Architectural Studies for Visual Processing

André J.S. Yakovleff

Thesis submitted for the degree of

Doctor of Philosophy

Department of Electrical and Electronic Engineering

The University of Adelaide

May 1995

Table of contents

List of figures

List of tables

Abstract

This thesis explores the issue of copying natural vision processes with a view to providing sensing information in real-time to the lowest control levels of an autonomous vehicle. The approach is first presented within the general context of cognition, and it is argued that the interpretation of sensory data should result in a level of perception which is tailored to the requirements of the control system.

The argument is supported by electrophysiological evidence suggesting that some species rely primarily on motion information in order to navigate and to avoid obstacles. In particular, the motion detection and motor control mechanisms of insects appear to be closely linked, thus implying that their responses to stimuli may be induced from a low level of perception. Hence, by utilising some of the underlying features of the insect visual system, the computational costs associated with some conventional visual processing methods may be bypassed in an innovative way.

In an attempt to assess the validity of the argument, it is proposed to build a concept demonstrator on a single VLSI chip, by exploiting analog and digital design techniques. The chip implements a model which consists of interpreting local changes in contrast in order to infer directional motion information. Moreover, the information is conveyed in a readily usable format, and new algorithms to estimate the relative velocities of moving objects are presented. Results obtained by testing the chip in real-time indicate that the directions of moving stimuli are correctly perceived and are consistent with their velocities. However, some aspects of the implementation should be improved upon to make the chip suitable to a wider range of applications.

The results indicate that directional and velocity data may be further interpreted in order to estimate the range of objects in the visual field, and to provide a collision avoidance mechanism. Finally, the interactions between sensing and control are discussed, with a view to enhancing sensory information through adaptive behaviour.

This work contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution, and to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text.

I give consent to this copy of my thesis, when deposited in the University Library, being available for loan and photocopying.

SIGNED : ..... *DATE :* 26 May 1995

Acknowledgements

"*A tout seigneur, tout honeur*"; my supervisor, Professor Robert Bogner was always available to discuss any matter, and often suggested new approaches and ideas. Bob's recurring question, "What are the criteria by which you can judge whether or not this thing works?" kept me on my toes. Professor Adrian Horridge, now retired from Australian National University, was instrumental in initiating the "bugeye" project. In fact, it was his seminal work on insects, coupled with a very contagious enthusiasm, which provided the original inspiration.

I am also indebted, in a somewhat pragmatic sense, to my employer, the Defence Science and Technology Organisation (DSTO), and in particular to every echelon of management in the Information Technology Division. In spite of the cyclical upheavals which are a feature of the public service, management's support never flinched more than a semi-quaver, and hence I wish to express my gratitude to Dr. Brian Billard, Dr. Ian Chessell, Peter Drewer, Dr. Stephen Hood, Dr. Don Sinnott, and John Ziukelis.

Still on the subject of management, but this time at the University of Adelaide, Derek Abbott, Dr. Abdessalam Bouzerdoum ("Salim"), Associate Professor Kamran Eshraghian, and Michael Liebielt, all played various roles. Kamran essentially set-up the project, Michael kept us financially responsible, Derek contributed his expertise in semi-conductor physics (as well as asking me probing questions concerning looming pictures), and Salim more or less ran the show, under Bob's general supervision.

Salim thus has an insider's understanding of the bugeye project, and happens to be an authority on visual processing. He was also one of the three "readers" of my numerous drafts, and kept asking me questions such as: "What has this got to do with your thesis?". The two other readers are both from DSTO. As a VLSI man, Mario Cavaiuolo would try to understand everything, would become annoyed if he could not, and would therefore immediately spot a fudged explanation ("I don't get it. How does it work?"). The third reader, Dr. Martin Burke, is somewhat recalcitrant when it comes to VLSI and insects. However, Martin contributed in a top-down fashion, by making me think about the presentation of the arguments. His catch phrase: "The flow of arguments leaves something to be desired", would sometimes leave me perplexed.

And now the bugeye team. Ali Moini and myself are credited with designing the original bugeye, but I freely admit that Ali carried out the trickiest part of the job. I also collaborated closely with Thong Nguyen, with whom I had many edifying arguments concerning the best way to use the chip. A late addition to the team, Andrew Blanksby brought humour, enthusiasm, basket-ball, and practical skills to the project.

In the course of this research, I had many fruitful discussions with friends and colleagues, notably at DSTO. In particular, Dr. Mark Nelson's expertise in robotics was inspirational. Mark also willingly answered my numerous queries concerning programming in C, without showing any sign of irritation. Dr. Paul Miller also showed a lot of interest in the bugeye project, and brought binary optics technology to our attention. As a fellow student and professional at DSTO, Peter Deer and I had frequent spirited discussions.

Other students and colleagues to whom I am indebted in one way or another include John Yesberg and Dr. Jim Grundy (DSTO), and Ook Kim (Seoul University).

Last, but not least, Helen Hayes provided enthusiastic support and encouragement. *Du fond du cœur*, thank you Helen.

Preamble

The compositional procedure known as "fugue" is one of western music's most intricate constructions. One could point out, though, that anybody who possesses a half-decent knowledge of certain counterpoint and harmony rules could compose a fugue. Then, an uncompromising analysis might reveal that all but the first few coefficients of a Fourier decomposition are absent, with the exception of a few tension-building dissonances peppered-in for good measure. Music connoisseurs, of course, would tend to disagree, and cite the fugue in J.S. Bach's "Well-tempered Clavier" as evidence against such preposterous trivialisations.

Fugues first appeared in the Fourteenth century, well before Bach formalised any rules, and are constructed by mixing human or instrumental "voices", generally four. These voices are simultaneous variations around a central theme, and are both melodious and harmonious with respect to each other. As different voices may be produced by the same instrument, the distinction between the melodic and harmonic aspects is important. While a computer analysis of the score grouped on a single staff might pick out consistencies in the patterns of "vertical" separations, and thereby infer some harmonic rules, the "horizontal" separations are another matter, as melodies may cross-over and create ambiguities. Yet, what makes a well composed - and executed - fugue remarkable, is most listeners' natural ability to chose whether to follow a particular melody, i.e., one voice, or listen instead to the ensemble.

The fugue metaphor may be applied outside the realm of music, and indeed Hofstadter[1] uses it to great effect in his "Ant Fugue", where several dialogues take place concurrently. Again, a listener may select a particular conversation from the "brouhaha" by making use of a number of cues such as voice pitch and timbre, spatial separation, or time correspondence, to determine which voices "belong" to the same conversation. Of course, another cue used by the listener is semantic context, which one could consider as a transposition of melodic context into words.

In many respects, visual perception processes are also fugue-like. For instance, the voices of a fugue can be associated with objects or particular areas in a visual scene, where contextual regions form uniformly textured surfaces delineated by edges, colours, motion differentials

1. Hofstadter D.R. (1979), in *Gödel, Escher, Bach: An Eternal Golden Braid*, Penguin Books

and so on. Focusing the attention, or tracking a moving object, would then be similar to the single voice perception in the musical fugue. In either case, what seems to occur is a top-down process, whereby the senses become attuned to making certain discriminations, reinforcing the perception, and in time the conscious attentional effort recedes until it becomes virtually unnoticeable.

However, discriminations sometimes appear to be automatic, almost hard-wired at the source, suggesting the existence of bottom-up processes which complement, but are not necessarily governed by, the contextual cues provided by cognition. For instance, in the musical fugue, no matter how focused the attention may be on a particular melody, the other voices constantly demand attention. Likewise, while concentrating on an object, a sudden movement occurring in the periphery of the visual field may elicit a reflex to shift the focus of attention.

From a biological perspective, receptive cells in the retina respond in specific ways to stimuli, and likewise, sound vibrations are transduced into nerve impulses by the cochlea. Perception can then be defined as the process by which these nerve impulses are transformed into patterns of frenetic activity capable of titillating parts of the conscious brain. In other words, perception may be considered as an "awareness interface", where electro-chemical reactions are somehow sorted out and assigned ethereal labels.

Thus, several questions may be asked. For instance, is the assignment of a label dictated by stored knowledge in a top-down fashion, or is it essentially a bottom-up, automatic, process? if both processes exist, at which stage and how do they mingle? are there circumstances where one process consistently dominates the other? Venturing answers to these questions is a decidedly ungratifying exercise, as answers are deemed to be either truisms or absurdities. Fugues may be employed as ammunition by either school of thought, as they evoke altered states of consciousness as well as somewhat dithering perception processes.

However, there is another reason why fugues are invoked in this preamble. The issue of using biological systems as a source of inspiration is a recurring theme in this thesis, and grabs the focus of attention at every opportunity.

CHAPTER 1 Introduction

The context surrounding the "bugeye" project, the assumed background of the reader, and the aims and significance of the work, are introduced. Then, the structure of the thesis is presented and the various contributions from my colleagues are clarified.

