipe	35.0	32.6

(mushroom) and an object made of acrylic plastic (Y-shaped pipe). These images were captured without cross-polarized filters to allow the specular reflections to be imaged. Notice the two specularities on the surface of the pipe. The quality of the reconstructed depth map of the integrated system as shown in Figures 4.12(d) and 4.12(f) has largely remained insensitive to the specular reflections. Figures 4.12(c) and 4.12(e) show the corresponding outputs of the stereo module.

Figures 4.13(a) and 4.13(b) show stereo images of a granny smith apple and a yellow pepper. The surfaces of both the objects do not possess a very uniform albedo. Again, as depicted in Figures 4.13(d) and 4.13(f) the integrated system performs better than the stereo module alone.

The segmentations obtained in the earlier images (Figs. 4.14(a)-(c)) by the grouping module were near perfect. We now demonstrate the efficacy of our approach in case of an imperfect segmentation of an image consisting of piecewise constant albedo

surfaces. Note that the grouping module fails to obtain a correct segmentation of this image (Fig. 4.14(d)). Figures 4.15(a) and 4.15(b) show stereo images of an unglazed ceramic object (*egg*) and a foam cup. Notice the lack of secondary reflections on the lower part of the egg. Figures 4.15(c) and 4.15(e), respectively, show the depth reconstruction and surface needle map obtained from the isolated stereo system. Figures 4.15(d) and 4.15(f) show the corresponding representations for the integrated system. Very bright (low depth values) regions between the objects and in the far left of the image are due to occlusions. Notice that the quality of surface reconstruction for the cup in case of the stereo module alone (Figures 4.15(c) and 4.15(e)) is comparable to that of the integrated system (Figures 4.15(d) and 4.15(f)). The depth reconstruction provided by the integrated system is relatively more accurate except for the slight deterioration on the lower part of the egg. The surface of the egg has been incorrectly reconstructed by the isolated stereo module (Figures 4.16(a)) compared to the integrated system (4.16(b)). Note that stereo module (alone) could reconstruct surface of glass correctly due to the presence of texture features.

Figure 4.17 shows the contours detected from the Mushroom and Pipe image by the line labeling module. The objects in this image do not have any surface markings and the image does not contain any shadows. The detected contours have been classified into limb and non-limb edges. Recall that these representations form an input to the junction labeling module. Figures 4.18(a) and 4.18(b) show the outputs of the line labeling algorithm proposed by Trytten [189] and our algorithm, respectively. Note that all the *curvilinear-L* junctions are correctly labeled by our algorithm. In addition, the labeling of the *T-junctions* is a gross approximation of

the physical reality [139], given the quality of output generated by the segmentation module.

4.4 Summary

The visual world can be ambiguous only in relatively contrived situations. In the real world a combination of cues (visual or other kinds) conveys a unique physical reality. Whether to reconstruct the entire visual input or to extract its components relevant to the given task, a reliable vision system is expected to integrate many visual cues to obtain an unambiguous output. However, the information provided by each cue is based on its own set of assumptions. This raises several important research issues in solving the problem of integration. What is the most reliable information provided by each visual cue? How to design an integrated system which can be easily maintained and extended? How to integrate vision modules so that the system performance does not critically depend on individual modules? Definitive answers to these questions do not appear to be in sight. In this chapter we have made an attempt to explore some of these issues in a somewhat limited context of an integrated system which reconstructs 3D information from a pair of (stereo) intensity images using the following four vision modules: perceptual organization, shape from shading, stereo, and line labeling.

Several integration strategies have been reported in the literature for primarily studying pairwise integration of vision modules. But only a few studies have comprehensively integrated more than two modules for a complete 3D reconstruction of *real images*. We have proposed and implemented an integration framework emphasizing

interaction and information exchange among the four vision modules. We have shown the reconstruction results using integrated system for both synthetic and real images. We also demonstrated the consistent performance of the integrated system even in the adverse situations where one or more assumptions made by the individual modules are violated. The numerical accuracy of the recovered depth is assessed in case of synthetically generated data. We have also quantitatively evaluated our approach by reconstructing geons from the depth data obtained from the integrated system.

In general, the potential of a system relying on low-level modules has been grossly underestimated because of the vulnerabilities of the individual modules. We have demonstrated that these shortcomings in the individual modules can be overcome in an integrated environment. In particular, the limitations of a stereo module in dealing with images displaying no significant intensity changes can be alleviated by the shading module. Our perceptual organization module completes the boundaries obscured due to low contrast and noise and facilitates in obtaining a reasonable line diagram. The perceptually completed boundaries also serve as feature propagation barriers and help in obtaining more reliable results for both shading and stereo modules. Difficulties in obtaining correct arc and junction labels can be overcome using depth information obtained from the stereo and shading modules.

The proposed system is far from ideal. The perceptual organization system does not get any feedback from the other modules. It is not obvious in what form the other modules can detect and convey the errors made by the perceptual organization module. Our design is largely inspired by psychophysical and neurological evidence. It is, therefore, not easy to extend the current integrated system to include an arbitrary

vision module without understanding the strengths and limitations of the module. A considerable amount of further research is needed to obtain better strategies of integration and establish a formal and more unified framework to ease the design of an integrated system. This constitutes the topic of the next chapter.

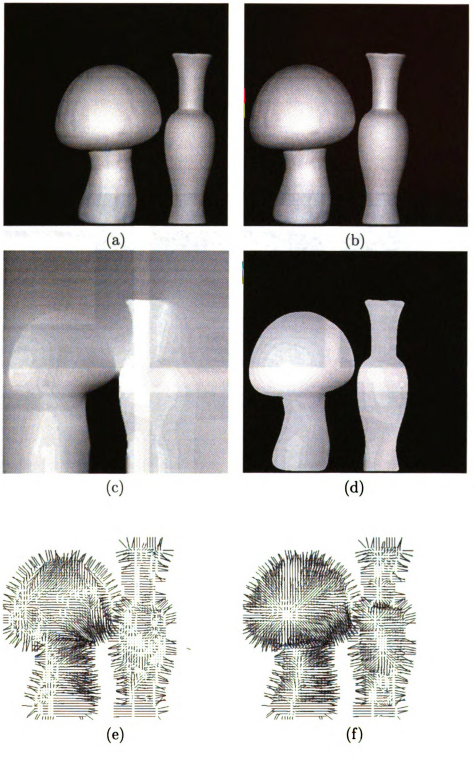


Figure 4.8: Mushroom and Vase image (size 512 × 512): (a) and (b) Left and right stereo images; (c) Recovered depth from stereo alone; (d) Recovered depth from the integrated system; (e) Recovered surface normals from stereo alone; (f) Recovered surface normals from the integrated system.

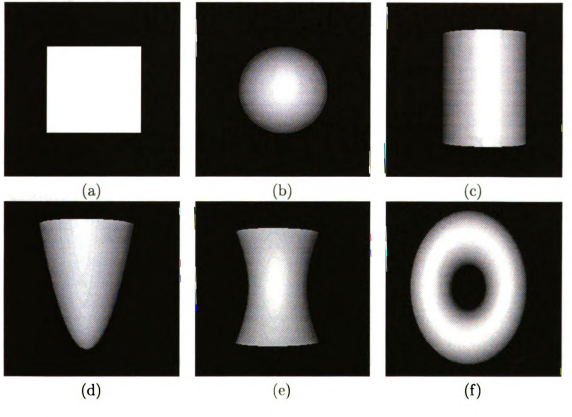


Figure 4.9: Synthetic surface primitives (Lambertian surfaces): (a) Parallelopiped; (b) Sphere; (c) Cylinder; (d) Paraboloid; (e) Hyperboloid; and (f) Torus.

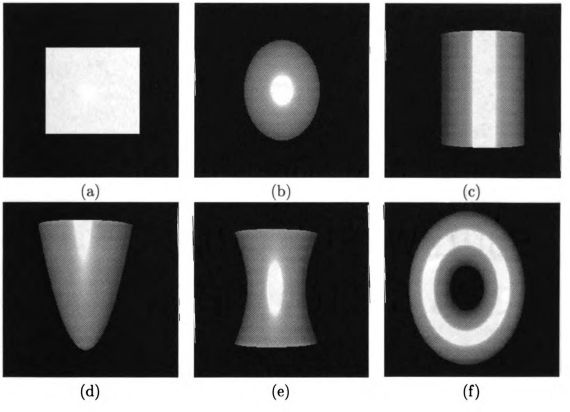


Figure 4.10: Synthetic surface primitives (Specular surfaces): (a) Parallelopiped; (b) Ellipsoid; (c) Cylinder; (d) Paraboloid; (e) Hyperboloid; and (f) Torus. All surfaces are shown with Phong shading.

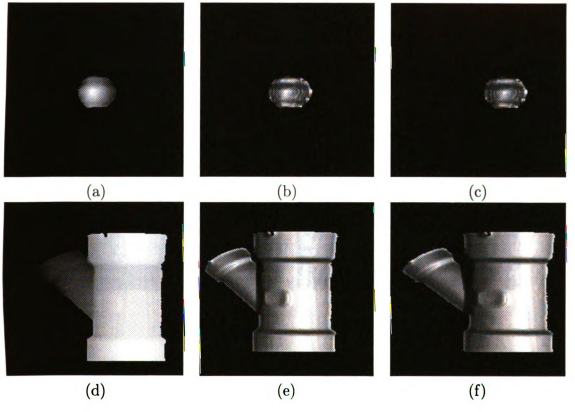


Figure 4.11: Synthetic stereo images: (a), (d): Range Images of Tomato and Pipe obtained from White scanner [110]; (b), (c): Left and right stereo images generated from (a); (e), (f): Left and right stereo images generated from (d).

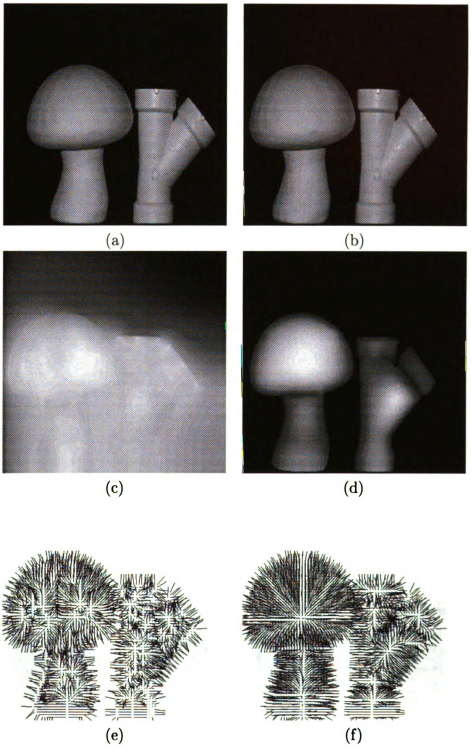


Figure 4.12: Mushroom and Pipe image (size 512 × 512): (a) and (b) Left and right stereo images; (c) Recovered depth from stereo alone; (d) Recovered depth from the integrated system; (e) Recovered surface normals from stereo alone; (f) Recovered surface normals from the integrated system.

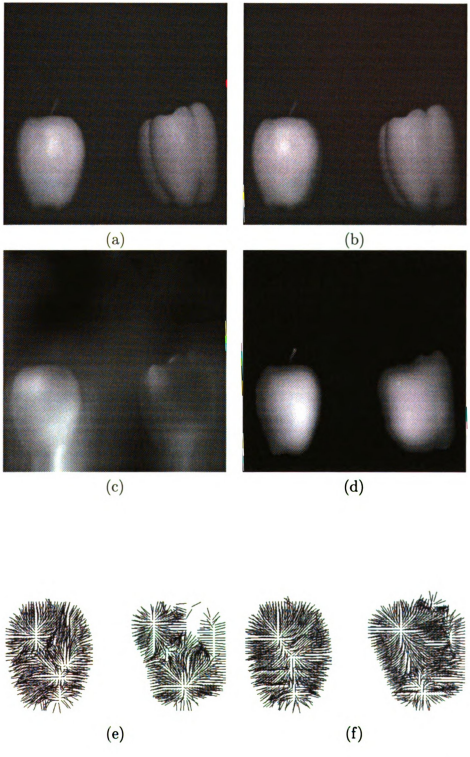


Figure 4.13: Apple and Pepper image (size 512 × 512): (a) and (b) Left and right stereo images; (c) Recovered depth from stereo alone; (d) Recovered depth from the integrated system; (e) Recovered surface normals from stereo alone; (f) Recovered surface normals from the integrated system.

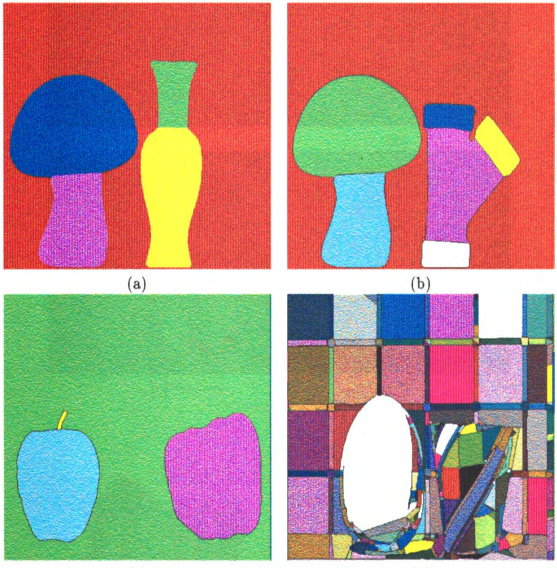


Figure 4.14: Segmentation results: (a) Mushroom and Vase 4.8(b); (b) Mushroom and Pipe 4.12(b); (c) Apple and Pepper 4.13(b); (d) Egg and Cup 4.15(b).

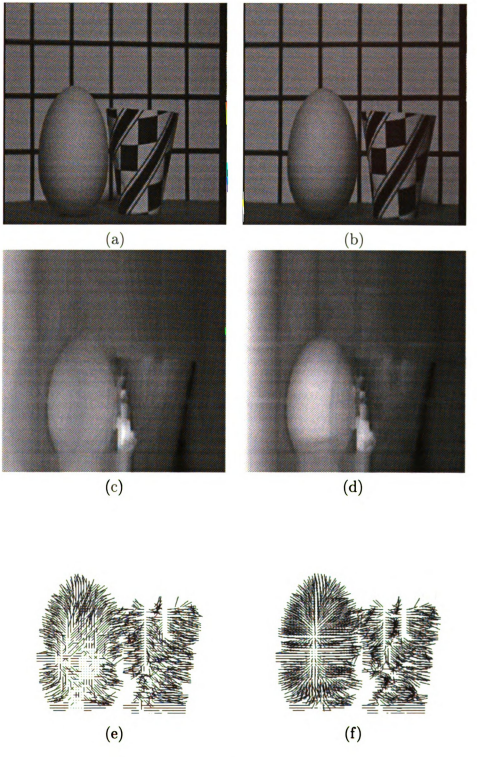


Figure 4.15: Egg and Cup image (size 512 × 512): (a) and (b) Left and right stereo images; (c) Recovered depth from stereo alone; (d) Recovered depth from the integrated system; (e) Recovered surface normals from stereo alone; (f) Recovered surface normals from the integrated system.

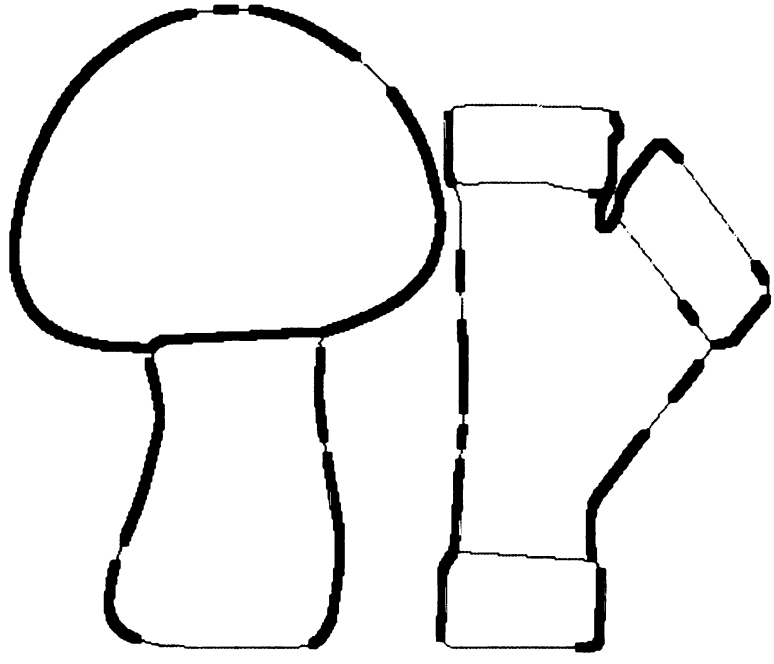
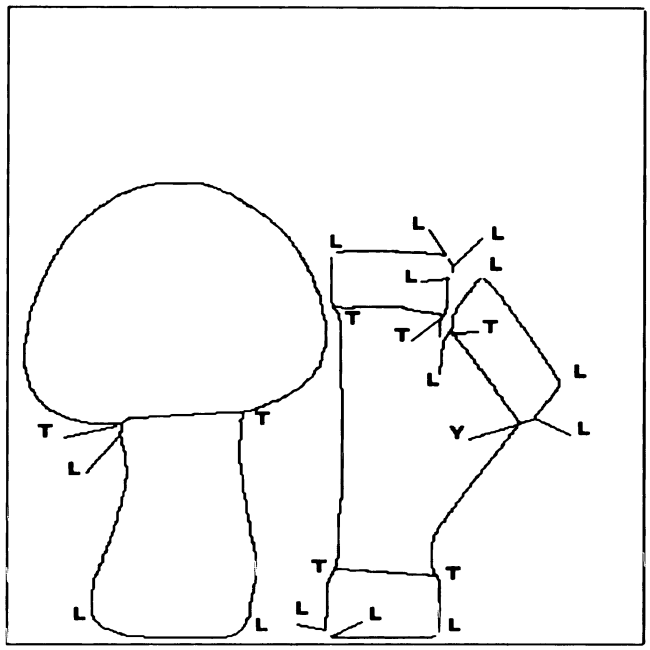
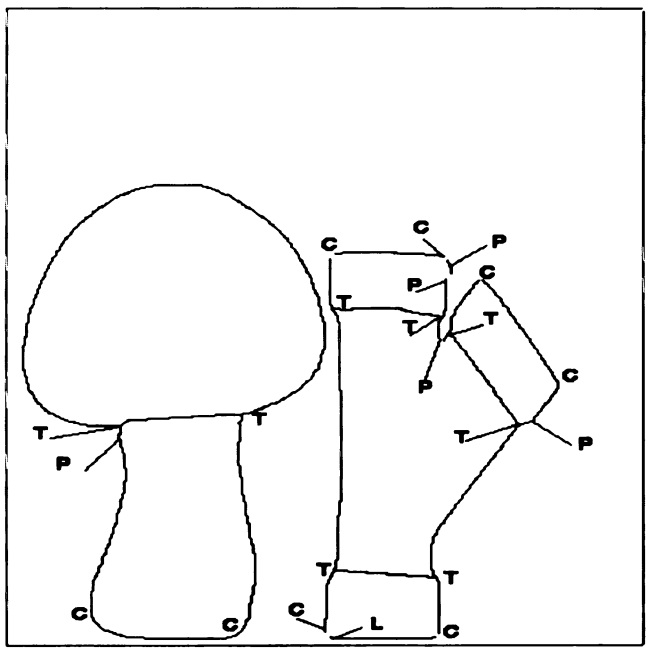


Figure 4.17: Limb edges detected in Mushroom and Pipe image (Figure 4.12). Limb boundary pixels are rendered as thick boundaries and non-limb boundary pixels are depicted as thinner edges.

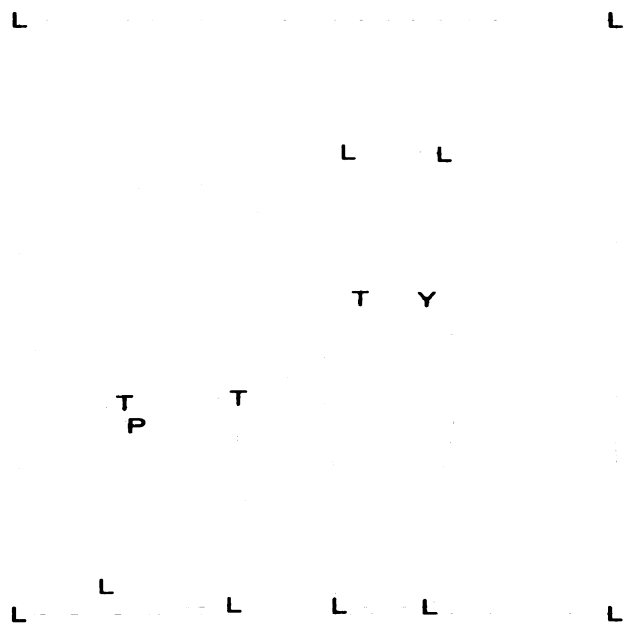


(a)

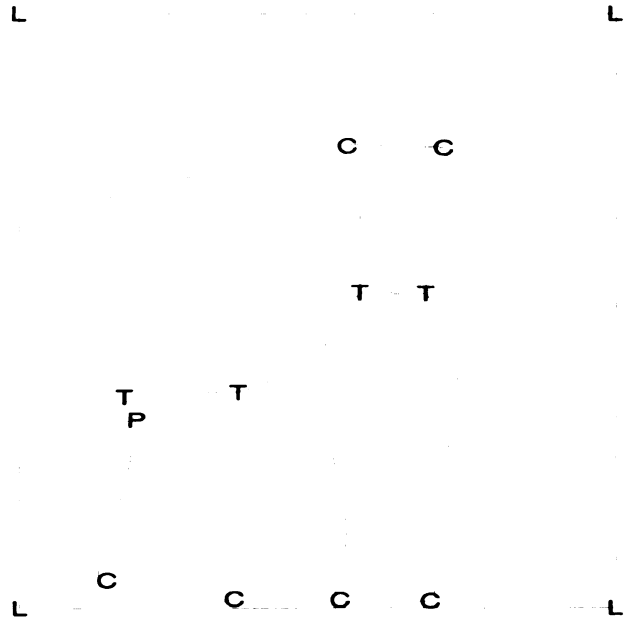


(b)

Figure 4.18: Junction labeling results for Mushroom and Pipe image (Figure 4.12): (a) using line diagram alone; (b) by the integrated system using the information provided by the depth modules and line diagram; L, C, T, Y, A, and P denote L, curvature-L, tangent, Y, arrow, and phantom junctions, respectively.