1. General

The research described in the thesis is based upon original ideas of the author, inspired from research by Professor Adrian Horridge, and suggested by discussions with Professor Robert Bogner and Doctor Abdessalam Bouzerdoum.

The development of the ideas, and the work described in this thesis, were undertaken while the author was a student at the University of Adelaide under the supervision of Professor Bogner, and a full-time employee of the Defence Science and Technology Organisation (Electronics and Surveillance Research Laboratories, Information Technology Division, Computer Systems Architectures Group).

This thesis describes the development and the implementation of an electronic device, whose design is inspired from natural visual systems, and whose application domain is the control of an autonomous vehicle. Hence, the subject matter is related in various ways to three disciplines: biological sciences, engineering, and robotics.

2. Context

2.1 Motivation

The work presented in this thesis is motivated by the desire to reduce the computational costs associated with processing sensory information, in order to meet the stringent power and size requirements of small autonomous vehicles.

Experimental studies of biological systems indicate that lightweight species make an efficient use of limited sensing and processing resources. In particular, it appears that some insects control their flight path primarily by responding to the indications provided by visual motion discrimination mechanisms. Thus it is conjectured that motion information alone may be sufficient for the control system of an autonomous vehicle to accomplish low-level tasks, such as collision avoidance.

2.2 Aims, methods, and significance

The objective of this thesis is to investigate the feasibility of utilising some of the characteristics of the insect sensory-motor systems in the control of an autonomous vehicle. To this end, it is proposed to design a silicon device which implements a motion detection scheme based on the insect visual system, and interprets motion information in an efficient and innovative fashion. Experimental evidence is then utilised to examine the manner in which motion information could be exploited at a low level of control, such as short-term path-planning and collision avoidance. It should be pointed out that the design of the control system itself is not investigated in this thesis, but is the object of ongoing work.

Biologically-inspired devices could have a significant impact on the design and applicability of autonomous systems. Firstly, devices such as that described in this thesis are usually small and have low power requirements. Secondly, sensing information can be provided in real-time, and may be readily interpreted, thus enhancing the control system's reactions to changes in the environment. Finally, these devices may also be used as tools permitting fresh investigations into matters such as the interactions between sensors and motor systems.

2.3 Assumed background

In this thesis, natural vision is the main source of inspiration behind the design of an integrated circuit, whose intended purpose is to assist in the control of an autonomous vehicle. Thus, the topic of this thesis is multi-disciplinary, and includes elements of biology, electrical engineering, and robotics.

Aspects of analog and digital VLSI, such as systems architectures, design, and implementation, constitute the main technical subject matter of this thesis. However, an in-depth knowledge of VLSI is not critical to understand the concepts which are presented.

It is assumed that the reader has some background knowledge of certain functional aspects of biological vision, particularly motion perception and basic neuronal mechanisms. Further information on these subjects can be found in the references, which are mainly to experimental studies. It is also assumed that the reader is familiar with the considerable body of work concerning computational methods of visual processing, even though the methods themselves are not central to the main argument of this thesis.

Finally, while the last part of this thesis concerns the sensing requirements of an autonomous vehicle, as distinct from vehicle control *per se*, it is nonetheless assumed that the reader comprehends the issues involved in designing robotics systems.

2.4 History of the "bugeye" project

The work presented in this thesis emanates from an auspicious meeting, held in early 1992 in Adelaide, at the instigation of Professor Adrian Horridge, from the Australian National University in Canberra, and Professor Robert Bogner.

Professor Horridge's extensive research on the insect visual system had led him to realise that some of the mechanisms found in the early stages of natural vision could be implemented artificially. His approach was different from that of other researchers, such as Carver Mead at Caltech, in that the aim was not to create an electronic version of these mechanisms, but instead to reproduce the *functions* underlying these mechanisms. In other words, Horridge wanted to build a "seeing" system capable of interpreting visual stimuli in a useful manner, as distinct from an "imaging" system.

The purpose of the meeting was to discuss ways in which Horridge's ideas could be put into practice. The outcome was that Professor Bob Bogner, Doctor Abdessalam Bouzerdoum, and the VLSI design team at the University of Adelaide headed by Associate Professor Kamran Eshraghian, would provide the necessary resources and engineering skills.

Ali Moini and myself then set about the task of mapping one of Horridge's models in a VLSI chip, which rapidly came to be known as the vernacular "bugeye", a term coined by Bob Bogner. The extent of our group's financial resources also became rapidly apparent, as we discovered that the available funds would barely cover the cost of a single fabrication run. Hence, we concluded that our first chip should incorporate as much functionality as possible, the idea being to attract additional funds on the strength of a compelling concept demonstrator.

3. Thesis structure and claims to contribution

The structure of the thesis comprises three main sections, as depicted in Figure 1.1. The first section, which comprises chapters 2, 3, and 4, is mainly concerned with introducing the main concepts and presenting the background. The second section (chapters 5, 6, and 7) focuses on hardware and testing. The last section (chapters 8 and 9) constitutes the link between testing results and the arguments of the first section. Throughout the thesis, each chapter opens with a short summary of its contents, followed by the main body, and, when appropriate, the issues and findings are discussed and summarised at the end of the chapter.

As the bugeye project involves a group of people, it is inevitable that various members of the group have contributed in one way or another to the work presented in my thesis. Moreover, for the sake of completeness, I have chosen to describe in some detail work which was carried out entirely by fellow students. However, while credit is given where it is due throughout the thesis, there may remain some ambiguities which could be misconstrued. The following list is meant to describe the structure of the thesis as well as to underline the various contributions, some of which have already been mentioned in the "Acknowledgements".

Section 1: background and inspiration

An autonomous mobile vehicle is viewed in the context of a cognitive system, which uses percepts derived from sensory information together with high-level objectives in order to act upon motor systems. In the main discussion, it is argued that employing biologically-inspired methods, as distinct from computational methods, may be beneficial to accomplish specific visual perception tasks. Computational visual processing is introduced, but not fastidiously elaborated upon, as such methods are not central to the thesis. For further details, the reader should thus refer to Appendix A.

Some functional aspects of motion perception are then reviewed in Chapter 3, which concentrates on motion detection models inspired by the insect visual system. The so-called "template model", which appears to be particularly suited to a hardware implementation, is described and characterised next (Chapter 4).

Contributions: The motion detection model described in Chapter 4 was devised by Professor Adrian Horridge, whose contribution is duly noted. The method used to characterise the model is a simplified version of more elaborate simulations effected by Thong Nguyen.

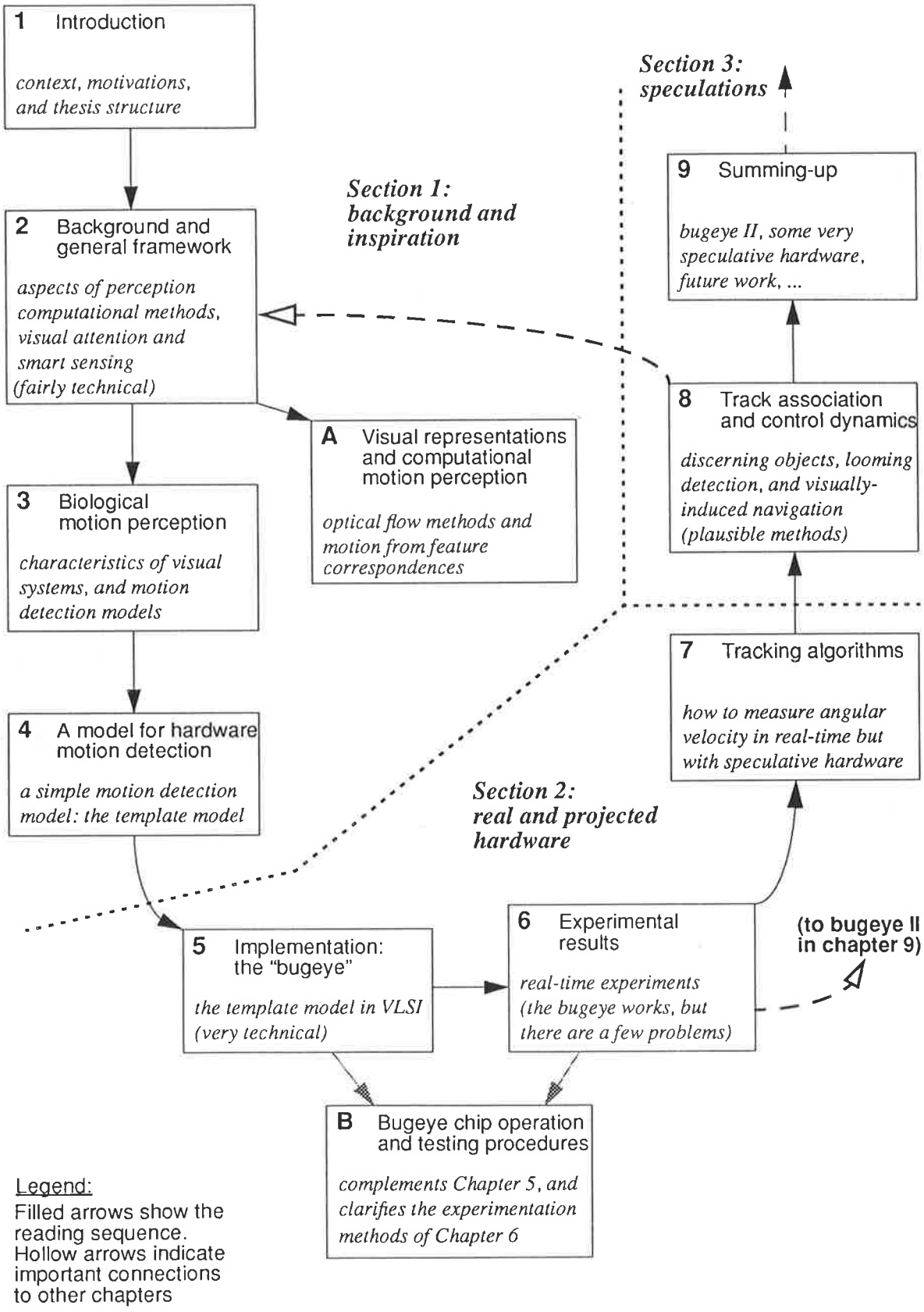


Figure 1.1 Thesis structure ("road map")

Section 2: real and projected hardware (implementation and results)

The VLSI implementation of the template model is presented in Chapter 5, and experimental results, where the motion of objects is detected in real-time, are presented in Chapter 6. Two algorithms which extract velocity information are then introduced (Chapter 7), as well as possible hardware implementations.

Contributions: The hardware architecture, timing scheme, and general operation of the "bugeye" chip (Chapter 5), result from an equally shared collaboration involving Ali Moini and myself. As regards the implementation, however, the respective contributions differ to some extent. The chip comprises three main blocks, namely, analog detection, digital interpretation, and processing. In addition to the routing between the various blocks and to the pads, Ali carried out the design and layout of all the analog components, admittedly the most difficult to implement properly. We designed the interpretation block together, but Ali was responsible for the layout. The design and layout of the processor were entirely my work.

The hardware interface and the computer software used for testing the chip (Chapter 6) were designed and implemented by Ali and myself, while Andrew Blanksby built the velocity measurement apparatus, and measured the optical characteristics of the lens. All members of the group participated in testing the chip and in obtaining experimental results.

The so-called "stair-step tracking" algorithm of Chapter 7 is entirely Thong Nguyen's idea, although we collaborated on hardware design aspects. The forward tracking algorithm and the rest of the chapter, including the simulations of both algorithms, describe my own work.

Section 3: application to robotics control and speculations

Chapter 8 exposes my thoughts concerning the use of the bugeye by the control system of an autonomous vehicle, and discusses some of the possible implications. Of particular interest are collision avoidance mechanisms and the interpretation of information induced by self-motion. Finally, the arguments are summarised in Chapter 9, and further implications of the work are speculated upon.

Contributions: In Chapter 8, the track association and "looming" detection mechanisms, as well as the manner in which such information could be utilised, are primarily my ideas. However, I have had many inspirational discussions with a number of people (see the "Acknowledgements"). Finally, Chapter 9 refers to a new version of the bugeye, which was designed entirely by Ali Moini.

CHAPTER 2 Background and general framework

Computational aspects of visual processing are presented in the broader context of cognitive processes. Next, the inherent restrictions of autonomous systems are used to highlight the advantages of the purposive vision paradigm, whereby visual information is interpreted at the sensor level. It is argued that the benefits gained from this concept outweigh the loss of generality.

1. A cognitive framework

The emerging field of cognitive science does not have a single traceable origin but instead brings together fields which can be described as "sciences of the mind" such as neuroscience, psychology, philosophy and linguistics. In a sense, cognitive science is gestaltist in that it attempts to merge different approaches: for instance, by combining philosophy's abstract representations of concepts with the underlying neural mechanisms.

Of particular interest to this work is the role of visual perception in a cognitive process. Not surprisingly, in light of its subjective nature, the problem has fuelled the more general debate about how perceived information is represented in the brain. Is information stored as retrievable "knowledge" in the form of a direct mapping obtained from physical information, or as a set of symbols that are used to *reconstruct* knowledge, as opposed to retrieving it? In the context of this thesis, the latter "computational" view is perhaps better suited to illustrate the interactions between perception and other processes.

Laird *et al.* (1987) have formalised the computational approach in the so-called "Soar*" cognitive architecture, a much simplified version of which is shown in Figure 2.1. The archi-

tecture includes a production memory which is long-term and used for knowledge, both declarative and procedural. New productions, i.e., more knowledge, are created from the experiences gained in solving problems through a process called "chunking". The working memory contains information about goals and how to achieve them, and interacts with the outside world through perceptual and motor systems. The reader is referred to Newell *et al.* (1989) for an examination of the role of such an architecture in cognition.

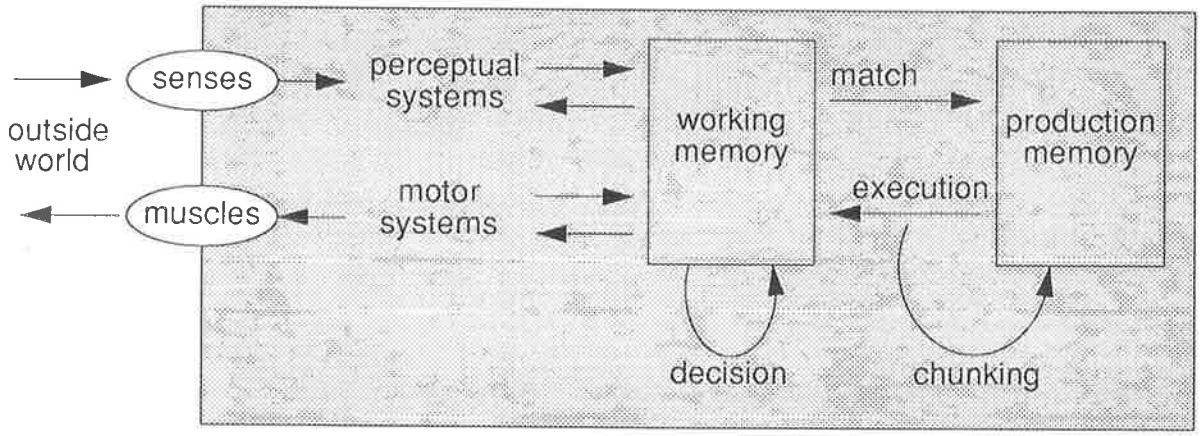


Figure 2.1 The Soar* cognitive architecture

In an attempt to illustrate cognitive processes, cognitive science may draw on examples provided by more practical, or "environmental" sciences such as biology, engineering and computing. In fact, the Soar* architecture may be interpreted as representing structural mechanisms in the mental sense, as well as "systems" in the physical or environmental sense.

This thesis focuses on the use of perception to achieve a level of behaviour adequate for a specific purpose. Instead of constructing a model of the environment, an attempt is made to interpret sensory information in such a way that it can be utilised by motor systems, and to avoid as much as possible the recourse to symbolic representations, in the spirit of methods advocated by Brooks (1991a). This is only one aspect of work which has been carried out by a group of individuals whose interests include biology, VLSI and semiconductor physics, signal processing and control systems. Hence it may be appropriate to present this work globally, as a "machine" version of a cognitive system: a biologically-inspired visual system which perceives motion through an interpretative process has been designed and implemented. Low-level, purely reactive, actions may be directly derived from the motion information, while further interpretations result in inferring activity at a higher level of behaviour. From this "conception" decisions are made, using a combination of reactive processes and procedural knowledge, as to what actions should be taken to guide an autonomous vehicle around obstacles.

The process by which detected information eventually results in the activation of motor systems is depicted in Figure 2.2. Note that in any system of only moderate complexity, processes

such as "perception" and "conception" may involve several steps which, for simplicity, are not represented here.