(a)



(b)

Figure 4.19: Junction labeling results for Mushroom and Vase image (Figure 4.8): (a) using line diagram alone; (b) by the integrated system using the information provided by the depth modules and line diagram; L, C, T, Y, A, and P denote L, curvature-L, tangent, Y, arrow, and phantom junctions, respectively.

Chapter 5

A Uniform Bayesian Framework for Integration

There is a substantial amount of literature on specific algorithms for pairwise integration of vision modules [8]. However, typically, these schemes lack a broader perspective of the integration problem and do not *systematically* deal with the problems encountered with images of real and complex scenes. Hence, these integration algorithms often can not be extended either to other domains or to different modules. A method of including additional vision modules in the non-uniform integration schemes similar to the one proposed in Chapter 4 is also not obvious. Several generic methodologies in the literature (Table 2.1) offer a diverse set of tools for the integration problem, but do not offer a ready-made solution. A considerable insight is needed in order to implement these techniques for solving a particular integration problem.

A uniform integration framework deals with the constraints provided by each

module in a systematic and regular fashion. One of the simplest examples of the uniform integration framework is Moravec's [130] work on fusion of data from multiple depth sensors to derive space occupancy information using Bayes rule. It is desirable to have a *uniform* framework for integrating information available from the vision modules. Such a framework will help us identify the essential components of an integration task and facilitate the incorporation of a given vision module into an existing system. In this chapter, we propose and evaluate a Bayesian framework for the recovery of structures specific to the 3D world.

Rest of this chapter is organized as follows. Section 5.1 formulates the 3D reconstruction problem as an estimation problem. Section 5.2 presents the four modules which we have integrated and their interactions with the intrinsic map. In Section 5.4, we describe the overall integration algorithm. Experimental results and imaging setup are described in Section 5.5. Section 5.6 concludes with a discussion of various issues pertaining to the visual integration, and accomplishments and limitations of the proposed scheme.

5.1 Bayesian Estimation

Let us illustrate the concept of Bayesian integration using the following simple example. Consider a Bayesian estimator integrating three depth observations, o_1, o_2, and o_3 at the same pixel site from three different vision modules, $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$,

respectively. Consider the following formulation[1]:

$$\begin{bmatrix} o_1 \\ o_2 \\ o_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \Theta + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix}, \tag{5.1}$$

where Θ is the (true) depth variable to be estimated based on the observed depths o_1, o_2, and o_3 contaminated with additive noise $n_1 \sim N(0, \sigma_1^2)$, $n_2 \sim N(0, \sigma_2^2)$, $n_3 \sim N(0, \sigma_3^2)$.

The Bayesian estimate is given by maximizing the a posteriori probability of Θ:

$$\hat{\Theta} = \arg\max P(\Theta | o_1, o_2, o_3). \tag{5.2}$$

Using Bayes' formula, the maximization problem in Eq. (5.2) can be restated as:

$$\hat{\Theta} = \arg\max \frac{P(o_1, o_2, o_3 | \Theta) P(\Theta)}{P(o_1, o_2, o_3)}. \tag{5.3}$$

Since the denominator $P(o_1, o_2, o_3)$ is independent of Θ, the maximization problem is simplified to:

$$\hat{\Theta} = \arg\max P(o_1, o_2, o_3 | \Theta) P(\Theta). \tag{5.4}$$

[1]In a more general case, the depth is a linear or non-linear function of the observed variables (see Section 5.3 for the linear case).

Assuming Θ follows a uniform distribution, Eq.(5.4) reduces to

$$\hat{\Theta} = \arg\max P(o_1, o_2, o_3|\Theta). \tag{5.5}$$

Assuming that observations are statistically independent leads to:

$$\hat{\Theta} = \arg\max P(o_1|\Theta)P(o_2|\Theta)P(o_3|\Theta). \tag{5.6}$$

Let

$$P \equiv P(o_1|\Theta)P(o_2|\Theta)P(o_3|\Theta) \tag{5.7}$$

$$\propto \frac{1}{\sigma_1^2}\exp\left(-\frac{(\Theta-o_1)^2}{\sigma_1^2}\right)\frac{1}{\sigma_2^2}\exp\left(-\frac{(\Theta-o_2)^2}{\sigma_2^2}\right)\frac{1}{\sigma_3^2}\exp\left(-\frac{(\Theta-o_3)^2}{\sigma_3^2}\right). \tag{5.8}$$

Differentiating $\ln(P)$ with respect to Θ and setting $\frac{\partial \ln(P)}{\partial \Theta}$ to zero, we obtain

$$0 = \frac{(o_1 - \Theta)}{\sigma_1^2} + \frac{(o_2 - \Theta)}{\sigma_2^2} + \frac{(o_3 - \Theta)}{\sigma_3^2}. \tag{5.9}$$

From Eq.(5.9), the MAP estimation of Θ turns out to be:

$$\hat{\Theta} = \frac{\frac{o_1}{\sigma_1^2} + \frac{o_2}{\sigma_2^2} + \frac{o_3}{\sigma_3^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} + \frac{1}{\sigma_3^2}}. \tag{5.10}$$

This Bayesian solution for integrating the three observations, o_1, o_2, and o_3 is intuitively appealing. The larger the variance in the module output, the lower the influence of the module in the final outcome of the integration. The reader may also

notice that this solution also corresponds to the least squares solution.

We now present the integration problem in the context of 3D scene reconstruction as a classical Bayesian estimation problem. Let $\mathcal{I} = (\mathcal{I}_r, \mathcal{I}_l) \in \mathcal{D}^N$ represent the right and left (stereo) images of size N $(= n \times n)$, where $\Re^N$ denotes an N–dimensional real vector. Let S represent the set of all sites (pixels) (x, y), $S = \{(x_i, y_i); i = 1, \ldots, n; j = 1, \ldots, n\}$. Let $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_K\}$, $\mathcal{D}_k \in \Re^N$, be the k^{th} component of the intrinsic map. For example, $\mathcal{D}_i$ could be the boundary, depth, or surface reflectance map. The generic problem of 3D reconstruction, given $\mathcal{I}$, can then be stated as finding the *maximum a posteriori* estimate of the intrinsic map:

$$\hat{\mathcal{D}} = \arg\max_{\mathcal{D}} P(\mathcal{D}|\mathcal{I}), \tag{5.11}$$

where $P(\mathcal{D}|\mathcal{I})$ denotes the probability of intrinsic map $\mathcal{D}$, given the observed intensity maps, $\mathcal{I} = \{\mathcal{I}_l, \mathcal{I}_r\}$ (see Fig. 5.1).

In a modular integration, this basic 3D reconstruction problem (Eq. 5.11) is modified as follows (see Fig. 5.2). Let us assume that the vision module M_r can reliably estimate a function $F_r : \mathcal{I} \rightarrow \Re^N$ of (often, a projection of) the intrinsic map component, $\mathcal{D}_j$, $j = 1, \ldots, K^2$. The corresponding imaging process is described by $R_r : \mathcal{D} \rightarrow \mathcal{I}$. The modular integration problem can then be stated as follows: given $\mathcal{I}$ and a set of modules $\{M_r, r = 1, \ldots, m\}$, find an accurate and stable estimate of

[2]For instance, some modules do not directly determine depth but only depth derivatives, *e.g.*, surface normals.

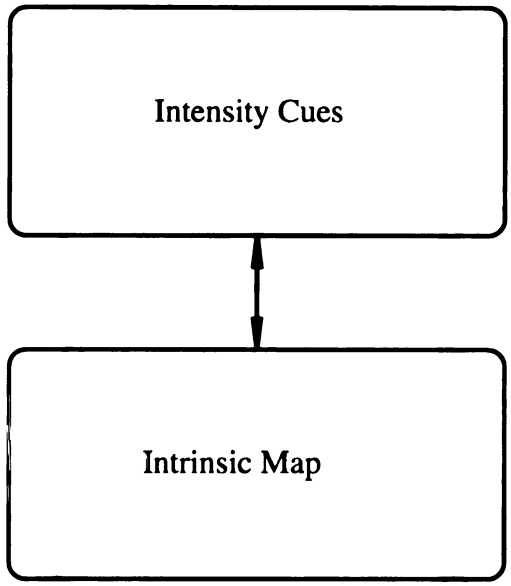


Figure 5.1: Generic Integration Problem.

the intrinsic map $\mathcal{D}$:

$$\hat{\mathcal{D}} = \arg\max_{\mathcal{D}} P(R_1(\mathcal{D}), \ldots, R_m(\mathcal{D})|\mathcal{I}), \tag{5.12}$$

where $P(R_1(\mathcal{D}), \ldots, R_m(\mathcal{D})|\mathcal{I})$ denotes the joint probability of modules $R_1, \ldots, R_m$ producing images *similar* to the observed intensity images $\mathcal{I}$.

The formulation in Eq. (5.12) is a very difficult optimization problem for the following reasons. First, the number of random variables involved is extremely large; for a $n \times n$ image, there are n^2 random variables in each $\mathcal{D}_i$. Secondly, the inverse optical imaging problem (function F_r) being solved by each vision module is often underconstrained. Finally, it is not easy to separately model the interactions among the modules; even if the joint density of random variables (outputs) associated with different modules can be estimated, the resultant model will be extremely complex

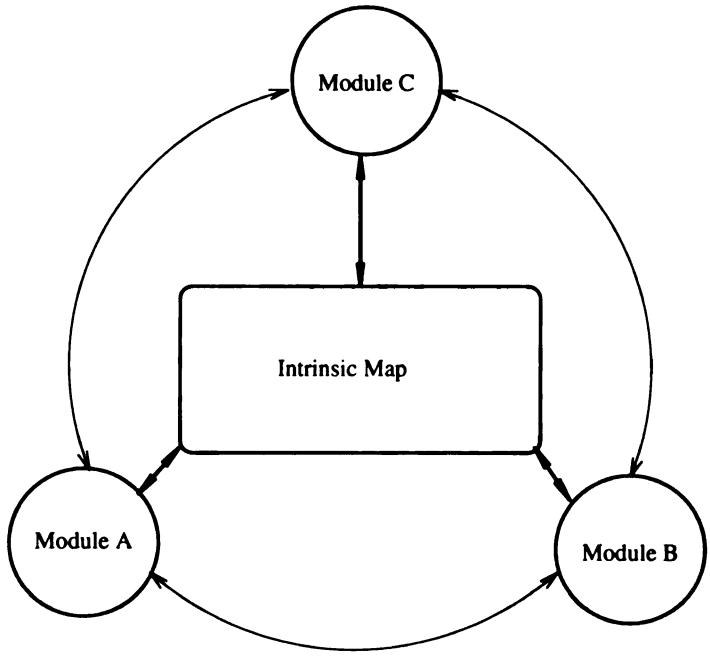


Figure 5.2: Modular Integration.

and difficult to optimize.

It is desirable to have a uniform interface mechanism to facilitate an extensible and a flexible overall system (even at the expense of accuracy). One pragmatic way to incorporate all these (integrated) system features is to make the operations of each module transparent to the operation of the other modules in the system. One of the simplifying assumptions is that at any stage, the intrinsic map could be updated by a given module by maximizing its (random variables associated with the intrinsic map) a posteriori probability independent of the other modules; the interaction of the modules is permitted only through a centralized intrinsic map. Each module observes the current state of the intrinsic map and the input intensity information to arrive at the best possible (MAP) refinement of the intrinsic map. The resultant iterative

strategy then is to obtain (see Fig. 5.3)

$$\hat{\mathcal{D}}^t_{r(t)} = \arg\max_{\mathcal{D}} P(R_{r(t)}(\mathcal{D})|\mathcal{D}^{t-1}_{r(t-1)}, \mathcal{I}), \tag{5.13}$$

where $r(t)$ defines a module $M_{r(t)}$ operating at time t and $\mathcal{D}^t_{r(t)}$ is an intermediate solution at time t proposed by module $M_{r(t)}$. Involving $\mathcal{D}^{t-1}_{r(t-1)}$ in the solution of $\mathcal{D}^t_{r(t)}$ makes the approach iterative and stable. Further, due to the distributed nature of this approach, it facilitates the implementation of each module *independent* of the implementation of the other modules. However, the solution of this new formulation is not guaranteed to coincide with that of Eq. (5.12) in all the situations. Under certain conditions, the iterative strategy converges to the correct solution[3]:

$$\lim_{t\to\infty} \hat{\mathcal{D}}^t_{r(t)} = \mathcal{D}. \tag{5.14}$$

Thus, in our Bayesian approach, the problem of *module integration* is formulated as the maximization of $P(R_r(\mathcal{D})|\mathcal{D}^{t-1}_{r(t-1)}, \mathcal{I})$, the posteriori density of a module M_r producing intensity maps similar to the observed $\mathcal{I}$ at time t (given the current solution $\mathcal{D}^{t-1}_{r(t-1)}$ and $\mathcal{I}$). Notice that since the operation of a typical vision module itself is very complex, finding the solution in Eq. (5.13) will essentially involve a (enumerative) search even if the module and its implementation were perfect. Our approach to obtain a practical solution to this problem consists of decomposing the

[3]The conditions under which the distributed solution corresponds to the exact solution are described in Chapter 2.

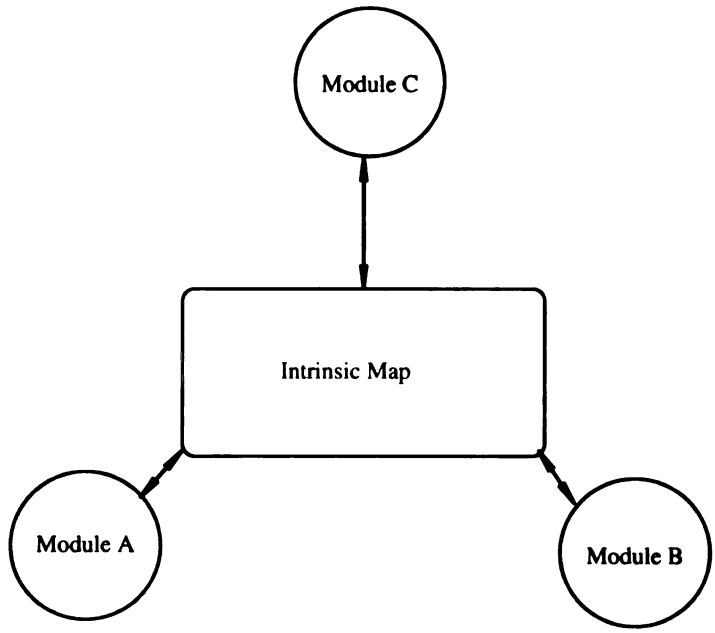


Figure 5.3: Modular Integration (restricted interactions).

original formulation into the three components described below. While each of these components could potentially make a suboptimal choice, we hope that an integrated system can tolerate these myopic decisions due to the multitude of constraints exerted by several modules, each compensating for the mistakes committed by the other. Such simplifications have been routinely (and successfully) incorporated into both natural and man-made system designs [182, 181, 156].

(i) **Intrinsic Map Estimation** The estimation of F_r by module M_r is known to be a difficult and an unstable problem. This can be considerably simplified by restating the original problem of *independently* estimating $F_r : \mathcal{I} \rightarrow \Re^N$ as an *incremental* (iterative) estimation problem:

$$\mathcal{U}_r : \mathcal{D}^{t-1}_{r(t-1)} \times \mathcal{I} \rightarrow \mathcal{D}^{t}_{r(t)}. \tag{5.15}$$

This function may determine a feature which completely constrains the depth values (*e.g.*, disparities in stereo module) or a depth-related feature which may partially constrain the depth map (*e.g.*, surface normals estimated by shape from shading). In some other modules (*e.g.*, perceptual organization module), this function determines the boundary segmentation which indirectly constrains the depth values [72]. The dynamics of such a feedback (involvement of $\mathcal{D}^{t-1}_{r(t-1)}$ in estimating $\mathcal{D}^{t}_{r(t)}$) often reduces sensitivity of the system to the violation of individual module assumptions and the operating parameters [42].

(ii) **Coherence Function**: To account for the spatial and systematic variations in the module performance, we associate a confidence map with each component of the intrinsic map and module outputs. These confidence maps are used to validate the individual module outputs.

Most vision modules are based on a simple imaging model. For instance, shape from shading and stereo modules often assume Lambertian surfaces. In some situations, model parameters are simply not available since they depend on (unknown) scene geometry. Many of the existing integration methodologies are biased against either data or synthetic constraints and, consequently, result in serious artifacts. It is desirable that erroneous outputs from the individual modules do not significantly degrade the output of the entire system and each additional module should leave the overall system output no worse than it originally found. In essence, we would like to assess the quality of the module output at each image site, and discard the noisy components of the feature map before

permitting this output to modify the intrinsic map. This problem can not be solved in its total generality. However, knowing that the features have originated from a physical surface helps us design a strategy based on the *principle of coherence* [154]. Simply stated, this principle hypothesizes that if there is a statistical interaction among an ensemble of estimates then they might have been derived from the same physical source(s).

Computation of coherence involves using the scene geometry and the existing intrinsic map. In our scheme, we validate a feature value at a location if it 'agrees' with its independent estimates derived from its neighbors. Note that we are not relating an estimate at a site with the estimates at its neighbors; we are relating an estimate of a variable at a site with the estimates of the *same* variable at that site derived from the information at the neighbors using a intrinsic map estimation function.

Let us illustrate a typical coherence function model:

$$\mathcal{C}_r : (\{\mathbf{P^1}, \mathbf{P^2}, \ldots, \mathbf{P^c}\}, \mathbf{P^0}, \rho) \rightarrow (\beta, \kappa, \hat{\mathbf{P}}),\ \beta, \kappa \in \Re.$$

Given a set of c attribute vectors $\{\mathbf{P^1}, \mathbf{P^2}, \ldots, \mathbf{P^c}\}$ and an attribute vector of the same type $\mathbf{P^0}$, a coherence function evaluates the likelihood that all the $(c+1)$ objects have originated from a single source and this likelihood is represented by the confidence value β. The parameter κ represents the confidence value that all members in $\{\mathbf{P^1}, \mathbf{P^2}, \ldots, \mathbf{P^c}\}$ are coherent. $\hat{\mathbf{P}}$ is the attribute vector which is considered to be a representative of the entire ensemble

$\{\mathbf{P^1}, \mathbf{P^2}, \ldots, \mathbf{P^c}\} \bigcup \{\mathbf{P^0}\}$, if $\beta > \rho$ and $\kappa > \rho$; ρ is a user-specified parameter. Given the intrinsic map, the updating mechanism seeks the opinion of each neighbor of a particular site (x, y) to predict the value of $F_r(x, y)$. If there is a consensus of opinions among the neighbors (according to the coherence function $\mathcal{C}_r$), and it is consistent with the $F_r(x, y)$ computed from $\mathcal{D}$ (since F_r is a known function of $\mathcal{D}$), then we increase our confidence in $\mathcal{D}(x, y)$. If there is a consensus of the opinions among the predictions of the neighbors, but it does not agree with $F_r(x, y)$ computed from $\mathcal{D}$, then we can replace $F_r(x, y)$ by the consensus value prescribed by $\mathcal{C}_r$. If there is no consensus among the predictions of neighbors of (x, y), then the value of $F_r(x, y)$ is not updated and we decrease the confidence value of $D_r(x, y)$.

(iii) **Consistency Mapping:** The problem of module integration (Eq. (5.13)) then reduces to obtaining the most likely and consistent (re)interpretation of the current solution $\mathcal{D}^{t-1}_{r(t-1)}$ in the light of new (validated) evidence $\mathcal{U}_r(\mathcal{D}^{t-1}_{r(t-1)}, \mathcal{I})$.

$$\hat{\mathcal{D}}^{t}_{r(t)} = \arg\max_{\mathcal{D}} P(D|\mathcal{U}_r(\mathcal{D}^{t-1}_{r(t-1)}, \mathcal{I}), \mathcal{D}^{t-1}_{r(t-1)}). \tag{5.16}$$

The integration scheme can now be expressed in terms of three major steps: (i) a method for module output estimation (Eq. (5.15)), (ii) a method for estimating the reliability of the module output, and (iii) a method for deriving the validated[4] module output using (Eq. (5.16))[5].

[4] A module output is validated using the associated confidence map.

[5] For conciseness, we will be referring to $\mathcal{D}^{t-1}_{r(t-1)}$ as $\mathcal{D}^{obs}$.

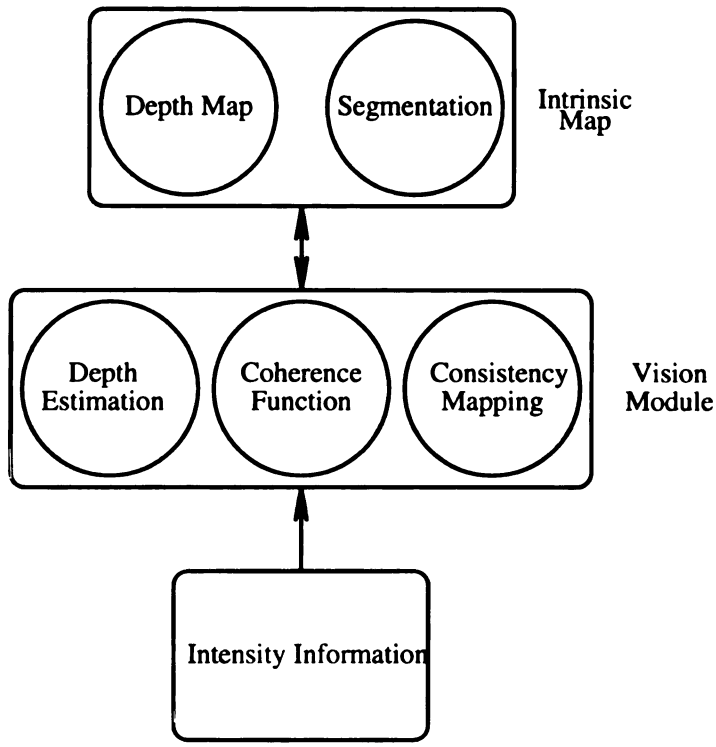


Figure 5.4: Interaction Model (for one module at one level of resolution).

Fig. 5.4 shows our interaction model for one module at one level of resolution. In this model, each module interacts only with the intrinsic map. Given a pair of intensity images and an initial estimate of intrinsic map, each module refines a component of the intrinsic map. Coherence module assesses the refined map by the degree of consensus among the independent estimates. Incoherent and noisy estimates are discarded and used to refine the segmentation of the image. Finally, the intrinsic map is updated using *consistency mapping*, so that it is consistent with the 'coherent' estimates of the module.

This basic model described above needs to be supplemented with the additional mechanisms to deal with the following issues:

1. **Scale of Integration** The interaction model will typically also involve **neigh-**

borhood definition $\mathcal{N}_r : S \rightarrow 2^S$, where 2^S is the set of all the subsets of S. Given a site $(x, y) \in S$, $\mathcal{N}_r$ defines a set of sites which are considered as *neighbors* of (x, y).

2. **Order of Module Operation:** The proposed integration scheme in itself does not impose any ordering constraints on the module operations. In fact, it is possible to implement the operation of the module concurrently. In our sequential implementation (see Fig. 5.5), we adopt the following sequence of module operations: perceptual grouping, stereo, shading, and shape from texture.

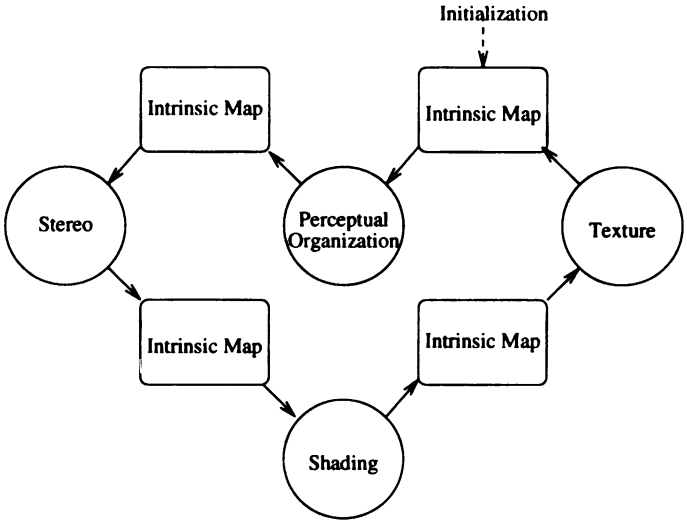


Figure 5.5: Sequence of Module Operation (at one level of resolution).

3. **Initial Intrinsic Map** How should one obtain the *initial* guess for the intrinsic map? This problem can be considerably alleviated by (i) the multi-grid, multi-resolution processing (see Fig. 5.6), (ii) default assumptions, and (iii) proper choice of models. Our system starts out with a planar depth map. A coarse-level solution guides the next finer level solution; the data-driven mechanisms

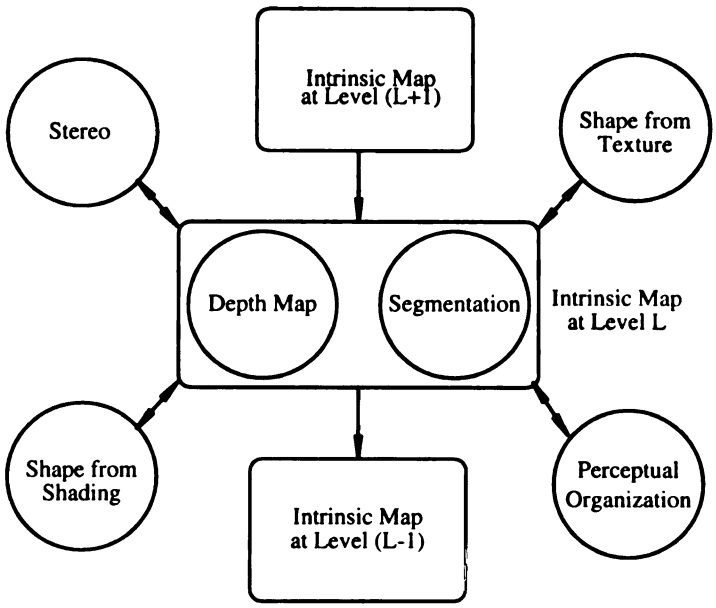


Figure 5.6: System Integration.

will iteratively refine the initial output into more accurate intrinsic maps at finer levels.

4. **Segmentation issues:** Most shape-from-X algorithms are sensitive to the domain assumptions. Indiscriminate application of a module to an arbitrary region in an image usually results in nonsensical results. While it is desirable to localize and restrict the scope of each module to appropriate regions in the image, it is difficult in practice to prespecify these regions. The principle of coherence permits not only the rejection of the unreliable results from a given module, but also a gradual refinement of the existing segmentation to eventually achieve an 'emergent' segmentation.

Table 5.1: Vision modules for the proposed integration.

Module	Strengths	Problems
Stereo	Reliable short-range depth information	Correspondence and Occlusion
Shape from Shading	Orientation estimation irrespective of distance	Mutual illumination and fine textures
Shape from Texture	Orientation estimation irrespective of distance	Texture segmentation, uniform albedo
Perceptual Organization	Boundary completion	Oversegmentation

5.2 Modules for Integration

Four modules have been chosen for demonstrating the utility of the proposed integration framework: shape from shading, shape from texture, stereo, and perceptual organization. Only the first three of these modules directly determine the depth; the importance of these low-level cues in determining depth has been recognized in the literature [116] and they display complementary strengths and limitations (see Table 5.1). Perceptual organization module has been known to interact with every other vision module and it helps determine depth by providing additional constraints in adverse situations. While these four modules form a good combination for illustrating the concept of the integration methodology, note that any other module can be incorporated into the implementation with equal ease because of our "uniform" integration methodology. In this section, we will describe the problem solved by each individual module. We will then present our formulation of interaction between the individual modules and the intrinsic map.

5.2.1 Perceptual Organization Module

The primary role of the perceptual organization module in our integrated system is to estimate and refine significant 3D boundaries. This boundary segmentation is difficult due to several confounding factors, including imaging noise and artifacts of the boundary detection operators. We have, therefore, collated different sources of information to reliably estimate these boundaries. A secondary role of the perceptual organization module is anisotropic diffusion of the depth features, where the estimated boundaries act as diffusion barriers [72].

Rewriting the generic modular integration equation (Eq. (5.16)) for grouping[6] module,

$$\hat{\mathcal{D}}^t_{gp} = \arg\max_{\mathcal{D}} P(\mathcal{D}|\mathcal{U}_{gp}(\mathcal{D}^{obs}, \mathcal{I}), \mathcal{D}^{obs}), \tag{5.17}$$

where, for conciseness, we have replaced $\mathcal{D}^{t-1}_{r(t-1)}$ by $\mathcal{D}^{obs}$. In our present implementation, grouping module does not affect the depth map and refines the boundary map only. Eq. (5.17) can, therefore, be written as:

$$\hat{\mathbf{B}}^t_{gp} = \arg\max_{\mathbf{B}} P(\mathbf{B}|\mathcal{U}_{gp}(\mathbf{B}^{obs}, \mathcal{I}), \mathbf{B}^{obs}). \tag{5.18}$$

$\mathbf{B}$ is the boundary component of the intrinsic map. For conciseness, we will refer to $\mathcal{U}_{gp}(\mathbf{B}^{obs}, \mathcal{I})$ as $\mathbf{B}^{gp}$. Thus, the term $P(\mathbf{B}|\mathbf{B}^{gp}, \mathbf{B}^{obs}))$ denotes the posteriori probability of boundary map $\mathbf{B}$, given the boundary map produced by grouping module ($\mathbf{B}^{gp}$) and the current state of boundary map ($\mathbf{B}^{obs}$).

[6] We will be using perceptual organization module and grouping module synonymously.

Boundary Estimation ($\mathcal{U}_{gp} : \mathcal{D}^{obs} \times \mathcal{I} \rightarrow \mathbf{B}$): We use (i) an edge segmentation of the intensity image using Canny edge operator [35]), (ii) a region segmentation of the intensity image using a split-and-merge algorithm (Pavlidis [149]), (ii) an edge segmentation of current depth map (from $\mathcal{D}^{obs}$) using a Sobel operator, and (iv) the current boundary map to obtain an initial estimate for the object boundaries. Each terminal boundary pixel in the current boundary map $\mathbf{B}^{obs}$ is extended in the tangential direction when supported by two of the three remaining segmentations. Let us call this representation $\mathbf{B}'$. We then detect corners and terminations in the resultant representation. A Voronoi tessellation $\mathbf{V}$ of the corner and termination pixels augments $\mathbf{B}'$. Thus, $\mathbf{B}^{\star} = \mathbb{B}' \bigcup \mathbf{V}$ is an initial estimate of the refined boundary map.

Coherence Function: The design of this component is based on the grouping module described in Section 5.2.1. The coherence at a boundary site $b_i \in \mathbf{B}'$ is estimated as:

$$\beta_i = \frac{\sum_{j=1}^{r} b_i^j}{r}, \tag{5.19}$$

where $r(=4)$ is the number of segmentations participating in the perceptual grouping and b_i^j is 1 if a boundary exists at the site i in the j^{th} segmentation.

The estimation of coherence of Voronoi edges $v_i \in \mathbf{V}$ at site i uses a different procedure[7]. The significance of each Voronoi edge is determined by the following factors:

1. **Proximity** Voronoi edges which are short will be considered perceptually more

[7]Some of this description is already described in Section 5.2.1

significant than those which are longer. The contribution of a Voronoi edge of length d connecting the edges of length D_1 and D_2 is defined as $E_p = w_p d^2/D_1 D_2$, where w_p is the relative significance of proximity attribute.

2. **Curvilinearity** Voronoi edges which are in the tangential direction of the edge terminations are more significant. A Voronoi edge which subtends angles θ_1 and θ_2 with the terminations of the edges at its either end will contribute $E_l = w_s(\theta_1 + \theta_2)/4\pi$ to the objective function, where w_s is the relative significance of the curvilinearity.

3. **Cotermination** It is the common point shared by terminations of two (or more) smooth boundaries. Coterminations are perceptually significant. We estimate the contribution of a cotermination to the objective function by $E_c = w_c 2/(n_1 + n_2)$, where n_1 and n_2 are the numbers of Voronoi neighbors of each termination and w_c is the weight indicating the perceptual significance of cotermination.

4. **Depth Gradient** Significance of a Voronoi edge is proportional to the average depth difference across its length: $E_d = w_d \frac{\sum \Delta D}{L}$, where ΔD is the difference of depth across the Voronoi edge of length L.

5. **Brightness Gradient** Significance of a Voronoi edge is proportional to the average intensity difference of the regions it abuts. Instead of computing a raw intensity gradient, we consider difference in means of intensity values of the adjoining regions as a reliable indicator of this criteria. The contribution of

the brightness gradient is measured by $E_g = w_g \frac{M}{G}$, where M is the maximum number of gray levels in the image and G is the average difference of intensity across the length of the edge.

Total perceptual significance of a Voronoi edge is

$$E = E_p + E_l + E_c + E_g + E_d. \tag{5.20}$$

The coherence of a Voronoi edge v_i, then, is determined as follows:

$$\beta_i = \frac{E}{w_p + w_l + w_c + w_g + w_d} \tag{5.21}$$

From the set of c Voronoi edge alternatives at an edge termination (or a corner) with the corresponding coherence s $\{\beta^1, \beta^2, \ldots, \beta^c\}$, all the Voronoi edges except with $\kappa_{gp} = \beta_{gp} = \max_{i=1}^{c} \beta^i$ are discarded, provided that its coherence is above a threshold ρ_{gp}.

Consistency Mapping

$$P(B|B^{gp}, B^{obs}) = \frac{P(B^{gp}, B^{obs}|B)P(B)}{P(B^{gp}, B^{obs})}, \tag{5.22}$$

where B is a binary map of the refined boundaries, and B^{gp} and B^{obs} are the output of the perceptual grouping module and the current boundary map (from the intrinsic map), respectively. As the denominator in Eq.(5.22) is independent of B, the maximization of $P(B|B^{gp}, B^{obs})$ essentially requires maximization of the product

$P(B^{gp}, B^{obs}|B)P(B)$. The first term in the product is the data term which depends on the output of perceptual organization module. The second (model) term imposes desirable characteristics of a boundary map. The present implementation assumes a spatially uniform distribution of boundary elements; our problem then reduces to maximization of $P(B^{gp}, B^{obs}|B)$.

The data term, $P(B^{gp}, B^{obs}|B)$ is given by (Chu and Aggarwal [41]):

$$P(B^{gp}(i,j), B^{obs}(i,j)|B(i,j)) \propto$$
$$\frac{1}{\sqrt{\beta_{gp}}} \exp\left(- \sum_{B^{gp}(k,l) \bigcup B^{obs}(k,l) \in \mathcal{N}(i,j)} \beta_{gp}(k,l) \| \left((k,l) - (i,j)\right) \| \right), \qquad (5.23)$$

where $\mathcal{N}(i,j)$ denotes the neighborhood of site (i,j). We make the simplifying assumption that the scale factor $\frac{1}{\sqrt{\beta_{gp}}}$ does not significantly influence the form of the $P(B^{gp}, B^{obs}|B)$ and consequently discard it[8]. By assuming spatial independence among the boundary elements, we have

$$P(B^{gp}, B^{obs}|B)) \propto \prod_{(i,j)} P(B^{gp}(i,j), B^{obs}(i,j)|B(i,j))$$
$$\propto \exp\left(- \sum_{(i,j)} \sum_{B^{gp}(k,l) \bigcup B^{obs}(k,l) \in \mathcal{N}(i,j)} \beta_{gp}(k,l) \left(\|(k,l) - (i,j)\| \right) \right) \qquad (5.24)$$

The overall optimization problem then reduces to maximizing individual terms at each site, $P(B^{gp}(i,j), B^{obs}(i,j)|B(i,j))$. Let $S(i,j)$ be the set of all boundary coordinates in a neighborhood of (i,j) in the set $B^{gp} \bigcup B^{obs}$. The required optimization

[8]This assumption about the scale factor is not often true and has negative impact on the performance of the integrated system. However, use of this assumption makes the solution tractable.

problem reduces to the following closed-form solution:

$$B(i,j) = \begin{cases} 1 & \exists\ (k,l) \text{ such that } (i,j) = \frac{\sum \beta_{gp}(k,l) S(k,l)}{|\beta_{gp}(k,l)|} \\ 0 & \text{otherwise} \end{cases} \tag{5.25}$$

5.2.2 Stereo Module

Rewriting the generic modular integration equation (Eq. (5.16)) for stereo module,

$$\hat{\mathcal{D}}^t_{st} = \arg\max_{\mathcal{D}} P(\mathcal{D}|\mathcal{U}_{st}(\mathcal{D}^{obs}, \mathcal{I}), \mathcal{D}^{obs}), \tag{5.26}$$

where, for conciseness, we have replaced $\mathcal{D}^{t-1}_{r(t-1)}$ by $\mathcal{D}^{obs}$. In our present implementation, stereo module does not affect the boundary map and refines the depth map only. Eq. (5.26) can, therefore, be written as:

$$\hat{\mathbf{D}}^t_{st} = \arg\max_{\mathbf{D}} P(\mathbf{D}|\mathcal{U}_{st}(\mathcal{D}^{obs}, \mathcal{I}), \mathcal{D}^{obs}), \tag{5.27}$$

where $\mathbf{D}$ is the depth component of the intrinsic map. For conciseness, we will refer to $\mathcal{U}_{st}(\mathcal{D}^{obs}, \mathcal{I})$ as $\mathbf{D}^{st}$. Thus, the term $P(\mathbf{D}|\mathbf{D}^{st}, \mathcal{D}^{obs}))$ denotes the posteriori probability of depth map $\mathbf{D}$, given the depth map produced by stereo ($\mathbf{D}^{st}$) and the current state of intrinsic map ($\mathcal{D}^{obs}$).

We use the stereo module proposed by Weng *et al.* [195] (see Section 3.3). The components for integrating the stereo module into the system are described below.

Disparity Estimation ($\mathcal{U}_{st} : \mathcal{D}^{obs} \times \mathcal{I} \rightarrow \mathcal{P}$): Let $\mathbf{d}^l$, $\mathbf{d}^r$ be the current right and left disparity maps derived from $\mathbf{D}^{obs}$. The two-way matching results in two

correspondence vector fields, $\mathcal{P}^l$ and $\mathcal{P}^r$ ($\mathbf{u} \equiv (i,j)$):

$$\mathcal{P}^l(\mathbf{u}) = \arg \min_{\mathbf{d} \in \mathcal{N}(\mathbf{d}^l(\mathbf{u}))} \sum_{\mathbf{u} \in \mathcal{I}_l} \sum_i \mathbf{w}_i [\mathbf{R}_i^2(\mathbf{u}, \mathbf{d})], \tag{5.28}$$

where $\mathcal{N}(.)$ denotes a neighborhood function. Similarly,

$$\mathcal{P}^r(\mathbf{u}) = \arg \min_{\mathbf{d} \in \mathcal{N}(\mathbf{d}^r(\mathbf{u}))} \sum_{\mathbf{u} \in \mathcal{I}_r} \sum_i \mathbf{w}_i [\mathbf{R}_i^2(\mathbf{u}, \mathbf{d})], \tag{5.29}$$

where $\mathbf{R_i}(\mathbf{u}, \mathbf{d})$ is the residual contributed by the i^{th} attribute image at location $\mathbf{u}$ due to correspondence vector $\mathbf{d}$, and $\mathbf{w}_i$ is a prespecified weight associated with the residual $\mathbf{R_i}$.

Neighborhood Definition: A standard 4-pixel neighborhood $\mathcal{N}$ is used.

Coherence Definition: Two coherence functions are defined, one each for left and right stereo matching. Here, we describe the coherence function for the right stereo matching. We hypothesize a correspondence at (x,y) defined by $\mathcal{P}_r(x,y)$ to be coherent if the back-projection of neighborhood of its correspondence, $\mathcal{N}((x,y)+\mathcal{P}_r(x,y))$, prescribed by $\mathcal{P}_l$ falls in the neighborhood $\mathcal{N}(x,y)$. Farther the back-projected correspondences fall, lower is the confidence.

$$C_{stereo}(\mathbf{F}(x,y), \tilde{\mathcal{P}}(x,y), \rho_{stereo}) = (\beta_{stereo}, \kappa_{stereo}, \hat{\mathcal{P}}(x,y)), \tag{5.30}$$

where $\tilde{\mathcal{P}}(x,y)$ is the disparity at (x,y) determined by the intrinsic map, $\mathbf{F}(x,y) = \{(m,n) \equiv (k,l) + \mathcal{P}_l(k,l); (k,l) \equiv (i,j) + \mathcal{P}_r(i,j); \forall (i,j) \in \mathcal{N}(x,y)\}$ is the set of disparity estimates at (x,y) by the stereo module in the neighborhood of (x,y). Like-

lihoods β_{stereo} and κ_{stereo} are measured using the variance estimates of $\mathbf{F}(x,y)$ and $\mathbf{F}(x,y) \bigcup \{\tilde{\mathcal{P}}(x,y)\}$, respectively.

$$\hat{\mathcal{P}}(x,y) = \begin{cases} \tilde{\mathcal{P}}(x,y) & \text{if } \rho_{stereo} > \beta_{stereo} \\ median(\mathbf{F}(x,y) \bigcup \{\tilde{\mathcal{P}}(x,y)\}) & \text{if } \rho_{stereo} > \kappa_{stereo} \\ median(\mathbf{F}(x,y)) & \text{otherwise} \end{cases} \tag{5.31}$$

We have used $\rho_{stereo} \leq 4$ pixels.

Consistency Mapping Given the depth maps $\mathbf{D^l}$ and $\mathbf{D^r}$ corresponding to the left and right (validated) disparity maps, $\mathcal{P}^l$ and $\mathcal{P}^r$ and their associated likelihoods $\beta^{\mathbf{l}}$ and $\beta^{\mathbf{r}}$, how to refine the estimate of the depths $\mathbf{D}$ in the intrinsic map? We resort to the Bayesian estimation. Let $\mathbf{D^{st}}$ be the depth map estimated by a (right or left) stereo module and $\beta^{\mathbf{st}}$ be the associated estimate of its likelihood [9] ($st \in \{l,r\}$):

$$P(\mathbf{D}|\mathbf{D}^{st}, \beta^{\mathbf{st}}, \mathbf{D}^{obs}) = \frac{P(\mathbf{D^{st}}, \beta^{\mathbf{st}}, \mathbf{D}^{obs}|\mathbf{D})P(\mathbf{D})}{P(\mathbf{D^{st}}, \beta^{\mathbf{st}}, \mathbf{D}^{obs})}. \tag{5.32}$$

Since the denominator in Eq. (5.32) is independent of $\mathbf{D}$, the MAP estimate of $\mathbf{D}$ involves maximizing the product of $P(\mathbf{D}^{st}, \beta^{st}, \mathbf{D}^{obs}|\mathbf{D})$ and $P(\mathbf{D})$.