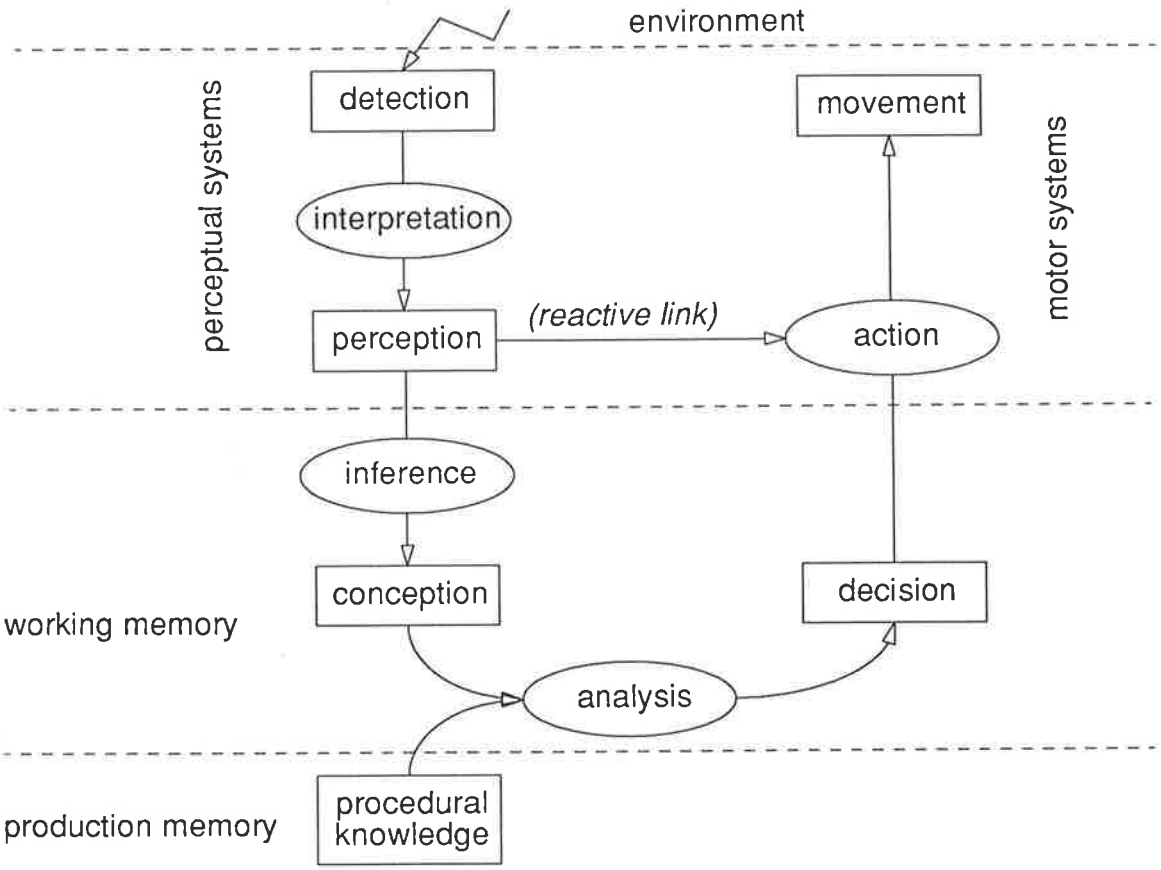


Figure 2.2 Cognitive framework

In the remainder of this chapter, the issue of defining a "machine" version of perception is addressed, followed by a discussion on general aspects of visual processing to explain why we chose the "purposive vision" paradigm for our system.

2. A criterion for perception

As mentioned earlier, the understanding of perception processes is a contentious issue among psychologists. However, taking sides in this debate is outside the scope of this thesis. Therefore, only those aspects that are of interest for establishing a criterion for perception are to be outlined.

Unlike sciences such as physics, where the observer enjoys the luxury of remoteness with respect to phenomena, the introspective nature of psychology tends to engender strong feelings, as illustrated by the antagonistic views expressed by Fodor & Pylyshyn (1981) and Turvey *et al.* (1981). The former are proponents of the traditional approach, which holds that

perception is computational and suggests that a representation of the outside world is *constructed* through inferences, memories and so on. By contrast, the alternative approach sometimes referred to as "direct realism" or "ecological", as expressed by its initiator (see Gibson, 1979, for a review), is essentially behaviourist and assumes that perception involves only detection, thereby eliminating any form of mental construction. In other words, visual perception would only require a simple selection, or "pick-up", of "invariant properties" present in the light impinging on the observer, implying that the encoding of external stimuli by the transducers, or detectors, is somehow driven by the environment. This contradicts the view that, as physical entities, detectors are in general quite primitive and must be constrained. For instance, detector cells in the retina respond non-uniformly and only to a restricted class of properties of the ambient light, and hence properties may not always be unequivocally perceived. Therefore the traditional view holds that there is a gap between transduction and perception which is bridged by some cognitive process (Pylyshyn, 1984), suggesting that inferences mediate the transducer responses to produce "percepts".

The computational approach is reflected in dictionaries, where the term *perception* is defined as "the process by which information acquired by sensory receptors is interpreted to produce some understanding of events that occur outside". This somewhat clinical definition is corroborated by that for the term *sense*: "The faculty by which the mind perceives external information". Further definitions reflecting everyday usage associate *sense* with "meaning", "judgement", and even "intelligence". Also, in the absence of a contextual precept, *perception* is often confused with "intuition" and "knowledge". The lack of a clear semantic distinction between the meanings of *sense* and *perception* is not really surprising given the fact that biological sensors react initially to outside phenomena in a "hard-wired" fashion, producing signals which then activate pre-determined regions of the cortex to eventually emerge as percepts. The confusion would appear to arise from the degree of circumstantial awareness of the process.

In fact, the debate has as much to do with what constitutes a percept as with the mechanism by which it is produced. Sayre (1986), for instance, argues that visual perception is "intentional" and postulates that elements of the visual pathways are constantly adjusting to the environment to first establish and then maintain visual representations. Moreover, the assumption is made that such adaptive perceptual control is embedded in each element, implying that all the necessary information is contained in the intermediate representations and that there is no recourse to inference. This also implies that increasing the range of the adaptive behaviour, or learning new percepts, is an intrinsic process. It is worth noting that "intentionality" parallels the unsupervised learning schemes of "self-organising" neural networks (some aspects of which are reviewed by Kohonen, 1990), and while these are suited to a number of applications, particularly in classification tasks such as early pattern recognition (e.g., Carpenter & Gross-

berg, 1987), there is no evidence that such methods constitute a panacea. Interestingly, the concomitant discussion along the lines of "pre-learning versus self-organising" in neural network research is reminiscent of the "computational versus ecological" debate in psychology.

Because of the diversity of opinions and the associated emotional undercurrents, the question of *machine* perception seems trivial by comparison. To paraphrase Dennett's sometimes parodic review of Sayre's contribution (see Sayre, 1986), working out the details of the implementation is "engineering's baby". However, designing a mechanical device or writing a computer program which carries out a perceptual task, irrespective of the mechanisms and principles involved, does not provide the answer to the philosophical question of whether a machine can perceive *intrinsically.* We, as thinking beings, are tempted to argue that a device may only react to phenomena, not to information, which is in the eye of the beholder. Thus perception, which we do not only relate to physical phenomena, is deemed to be an inherent quality of organised living beings, thereby giving an oxymoronic connotation to the expression "machine perception".

Therefore, from the point of view of the engineer, for whom - apologetically - guidelines are more important than absolute truth, it is useful to proffer a criterion against which the attribution of a perceptual quality to a machine will be assessed. In a nutshell, the following criterion is proposed: a mechanical device perceives if it encodes phenomena into a calibrated measurement which can be validated by an observer. "Calibrated measurement" means that phenomena result in values or symbols which are interpretable by another machine or a human observer, while the process of calibration implies prior knowledge of the device's physical parameters. Note that a system may be designed in such a way that repetition of the same phenomenon does not necessarily produce identical values, as it may be desirable to incorporate a measure of adaptability.

An electrical current meter provides a very simple example of machine perception. The meter senses by transducing current into a certain position of a needle, whose magnetised tip is centred along the axis of a coil which produces a magnetic field when a current passes through it. In the computational sense, the information given by the needle's position is useless without the knowledge of its previous position and of the device's calibration. Hence the percept, in this case a measure of the current in the form of a symbolic value, or a rate of change and so on, results from the contextual interpretation of sensory information. (One could argue that, in the ecological sense, if the device were built in such a way that it were able to provide a direct mapping of the current, its direction *and* rate of change and so on, then a current meter would indeed possess a perceptive quality. The main difficulty, however, lies with the environment-driven nature of the mapping).

Viewed in this way, context, or more specifically prior knowledge, is a fundamental aspect of machine perception. In the current meter example, context is provided through a calibration process which produces a scale upon which measurements are based. Similarly, a visual recognition system may be based on a set of stored features which are used in a matching process, while for motion perception, prior knowledge may consist of the possible interpretations of a sequence of events. More generally, one can view prior knowledge as "postulated data" from which hypotheses are made about the outside world, as expressed by Gregory (1980).