Let $\mathbf{B} \equiv \{B(i,j)|i = 1,\ldots,n; j = 1,\ldots,n\}$ represent the binary boundary map. $B(i,j) = 1$ indicates the presence of a boundary at location (i,j) and $\mathbf{B}$ defines closed regions $\mathcal{R}^r$, $r = 1,\ldots,p$. The data dependent term $P(\mathbf{D^{st}}, \beta^{st}, \mathbf{D}^{obs}|\mathbf{D})$ at location

[9] Eq. (5.32) is applied twice; once for the left stereo matching and once for the right stereo matching.

(i,j) can be estimated as

$$P(\mathbf{D^{st}}(i,j),\beta^{st}(i,j),\mathbf{D}^{obs}(i,j)|\mathbf{D}(i,j)) \propto \frac{1}{\sqrt{\beta_{st}(i,j)}}\exp(-k_d e_d(i,j)), \tag{5.33}$$

$$e_d(i,j) = \beta^{st}(i,j)(D^{st}(i,j) - D(i,j))^2. \tag{5.34}$$

We make the simplifying assumption that the effect of the scale factor $\frac{1}{\sqrt{\beta_{st}(i,j)}}$ is negligible and discard it. By assuming that the likelihoods $e_d(i,j)$ are spatially independent, we obtain,

$$P(\mathbf{D}|\mathbf{D}^{st},\beta^{\mathbf{st}},\mathbf{D}^{obs}) \propto exp(-k_d e_d) = \prod_{(i,j)} \exp(-k_d e_d(i,j)) \tag{5.35}$$

$$e_d = \sum_i \sum_j \beta^{st}(i,j)(D^{st}(i,j) - D(i,j))^2. \tag{5.36}$$

The prior term $P(\mathbf{D})$ is chosen to favor smooth depth profiles:

$$P(\mathbf{D}(i,j)) \propto exp(-k_s e_s(i,j)), \tag{5.37}$$

where

$$e_s(i,j) = D_x^2(i,j) + D_y^2(i,j), \tag{5.38}$$

and $D_x(i,j)$ and $D_y(i,j)$ denote partial derivatives of the depth map in x and y direction, respectively. These partial derivatives are estimated as follows.

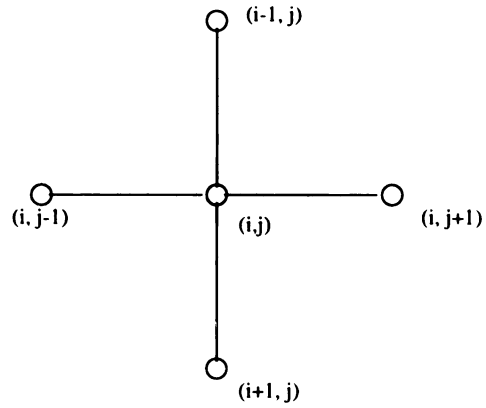


Figure 5.7: 4-neighborhood system.

$$D_x(i,j) = \begin{cases} D(i+1,j) - D(i,j) & if\ (i+1,j),(i,j) \in \mathcal{R}^p, \\ 0 & otherwise \end{cases} \tag{5.39}$$

$$D_y(i,j) = \begin{cases} D(i,j+1) - D(i,j) & if\ (i,j+1),(i,j) \in \mathcal{R}^p, \\ 0 & otherwise \end{cases} \tag{5.40}$$

By assuming that the likelihoods $e_s(i,j)$ are spatially independent, we obtain,

$$P(\mathbf{D}) \propto exp(-k_s e_s) = \prod_{(i,j)} exp(-k_s e_s(i,j)), \tag{5.41}$$

where

$$e_s = \sum_i \sum_j \left[D_x^2(i,j) + D_y^2(i,j) \right]. \tag{5.42}$$

Thus MAP estimation in Eq. (5.32) is identical to maximizing

$$P(\mathbf{D}|\mathbf{D}^{st}, \beta^{\mathbf{st}}, \mathbf{D}^{obs}) \propto \exp(-k_d e_d) \exp -(k_s e_s) \tag{5.43}$$

$$\propto \exp(-(k_d e_d + k_s e_s)) \tag{5.44}$$

$$\propto \exp(e_d + \lambda_s e_s). \tag{5.45}$$

From Eq. (5.45), we observe that the problem of MAP estimation is identical to minimizing

$$e_{st} = e_d + \lambda_{st} e_s \tag{5.46}$$

$$e_{st} = \sum_i \sum_j \left[\beta^{st}(D^{st}(i,j) - D(i,j))^2\right] + \lambda_{st} \sum_i \sum_j \left[D_x^2(i,j) + D_y^2(i,j)\right]. \tag{5.47}$$

If we assume a 4-neighborhood system, then the resultant stereo estimate is given by the following equation [15]:

$$D(i,j) = \frac{\beta^{st}(i,j) D^{st}(i,j) + \lambda_s \overline{D}(i,j)}{\beta^{st}(i,j) + \lambda_s}, \tag{5.48}$$

where $\overline{D}(i,j)$ is the local (4-neighborhood) average at $\mathbf{D}(i,j)$. The true average ($\overline{D}(i,j)$) is not known and is approximated by $\overline{D}^{obs}(i,j)$.

5.2.3 Shape From Shading

Rewriting the generic modular integration equation (Eq. (5.16)) for the shading module,

$$\hat{\mathcal{D}}_{sh}^t = \arg\max_{\mathcal{D}} P(\mathcal{D} | \mathcal{U}_{sh}(\mathcal{D}^{obs}, \mathcal{I}), \mathcal{D}^{obs}), \tag{5.49}$$

where, for conciseness, we have replaced $\mathcal{D}_{r(t-1)}^{t-1}$ by $\mathcal{D}^{obs}$. The shape from shading module does not affect the boundary map and refines the depth map only. Eq. (5.49)

can, therefore, be written as:

$$\hat{\mathbf{D}}^{t}_{sh} = \arg\max_{\mathbf{D}} P(\mathbf{D}|\mathcal{U}_{sh}(\mathcal{D}^{obs}, \mathcal{I}), \mathcal{D}^{obs}), \tag{5.50}$$

where $\mathbf{D}$ is the depth component of the intrinsic map. For conciseness, we will refer to $\mathcal{U}_{sh}(\mathcal{D}^{obs}, \mathcal{I})$ as $\mathbf{D}^{sh}$. Thus, the term $P(\mathbf{D}|\mathbf{D}^{sh}, \mathcal{D}^{obs}))$ denotes the a posteriori probability of depth map $\mathbf{D}$, given the depth map produced by shading ($\mathbf{D}^{sh}$) and the current state of intrinsic map ($\mathcal{D}^{obs}$).

Given a Lambertian image surface with constant albedo η, the image irradiance equation relates the brightness (E) at a position (x, y) with the surface normal, $\mathbf{n} \equiv (n_x, n_y, n_z)$, by $E(x, y) = \eta \mathbf{n_s}.\mathbf{n}$, where $\mathbf{n_s}$ is the unit vector pointing towards the illumination source. The goal of shape from shading module is to recover the surface orientation (normal) $\mathbf{n}$ at each pixel, given the intensity image.

Surface Normal Estimation ($\mathcal{U}_{sh} : \mathcal{D}^{obs} \times \mathcal{I} \rightarrow \mathcal{O}$): Given a surface orientation $\mathbf{n}(x, y)$ at site (x, y), usually two solutions exist for surface normals for the neighbors in the directions ϕ and $-\phi$. These solutions correspond to the maxima of a constant brightness equation [80]. Given the current estimates of the depth (from the intrinsic map) and intensity image, our shape from shading module directly estimates the depth at each pixel using Oliensis's algorithm [144]. However, the depth information thus obtained is only qualitatively correct; the consistency mapping function is designed to utilize the relative depth information from the depths obtained by the shape from shading module.

Neighborhood definition: We use a standard 8-pixel neighborhood, $\mathcal{N}$.

Coherence Function:

$$C_{shading}(\mathbf{B}(x,y), \tilde{\mathbf{n}}(x,y), \rho_{shading}) = (\beta_{shading}, \kappa_{shading}, \hat{\mathbf{n}}(x,y)), \tag{5.51}$$

where $\tilde{\mathbf{n}}(x,y)$ is the surface orientation at (x,y) computed from intrinsic map, $\mathbf{B}(x,y) = \{\mathbf{n}(i,j), (i,j) \in \mathcal{N}(x,y)\}$ is the set of surface orientation estimates at (x,y) by shape from shading module in the neighborhood of (x,y). Likelihoods $\beta_{shading}$ and $\kappa_{shading}$ are measured using the variance estimates of $\mathbf{B}(x,y)$ and $\mathbf{B}(x,y) \bigcup \{\tilde{\mathbf{n}}(x,y)\}$, respectively. We define the median of a set of vectors to be the nearest Euclidean neighbor of median of individual components and variance of the set to be an arithmetic mean of the individual component variances.

$$\hat{\mathbf{n}}(x,y) = \begin{cases} \tilde{\mathbf{n}}(x,y) & \text{if } \rho_{shading} > \beta_{shading} \\ median(\mathbf{B}(x,y) \bigcup \{\tilde{\mathbf{n}}(x,y)\}) & \text{if } \rho_{shading} > \kappa_{shading} \\ median(\mathbf{B}(x,y)) & \text{otherwise} \end{cases} \tag{5.52}$$

Consistency Mapping Usually, albedo of the scene surfaces is not known. In such a situation, the depth map obtained from the shape from shading module is only qualitatively correct and can not be directly related to the depth map obtained from stereo module. Then, how should the information obtained from the shape from shading module be meaningfully used to improve the depth map obtained from stereo? Given a current estimate of the depth, the boundaries from the intrinsic map and an estimate of the shape from the shading module, we propose to use a Bayesian strategy to arrive at a refined estimate of the depth map.

Let $\mathbf{D} \equiv \{D(i,j)|i = 1,\ldots,n; j = 1,\ldots,n\}$ represent depth map random variables[10]. Let $\mathbf{B} \equiv \{B(i,j)|i = 1,\ldots,n; j = 1,\ldots,n\}$ represent the binary boundary map. $B(i,j) = 1$ indicates presence of a boundary at location (i,j) and $\mathbf{B}$ defines closed regions $\mathcal{R}^r$, $r = 1,\ldots,p$. $\mathbf{D}^{sh} \equiv \{D^{sh}(i,j)|i = 1,\ldots,n; j = 1,\ldots,n\}$ denotes the 'depth' map obtained using shape from shading module.

$$P(\mathbf{D}|\mathbf{D}^{sh},\mathbf{D}^{obs}) = \frac{P(\mathbf{D}^{sh},\mathbf{D}^{obs}|\mathbf{D})P(\mathbf{D})}{P(\mathbf{D}^{sh})} \tag{5.53}$$

The denominator is independent of $\mathbf{D}$, therefore, the estimation of refined depths reduces to computation of $P(\mathbf{D}^{sh},\mathbf{D}^{obs}|\mathbf{D})$ and $P(\mathbf{D})$. The first term is referred to as the *data* term and the latter term is referred to as the *model* term or the *prior* term. We model the data term as the consistency between the existing depth map and the (relative) depth map obtained from the shading module; the prior term is modeled as the piecewise smooth depth map. These terms are described below.

1. *Prior density of* $\mathbf{D}$, $P(\mathbf{D})$: The surface depth varies smoothly over each image region of constant albedo. The measure of departure from smoothness of the surface at pixel (i,j) can be expressed by $(\mathbf{D}_x(i,j)^2 + \mathbf{D}_y(i,j)^2)$, where $\mathbf{D}_x(i,j)$ and $\mathbf{D}_y(i,j)$ represent partial derivatives of depth map at $\mathbf{D}(i,j)$, in x and y direction, respectively.

$$P(D(i,j)) \propto \exp(-k_s e_s(i,j)), \tag{5.54}$$

$$e_s(i,j) = \sum_i \sum_j (D_x(i,j)^2 + D_y(i,j)^2), \tag{5.55}$$

[10]Recall that $N(= n \times n)$ is the image size.

where k_s is a constant. The partial derivatives are estimated as follows.

$$D_x(i,j) = \begin{cases} D(i+1,j) - D(i,j) & if\ (i+1,j),(i,j) \in \mathcal{R}^p, \\ 0 & otherwise \end{cases} \tag{5.56}$$

$$D_y(i,j) = \begin{cases} D(i,j+1) - D(i,j) & if\ (i,j+1),(i,j) \in \mathcal{R}^p, \\ 0 & otherwise \end{cases} \tag{5.57}$$

By assuming that the likelihoods $e_s(i,j)$ are spatially independent, we obtain,

$$P(\mathbf{D}) \propto exp(-k_s e_s) = \prod_{(i,j)} exp(-k_s e_s(i,j)), \tag{5.58}$$

$$e_s = \sum_i \sum_j \left[D_x^2(i,j) + D_y^2(i,j) \right]. \tag{5.59}$$

2. *Data Term,* $P(\mathbf{D}^{sh}, \mathbf{D}^{obs}|\mathbf{D})$: The resultant shape obtained by the integrated system should conform with the shape obtained by the shading module. More specifically, the surface orientation at each point (i,j) of the reconstructed surface, D, should be consistent with the orientation of the surface estimated at that point by the shading module alone. The aforementioned consistency can be measured using:

$$P(\mathbf{D}^{sh}(i,j), \mathbf{D}^{obs}(i,j)|\mathbf{D}(i,j)) \propto$$

$$\frac{1}{\sqrt{\beta_{shading}}} \exp\left(-k_c \beta_{shading}(i,j) e_c(i,j)\right), \tag{5.60}$$

(5.61)

where

$$e_c(i,j) = \; (D_x(i,j) - D_x^{sh}(i,j))^2 + \\ (D_y(i,j) - D_y^{sh}(i,j))^2, \tag{5.62}$$

where k_c is a constant and D_x^{sh} and D_y^{sh} are defined similar to D_x and D_y (Eqs.(5.56) and(5.57)), respectively. We make the simplifying assumption that the effect of the scale factor $\frac{1}{\sqrt{\beta_{shading}}}$ is negligible and hence discard it.

By assuming that the likelihoods $e_c(i,j)$ are spatially independent, we obtain,

$$P(\mathbf{D}^{sh}, \mathbf{D}^{obs}|\mathbf{D}) \propto \; \exp\left(-k_c \beta_{shading} e_c\right) = \\ \prod_{(i,j)} \exp\left(-k_c \beta_{shading}(i,j) e_c(i,j)\right), \tag{5.63}$$

(5.64)

where

$$e_c \; = \sum_i \sum_j \left[\, (D_x(i,j) - D_x^{sh}(i,j))^2 \right. \\ \left. + (D_y(i,j) - D_y^{sh}(i,j))^2 \right], \tag{5.65}$$

Combining Eqs. (5.53), (5.54), and (5.63), we have

$$P(\mathbf{D}|\mathbf{D}^{sh}, \mathbf{D}^{obs}) \propto \; \exp\left(-k_s e_s\right) \exp\left(-k_c \beta_{shading} e_c\right) \tag{5.66}$$

$$\propto \; \exp\left(-k_c \beta_{shading} e_c\right) \exp\left(-k_s e_s\right) \tag{5.67}$$

$$\propto \quad \exp -(k_c\beta_{shading}e_c + k_s e_s) \tag{5.68}$$

$$\propto \quad \exp -(k_c\beta_{shading}e_c + k_s e_s) \tag{5.69}$$

$$= \quad K \exp -(e_c + \lambda e_s), \tag{5.70}$$

$$\tag{5.71}$$

where K is a constant and λ decides the significance of the shading module output at each site. Bayesian (optimal) estimate corresponds to $\mathbf{D}$ which maximizes the *a posteriori* probability $P(\mathbf{D}|\mathbf{D}^{sh})$. Eq.(5.71) implies that the problem of obtaining a *maximum a posteriori* estimate of $\mathbf{D}$ is identical to minimizing

$$e = e_c + \lambda e_s. \tag{5.72}$$

Differentiating e with respect to $D(i,j)$, we obtain

$$\frac{\partial e}{\partial D(i,j)} = \frac{\partial e_c}{\partial D(i,j)} + \lambda(i,j)\frac{\partial e_s}{\partial D(i,j)}, \tag{5.73}$$

where

$$\begin{aligned}\frac{\partial e_s}{\partial D(i,j)} &= -0.5(D(i+1,j) - D(i,j)) - 0.5(D(i,j+1) - D(i,j)) \\ &\quad +0.5(D(i,j) - D(i,j-1)) + 0.5(D(i,j) - D(i-1,j)).\end{aligned} \tag{5.74}$$

After differentiation[11] and rearrangement, we have

$$\frac{\partial e_s}{\partial D(i,j)} = 2\left(D(i,j) - \overline{D}(i,j)\right), \tag{5.75}$$

where $\overline{D}(i,j)$ denotes the local 4-neighbor arithmetic mean of $D(i,j)$, and

$$\begin{aligned}\frac{\partial e_c}{\partial D(i,j)} &= -0.5(D(i+1,j) - D(i,j)) + 0.5(D^{sh}(i+1,j) - D^{sh}(i,j)) \\ &\quad -0.5(D(i,j+1) - D(i,j)) + 0.5(D^{sh}(i,j+1) - D^{sh}(i,j)) \\ &\quad +0.5(D(i,j) - D(i,j-1)) - 0.5(D^{sh}(i,j) - D^{sh}(i,j-1)) \\ &\quad +0.5(D(i,j) - D(i-1,j)) - 0.5(D^{sh}(i,j) - D^{sh}(i-1,j)).\end{aligned} \tag{5.76}$$

Again, rearrangement results in

$$\begin{aligned}\frac{\partial e_c}{\partial D(i,j)} &= 2\left(D(i,j) - \overline{D}(i,j)\right) \\ &\quad -2\left(D^{sh}(i,j) - \overline{D}^{sh}(i,j)\right).\end{aligned} \tag{5.77}$$

Combining Eqs. (5.75) and (5.77), we have,

$$\begin{aligned}\frac{\partial e}{\partial D(i,j)} &= (1+\lambda(i,j))\left(D(i,j) - \overline{D}(i,j)\right) \\ &\quad -\lambda(i,j)\left(D^{sh}(i,j) - \overline{D}^{sh}(i,j)\right).\end{aligned} \tag{5.78}$$

[11] We use finite differences (on 4-neighborhood system) to approximate one-sided differentiation.

Setting $\frac{\partial e}{\partial D(i,j)}$ to zero gives us an iterative solution of the depth estimate:

$$(1+\lambda(i,j))(D(i,j)-\overline{D}(i,j)) \;\; = \lambda(i,j)(D^{sh}(i,j)-\overline{D}^{sh}(i,j)) \tag{5.79}$$

This results in an iterative solution of the depth estimate:

$$D^{n+1}(i,j) = \overline{D}^{n}(i,j) - \frac{\lambda(i,j)}{1+\lambda(i,j)}\Delta D^{sh}(i,j)^{n}, \tag{5.80}$$

where $\Delta D^{sh}(i,j)^{n}$ is the magnitude of the mean gradient at (i,j) in D^{sh} at the n^{th} iteration and is given by

$$\begin{aligned}\Delta D^{sh}(i,j)^{n} = \;& \tfrac{1}{4}(D^{sh}(i+1,j)+D^{sh}(i,j+1)+ \\ & D^{sh}(i-1,j)+D^{sh}(i,j-1)) \\ & -D^{sh}(i,j),\end{aligned} \tag{5.81}$$

and $D^{0} = D^{obs}$. In practical situations, albedo of the scene surfaces is not known *a priori* and hence shape of the surface obtained from shading information is only qualitatively correct. In order to avoid the instability in the reconstruction process, Eq. (5.80) is applied only when the sign of $\Delta D^{sh}(i,j)^{n}$ differs from the sign of $\Delta D(i,j)^{n}$, where

$$\begin{aligned}\Delta D(i,j)^{n} \;\; & = \tfrac{1}{4}(D(i+1,j)+D(i,j+1)+ \\ & D(i-1,j)+D(i,j-1))-D(i,j).\end{aligned} \tag{5.82}$$

We will refer to $\frac{\lambda(i,j)}{1+\lambda(i,j)}$ in Eq. (5.80) as $\alpha_{shading}(i,j)$. Parameter $\alpha_{shading}(i,j)$ is the coupling coefficient between the shape from shading module and the intrinsic map at the site (i,j). Notice that the correction term in Eq. (5.80) is not based on any precise calibration, but is set to an arbitrary monotonic function of the depth depending on the value of $\alpha_{shading}$. In practice, we have seen that the performance of the system does not critically depend on the value of $\alpha_{shading}$ as long as it is sufficiently small ($\alpha_{shading} \leq 0.05$). In the present implementation, we assume $\alpha_{shading}$ to be a global constant. We have used $\rho_{shading} \leq 0.2$.

5.2.4 Shape from Texture

Rewriting the generic modular integration equation (Eq. (5.16)) for texture module,

$$\hat{\mathcal{D}}_{tx}^{t} = \arg\max_{\mathcal{D}} P(\mathcal{D}|\mathcal{U}_{tx}(\mathcal{D}^{obs},\mathcal{I}),\mathcal{D}^{obs}), \tag{5.83}$$

where, for conciseness, we have replaced $\mathcal{D}_{r(t-1)}^{t-1}$ by $\mathcal{D}^{obs}$. In our present implementation, texture module does not affect the boundary map and refines the depth map. Eq. (5.83) can, therefore, be written as:

$$\hat{\mathbf{D}}_{tx}^{t} = \arg\max_{\mathbf{D}} P(\mathbf{D}|\mathcal{U}_{tx}(\mathcal{D}^{obs},\mathcal{I}),\mathcal{D}^{obs}), \tag{5.84}$$

where $\mathbf{D}$ is the depth component of the intrinsic map. For conciseness, we will refer to $\mathcal{U}_{tx}(\mathcal{D}^{obs},\mathcal{I})$ as $\mathbf{D}^{tx}$. Thus, the term $P(\mathbf{D}|\mathbf{D}^{tx},\mathcal{D}^{obs}))$ denotes the posteriori probability

of depth map $\mathbf{D}$, given the depth map produced by shape from texture module ($\mathbf{D}^{tx}$) and the current state of intrinsic map ($\mathcal{D}^{obs}$).

The algorithm we adopt here is a variation of Super and Bovik's [178] shape from texture algorithm described in Section 3.5. This algorithm assumes that the surface texture is uniform and can be correctly segmented.

Surface Normal Estimation ($\mathcal{U}_{tx} : \mathcal{D}^{obs} \times \mathcal{I} \rightarrow \mathcal{O}$): Let $\mathbf{g}(x,y) = (g_{\mathbf{x}}(x,y), g_{\mathbf{y}}(x,y), g_{\mathbf{xy}}(x,y))$ represent x, y, and xy squared image moments of a texture feature[12] at (x,y). Given image moments $\mathbf{g}(x,y)$ and surface orientation $\mathbf{n} \equiv (\sigma,\tau)$[13] at a site (x,y), surface moments can be estimated by:

$$\mathbf{g}^s(x,y) = \mathbf{T} \star \mathbf{g}(x,y), \tag{5.85}$$

where matrix $\mathbf{T}$ is given by

$$\mathbf{T} = \begin{vmatrix} \cos^2\sigma\cos^2\tau & \cos^2\sigma\cos\tau\sin\tau & \cos^2\sigma\sin^2\tau \\ -2\cos\sigma\cos\tau\sin\tau & \cos\sigma(\cos^2\tau - \sin^2\tau) & 2\cos\sigma\cos\tau\sin\tau \\ \sin^2\tau & -\cos\tau\sin\tau & \cos^2\tau \end{vmatrix} \tag{5.86}$$

Canonical image moments (M,m) and angle θ are determined from image moments $\mathbf{g}(x,y)$ by (after dropping the qualifier (x,y) for brevity)

$$[M,m]^T = \tfrac{1}{2}\left[\left(g_{\mathbf{x}} + g_{\mathbf{y}} + \sqrt{g_{\mathbf{y}}^2 + (g_{\mathbf{x}} - g_{\mathbf{xy}})^2}\right), \left(g_{\mathbf{x}} + g_{\mathbf{y}} - \sqrt{g_{\mathbf{y}}^2 + (g_{\mathbf{x}} - g_{\mathbf{xy}})^2}\right)\right] \tag{5.87}$$

[12] Our implementation uses intensity gradients.

[13] For conciseness, we represent orientation as (slant, tilt) pair.

$$\theta = \frac{1}{2}\arctan\frac{g_y}{g_x - g_{xy}} \tag{5.88}$$

Similarly, surface canonical moments can be computed. Let $p(x_1, y_1)$ and $q(x_2, y_2)$ be two sites located on a uniform textured surface. Given surface orientation $\mathbf{n^q}$ at a site (x_2, y_2), and image moments of a texture feature at site p and q, surface orientation $\mathbf{n^p} = (n_x, n_y, n_z)$ at p can be constrained by the following system of equations:

$$n_z = \sqrt{\frac{M^s(x_2,y_2)m^s(x_2,y_2)}{M(x_1,y_1)m(x_1,y_1)}}, \tag{5.89}$$

$$n_x = \cos\tau, \tag{5.90}$$

$$n_y = \sqrt{1-(n_x)^2-(n_z)^2}, \tag{5.91}$$

where $M^s(x_2, y_2)$ and $m^s(x_2, y_2)$ are maximum and minimum canonical *surface* moments of the selected features at site (x_2, y_2); $M(x_1, y_1)$ and $m(x_1, y_1)$ are the corresponding *image* moments of the same feature at site (x_1, y_1).

$$\tau = \begin{cases} \theta \pm \frac{1}{2}\arccos\lambda \\ \theta \pm \frac{1}{2}\arccos\lambda + \pi \end{cases} \tag{5.92}$$

$$\lambda = \frac{(\cos^2\sigma+1)[M(x_1,y_1)+m(x_1,y_1)]-2[M^s(x_2,y_2)+m^s(x_2,y_2)]}{\sin^2\sigma(M_A-m_A)}, \quad \sin\sigma \neq 0, M_A \neq m_a, \tag{5.93}$$

where θ is the orientation determined by canonical image moments $M(x_1, y_1)$ and $m(x_1, y_1)$[14].

Since shape from texture module also estimates surface orientation (although from

[14]Among the four estimates of τ, only the one which is the closest to the current surface normal estimate is considered.

different features), we have essentially used the same neighborhood definition and coherence mapping described in Section 5.2.3. We will denote the corresponding parameter set associated with shape from texture module by $(\alpha_{texture}, \rho_{texture})$. The values of these parameters are identical to the values of the corresponding shape from shading parameters.

5.3 Computation of Reliabilities

Given input intensity images, a vision module outputs a map of intrinsic variables or of variables which partially constrain the intrinsic map. The reliability of the module output is not spatially uniform due to several reasons. First, not all parts of the input images have the information desired by a module; the module output from the corresponding parts of the image are erroneous. Secondly, the assumptions made by a given module might be violated in some regions of an image invalidating the module output in those regions. Finally, recovery of structural information from intensity images inherently depends upon the imaging geometry and the reliability of resultant module output might systematically depend upon both the scene and imaging geometry.

Assessing the reliability of a module at each pixel is important since such information directly helps in determining to what extent each portion of a module output is to be believed. It would be useful to obtain a measure of reliability of a module output for each part of the input images. Such an index will associate a confidence value $R(x, y) \in \Re$ with output of the module at each site (x, y) in an image. This

reliability information might then be used for discarding highly unreliable portions of a module output or for relating a module output with the other module outputs. Unfortunately, consistent and accurate estimation of reliability of a module is an extremely difficult task in its most general form. We briefly discuss a few approaches to solve this problem and the limitations associated with each of these approaches.

1. **Intramodular approaches:** Typically, these methods make assumptions about the imaging process and estimation of reliability is based on operation of the module itself, *i.e.,* the method of extraction of the requisite information by the module. The intramodule assessment of the output usually does neither verify whether the required information is available nor question the module assumptions. For instance, in Figure 5.8, we show an image of a cylindrical surface with circularly shaped stripes along its circumference. a typical shape from texture module will infer a flat surface in the image with a high degree of reliability. Similarly, a typical shape from shading algorithm would predict a curved surface after analyzing the intensity profiles across the photo frame in Figure 5.9. Therefore, the reliability of a module output (as estimated by an intramodular approach) is not always an indicator of its correct behavior. We now describe an intramodular method of assessing the reliability of the stereo module.

   In this section we will briefly describe a method of assessing the reliability of a module using analyses of the stereo module presented in Weng *et. al* [196]. Let

Figure 5.8: An Image of a cylinder.

us first consider the following generic linear problem:

$$\mathbf{y} = A\mathbf{p} + \epsilon_{\mathbf{y}}, \tag{5.94}$$

where we need to estimate the parameter p. Note that $\epsilon_{\mathbf{y}}$ is a random vector contaminating the measurements $\mathbf{y}$. Let the expectation of $\epsilon_{\mathbf{y}}$ be zero ($\mathcal{E}(\epsilon) = 0$) and its covariance matrix be $\Gamma_{\mathbf{y}} = \mathcal{E}(\epsilon_{\mathbf{y}}\epsilon_{\mathbf{y}}{}^t)$. According to Gauss-Markov theorem [175, 65], the unbiased linear minimum variance estimator of $\mathbf{p}$ is

$$\hat{\mathbf{p}} = \left(A^t\Gamma_{\mathbf{y}}^{-1}A\right)A^t\Gamma_{\mathbf{y}}^{-1}\mathbf{y} \tag{5.95}$$

Figure 5.9: Image in an image.

whose error covariance matrix is

$$\Gamma_{\hat{\mathbf{p}}} \equiv \mathcal{E}((\hat{\mathbf{p}} - \mathbf{p})(\hat{\mathbf{p}} - \mathbf{p})^t = \left(A^t \Gamma_{\mathbf{y}}^{-1} A\right)^{-1}. \tag{5.96}$$

This estimator is equivalent to the least-squares estimator with weight matrix $\Gamma_{\mathbf{y}}^{-1}$ minimizing objective function (Weng *et. al* [196]):

$$(\mathbf{y} - A\mathbf{p})^t \Gamma_{\mathbf{y}}^{-1} (\mathbf{y} - A\mathbf{p}). \tag{5.97}$$

Let us now consider the estimation of disparities by the stereo module. This algorithm matches four different attributes of the intensity images (see Section 3.3). The stereo module assumes Lambertian scene surfaces and posits

that the image attributes of the *corresponding* pixels in a stereo image pair are identical. The difference in these image attribute values at the corresponding pixels (residuals) contribute to an error function which guides the matching process.

Given an estimate of the correspondence (disparity) vector field, the stereo matcher at a given level of resolution obtains a refinement of the correspondence (disparity) vector field by minimizing a weighted sum of squared residuals (errors):

$$\min_{\mathbf{d}} \sum_{\mathbf{u}} \sum_{i} \mathbf{w}_i [\mathbf{R}_i(\mathbf{u}, \mathbf{d})]^2, \tag{5.98}$$

where $\mathbf{d}$ is the correspondence (disparity) vector field, $\mathbf{R_i}(\mathbf{u}, \mathbf{d})$ is the residual contributed by the i^{th} attribute image at location $\mathbf{u}$ due to correspondence vector $\mathbf{d}$, and $\mathbf{w_i}$ is a pre-specified weight associated with the residual $\mathbf{R_i}$ (with $\mathbf{W} = diag(\{\mathbf{w_i}\})$). For instance, intensity residual is defined as follows:

$$\mathbf{R_{int}}(\mathbf{u}, \mathbf{d}) = \mathcal{I}_l(\mathbf{u} + \mathbf{d}) - \mathcal{I}_r(\mathbf{u}). \tag{5.99}$$

Expanding $\mathbf{R}$ at $\mathbf{d} = \mathbf{d}' + \delta_{\mathbf{d}}$ and assuming that higher order terms of $\|\mathbf{d} - \mathbf{d}'\|$ are negligible, we have

$$\mathbf{R}(\mathbf{u}, \mathbf{d}) = \mathbf{R}(\mathbf{u}, \mathbf{d}') + J(\mathbf{d} - \mathbf{d}'), \tag{5.100}$$

where $J = \frac{\partial \mathbf{R}(\mathbf{u},\mathbf{d})}{\partial \mathbf{d}}$.

Rewriting Eq. (5.100), we obtain

$$-\mathbf{R}(\mathbf{u},\mathbf{d}') + \frac{\partial \mathbf{R}(\mathbf{u},\mathbf{d})}{\partial \mathbf{d}}\mathbf{d}' = \frac{\partial \mathbf{R}(\mathbf{u},\mathbf{d})}{\partial \mathbf{d}}\mathbf{d} - \mathbf{R}(\mathbf{u},\mathbf{d}). \tag{5.101}$$

This equation can be related to Eq. (5.94) where the term $-\mathbf{R}(\mathbf{u},\mathbf{d}') + \frac{\partial \mathbf{R}(\mathbf{u},\mathbf{d})}{\partial \mathbf{d}}\mathbf{d}'$ could be considered as $\mathbf{y}$, $A = \frac{\partial \mathbf{R}(\mathbf{u},\mathbf{d})}{\partial \mathbf{d}}$, $\mathbf{p} = \mathbf{d}$, and $\epsilon_{\mathbf{y}} = -\mathbf{R}(\mathbf{u},\mathbf{d})$. The least squares solution (Eq. (5.98)) and the minimum variance estimator of $\mathbf{p}$ in Eq. (5.94) are related through Eq. (5.97). If we make the simplifying assumption that $\epsilon_{\mathbf{y}} = \sigma^2 I$ and $W = wI$, it can be observed from Eq. (5.96) that

$$\Gamma_{\hat{\mathbf{d}}} \propto (J^T J)^{-1}. \tag{5.102}$$

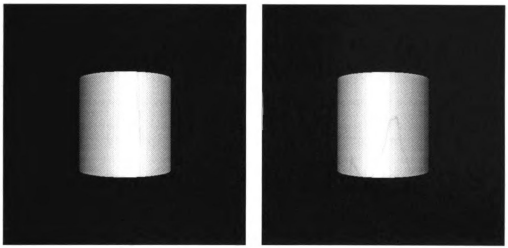
Thus, the confidence in the stereo module output is directly proportional to the magnitude of J (evaluated at the final solution). Relating J to the reliability of the module is intuitive: since steeper the $\mathbf{R}(\mathbf{u},\mathbf{d})$ profile at $\mathbf{u}$ with the variation in $\mathbf{d}$, the more likely that the salient image features (as opposed to noise fluctuations) are contributing to the matching process. In addition, a large residual value $\mathbf{R}(\mathbf{u},\mathbf{d})$ could indicate a larger likelihood of the solution being spurious. However, all these heuristics are merely gross indicators of the reliability of the module and tend to be inconsistent and even misleading in rather simplistic images.

Consider a simple synthetic stereo image pair of a cylinder shown in Figure 5.10.

Figure 5.11 illustrates some of results derived from matching the stereo pair in Figure 5.10 using the stereo module. Figure 5.11(a) shows the intensity profiles of the cylinders in the 256^{th} row. Figure 5.11(b) shows the error in the depth computed from the stereo module for the 256^{th} row. In this illustration, we will only consider the intensity attribute of the image. Figure 5.11(c) shows the intensity residuals computed in the 256^{th} row at the final solution. Figure 5.11(d) depicts $J^{x}_{int} = \frac{\partial \mathbf{R_{int}(u,d)}}{\partial \mathbf{d_x}}$, where d_x is the x-axis component of the disparity. From these illustrations, it could be observed that (i) the large final values of the residuals are not directly related to the error in the disparity computation and (ii) the higher magnitude of J^{x}_{int} does not necessarily equate to more accurate disparity values.

In our experience, one of the the primary sources of error in the stereo module output is due to the violations of the implicit assumptions embedded in the stereo module. For instance, the stereo module we use assumes that all the pixels in each stereo image are visible to both the cameras and the corresponding pixels possess similar image features. Image regions representing occlusion, limb edges, *etc.*, conspicuously violate these assumptions and result in an erroneous disparity output which can not be assessed using this method of reliability estimation.

Some other criteria for computing the confidence measures have been suggested in the literature based on a spatial variation in the magnitude of residuals with respect to change in observed disparity. For instance, Singh and Allen [173]



(a) (b)

Figure 5.10: Synthetic stereo pair image of a cylinder (size 512 × 512).

propose a confidence measure based on the eccentricity of this spatial distribution. These and other measures suffer from similar limitations when the correspondence estimation based on similarity of the image attributes is not adequate. Hence, caution needs to be exercised in the use of confidence measures based on intramodule approach alone.

2. **Empirical approaches:** These approaches are based on the known facts about a given vision module. These facts may have been either derived from psychophysical experiments or from the human neurophysiological data. For instance, there is some evidence that the weight assigned to each cue is computed from an unrelated "ancillary" cue [72]. In some situations, the human visual system might have empirically learnt to weight the stereo module output more heavily than the output from shape from shading module. However, such heuristics are not readily known and tend to display inconsistent performance in real

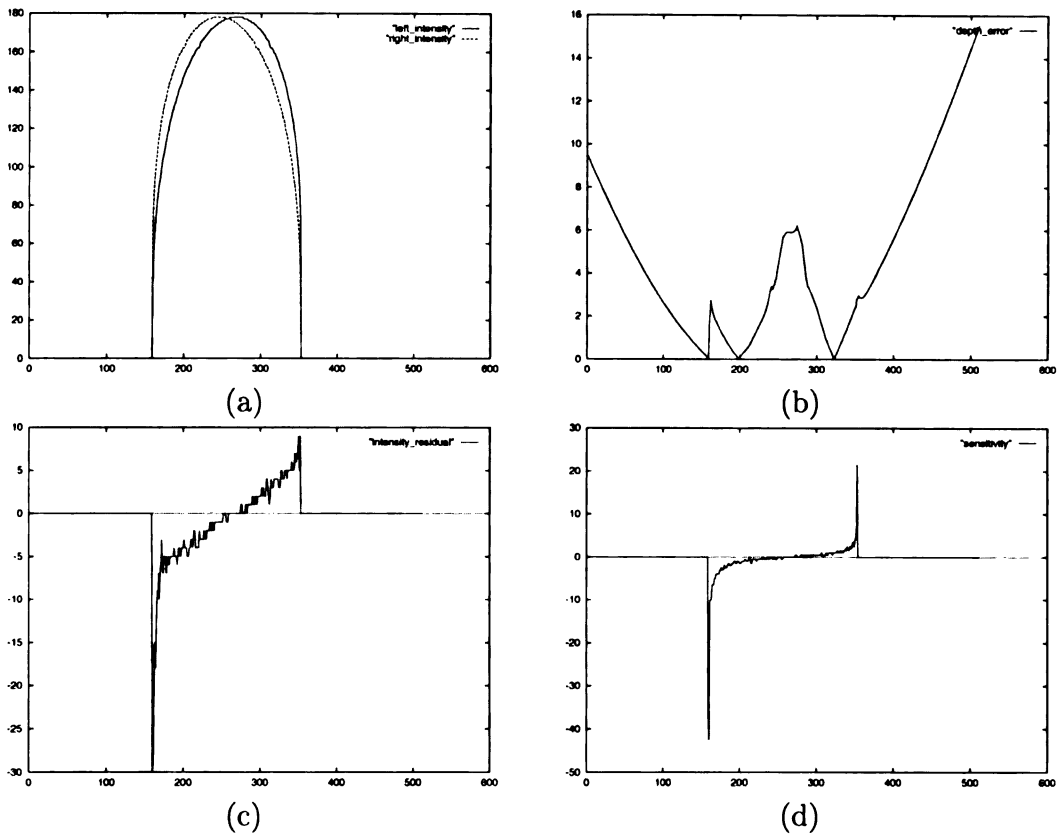


Figure 5.11: Error analysis of synthetic stereo image pair in Figure 5.10. This figure shows the results for the 256^{th} row only: (a) Intensity profiles of the left and right stereo image in the 256^{th} row. (b) Error between the observed and the true depths; (c) Intensity residuals; (d) Sensitivity of the intensity residuals.

situations.

For instance, it is commonly known that the stereo modules perform better in the regions of high contrast. Consider Figure 5.12. Figure 5.12(a) shows the right image of a stereo pair of sphere. Figure 5.12(b) depicts the edge output from Canny operator (only 10% of the strongest edges shown). Figures 5.12(c) and (d) show the depth map obtained from the stereo module and ground truth depths, respectively. Note that the regions of large error in estimates do not

directly correspond with the regions of higher edge density.

3. **Intermodular approaches:** Here the communication among the modules themselves is used for resolving the uncertainty about the overall system output. As mentioned in Chapter 1, there are several methods of resolution: accumulation, cooperation, competition, veto, disambiguation, promotion, consensus, coherency, *etc.* Since the information from several independent modules are juxtaposed in this approach, such an approach has better opportunities to correctly assess output of individual module. In this thesis we have used a novel method of assessing the reliability of the module output based on coherency. In the present implementation we do not weigh the module output at a site by the corresponding confidence value (except in the case of stereo integration (Eq. (5.48)) and uniformly treat the outputs at all sites. This means that the values of λ and β_{gp} in Eqs. (5.80) and (5.25) are assumed to be globally constant.

   The confidence maps of the module output influence the weighing of the evidence gathered by that module. The present approach does not estimate the reliability of the intrinsic map itself and the information regarding the reliability of a given solution is lost after each module performs its update. In a more general solution, a confidence map could be associated with each component of the intrinsic map. Given a module output along with its confidence map, the current state of intrinsic map, and the current state of the confidence map associated with the intrinsic map, it is possible to use a Bayesian framework to updating not only the intrinsic map but also the confidence map associated

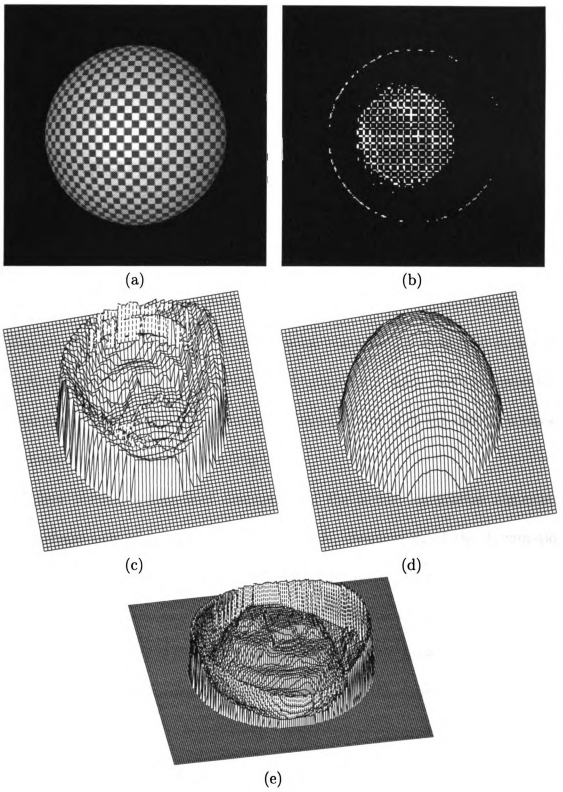


Figure 5.12: Empirical method for stereo evaluation (only the right stereo image is shown): (a) intensity image of sphere; (b) output from Canny edge applied to (a); (c) depth output from stereo module; (d) ground truth depths; (e) error in (c).

with it. These representation and updation schemes are similar to the Kalman filter approach. The advantage of such a scheme is that the confidence map associated with the intrinsic map indicates the reliability of the final solution.