Hence to summarise, "perception" will be used to describe a process which uses prior knowledge to interpret sensory information, and produces some measurement, or, in a wider sense, a symbolic representation. As stated earlier, the computational and "direct realism", or ecological, approaches are not compared on a theoretical basis, but on the grounds of feasibility in a purely practical sense. In other words, building a machine which implements the level of causality implied by Gibson's ecological approach seems to lie beyond present technological capabilities. However, the application of biological principles to the emulation of perceptual functions may yet contribute to closing the gap between both approaches.

3. Hierarchical levels of perception

In many cases, the perception process may be broken down into several sub-processes, each producing intermediate results which then become the "sensory inputs" to the next stage. Since each step represents a different level of abstraction, and thus requires a distinct context, the overall process is non-recursive. It should be noted that this may not be the case for biological processes, which apparently involve reciprocal connections between cortical areas identified with different functions (Coogan & Burkhalter, 1993). Significantly, however, there is evidence that a form of hierarchical decomposition exists in biological systems, as suggested for instance by Van Essen & Maunsell (1983), who have identified separate hierarchies for motion and form perception in monkeys, and Ballard (1986), who uses a connectionist approach to relate neuro-anatomical and physiological knowledge to different levels of abstract computations.

In terms of machine perception, the hierarchy can be illustrated by the following example, derived from classical computational methods. Consider an artificial vision system consisting of a video camera and a number of devices which process the video signal with a view to extract motion information from a succession of images. This process is depicted in Figure 2.3 which shows the correspondence between levels of perception and the results of operations effected on sensory data. For the sake of the argument, it is assumed that the images are smooth and devoid of spurious noise, the purpose of this example being to illustrate a principle rather than to propose a concrete mechanism.

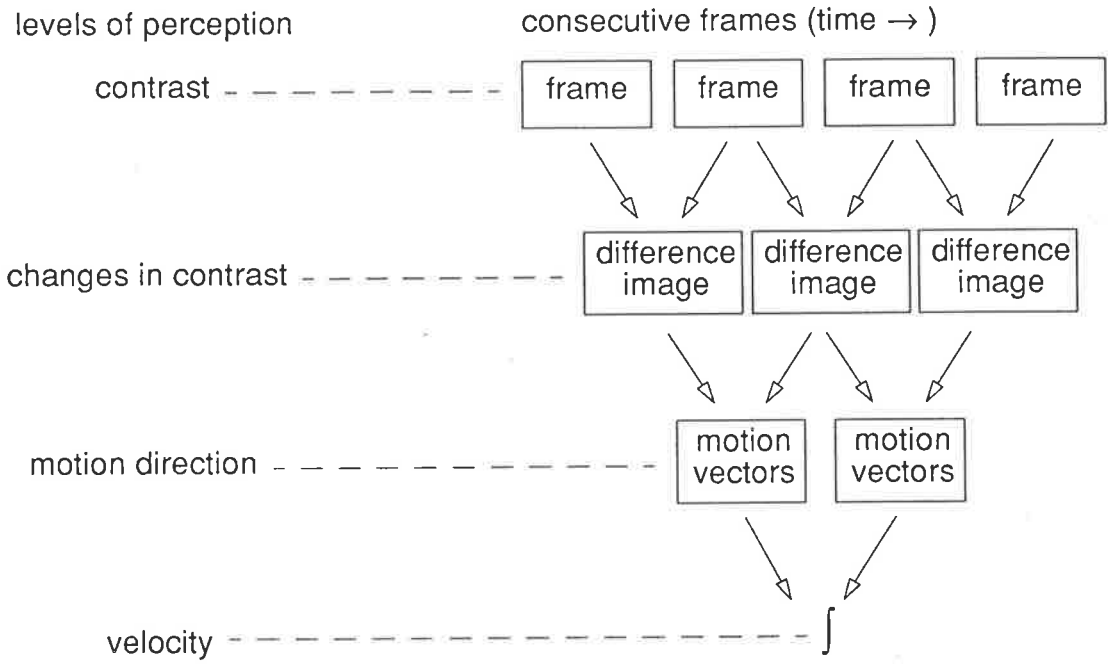


Figure 2.3 Hierarchical structure of motion perception

The camera transduces frequencies, energies and so on, into a binary representation of contrast levels for each pixel. It could be argued, simplistically, that the camera *perceives* contrast, as the value of each pixel is a calibrated measure of contrast. The video signal can then be used to detect changes in contrast. Thus, the next step consists of producing a "difference" image, where the new pixels are the contrast levels obtained by evaluating, pixel by pixel, the difference between the contrasts measured for consecutive frames. This constitutes a perception of changes in contrast (again, there is a measure) and a detection of motion. Consecutive changes in contrast can then be associated to produce "motion vectors", which constitute elementary measures of the direction of motion, hence motion has been perceived. The next level of abstraction, velocity, is obtained by a form of summation, or temporal integration, of the motion information.

In the scheme depicted in Figure 2.3, each step is relatively straightforward in terms of computational complexity and to some extent certain intermediate measurements reflect biological principles. For instance, a characteristic of biological vision is that motion appears to be coherent if the velocity is below a certain threshold (see Chapter 3), and hence the orientations and amplitudes of the motion vectors are more reliably obtained from combinations of changes in contrast which are noted within small areas of adjacent pixels. Other schemes may use a "brute force" approach, such as obtaining the motion vectors by matching previously extracted features in consecutive frames and evaluating their displacements. Whichever methods are used, however, there may occur some unresolvable ambiguities commonly known as the "aperture" and "correspondence" problems, which are examined in Appendix A.

4. Visual processing and perception

Visual sensing is usually accomplished by projecting three-dimensional information onto a planar array of photo-sensitive elements, each providing a measured grey level, or pixel. The task of early visual processing is to establish relationships between these pixels so that higher processing levels may assign "content" to a collection of seemingly unrelated primitives. However, the problem of recovering volumetric information from two-dimensional projections is under-constrained, and requires that a number of assumptions be made in order to constrain the solution space. The challenge is thus to make the "correct" assumptions, without which the problem would be ill-posed and result in ambiguous solutions (Poggio *et al.*, 1985). Also, it is sometimes difficult to make assumptions that are independent of context, as, for instance, restrictions on the possible shapes of objects in an indoor visual scene may not apply to objects in an outdoor scene.

Using traditional visual processing methods (reviewed in Appendix A), the amount of computations required to solve what would be considered as fairly straightforward tasks is in marked contrast with the apparent ease with which natural systems process visual information (Koch, 1989). In fact, many species, from insects to mammals, seem to be particularly adept at making the correct assumptions about the environment under varying circumstances. Hence, biological evidence may support the argument, developed in this section, that incorporating elementary discrimination mechanisms at the sensor level could have a significant impact on the overall processing requirements. The necessary assumptions and constraints would thus be expressed implicitly by the sensor's response to visual stimuli.

Obtaining visual representations is an arduous task ...

Representation is a major concern of computational vision. For instance, under certain circumstances shapes may be represented in terms of intensity, colour or texture distributions, or conversely as contours formed by joined segments or defined by geometrical functions. Such high level descriptions, however, are obtained from basic measurements which are effected on the light impinging on an observer. Hence, to account for the myriad of possible representations, Adelson & Bergen (1991) have proposed that all the potential measurements be expressed in a unified context in terms of a "periodic table" obtained from what they call the *plenoptic function*. The parameters of this function include the coordinates of the intensity distribution and the wavelength at each point, as well as the observable light intensities at every viewing position, and finally the time dimension. In effect, the plenoptic function consists of a time-varying "holographic" representation from which any property such as orientation, motion and colour can be extracted, and in many respects is inspired by Gibson's "ecological" approach to perception.

The analogy previously made with chemistry is compelling and leads one to envisage establishing relationships between "atomic" visual properties in a manner akin to chemical reactions, thus hinting at the existence of properties yet to be discovered or even created. From a computational perspective, every possible property may be extracted in parallel and then, depending on what one is looking for, a subset of those combined. For instance, by associating compatible properties one could create new features which may or may not be biologically plausible, but would be computationally appealing. The scope of such a global approach is enormous, as are, unfortunately, the demands placed on computing resources. By contrast, previous approaches have been more linear in that they focused on constructing symbolic representations from single measurements which could then be discarded. One of those approaches is briefly expanded upon, firstly due to its seminal nature, and secondly because it provides useful insights into vision's practical computational requirements.

Marr's work on early vision processing (Marr, 1976 - see also Marr, 1980, for a review) marked a watershed in the computational analysis of visual information in that it introduced a formal way of describing primitive features present in an image. That description, the "primal sketch", serves as a basis upon which subsequent processes operate. A most important aspect is that the processes by which the primal sketch is computed are content-independent, in the sense that grey levels are directly mapped into features defining edges, lines and blobs, with their positions and orientations. The main objective of the primal sketch, therefore, is to act as the interface at which subsequent processes manipulate symbols, as distinct from raw data.