5.4 Integration Algorithm

Our representation of the input images and the extracted features uses seven levels of resolution; each coarser level of resolution reduces the number of pixels by a factor of 4 (half in each dimension, x and y).

The integration system starts with an initial planar depth map (and no boundaries) at the coarsest level of resolution (8×8 image size). Each module sequentially updates the intrinsic map a fixed number of times (20) and then the control is passed to the next finer level of resolution (Fig. 5.6).

We now present a high-level description of the overall integration algorithm. Given a stereo pair of intensity images, I_l and I_r, direction $\mathbf{n_s}$ of the illumination source, weight vector $\mathbf{w}$ (Eqs. (5.28)), thresholds related to the coherence functions ($\rho_{shading}$, ρ_{stereo}, $\rho_{texture}$, and ρ_{gp}) (Eq. 5.52), coupling coefficients ($\alpha_{shading}$ and $\alpha_{texture}$) (Eq. 5.80), and the weight vector associated with perceptual grouping module (Section 5.2.1), the depth values are reconstructed using the algorithm in Figure 5.13.

Note that our uniform framework facilitates extension of the current system to include additional modules. All components of the integration system require *local* computations. The integration of the perceptual grouping and stereo modules at

1. Compute the four attribute (smoothed, gradient, positive curvature, and negative curvature) images. Initialize the disparity at each pixel at the coarsest level ($l = 6$) to zero.

2. Starting with the coarsest level ($l = 6$), do at each level l:

   A. Obtain the four attribute images at level l by blurring the attribute images at level 0 (input image).

      Do steps (i) to (iv) N ($= 20$) times.

      (i) Apply perceptual organization module to refine the boundary map. Do a fixed number (5) of anisotropic diffusion iterations of depth values.

      (ii) Update the disparities at level l by one iteration of stereo module (Eqs. (5.28)).

      (iii) Apply shape from shading module to refine the depth map.

      (iv) Apply shape from texture module to refine the depth map.

   B. Project the disparities to level $(l - 1)$ by (quadratically) interpolating disparities at level l. Project the regions from level l to $(l - 1)$.

3. Output the final depth map and boundaries. The surface orientation (normals) at each point can be computed from the depth map.

Figure 5.13: Uniform Bayesian Integration Algorithm.

each level of resolution for each iteration needs one pass over the entire image and its computational complexity is $O(n^2)$, for an $n \times n$ image. The complexities of integrating shape from shading and texture modules are somewhat data-dependent and, from our experience, require a few dozen iterations before convergence. The computations required for the integration constitute about 25% of the overall computations of the system. The modular and distributed organization facilitates parallel implementation. Table 5.4 presents actual timing statistics for a typical pair of 512×512 intensity images on a Sun Sparc 20.

Table 5.2: CPU times for a typical image on Sun Sparc 20.

Computation	Time
Perceptual Grouping	1830s
Shape from shading	810s
Stereo	1188s
Shape from texture	3360s
Integrated System	8439s

5.5 Experimental Results

Our experimental results will be discussed in the context of quality of reconstruction obtained from the integrated system. The experiments are primarily designed to demonstrate the graceful deterioration in the performance of the system in situations where assumptions made by the individual modules are violated. Many results are presented for subjective evaluation by the reader. Objective evaluation of the quality of surface reconstruction is presented in case of synthetic images. We will now briefly describe our imaging setup before presenting our results.

The objective of the experiments described in this section is to demonstrate the improvements in the recovery of the 3D structure of the scene due to integration as well as the graceful degradation in the performance of the system under adverse situations. In most of the experiments, we have chosen to compare the reconstruction results obtained from the integrated system with corresponding results from stereo module alone; the reconstruction results from the other individual modules were not as reliable as those of the stereo module. The experiments are organized as follows. First, we describe the improvement in numerical accuracy of 3D reconstruction due

to integration for synthetic images rendered under a variety of controlled imaging conditions. The second part of the section illustrates the qualitative improvement in the recovery of 3D structure due to integration for several real images.

Synthetic Data: Six volumetric primitives (parallelopiped, cylinder, hyperboloid, paraboloid, ellipsoid, and torus) were used for generating photo-realistic stereo images. This set of primitives was selected primarily due to its wide range of surface profiles. This data set was supplemented by the range image of a Mozart bust. Texture-mapped stereo images of these surfaces were obtained using a photo-realistic renderer, `pov-ray`.

Imaging Conditions: In all the experiments involving synthetic images, the *image centers* of the left and right cameras were located at $(-9, 0, -90)$ and $(9, 0, -90)$, respectively. The optical axes of both the cameras pointed towards the origin and were located in X–Z plane. A pin hole camera geometry determined the imaging projections. A point illumination source was placed directly behind the camera at $(0, 0, -1000)$. The sensing noise was simulated by 10% *i.i.d* Gaussian jitter in the projected intensity of pixels.

The first set of experiments was designed to demonstrate the soundness of the integration scheme to recover the 3D structure for the entire data set (which encompasses a wide range of surfaces) from their stereo images. Each object surface was texture-mapped with a synthetic checkerboard texture and was modeled as a perfect Lambertian surface. The right stereo image for each of the surface primitives and their reconstructions obtained from stereo module alone and from the integrated system are depicted in Figs. 5.14 and 5.15. The regularity of the fine-grained texture often

presented a severe challenge to the stereo module in obtaining the correct correspondence [76]. In an integrated system, some of the mistakes committed by the stereo module were corrected by the information provided by the other modules. However, the performance of the integrated system is far from perfect. The errors in the reconstruction obtained by the integrated system can be primarily attributed to the late introduction of the other modules in the processing of data [15]. Consequently, the errors committed by the stereo module at the coarser levels of resolution could not always be reversed. The 3D reconstructions obtained from stereo alone and from the integrated system were compared with the ground truth (given), using a squared error function based on differences in true and estimated depth and shape (surface normals) measurements. Percentage reductions in this squared error are used for assessing the performance of the proposed integration strategy. Table 5.3 shows that the surface reconstruction using the integrated method is superior to that obtained by the stereo module alone.

Table 5.3: Improvement in surface reconstruction due to integration: Lambertian surfaces.

Surface primitive	**% Reduction in shape estimation error**	**% Reduction in depth estimation error**
Parallelopiped	31	28
Sphere	20	23
Cylinder	22	26
Paraboloid	17	22
Hyperboloid	8	14
Torus	5	12

Violation of nominal assumptions made by individual modules should not signif-

[15] At very coarse resolutions (8×8, 16×16, and 32×32), no meaningful features could be extracted from the image pair for the modules other than stereo.

icantly deteriorate the overall system performance. The second experiment focussed on the reconstruction of a surface of an arbitrary complexity (Mozart bust) under various "adverse" conditions. Each of the confounding situations is shown in (a) and (b) parts Figs. 5.16-5.20. Fig. 5.16(a) is the right stereo image of a Lambertian surface with no texture and hence the performance of shape from texture module was not expected to be reliable. The resultant integrated system performance remained stable. The improvement in performance (Table 5.4) was primarily contributed by the shape from shading module and was not significant since considerable variation in brightness across the image surface helped stereo module to obtain the correct correspondences. Stereo images in Figs. 5.16-5.19 present challenging conditions for not only the shape from shading module (due to violation of single surface albedo assumption), but also to the shape from texture module. First, the texture is not uniform and the variation of the texture features across a physical surface does not systematically relate to its orientation. Further, in Fig. 5.19 the surface is mapped with two similar yet distinct textures. All these confounding conditions did not significantly interfere with the performance of the integrated system. We primarily observed an improved shape estimation rather than improved depth estimation under these situations. Figs. 5.20(a) and (b) depict a more complex situation. Here, the specularity violates the "Lambertian surface" assumption made by the shape from shading module in addition to the unique surface albedo. The specular patches also interfere with the performance of the stereo module (since the sites with similar intensity attributes do not necessarily imply correspondence). Finally, the non-uniform texture violated the assumptions made by the texture module. The violation of assumptions made by all the three

modules results in a very low improvement in the overall system performance (see Table 5.4).

Table 5.4: Improvement in surface reconstruction in texture-mapped Mozart images.

Surface	**% Reduction in shape estimation error**	**% Reduction in depth estimation error**
No texture (Fig. 5.16)	8	4
Partial texture (Fig. 5.17)	27	12
Full texture (Fig. (5.18)	32	14
Two textures (Fig. 5.19)	37	18
Specular texture (Fig. 5.20)	16	14

Finally, Figures 5.21 and 5.22 illustrate (visually) how the performance of the integrated system improves the 3D reconstruction of real images. In the fruit image (Figs. 5.21(a) and (b)), the integrated system improved the shape estimation (improvement is conspicuous over the cantaloupe and table-cloth surfaces) and the depth estimation (especially over the cantaloupe surface). Figures 5.24(a) and (b) show superquadric fits to the depths obtained from stereo module and the integrated system, respectively.

The image of Egg and Cup (Figs. 5.22(a) and (b)) was captured by an inexpensive CCD camera (Panasonic GP-KR202, $f = 25\ cm$, maximum aperture). The images were subsequently gamma corrected with $\gamma = 2.0$ and normalized to 256 gray levels. The stand off was approximately 80 cm. To obtain a stereo pair of images, the camera was translated and rotated. The translation was in the direction of x-axis and rotation was about the y-axis (the z-axis being approximately aligned with the optical axis). The rotation of the camera was effected to bring the disparity of the region of interest close to zero. The scene was illuminated with ambient light and a single incandescent

light source was located (30 cm) behind the camera (approximately in the x-z plane) pointing in the direction $(0, 0, 1)$. The integrated system improved the estimation of the shape and depth features over the egg surface.

Fig. 5.23 shows segmentation results for Fruit image (Fig. 5.21(b)) and Egg and Cup image (Fig. 5.22(b)).

5.6 Summary

The visual world can only be ambiguous in relatively contrived situations, but in real world it is the combinations of the cues which convey a unique physical reality. Whether in reconstructing the entire visual input or in extracting its component relevant to the given task, the designer of a vision system confronts the challenging problem of integrating all visual cues to obtain a reliable performance. Several integration strategies have been reported in the literature which primarily study pairwise integration of vision modules. But, there is a dearth of results on integrating more than two modules for *real* images. We have presented a unified framework for integrating vision modules which facilitates a design of a flexible and extensible integrated system for 3D reconstruction from a pair of stereo images. Based upon this framework, we have implemented a system for integrating four vision modules: perceptual organization, shape from shading, stereo, and shape from texture. We show the reconstruction results using the integrated system for both synthetic and real images. We also demonstrate the consistent performance of the integrated system even in the adverse situations where one or more assumptions made by the individual modules

are violated. The numerical accuracy of the recovered depth is assessed in case of synthetically generated data.

Finally, we note some of the limitations of our current system which are significant topics for further research: We have only provided some empirical evidence for the convergence and stability of the integrated system. A rigorous analysis of these issues needs to be undertaken which is a subject of our ongoing exploration. Further, the present flow of control is fixed and is not suitable for partial and dynamic reconstruction. A more flexible control might provide a congenial environment for many active and purposive vision systems.

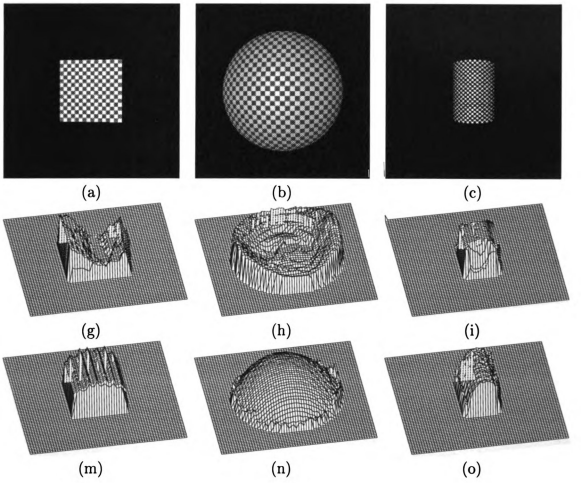


Figure 5.14: Synthetic texture-mapped surface primitives (Only the right stereo image is shown): (a) parallelopiped (31%); (b) sphere (20%); (c) cylinder (22%); (g)–(i) and (m)–(o) depict the depth reconstruction for these primitives from stereo module and from the integrated system, respectively. Figures in the parentheses show improvements in the depth estimate due to integration (see Table 5.3).

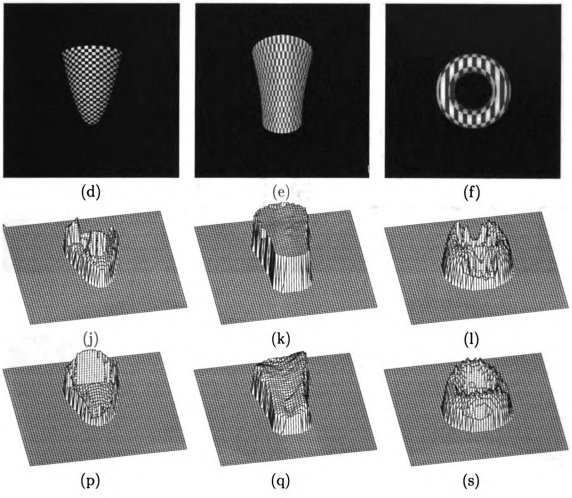


Figure 5.15: Synthetic texture-mapped surface primitives (Only the right stereo image is shown): (d) paraboloid (17%); (e) hyperboloid (8%); and (f) torus (5%); (j)–(l) and (p)–(s) depict the depth reconstruction for these primitives from stereo module and from the integrated system, respectively. Figures in the parentheses show improvements in the depth estimate due to integration (see Table 5.3).

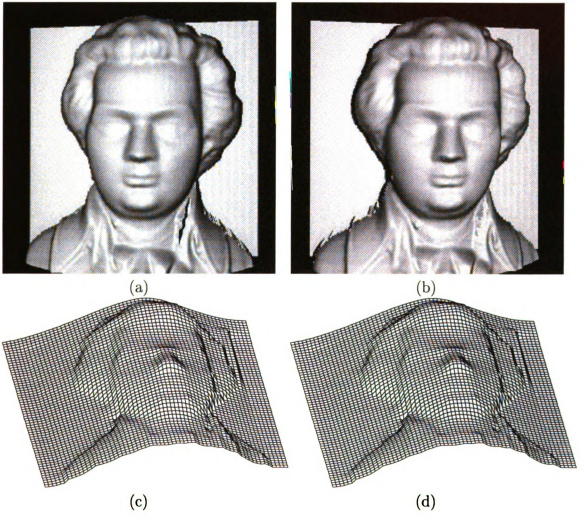


Figure 5.16: A Lambertian surface with no texture: (a) and (b) Mozart stereo images (size 512x512) synthesized with no surface texture; (c) reconstruction of (a) and (b) using stereo module only; (d) reconstruction of (a) and (b) using the integrated system. Depth estimation was improved by 8% due to integration (see Table 5.4).

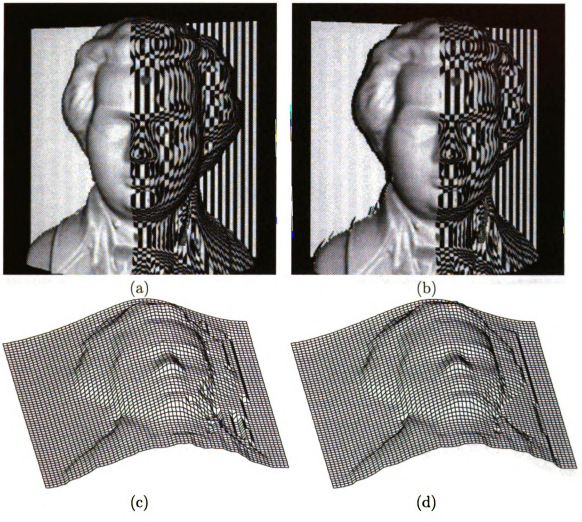


Figure 5.17: A surface with partial texture: (a) and (b) Mozart stereo images (size 512x512) with texture-mapped surface juxtaposed with a surface with no texture; (c) reconstruction of (a) and (b) using stereo module only; (d) reconstruction of (a) and (b) using the integrated system. Depth estimation was improved by 27% due to integration (see Table 5.4).

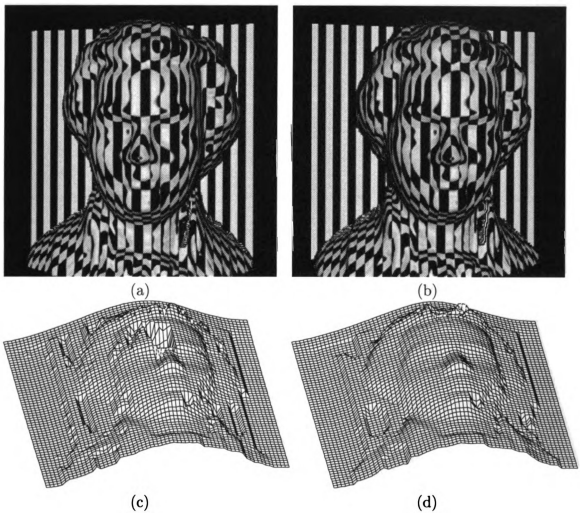


Figure 5.18: A surface with full texture: (a) and (b) Mozart stereo images (size 512x512) texture-mapped with a homogeneous texture; (c) reconstruction of (a) and (b) using stereo module only; (d) reconstruction of (a) and (b) using the integrated system. Depth estimation was improved by 32% due to integration (see Table 5.4).

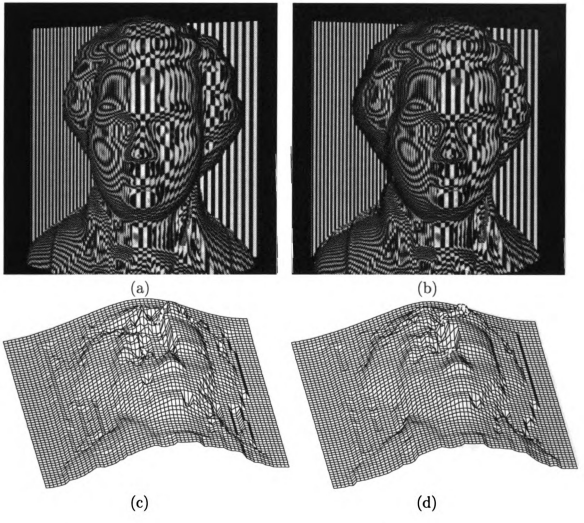


Figure 5.19: A surface with two textures: (a) and (b) Mozart stereo images (size 512x512) texture-mapped with two distinct textures; (c) reconstruction of (a) and (b) using stereo module only; (d) reconstruction of (a) and (b) using the integrated system. Depth estimation was improved by 37% due to integration (see Table 5.4).

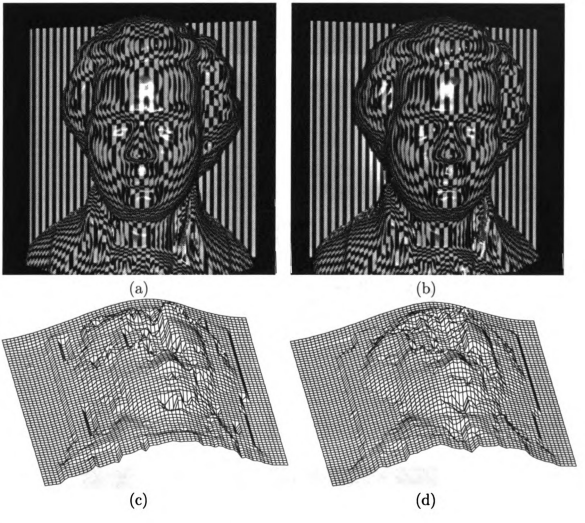


Figure 5.20: Specular surface: (a) and (b) Mozart stereo texture-mapped images (size 512x512) for a specular surface; (c) reconstruction of (a) and (b) using stereo module only; (d) reconstruction of (a) and (b) using the integrated system. Depth estimation was improved by 16% due to integration (see Table 5.4).

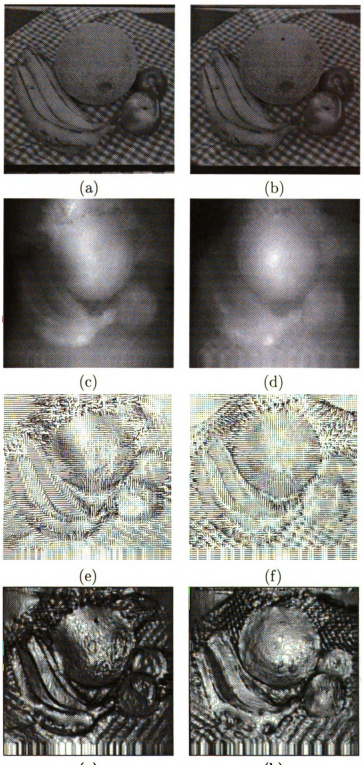


Figure 5.21: Fruit image (size 512x512): (a) and (b) Left and right stereo images; (c) Recovered depth from stereo alone; (d) Recovered depth from the integrated system; (e) Recovered surface normals from stereo alone; (f) Recovered surface normals from the integrated system; (g) and (h) show (e) and (h) as shaded perspective views [76]. Fruit image was provided by Prof. Narendra Ahuja.

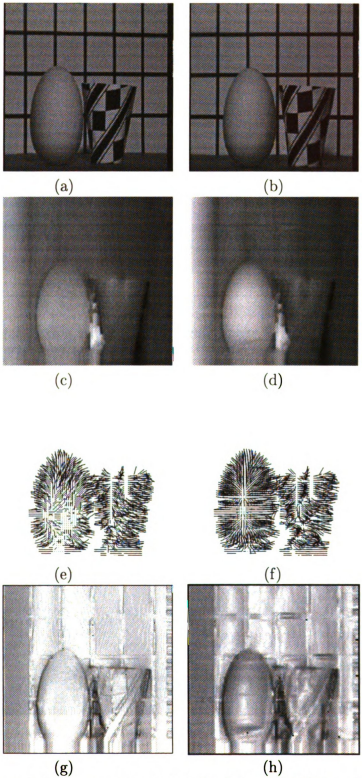
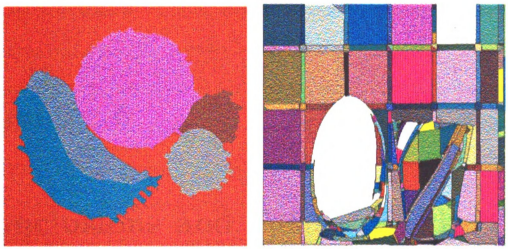
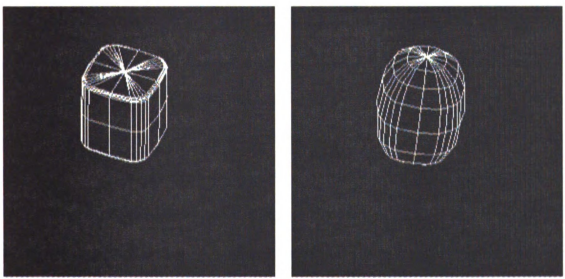


Figure 5.22: Egg and Cup image (size 512x512): (a) and (b) Left and right stereo images; (c) Recovered depth from stereo alone; (d) Recovered depth from the integrated system; (e) Recovered surface normals from stereo alone; (f) Recovered surface normals from the integrated system; (g) and (h) show (e) and (h) as shaded perspective views [76].



(a) (b)

Figure 5.23: Segmentation results for (a) Fruit image (Fig. 5.21(b)) and for (b) Egg and Cup image (Fig. 5.22(b)).



(a) (b)

Figure 5.24: Recovery of 3D shape of cantaloupe in the Fruit image (Fig. 5.21): (a) Recovered superquadrics from stereo alone; (b) Recovered superquadrics from the integrated system.

Chapter 6

Conclusions and Future Work

Information integration is an important research problem and is a constant theme of exploration in many scientific fields, including computer vision. Integration of vision modules for 3D surface reconstruction is the focus of research presented in this thesis.

In this chapter we present a brief overview of our work (Section 6.1), a list of directions for future research (Section 6.2), and some concluding remarks (Section 6.3).

6.1 Overview

Individual visual cues are often unreliable and ambiguous. It is extremely difficult to overcome the limitations in implementations of the individual vision modules in an isolated system; integrated vision systems are necessary to obtain a reliable interpretation of complex scenes. Design of such systems is challenging since each vision module works under a different and possibly conflicting sets of assumptions; an effective integration scheme must not only deal with noisy input images but must also

overcome the artifacts and restrictive assumptions of the individual modules.

Research reported in this thesis emphasizes modeling the interaction and information exchange among the vision modules to overcome the limitations in their individual performance in isolation. In a detailed case study presented in Chapter 4, we demonstrated how the interactions among the vision modules (*viz.*, shape from shading, stereo, line labeling, and perceptual organization) can improve 3D reconstruction from a pair of stereo intensity images.

In Chapter 5 we made an attempt at systematizing design of integration of vision modules using a simple control structure. We proposed a unified Bayesian integration framework for interaction among the vision modules to obtain a complete 3D reconstruction from a pair of intensity (stereo) images. The proposed integration architecture allows a parsimonious modeling of various interactions. Novel features of the proposed scheme include, (i) interaction of each module with *intrinsic map*, (ii) multi-level, multi-resolution representations and hierarchical coarse-to-fine control, (iii) fine–grained feedback mechanisms, and (iv) robust estimation procedures based on the principle of *coherence*. We have integrated perceptual grouping, stereo, shape from shading, and shape from texture modules under the proposed framework. We demonstrated the efficacy of our approach using real images of several different scenes and observed improvements in the quality of recovered 3D structure as a result of integration. The output of the integrated system is shown to be insensitive to violations of individual module assumptions. The numerical accuracy of the recovered depth is assessed for photo-realistically rendered images from several scenes containing a variety of generic surfaces. We have also qualitatively evaluated our approach

by reconstructing geons from the depth data obtained from the integrated system.

6.2 Future Directions

Whether to reconstruct the entire visual input or to extract its component relevant to the given task, a reliable vision system is required to consider all the available visual cues to obtain an unambiguous output. This raises several important research issues in solving integration problem. What is the most reliable information provided by each visual cue? How to design an integrated system which can be easily maintained and extended? How to integrate vision modules so that the system performance does not critically depend on the performance of individual modules? How much weight should be assigned to the information provided by each module? We have made an attempt to address some of these research issues in our work, but a definitive solution to these problems will need more extensive research. Here we list some possible enhancements to the present system:

1. Color is an important source of information in the human visual system. Until now, we did not consider this cue in order to keep the magnitude of the image data manageable. As seen in Section 3.2, the recovery of surface orientation from shading information alone is an underconstrained problem. Inclusion of color information will provide additional *physical* constraints (one irradiance equation per channel). Such an augmentation is likely to improve the performance of the integrated system.

2. The current system does not include a systematic model for specular reflectance. Our experiments indicate that a limited amount of specularity does not critically degrade the performance of the system. Inclusion of even a simple model for specularity could improve the performance of the system for scenes with highly specular surfaces.

3. The present implementation separately models interaction between each cue and the intrinsic map to derive a simple control structure. How effective is this model, in practice, compared to the pairwise modeling of interactions among the modules?

4. Our system is relatively insensitive to the various module and interaction parameters due to the manner in which we have incorporated feedback into the system. It would be desirable to automatically learn the parameters of the system.

5. The flow of control in our system is fixed and is not suitable for partial and dynamic reconstruction. A more flexible control might provide a congenial environment for many active and purposive vision systems.

6. The present approach, in a sense, egalitarian; it assumes that all modules perform equally well in all regions of an image. This situation can be vastly improved by our knowledge about the performance of a module on the image features. For instance, Karu, Jain, and Bolle have recently shown that it is possible to identify textured regions in an image. Such mechanisms are expected

to vastly improve the system performance [93]. The present approach does not utilize the common knowledge about the performance of various modules.

7. The present system proposes an ambitious goal of generic 3D reconstruction. A given application domain often offers opportunities to recover from the errors in the sensing and intermediate processing by coupling goals of the system with the object models in the domain. Such an augmentation of the current integrated system in the context of a given application (*e.g.*, 3D object recognition for a limited class of objects) would be useful.

6.3 Conclusions

The visual world can only be ambiguous in relatively contrived situations (*e.g.* Necker cube) but the visual input from the scenes in real world abounds with a number of visual cues. It is only the combination of several cues that permits a reliable interpretation[1].

There is a general agreement among the computer vision researchers about the need for information integration. However, the potential of a vision system relying on low-level modules has been grossly underestimated because of individual vulnerabilities of the modules. While much of the research is directed at improving the performance of individual modules, a relatively few studies emphasize the importance of integration. Our results from a limited scene domain demonstrate that an integrated system comprising of several low-level modules can provide a better 3D

[1]This line of thinking is associated with the theory of *direct* perception.

reconstruction than the individual modules without using any top-down knowledge. However, attaining the performance of the human visual system using computers appears to be a far cry.

Integrating vision modules for 3D reconstruction from a stereo pair of intensity images is a difficult problem primarily due to our lack of understanding of two underlying issues: (i) an accurate assessment of the strengths and limitations of individual modules; (ii) the representations and control structures which can exploit complementary constraints provided by the imperfect modules to recover the true structure in the data. We have attempted to systematize the design procedure for an integrated system which takes into account these research issues and demonstrated that an integrated system thus designed leads to improved results in a limited scene domain. Much more research is needed to obtain definitive and robust solutions to the integration problem.

Appendices

Appendix A

Notation and Conventions

R: set of real numbers.

R^d: d-dimensional real space.

C^n: A function differentiable n times.

$f(.)$: variable probability distributions.

$P(.)$: specific named probabilities.

X_{ij}: j^{th} observation (measurement) at the i^{th} location (site).

Y_{ij}: j^{th} intrinsic parameter of the i^{th} location (site).

$\mathbf{P_i}$: Decision vector variable.

$\mathbf{p_i}$: A decision vector.

$\mathbf{P_i^\star}$: Optimal decision vector variable.

$Sgn(x)$: $Sign$ function; $Sgn(x) = 1$ if $x > 0$ else zero.

$Id\_Sgn(x, y)$: 1 if $Sgn(x) = Sgn(y)$ else zero.

$Z_{\setminus y}$: All members of set Z except y.

Appendix B

Glossary of Assumptions in Computer Vision

1. Transparency: Depth map could be a multi-valued function.

2. Opacity: Depth map is a single-valued function.

3. General Viewpoint: Statistically significant structural relationships among image features are unlikely to have resulted due to accidental viewpoint of the observer.

4. Coherence: Features resulting from a single physical event display statistical interaction.

5. Cohesiveness: Objects are usually compact and opaque.

6. Continuity: A given property is smoothly varying and is differentiable. Usually, used in the context of the linear image (or scene) features, *e.g,* boundaries.

7. Isotropy: all edge orientations are equally likely in the scene.

8. Homogeneity: The surface is covered with a single texture.

9. Independence: The edgel orientations extracted from the input image are independent.

10. Regularity: The texture is periodic and not stochastic.

11. Regular projective geometry: The unprojected texture in the scene and the imaging geometry are not conspiring to compensate for their individual contributions.

12. Smoothness: Differentiable property. Typically used for surface properties.

13. Integrability: The reconstructed surface is required to be physically meaningful.

14. Local spatial interaction: Spatially distant image features are less likely to display statistical interaction.

15. Lambertian Surface: Irradiance is independent of viewing direction and follows Lambert's cosine law.

16. Gaussian noise: Noise process follows Gaussian distribution.

17. Perfect segmentation: The objects of "interest" (foreground) can be accurately identified in the given image(s).

18. Single source of illumination: Objects in the image are illuminated by a single source of light.

19. Small motion: Infinitesimal motion of each image feature between successive image frames.

20. Polyhedral world: Objects in the image are polyhedral and possess definite volume.

21. Paraperspective projection: Imaging projection process is considered to be a composition of two projections: (i) first, all features on surface of an object are orthographically projected onto a a plane parallel to the image plane and passing through the center of mass of that object, (ii) the projected surface features are then projected on to the image plane using perspective projection.

22. Sufficient context: The projective distortion of surface markings due to imaging is not trivially degenerate.

23. Symmetry: Objects in the image are symmetric.

24. Rigidity: Objects in the image are rigid. No relative motion of any part of an object with respect to any other part of the object.

25. Locally spherical surface: Maximum and minimum curvature at every point on the object surface are identical.

26. No interreflection: The amount of illumination at each point on an object surface is accounted for by the source of direct illumination.

27. Piecewise continuity: Image features (boundaries) are piecewise continuous.

28. Dichromatic reflectance: Spectral distribution of the diffuse component is determined by the colorant in the surface whereas the specular component preserved the spectral distribution of the incident light.

29. Fractal surface model: The surface shape can be adequately described by a fractal.

30. Separable colors: Piecewise constant albedo.

31. Constant albedo: Albedo of all the surfaces in an image is identical and constant.

32. Gaussian blurring: Camera's point spread function can be approximated by a two dimensional Gaussian function.

Bibliography

[1] A. L. Abbott and N. Ahuja. Surface reconstruction by dynamic integration of focus, camera vergence, and stereo. In *Proceedings of the Second International Conference on Computer Vision*, pages 532–543, Tarpon Springs, 1988.

[2] N. Ahuja. A transform for detection of multiscale image structure. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 780–781, New York, 1993.

[3] N. Ahuja and M. Tuceryan. Extraction of early perceptual structure in dot patterns: integrating region, boundary, and component Gestalt. *Computer Vision, Graphics, and Image Processing*, 48:304–356, 1989.

[4] J. Aloimonos. Detection of surface orientation from texture i: The case of planes. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 584–593, Miami Beach, FL, 1986.

[5] J. Aloimonos. Purposive and qualitative vision. In *Proceedings of the 10th International Conference on Pattern Recognition*, pages 346–360, Atlantic City, New Jersey, 1990.

[6] J. Aloimonos and A. Bandopadhyay. Active vision. In *Proc. First IEEE International Conference on Computer Vision*, pages 35–54, London, 1987.

[7] Y. Aloimonos, editor. *Active Perception.* Lawrence Erlbaum Assoc., Hillsdale, NJ, 1993.

[8] Y. Aloimonos and D. Shulman. *Integration of Visual Modules: An Extension of the Marr Paradigm.* Academic Press, San Diego, CA, 1989.

[9] T. Başar and G. J. Olsder. *Dynamic Noncooperative Game Theory.* Academic Press, New York, NY, 1982.

[10] D. H. Ballard and C. M. Brown. *Computer Vision.* Prentice Hall, New Jersey, 1982.

[11] S. T. Barnard and W.B. Thompson. Disparity analysis of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(4):333–340, June 1980.

[12] H.G. Barrow and J.M. Tenenbaum. Recovering intrinsic characteristics from images. In A. Hanson and E. Riseman, editors, *Computer Vision Systems*, pages 3–26. Academic Press, New York, NY, 1978.

[13] H.G. Barrow and J.M. Tenenbaum. Interpreting line drawings as three dimensional surfaces. *Artificial Intelligence*, 17:47–75, August 1981.

[14] H.G. Barrow and J.M. Tenenbaum. Retrospective on interpreting line drawings as three dimensional surfaces. *Artificial Intelligence*, 59:71–80, 1993.

[15] P.N. Belhumeur and D. Mumford. A Bayesian treatment of the stereo correspondence problem using half-occluded regions. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 506–512, Champaign, Illinois, 1992.

[16] J. A. Benediktsson and P. H. Swain. Consensus theoretic classification methods. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(4), July/August 1992.

[17] J. Besag. On the statistical analysis of dirty pictures. *Journal of Royal Statistical Society Ser.B*, 48:259–302, 1986.

[18] T. O. Binford. Inferring surfaces from images. *Artificial Intelligence*, 17:205–244, August 1981.

[19] A. Blake and A. Yuille, editors. *Active Vision.* The MIT Press, Cambridge, MA, 1992.

[20] A. Blake and A. Zisserman. *Visual Reconstruction.* The MIT Press, Cambridge, MA, 1987.

[21] A. Blake, A. Zisserman, and G. Knowles. Surface descriptions from stereo and shading. In B. K. P. Horn and M. J. Brooks, editors, *Shape from Shading*, chapter 2, pages 29–52. The MIT Press, Cambridge, MA, 1989.

[22] D. Blostein and N. Ahuja. Shape from texture: Integrating texture-element extraction and surface estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(12):1233–1251, December 1989.

[23] R.M. Bolle and D.B. Cooper. Bayesian recognition of local 3-D shape by approximating image intensity functions with quadric polynomials. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:418–429, July 1984.

[24] R.M. Bolle and D.B. Cooper. On optimally combining pieces of information, with application to estimating 3-D complex-object position from range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:619–638, September 1986.

[25] R. C. Bolles and P. Horaud. 3DPO: A three–dimensional part orientation system. *The International Journal of Robotics Research*, 5(3), 1986.

[26] H.I. Bozma. *A Decentralized Integration in Modular Systems Using a Game Theoretic Framework.* PhD thesis, Yale University, New Haven, CT, 1992.

[27] H.I. Bozma and J.S. Duncan. Integration of vision modules: A game-theoretic framework. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 501–507, Maui, Hawaii, 1991.

[28] H.I. Bozma and J.S. Duncan. A modular system for image analysis using a game theoretic framework. *Image and Vision Computing*, 6(10):431–443, July–August 1992.

[29] H.I. Bozma and J.S. Duncan. A game-theoretic approach to integration of modules. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(11):1074–1086, November 1994.

[30] J. Brolio, B. A. Draper, J. R. Beveridge, and A. R. Hansen. ISR: A database for symbolic processing in computer vision. *Computer*, pages 22–30, Dec 1989.

[31] D. Buckley and J. Frisby. Interaction of stereo, texture and outline cues in the shape perception of three-dimensional ridges. *Vision Research*, 33(7):919–933, May 1993.

[32] D. Buckley, J. P. Frisby, and J. E. W. Mayhew. Interaction of texture and stereo cues in perception of surface slant: Evidence for surface orientation anisotropy in cue integration. *Perception*, 17:384, 1988.

[33] D. Buckley, J. P. Frisby, and J. E. W. Mayhew. Integration of stereo and texture cues in the formation of discontinuities during three-dimensional surface interpolation. *Perception*, 18(5):563–588, 1989.

[34] H. H. Bulthoff and H. A. Mallot. Integration of depth modules: Stereo and shading. *Journal of Optical Society of America*, 5(10):1749–1758, October 1988.

[35] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986.

[36] I. Chakravarty. A generalized line and junction labeling scheme with applications to scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2), April 1979.

[37] B. Chandrasekaran. What kind of information processing is intelligence? a perspective on AI paradigms and a proposal. In Patridge and Wilks, editors,

Source Book on the Foundations of AI. Cambridge University Press, Cambridge, England, 1987.

[38] Y.L. Chang and J.K. Aggarwal. Reconstructing 3D lines from a sequence of 2D projections: Representation and estimation. In *Third IEEE International Conference on Computer Vision*, pages 101–105, Osaka, Japan, 1990.

[39] P. Chou and C. Brown. The theory and practice of Bayesian image labelling. *International Journal of Computer Vision*, 4:185–210, 1990.

[40] P.H. Christensen and L.G. Shapiro. Three-dimensional shape from color photometric stereo. *International Journal of Computer Vision*, 13:213–227, 1994.

[41] C.-C. Chu and J. K. Aggarwal. The integration of image segmentation maps using region and edge information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(12):1241–1252, December 1993.

[42] J. J. Clark and A. L. Yuille. *Data Fusion for Sensory Information Processing Systems.* Kluwer Academic Publishers, Boston, MA, 1990.

[43] M. B. Clowes. On seeing things. *Artificial Intelligence*, 2:79–116, 1971.

[44] G. F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 41:393–405, 1990.

[45] R. Cowie and R. Perrott. From line drawings to impressions of 3D objects: Developing a model to account for the shapes that people see. *Image and Vision Computing*, 11:342–352, 1993.

[46] F. Crick. Function of the thalamic reticular complex: The searchlight hypothesis. In *Proceedings of the National Academy of Sciences*, pages 4586–4590, 1984.

[47] J. Cryer, P.S. Tsai, and M. Shah. Integration of shape from X modules: Combining stereo and shading. In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 720–721, New York, 1993.

[48] M. Daily. Color segmentation using Markov random fields. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 304–312, San Diego, June 1989.

[49] L. S. Davis and A. Rosenfeld. Cooperating processes for low-level vision: A survey. *Artificial Intelligence*, 17(1–3):245–263, August 1981.

[50] U. R. Dhond and J.K. Aggarwal. Structure from stereo – A review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510, June 1989.

[51] B. A. Draper, R. T. Collins, J. Brolio, A. R. Hanson, and E. M. Riseman. Issues in the development of a blackboard-based schema system for image understanding. In R. Engelmore and T. Morgan, editors, *Blackboard Systems*, chapter 8, pages 189–218. Addison Wesley, 1988.

[52] D. Dubois and H. Prade. *Possibility Theory: An Approach to Computerized Processing of Uncertainty.* Plenum Press, New York, 1988.

[53] J. Duncan and G. W. Humphreys. Visual search and stimulus similarity. *Psychological Review*, 96:433–458, 1989.

[54] R. Engelmore and T. Morgan. *Blackboard Systems.* Addison Wesley, New York, 1988.

[55] J. Ens and P. Lawrence. An investigation of methods for determining depth from focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(2):97–108, February 1993.

[56] L. D. Erman, F. Hayes-Roth, V. R. Lesser, and D. R. Reddy. The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty. *Computing Surveys*, 12(2):213–253, June 1980.

[57] E. A. Feigenbaum. The art of artificial intelligence: themes and case studies in knowledge engineering. In *Fifth Joint International Conference on Artificial Intelligence*, pages 1014–1029, Menlo Park, CA, 1977.

[58] F. P. Ferrie and M. D. Levine. Where and why local shading works. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1), 1989.

[59] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. In B. K. P. Horn and M. J. Brooks, editors, *Shape from Shading*, chapter 5, pages 89–122. The MIT Press, 1989.

[60] E. B. Gamble, D. Geiger, T. Poggio, and D. Weinshall. Integration of vision modules and labeling of surface discontinuities. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1576–1581, November/December 1989.

[61] J. Garding. Shape from texture for smooth curved surfaces in perspective projection. *Journal of Mathematical Imaging and Vision*, 2:327–350, 1992.

[62] J. Garding. Shape from texture and contour by weak isotropy. *Artificial Intelligence*, 64:243–297, 1993.