Marr & Nishihara (1978) then proposed that an intermediate representation between the primal sketch and a full three-dimensional model be constructed, and consequently called it the "$2^1/_2$D sketch", the objective of which is to make explicit the properties of surfaces, such as orientation and discontinuities, without dealing with objects *per se*. Further work, notably by Marr & Ullman (1981) and Harris (1986), showed that motion information may also be obtained from the raw image, and that motion would thus be used to enhance the various levels of representation.

In fact, a number of strategies have been developed, some of which are reviewed in Appendix A along with the appropriate references. Briefly, shapes of objects can be characterised by contours obtained by joining small edge segments or from texture information, which also makes use of intermediate representations such as colour, blobs and edges with their orientations, aspect ratios, and terminating points (Julesz, 1981). Other methods include shape from shading, which uses the relationships between image brightness and object shape, and optical flow, which is the velocity distribution of the motion of brightness patterns. Shape may also be obtained from stereo, which enhances feature discrimination by providing depth cues, and motion, by analysing the discrepancies in a sequence of images.

The underlying characteristic of most classical methods is that computations are applied uniformly over an entire image, and there is generally no initial attempt at discrimination, the tendency being to extract features first, and optimise later. Moreover, several computations are often required for every pixel, followed by the application of sometimes complex algorithms to select those pixels that may be of interest in order to form features. Finally, to provide a comprehensive account of a visual scene, it is likely that features detected separately may have to be combined (Aloimonos, 1988) in a manner akin to fusing data obtained from different sensors. In practice, however, the high computational requirements often render the application of such methods to real-time problems unrealistic.

... and some of the methods employed to alleviate the computational load ...

In the general case where there is no *a priori* knowledge of an image's contents, the requirement that the raw data be processed in its entirety, at least superficially, is unavoidable. However, this initial processing may be conducted with a view to detect where further processing is most likely to be of use. In practice, selecting areas of interest generally consists of a coarse examination of the image in order to discriminate between more or less uniformly textured areas and regions presenting a sharp change in intensity, colour, and so forth. The literature abounds with image segmentation techniques, as this field is by no means confined to light intensity images, and the methods vary substantially depending on the type of images such as those obtained from magnetic resonance or range-finding instruments (see Pal & Pal, 1993, for a review).

In general, there is no consensus as to what one should look for, even within a particular class of images. In natural scenes, it is unclear whether the search can be reduced to distributions of any particular texture features, or "textons", as Julesz (1980) calls them. For instance, Caelli & Julesz (1979) have found that the visual system more readily discriminates the orientation of paired dots (dipoles) than their lengths, while other experiments (e.g. Bergen & Adelson, 1988) suggest otherwise when patterns consisting of segments are present instead. Hence, from a computational perspective, the observation made by Haralick (1979) that texture discrimination techniques are mostly *ad hoc* and are designed for specific types of images still seems to hold.

Arguably, the most promising segmentation methods are those based on pre-attentive vision mechanisms (so-called because the higher cognitive levels are normally unaware of their operation), where the clues provided by early discriminations consist of temporal or spatial changes which are sufficiently large to cause a shift in the focus of attention for further scrutiny. To determine whether a change is significant, one way is to increase the discrimination thresholds. However, since the changes must still be evaluated in the first place, the computational require-

ments remain largely unaffected. A more economical method, therefore, consists of progressively decreasing the spatial and temporal resolutions, and hence if a change is noted, it is likely to be significant enough to warrant full attention.

Unfortunately, detection instruments such as cameras are designed to provide a relatively uniform resolution across the aperture, and aberrations towards the periphery are deemed to be undesirable characteristics because they introduce extra parameters which render processing more complex. It seems, though, that eccentric resolution coarseness is a characteristic exploited by natural systems. For instance, in the human visual system the detectors (i.e., rods and cones) are placed further apart in the periphery of the retina than in the fovea, but the relative sensitivity to motion is more or less even across the eye (McKee & Nakayama, 1984), therefore if motion is detected in the periphery, attentional mechanisms may be alerted.

A logical extension of pre-attentive mechanisms is the "active vision" paradigm, whereby a vision system need not be entirely passive. Cues provided by these mechanisms may be utilised to concentrate the bulk of the processing onto a small area, or "window of attention". In addition to position, control of the window may extend to its size and spatial frequency characteristics, as is believed to be the case in primates (Van Essen *et al.*, 1991), although within anatomical constraints. The overall effect is that only a very small proportion of the information present at the retina filters through to the processing levels. Hence, by implementing a form of dynamic control in an artificial system, including the capability to rapidly shift the focus of attention, it may be possible to achieve a significant reduction in bandwidth requirements while maintaining an acceptable level of performance. Also, if the system possesses some degree of mobility, it may be advantageous to control its behaviour in such a way that additional cues can be obtained. For instance, the three-dimensional structure of a stationary object may be recovered by combining features of the same object as seen from different viewpoints. Significantly, this can be achieved with just a few linear combinations, and hence be relatively cheap in computational terms (Ullman, 1992). At a lower level of perception, egomotion may be used to provide depth cues, while peering may bring about sufficient parallax to permit discrimination between elements present in a scene. In nature, a striking example is provided by insects such as bees, praying mantes and locusts, whose behaviour is closely related to the capabilities of their visual system. When stationary, many insects appear to use the differences in velocities induced by peering motion to distinguish objects from the background, while distance seems to be inferred from self-motion (Horridge, 1986, Sobel, 1990).

Finally, many algorithms used both for motion and stereo require that features present in distinct images at possibly different positions and orientations be matched. For motion and velocity estimation, commonly used methods to solve this "correspondence problem" often draw upon assumptions which reflect biological vision and have the effect of limiting the

number of possible matches. Chief among those are elements of predictability, such as inertia, as objects in motion tend to continue along more or less the same path, object rigidity, and covering and uncovering of background, whereby it is assumed that objects continue to exist, at least for a while, even after occlusion (see Chapter 3).

... lead to a re-appraisal of the problem.

Many computational methods rely on sensing devices which encode visual stimuli into intensity distributions upon which computations are carried out with a view to extract information about structure, motion, and so on. Also, there is a tendency to conserve the integrity of the data at each transformation, from intensity maps to feature points, to structures and then shapes, in an attempt to be able to utilise the same representations in a variety of tasks such as recognition and motion estimation. In essence, the philosophy is that all the "detectable" information be present before proceeding with higher level processing. One justification for such a general purpose approach is that by taking into account all the available information, it is assumed that errors, false interpretations and ambiguities will be minimised, albeit at the expense of high computational requirements, as the main drawback is the time-consuming task of having to sift through large amounts of data in the initial stages.

As has been pointed out earlier on several occasions, many of the proposed techniques to reduce processing requirements reflect physiological findings. The idea of turning to biological systems for inspiration may be taken further, leading to a rather fundamental reconsideration of the problem. Specifically, it is proposed that elementary discrimination mechanisms be built into the detection stage of an artificial vision system.

Generally speaking, biological species from insects to mammals live in a constantly moving world, and hence, as argued by Lee (1980), optic flow should be considered as a fundamental aspect of visual processing. It is therefore somewhat paradoxical that so many methods focus on the analysis of the intensity distributions obtained from *static* images. There may not be a clearly identifiable reason for this to be the case, except possibly for the fact that, in historical terms, motion pictures have only recently become available. By noting that even the fastest supercomputers are regularly humiliated by human performance for what appears to be simple tasks, it might also be suggested that adding the time dimension to the analysis would only increase its complexity. This may be the case insofar as high-level recognition tasks are concerned, but even then the additional clues provided by motion may have a mitigating effect on the amount of computation.

It is fair to assume that the visual systems of the lowest level species would be geared towards survival tasks, thus emphasising natural vision's fundamental functions. However, given nature's tendency to be economical, there is also little doubt that the underlying mecha-

nisms are reproduced elsewhere. The perceptual requirements of species ranging from insects to mammals vary widely, and yet there is increasing evidence suggesting that elementary motion discrimination mechanisms exist in both vertebrate and invertebrate species (Borst & Egelhaaf, 1989). For instance, experimental studies reported by Srinivasan (1992) illustrate how bees carry out depth perception and navigational tasks using optical flow information obtained from elementary movement detectors. Also, it appears that the human visual system has specific sensitivities to different kinds of motion, such as the velocity difference between two moving points, rotational motion, and so on (Regan, 1986). Thus, the differences between the various visual systems seems to be a matter of sophistication, rather than principle. In other words, it could be argued that similar mechanisms are at the source of percepts necessary for survival tasks as well as more "intellectual" tasks, such as recognition.

Hence, the reason for considering early discriminations as a viable alternative to the classical, or "globalistic", approach may be justified as follows: firstly, the detection mechanisms involved are simple, thus minimising computations, and secondly the potential exists to recover higher-level information through association with other discriminatory mechanisms, in the expectation that the necessary data is implicitly contained in the elementary representations. As a result of the application of these principles, it is hoped that an adequate level of perception, at least for simple tasks, may be attained while avoiding some computational pitfalls.

5. Task-oriented sensing

Science and technology tend to feed upon one another, and, as with many symbiotic relationships, the overall effect is generally positive. As a result, challenges to "conventional wisdom" continuously provide researchers of either aspiration with new avenues of thought, stimulated by arguments exemplified by the perception debate mentioned in Section 2. For instance, with the coming of age of Very Large Scale Integration (VLSI), the past decade has seen the emergence of "biologically inspired" devices such as the silicon retina and the cochlea (Mead, 1989), and it is becoming increasingly apparent that this may constitute the most feasible approach towards robust artificial perception systems (see for instance, Mead, 1990, and Gupta, 1991).

The successful implementation of such devices, even as concept demonstrators, is in accordance with the observation that pre-attentive mechanisms, which in human vision appear to be based on texture discrimination, motion, and depth perception through stereo, are essentially bottom-up processes (Julesz, 1991). On the basis of experimental studies, Kröse & Julesz (1989) have speculated that feature extraction occurs automatically and in parallel, thus revealing possible targets, which may then be checked by a possibly slower and serial attentional process (Treisman & Gelade, 1980). What controls this process, however, is anybody's guess,

and there are indeed a number of conflicting theories concerning the interactions between pre-attentive mechanisms and high-level processes, and under which conditions shifts in the focus of attention may occur (Eriksen & Murphy, 1987).

In reviewing the subject, Johnston & Dark (1986) have found that in many cases attention control is somewhat nebulously attributed to intelligent processes, whether explicitly or implicitly. Observed phenomena can indeed be explained in many ways, and be interpreted differently to fit a number of theories, as in forming hypotheses concerning the possible locations of elements in a jigsaw puzzle. Given the complexity of the problem, a glimpse of the answer may be provided by lower-level species. Early studies of the "optomotor response", i.e., the motor reaction elicited in insects by simple visual stimuli, strongly suggest the existence of hard-wired mechanisms linking sensory receptors to motor functions (Reichardt, 1961). In other words, an input stimulus may *directly* cause an output in the form of a motor reaction. Arguably, a hierarchical structure built along similar principles would imply that even sophisticated behaviour could ensue, at least partly, from bottom-up processes. A further implication might be that the control of attention should be distributed across the perceptual systems. Interestingly, Posner & Petersen (1990) have postulated that the human attention system is anatomically distinct from other functions, and spread over several cortical areas. However, attentional control is almost certainly not fully independent from high level mechanisms, and some of the physiological and experimental evidence may indeed be interpreted as a strong indication that conscious attention relies primarily on context-specific knowledge (Tsotsos, 1990). Unfortunately, determining how, and at which stage, top-down and bottom-up processes could mingle, seems to be a fairly intractable problem.

Nevertheless, the principle of distributed control for autonomous systems, where perceptual mechanisms contribute towards behaviour, is currently being promoted as a viable alternative to traditional and centralised approaches to robotics (Brooks, 1991b). In the case of an autonomous vehicle, behaviour may be organised hierarchically, with increasing levels of competence, from reflexes to piloting, or localised path-planning, to high-level goals and objectives (Brooks, 1986). The higher levels may thus provide the general heading direction towards a given location defined by certain landmarks, while the lower levels carry out the mundane task of steering the vehicle while avoiding collision with obstacles. Thus, sensing at the reactive level may be viewed as a pre-attentive mechanism which assists higher-level tasks (Granlund *et al.*, 1994). It is worth noticing that such a scheme is in marked contrast with earlier methods, which essentially consisted of building a model of the environment before planning an obstacle-avoiding path (e.g., Moravec, 1983).

The sensing and timing requirements vary with the level of competence, and are thus geared towards the intended class of behaviour for each level. Hence at the lowest, which is mostly

reactive, obstacles must be detected quickly and effectively, but need not be recognised specifically as objects. Conversely, landmarks may have to be identified in order to determine heading direction, although under less stringent constraints since the time-critical navigation tasks are being carried out at the appropriate control level.

In light of the previous discussion, the close parallels between the sensing requirements of machines and biological species reinforce the notion that the most promising solutions may well be those that are inspired by natural systems. It has been argued that the elementary movement detection mechanism found in insects might constitute a suitable model and a good starting point towards building a perceptual system (Horridge, 1991), and indeed investigations on founding system design on insect visual behaviour have begun (e.g., Franceschini *et al.*, 1992). Considering that the low-level behavioural abilities of the insect and mammalian visual systems present a number of similarities (Osorio & Sobey, 1992), simple systems should still be capable of contributing to bottom-up perceptual processes, which is in keeping with the conclusion of the previous section, where it was suggested that early discriminations do not preclude the recovery of higher-level information.

6. Summary

The control architecture of an autonomous vehicle which exhibits "smart" behaviour may consist of a hierarchical structure, comprising mechanisms whereby sensing information is combined with high level goals and objectives with a view to control motor systems. From a broad perspective, such an architecture can be thought of as forming the outlines of a cognitive system, where behaviour is induced by the interaction of perception and prior knowledge.

Of particular interest to this thesis is the role played by visual perception in motor control. For a machine, perception may be considered as the process by which data is transformed into information that can be understood at higher levels (the "criterion for perception"). Visual perception thus consists of interpreting quantities such as wavelengths and changes in contrast, as a means of recognising phenomena such as colour and motion. This process is far from easy, due mainly to the difficulty of forming global percepts from a sometimes disparate collection of measurements effected over space and time. While, in principle, traditional image acquisition techniques using cameras lend themselves to computational analysis, the costs in terms of processing requirements may be prohibitive, particularly when real-time applications are envisaged. Rather unsurprisingly, many of the methods employed to alleviate the computational load are based on shortcuts and assumptions inspired from biological systems.

Natural systems appear to manage the flow of information by making discriminations at the earliest stages of visual processing. In other words, detected stimuli are almost immediately

interpreted, and hence elementary percepts are obtained regardless of their significance to higher levels of processing. Moreover, stimuli may sometimes directly induce a response on the part of the motor systems. In contrast, computational methodologies essentially consist of extracting as much information as possible from stimuli, prior to establishing an appropriate response.

Several general issues may therefore be raised, such as the possible implications as regards the control of visual attention, and the recovery of high-level percepts, in spite of a large proportion of sensory data being discarded. Central to this thesis, however, is the issue concerning the motor responses elicited by the perception of motion. To that effect, biological motion discrimination mechanisms are investigated next (Chapter 3), followed by a detailed description of a motion detection model inspired from insect vision, which appears to be particularly suited to a VLSI implementation (Chapters 4 and 5). Experimental results are then discussed in relation with motor control (Chapters 6, 7, and 8).

CHAPTER 3 Biological motion perception

In nature, motion plays a critical role which goes beyond the passive interpretation of visual stimuli. The many roles of motion perception are briefly reviewed, with an emphasis on self-motion and looming. This is followed by a description of some relevant characteristics of visual systems across several species, such as the combination of perception processes, the correspondence problem, and the existence of specialised mechanisms. Then, the issue of modelling the most desirable characteristics is addressed, in particular directional discrimination and velocity tuning, in order to provide a basis upon which hardware systems could be constructed.

1. The roles of motion perception

Motion perception is of critical importance for a wide variety of tasks involving quite different behavioural aspects, such as collision avoidance and posture maintenance. There is also increasing physiological evidence that elementary movement detection takes place in the early stages of biological vision, in effect immediately following the detection of visual stimuli and before the information reaches higher cognitive levels. Following is a brief description of the fundamental role played by motion perception, firstly as a provider of cues from which structural information about the environment is obtained, and secondly as a stimulus affecting behaviour (see Nakayama, 1985, for a comprehensive review).

As mentioned in Chapter 2, the three-dimensional visual field is projected onto the sensory system such as the retina or an array of photo-receptors. Movement in the visual field causes time-varying changes in the projected image, and these changes may be interpreted to obtain information which may not be available otherwise, due to circumstances such as colour blind-

ness or monocularity. For instance, directional or velocity differences in the retinal image may be converted into separation of moving objects from the background, thereby possibly providing their shapes, and experiments on the fly visual system have shown that such discriminations may result from a fairly simple interpretation of discontinuities in the motion field (Reichardt & Poggio, 1979). Furthermore, depth information may be inferred from the phenomenon of "motion transparency", when the same region of the retina is stimulated by different velocities. The apparent contradiction of more than one object occupying the same place at the same time may be resolved by assigning the velocities to different depth planes, thus segregating objects from one another (see, for instance, Mulligan, 1993).