[63] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:721–741, November 1984.

[64] J. Gibson. *The Ecological Approach to Visual Perception.* Houghton-Mifflin, 1979.

[65] A. A. Giordano and F. M. Hsu, editors. *Least squares estimation with applications to digital signal processing.* Wiley, New York, 1985.

[66] A.S. Glassner, editor. *An introduction to Ray tracing.* Academic Press, New York, 1989.

[67] F. Glazer. Multilevel relaxation in low level computer vision. In A. Rosenfeld, editor, *Multiresolution Image Processing and Analysis*, pages 312–330. Springer–Verlag, New York, 1984.

[68] W. E. L. Grimson. *From Images to Surfaces.* MIT Press, Cambridge, MA, 1981.

[69] W. E. L. Grimson. *Object Recognition by Computer: The Role of Geometric Constraints.* MIT Press, Cambridge, MA, 1990.

[70] W.E.L. Grimson. Binocular shading and visual surface reconstruction. *Computer Vision, Graphics, and Image Processing*, 28:19–43, 1984.

[71] A. D. Gross. Shape from a symmetric universe. Technical Report CUCS–065–90, Columbia University, New York City, New York, 1990.

[72] Stephen Grossberg, editor. *Neural Networks and Natural Intelligence.* The MIT Press, Cambridge, MA, 1988.

[73] G. Healey and T. Binford. A color metric for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 10–17, Ann Arbor, MI, 1988.

[74] G. Healey and T.O. Binford. Local shape from specularity. In *Proc. First IEEE International Conference on Computer Vision*, pages 151–160, London, UK, 1987.

[75] J. E. Hochberg. *Perception.* Prentice-Hall, Englewood Cliffs, N.J., 1964.

[76] W. Hoff and N. Ahuja. Extracting surface from stereo: An integrated approach. In *Proc. First IEEE International Conference on Computer Vision*, pages 284–294, London, UK, 1987.

[77] R. L. Hoffman and A. K. Jain. Segmentation and Classification of Range Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5):608–620, September 1987.

[78] R. Horaud and M. Brady. On the geometric interpretation of image contours. In *The 1987 IEEE International Conference on Computer Vision*, pages 374–382, 1987.

[79] B. K. P. Horn. Obtaining shape from shading information. In *Shape from Shading*, pages 123–173. The MIT Press, Cambridge, MA, 1989.

[80] B. K. P. Horn and M. J. Brooks, editors. *Shape from Shading.* The MIT Press, Cambridge, MA, 1989.

[81] B. K. P. Horn and B. G. Schunck. Determining optic flow. *Artificial Intelligence*, 17, 1981.

[82] R. A. Hummel and S. W. Zucker. On the foundation of relaxation labeling processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5(3):267–287, May 1983.

[83] T. L. Huntsberger and S. N. Jayaramamurthy. A framework for multi-sensor fusion in the presence of uncertainty. In *Proceedings of 1987 Workshop on Spatial reasoning and Multi-sensor fusion*, pages 345–350, St. Charles, IL, October 1987. Morgan Kaufman, Los Altos.

[84] K. Ikeuchi. Generating an Interpretation Tree from a CAD Model for 3-D Object Recognition in Bin-Picking Tasks. *International Journal of Computer Vision*, 1(2):145–165, 1987.

[85] K. Ikeuchi and B. K. P. Horn. Numerical shape from shading and occluding boundaries. In *Shape from Shading*, pages 245–301. The MIT Press, 1989.

[86] D.W. Jacobs. Space efficient 3D model indexing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 439–444, Urbana, Illinois, June 1992.

[87] D.W. Jacobs. Robust and efficient detection of convex groups. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–771, New York, New York, June 1993.

[88] A.K. Jain and P. Flynn, editors. *3D Object Recognition Systems.* Elsevier, New York, 1993.

[89] A. Jepson and W. Richards. A lattice framework for integrating vision modules. *IEEE Transactions on Systems, Man, and Cybernetics*, 22:1087–1096, 1992.

[90] E. B. Johnston, B. G. Cumming, and A. J. Parker. Integration of depth modules: Stereopsis and texture. *Vision Research*, 33(5-6):813–826, 1993.

[91] B. Julesz. *Foundations of Cyclopean Perception.* University of Chicago Press, Chicago, Ill., 1971.

[92] T. Kanade. A theory of origami world. *Artificial Intelligence*, 13:279–311, 1980.

[93] Kalle Karu, A. K. Jain, and R. M. Bolle. Is there a texture in the image? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995. To appear.

[94] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331, 1988.

[95] J. Kender and E.M. Smith. Shape from darkness: Deriving surface information from dynamic shadows. In *Proceedings of the fifth National Conference on Artificial Intelligence*, pages 664–669, Philadelphia, PA, August 1986.

[96] J. Kittler and J. Illingworth. Relaxation labelling algorithms — a review. *Image and Vision Computing*, 3:206–216, 1985.

[97] J. J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13:321 - 330, 1984.

[98] W. L. Lakin, J. A. H. Miles, and C. D. Byrne. Intelligent data fusion for naval command and control. In R. Engelmore and T. Morgan, editors, *Blackboard Systems*, chapter 22, pages 443–458. Addison Wesley, 1988.

[99] S. Lakshmanan and K. C. Kluge. Lane detection for automotive sensors. In *IEEE International Conference on Acoustics Speech and Signal Processing*, Detroit, May 1995. To appear.

[100] D. Lamb and A. Bandopadhay. Shape from line drawings: Beyond Huffman-Clowes labeling. *Pattern Recognition*, 14:213–219, 1993.

[101] Y.G. Leclerc. Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision*, 3:73–102, 1989.

[102] Y.G. Leclerc and A.F. Bobick. The direct computation of height from shading. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 552–558, Maui, Hawaii, 1991.

[103] Y.G. Leclerc and M.A. Fischler. An optimization-based approach to the interpretation of single line drawings 3D wire frames. *International Journal of Computer Vision*, 9:113–136, 1992.

[104] C. Lee and A. Rosenfeld. Improved methods of estimating shape from shading using the light source coordinate system. In B. K. P. Horn and M. J. Brooks, editors, *Shape from Shading*, chapter 11, pages 323–348. Massachusetts Institute of Technology, Cambridge, MA, 1989.

[105] E.L. Lehmann. *Nonparametrics: Statistical Methods Based on Ranks.* Holden-Day, San Francisco, 1975.

[106] M. D. Levine. *Vision in Man and Machine.* McGraw-Hill, New York, New York, 1985.

[107] M. D. Levine. *Vision in Man and Machine.* McGraw-Hill, New York, New York, 1985.

[108] M. Livingstone and D. Hubel. Segregation of form, color, movement and depth: Anatomy, physiology, and perception. *Science*, 240:740–749, 1988.

[109] D. G. Lowe. *Perceptual Organization and Visual Recognition.* Kluwer Academic Publishers, Boston, MA, 1985.

[110] Tomàs Lozano-Perèz, W. Eric L. Grimson, and Steven J. White. Finding cylinders in range data. In *Proceedings of the 1987 IEEE International Conference on Robotics & Automation*, pages 202–207, Raleigh, NC, 1987.

[111] S. Madarasmi, D. Kersten, and T.C. Pong. Multi-layer surface segmentation using energy minimization. In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 774–775, New York, 1993.

[112] J. Malik. Interpreting line drawings of curved objects. *International Journal of Computer Vision*, 1:73–103, 1987.

[113] J. Malik and D. Maydan. Recovering three–dimensional shape from a single image of curved objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):555–566, June 1989.

[114] J. Malik and R. Rosenholtz. A differential method for computing local shape-from-texture for planar and curved surfaces. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 267–273, New York, 1993.

[115] B.S. Manjunath and R. Chellappa. A unified approach to boundary perception: Edges, textures, and illusory contours. *IEEE Transactions on Neural Networks*, 4:96–108, 1993.

[116] D. Marr. *Vision.* W.H. Freeman and Co., San Francisco, 1982.

[117] D. Marr and E. Hildreth. Theory of edge detection. *Proc. Royal Society London (B207)*, pages 187–217, 1980.

[118] D. Marr and H. K. Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proc. Royal Society of London (B200)*, pages 269–294, 1978.

[119] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, 194:283–287, October 1976.

[120] J. Marroquin. Random measure fields and the integration of visual information. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(4):705–716, July 1992.

[121] J. Marroquin, S. Mitter, and T. Poggio. Probabilistic solution of ill-posed problems in computational vision. *Journal of the American Statistical Association*, 82:76–89, 1987.

[122] L. Matthies and A. Elfes. Integration of sonar and stereo range data using grid representation. In *IEEE International Conference on Robotics and Automation*, pages 723–733, Philadelphia, PA, 1988.

[123] J. E. W. Mayhew and J. P. Frisby. Psychological and computational studies towards a theory of human stereopsis. *Artificial Intelligence*, 17:349–385, 1981.

[124] J. D. McCafferty. *Human and Machine Vision: Computing Perceptual Organization.* Ellis Horwood, New York, 1990.

[125] J. McClelland and D. Rumelhart, editors. *Parallel Distributed Processing*, volume 1,2. MIT Press, 1986.

[126] G. Medioni and R. Nevatia. Segment-based stereo matching. *Computer Vision, Graphics, and Image Processing*, 31:2–18, 1985.

[127] J. Modestino and J. Zhang. A Markov random field model-based approach to image interpretation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 458–465, San Diego, June 1989.

[128] M.L. Moerdler and T.E. Boult. The integration of information from stereo and multiple shape-from texture cues. In *Proc. 1988 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 524–529, Ann Arbor, Michigan, June 1988.

[129] H. Moravec. Towards automatic obstacle avoidance. In *Proceedings 5th Joint Conference on Artificial Intelligence*, page 584, Los Angeles, LA, 1977. William Kaufmann.

[130] H. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, pages 61–74, Summer 1988.

[131] P.G. Mulgaonkar, L.G. Shapiro, and R.M. Haralick. Shape from perspective: A rule-based approach. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 298–320, Miami Beach, FL, 1986.

[132] S. Nadabar. *MRF Model-Based Segmentation of Range Images.* PhD thesis, Michigan State University, 1992.

[133] S. G. Nadabar and A. K. Jain. Edge Detection and Labeling by Fusion of Intensity and Range Images. *Proc. of SPIE Conf. on Applications of AI: Machine Vision and Robotics*, 1708:108–119, 1992.

[134] S. G. Nadabar and A. K. Jain. Parameter estimation in MRF line process. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 528–533, Champaign, Illinois, 1992.

[135] M. Nagao and T. Matsuyama. *A Structural Analysis of Complex Aerial Photographs.* Plenum Press, New York, 1980.

[136] M. Nagao, T. Matsuyama, and H. Mori. Structural analysis of complex aerial photographs. In R. Engelmore and T. Morgan, editors, *Blackboard Systems*, chapter 9, pages 219–230. Addison Wesley, 1988.

[137] H.-H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:565–593, 1986.

[138] K. Nakayama and S. Shimojo. Experiencing and perceiving visual surfaces. *Science*, 257:1357–1363, 1992.

[139] V. S. Nalwa. Line–drawing interpretation: A mathematical framework. *International Journal of Computer Vision*, 2:103–124, 1988.

[140] S. K. Nayar, X.-S. Fang, and T. Boult. Removal of specularities using color and polarization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 583–590, New York, 1993.

[141] U. Neisser. *Cognitive Psychology.* Appleton-Century-Crofts, New York, NY, 1967.

[142] Q.L. Nguyen and M.D. Levine. 3-d object representation in range images using geons. In *Proc. the 11th IAPR International Conference on Pattern Recognition*, volume ICPR A, pages 149–153, The Hague, 1982.

[143] Y. Ohta, K. Mennobu, and T. Sakai. Obtaining surface orientation from texels under perspective projection. In *Proc. 7th. Int. Jt. Conf. Artificial Intell.*, pages 746–751, Oulu, Finland, 1981.

[144] J. Oliensis. Shape from shading as a partially well-constrained problem. *Computer Vision, Graphics, and Image Processing*, 54:163–183, September 1991.

[145] J. Oliensis and P. Dupuis. Direct method for reconstructing shape from shading. In *Proc. of SPIE conference on Geometric Methods*, volume 1570, pages 116–128, San Diego, California, July 1991.

[146] S. Pankanti, C. Dorai, and A. K. Jain. Robust feature detection for 3d object recognition. In *Proceedings of 2031 SPIE Conference on Geometric Methods In Computer Vision II*, pages 366–377, San Diego, CA, July 1993.

[147] S. Pankanti and A. K. Jain. On integration of vision modules. Technical Report TR–CS, Michigan State University, June 1994.

[148] S. Pankanti, A. K. Jain, and M. Tuceryan. On integration of vision modules. In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 316–322, Seattle, WA, June 1994.

[149] T. Pavlidis. *Algorithms for Graphics and Image Processing.* Computer Science Press, Rockville, Maryland, 1982.

[150] T. Pavlidis and C. Liow. Integrating region growing and edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI–12:225–233, March 1990.

[151] A. P. Pentland. Shading into texture. *Artificial Intelligence*, 29:147–170, 1986.

[152] A. P. Pentland. Local shading analysis. In B. K. P. Horn and M. J. Brooks, editors, *Shape from Shading*, chapter 15, pages 443–488. The MIT Press, Cambridge, MA, 1989.

[153] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317:638–643, 1985.

[154] K. Prazdny. Detection of binocular disparities. *Biological Cybernetics*, 52:93–99, 1985.

[155] V. S. Ramachandran. The utilitarian theory of perception. In *Proceeedings of APA Symposium: Theories of Perception*, Washington, DC, 1986.

[156] V. S. Ramachandran. Interaction between motion, depth, color and form: the utilitarian theory. In A. Blake and T. Troscianko, editors, *AI and the Eye*, chapter 31, pages 348–360. John Wiley and Co., New York, 1990.

[157] V. S. Ramachandran. Visual perception in people and machines. In A. Blake and T. Troscianko, editors, *AI and the Eye*, chapter 3, pages 21–77. John Wiley and Co., New York, 1990.

[158] V. S. Ramachandran. Biological perspective. In G. Carpenter and S. Grossberg, editors, *Neural networks for vision and image processing*, chapter 3, pages 46–91. The MIT Press, 1992.

[159] B. S. Rao and H. Durrant-Whyte. A decentralized Bayesian algorithm for identification of tracked targets. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-23(6):1683–1698, November/December 1993.

[160] R. Reiter and A. K. Mackworth. A logical framework for depiction and image interpretation. *Artificial Intelligence*, 41:125–155, 1990.

[161] I. Rigoutsos and R. Hummel. Distributed Bayesian object recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 180–186, New York, 1993.

[162] J. Rissanen. A universal prior for integers and estimation by minimal description length. *Annals of Statistics*, 11(2):416–431, 1983.

[163] L. G. Roberts. Machine perception of three-dimensional solids. In J. Tippett et al., editor, *Optical and Electro-Optical Information Processing*. The MIT Press, Cambridge, MA, 1965.

[164] A. Rosenfeld. 'Expert' Vision systems: Some issues. *Computer Vision, Graphics, and Image Processing*, 34(2):99–102, 1986.

[165] A. Rosenfeld and R. A. Hummel. Scene labeling by relaxation operations. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(6):420–433, June 1976.

[166] S. Sarkar and K.L. Boyer. Integration, inference, and management of spatial information using Bayesian networks: Perceptual organization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:256–274, 1993.

[167] A. H. Schistad-Solberg, A. K. Jain, and T. Taxt. Multisource classification of remotely sensed data: Fusion of Landsat TM and SAR images. *IEEE Transactions on Geoscience and Remote Sensing*, 32(4):768–778, July 1994.

[168] R. Sekuler and R. Blake. *Perception.* McGraw Hill, New York, 1990.

[169] G. Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, NJ, 1976.

[170] S. A. Shafer, A. Stentz, and C. E. Thorpe. An architecture for sensor fusion on mobile robot. In *IEEE International Conference on Robotics and Automation*, pages 2002–2011, San Francisco, CA, 1986.

[171] W.J. Shomar and T.Y. Young. Three-dimensional shape recovery from line drawings. In Tzay Y. Young, editor, *Handbook of Pattern Recognition and Image Processing*, volume 2 (Computer Vision), pages 53–100. Academic Press, San Diego, CA, 1994.

[172] W. Singer, C. Gray, P. Konig, A. Artola, and S. Brocher. Formation of cortical assemblies. In *Cold Spring Harbor Symposia on Quantitative Biology*, volume 50, pages 939–952, 1990.

[173] Ajit Singh and Peter Allen. Image-flow computation: An estimation-theoritic framework and a unified perspective. *Computer Vision, Graphics, and Image Processing*, 56(2):152–177, September 1992.

[174] F. Solina and R. Bajcsy. Recovery of parametric models from range images: The case for superquadrics with global deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):131–147, February 1990.

[175] H. W. Sorenson, editor. *Parameter estimation: Principles and problems.* Marcel Dekker, New York, 1980.

[176] G. Stockman, G. Lee, and S. W. Chen. Reconstructing line drawings from wings: The polygonal case. In *Proceedings of the Third IEEE International Conference on Computer Vision*, pages 526–529, Osaka, Japan, 1990.

[177] K. Sugihara. *Machine Interpretation of Line Drawings.* The MIT Press, Cambridge, MA, 1986.

[178] B. J. Super and A. C. Bovik. Shape from texture using local spectral moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(4):333–343, 1995.

[179] R. Szeliski. Regularization uses fractal priors. In *National Conference on Artificial Intelligence*, pages 749–754, Seattle, WA, 1987.

[180] M. J. Tarr and M. J. Black. A computational and evolutionary perspective on the role of representation in vision. In preparation, June 1992.

[181] R. R. Tenney and N. R. Sandell. Strategies for distributed decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(8):527–538, 1981.

[182] R. R. Tenney and N. R. Sandell. Structures for distributed decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(8):517–526, 1981.

[183] D. Terzopoulos. Multilevel computational processes for visual surface reconstruction. *Computer Vision, Graphics, and Image Processing*, 24:211–221, 1983.

[184] D. Terzopoulos. Multilevel reconstruction of visual surfaces: variational principles and finite-element representations. In A. Rosenfeld, editor, *Multiresolution Image Processing and Analysis*, pages 237–310. Springer–Verlag, New York, 1984.

[185] D. Terzopoulos. Image analysis using multigrid relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(2), March 1986.

[186] D. Terzopoulos, A. Witkin, and M. Kass. Symmetry-seeking models and 3D object recognition. *International Journal of Computer Vision*, 1:211–221, 1987.

[187] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-Posed Problems.* Winston & Sons, Washington, D.C., 1977.

[188] J. Ton and Anil K. Jain. Registering Landsat image by point matching. *IEEE Transactions on Geoscience and Remote Sensing*, 27(5):642–651, 1989.

[189] D. A. Trytten. *Integrating Diverse Perceptual Modules to Create a 2.5 Dimensional Sketch.* PhD thesis, Michigan State University, E. Lansing, Michigan, 1992.

[190] D. A. Trytten and M. Tuceryan. Segmentation and grouping of object boundaries using energy minimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 730–731, Maui, Hawaii, June 1991.

[191] D. A. Trytten and Mihran Tuceryan. Construction of labeled line drawings from intensity images. *Pattern Recognition*, 28(2):171–198, February 1995.

[192] P.S. Tsai and M. Shah. A fast linear shape from shading. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–718, Champaign, Illinois, 1992.

[193] Ch. von der Malsburg and W. Schneider. A neural cocktail–party processor. *Biological Cybernetics*, 54:29–40, 1986.

[194] D. Waltz. Understanding line drawings of scenes with shadows. In P. H. Winston, editor, *Psychology of Computer Vision*, pages 19–91. McGraw-Hill, New York, NY, 1975.

[195] J. Weng, N. Ahuja, and T. S. Huang. Matching two perspective views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):806–825, August 1992.

[196] J. Weng, N. Ahuja, and T. S. Huang. Optimal motion and structure estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):864–884, September 1993.

[197] Juyang Weng, Thomas S. Huang, and Narendra Ahuja, editors. *Motion and structure from image sequences.* Springer-Verlag, New York, 1993.

[198] A. P. Witkin. Recovering surface shape from texture. *Artificial Intelligence*, 17:17–45, August 1981.

[199] A. P. Witkin and J. M. Tenenbaum. On the role of structure in vision. In J. Beck, B. Hope, and A. Rosenfeld, editors, *Human and Machine Vision*, pages 481–543. Academic Press, New York, 1983.

[200] J. M. Wolfe and K. R. Cave. Deploying visual attention: The guided search model. In A. Blake and T. Troscianko, editors, *AI and the Eye*, chapter 4, pages 79–103. John Wiley and Co., New York, 1990.

[201] L.B. Wolff and T.E. Boult. Constraining object features using a polarization model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:635–657, August 1991.

[202] Robert J. Woodham. Determining Surface Curvature with Photometric Stereo. In *Proc. 1989 IEEE International Conference on Robotics and Automation*, pages 36–42, May 1989.

[203] G. Yang and T.S. Huang. Human face detection in a scene. In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 453–458, New York, 1993.

[204] Semir Zeki. *A vision of the brain.* Blackwell Scientific Publications, Boston, 1993.

[205] R. Zhang, P.S. Tsai, J. E. Cryer, and M. Shah. Analysis of shape from shading techniques. In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 377–384, Seattle, 1994.

[206] S. W. Zucker. Vertical and horizontal processes in low level vision. In A. Hanson and E. Riseman, editors, *Computer Vision Systems*, pages 187–195. Academic Press, New York, NY, 1978.

[207] S. W. Zucker and R. A. Hummel. Toward a low-level description of dot clusters: Labeling edge, interior and noise points. *Computer Graphics and Image Processing*, 9:213–233, 1979.