While some species benefit from stereopsis to perceive depth, other species that do not have binocular vision are nonetheless able to navigate efficiently in a three-dimensional environment. A monocular cue, motion parallax, provides information about the position of a moving observer relative to objects or surfaces, and is most notably used by some insects (Sobel, 1990). Motion parallax is illustrated in Figure 3.1a, where an observer can infer the distance (R) from itself to an object by using the relative angular velocity ($d\varphi/dt$) induced by its own motion (v). It should be noted that this formulation differs from that proposed by Sobey (1990), which relies on the angle subtended by a single photodetector.

Since relative motion alone does not provide an absolute measure of distance, many species, including humans, make use of the rate of expansion of an object relative to the visual receptor to estimate the time to contact. Mathematically, the time to contact (T) can be expressed as the ratio of the angle subtended at the receptor by an object, to the rate of increase of that angle, provided the angle is small (Figure 3.1b). Experimental studies (e.g., Regan & Beverley, 1978 and more recently Regan & Hamstra, 1993) suggest that channels of the visual pathways are sensitive to this "looming" effect. Considering that neither the receptor's motion nor the distance between the observer and the object are known, the result is quite remarkable, as it can be utilised directly by collision avoidance mechanisms. However, it should be pointed out that it is unclear if the collision avoidance reflex is itself triggered by the time to contact having decreased to a particular value. In fact, experimental studies on locusts show that the insect may alter its flight path when an obstacle subtends more than ten degrees of the field of view, apparently irrespective of the time to contact (Robertson & Johnson, 1993).

There are many other functions for which motion plays an important role such as oculomotor pursuit (or tracking, to use an engineering term), attention control and posture, where motion complements other proprioceptive senses. Moreover, motion appears to be a fundamental component of developmental processes, as it is believed that infants employ motion coherence to perceive objects before learning to make distinctions based on other cues such as colour or texture (Spelke, 1990).

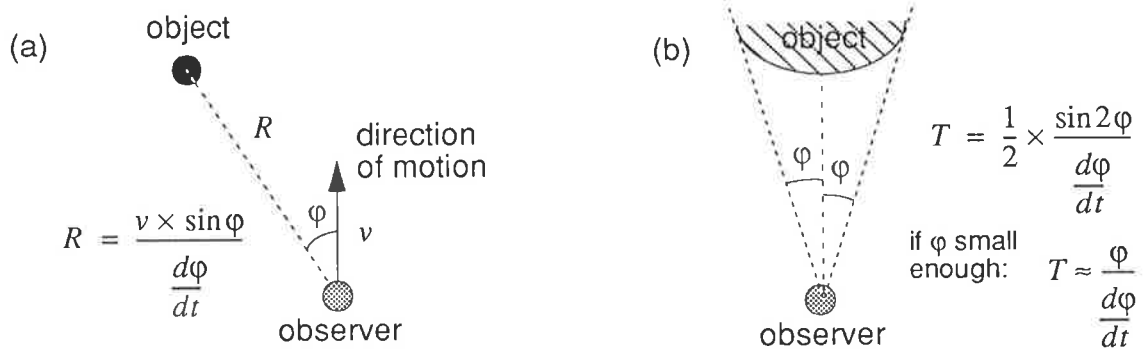


Figure 3.1 (a) Motion parallax, (b) Looming

In the context of this thesis, however, visual stimuli are interpreted primarily with a view to measure relative motion and to detect looming, the objective being to guide an autonomous vehicle around obstacles in a manner similar to which some insects control their flight paths (Egelhaaf *et al.*, 1988). For navigational purposes, only the positions of nearby obstacles relative to the vehicle need to be known. Hence, as will be shown in Chapter 8, the detection of looming elicits an evasive manoeuvre, while relative angular velocities may be utilised to estimate the heading direction by evaluating the distances between the sensor and objects, irrespective of their sizes (Lehrer *et al.*, 1988). But first, motion detection mechanisms are to be investigated in order to determine how such principles can be applied to the design of artificial systems.

2. Characteristics of visual systems

Much insight into understanding the underlying structure of motion perception systems can be gained by first describing some of the perceptual phenomena arising from apparent motion. These phenomena expose a number of rules by which perception processes interpret motion information, which may be explained in terms of neuronal mechanisms.

2.1 Perceptual phenomena and process combinations

Natural visual systems seem to elicit motion percepts under quite varied circumstances. Human observers have no trouble in perceiving apparent motion from a discrete sequence of images, provided that the frame rate is fast enough. Motion is also perceived to be smooth if the interframe displacements are small, and even if this is not the case, the perception of continuous motion may be recovered if large displacements can be integrated over time. Moreover, the perception may be re-enforced if several features are displaced coherently, as visual systems have a tendency to combine localised motion vectors whose directions and amplitudes are similar (Williams & Sekuler, 1984).

The study of apparent motion has led to the observation that motion perception may be mediated by two distinct processes (Anstis, 1980). The first process, often referred to as "short range", appears to operate at a low level to establish point-to-point correlations between images, while the second, "long range", process would more explicitly involve finding correspondences between features which are displaced from one image to another. Note that it is debatable whether motion detection mechanisms operate primarily on shapes or on salient features such as corners (He & Nakayama, 1994).

While it seems reasonable to assume that the long range process at least partly exploits information provided by the short range process (Braddick, 1980), the precise manner in which these processes interact is unclear. If the interpretation of motion information proceeded in a purely hierarchical fashion, one would expect the performance limitations of the lower level to be carried through to the next level. However, psychophysical data suggests that the nature of the interactions is not so straightforward, as motion perception processes seem to combine information from different sources, even from some sources normally associated with other perception processes (Cavanagh *et al.*, 1989).

The perception of motion seems to be influenced not only by the direction and velocity of moving patterns, but also by the stimuli's spatial and temporal characteristics, luminance, contrast, colour, exposure duration, background, and so forth. For instance, in the human visual system, it has been found that colour seems to play a role in resolving ambiguities at equi-luminance (Papathomas *et al.*, 1991). Recent results, notably by Gorea *et al.* (1993), suggest the existence of distinct mechanisms for colour and luminance, whereby information is first collected from similar sources (e.g., all colour) before being combined (see also Ramachandran, 1987). Similar hypotheses have been postulated regarding the existence of separate mechanisms for contrast and luminance variations, even under quite different experimental conditions (see Derrington *et al.*, 1992, and Ledgeway & Smith, 1994).

Remarks on motion colour blindness

The use of loosely related sources of information to complement processes dedicated to motion perception raises a peripheral, but nonetheless interesting issue. The notion that motion perception is a colour-blind process seems to be a question of degree. In humans, for instance, small displacements are more readily perceived in the direction of chromatically-defined borders, while contrast polarity appears to dominate for large displacements (Dobkins & Albright, 1993). For some insects, such as bees and dragonflies, it is believed that self-motion detectors are only sensitive to wavelengths within the green band (Horridge *et al.*,1990; see also Lehrer, 1990). Therefore, it may be argued that colour-blindness enhances motion sensitivity, since the spectral properties of receptors should be closely matched for stimuli rich in colour (Srinivasan, 1985). Intuitively, one could also argue that, by participating as secondary, or support,

processes for motion perception, colour-sensitive mechanisms may partly lose their primary purpose. Even in highly evolved species, a form of partial colour-blindness may thus be explained by the fact that large displacements necessitate a greater effort on the part of the visual system in order for it to compensate for the limitations of the short range processes.

2.2 Maximum displacement (d_{max}) and heuristics of motion perception

In a random dot motion display, dots displaced from frame to frame elicit a perception of motion if the displacements are small enough. Beyond a certain displacement, however, the dots seem to disappear and reappear elsewhere, and hence the sensation of coherent movement is lost. In other words, there exists a limit beyond which the visual system can no longer solve the correspondence problem. The limit, known as d_{max}, has first been measured by Braddick (1974), who then postulated the existence of the short range process in apparent motion. Note that in order to characterise the short-range process, the use of random dots is particularly important since it eliminates the possibility of features such as edges being recognised independently by other processes, before assisting the long range motion perception process.

The value of d_{max} depends on the characteristics of visual systems as well as on the nature of the stimuli. For instance, by looking straight ahead at a display where displacements are greater than d_{max}, an observer can recover the sensation of coherent movement by squinting, which, in effect, blurs the image and reduces the spatial sensitivity. It therefore appears that d_{max} is a function of physical characteristics such as the distance between receptors, and would thus increase with eccentricity (Baker & Braddick, 1985). Temporal integration also seems to play a significant role, as shown by Williams *et al.* (1986) who found that a percept, once established, exhibited resistance to change, suggesting a form of cooperation between motion detection units. Hence, it appears that d_{max} increases with stimulus duration, albeit asymptotically (Snowden & Braddick, 1990). Another aspect concerns the number of motion detection units which may be simultaneously activated. For instance, Van Doorn & Koenderink (1984) have shown that the time required for a percept to become established is inversely proportional to the area stimulated by similar displacements, thus suggesting a form of spatial as well as temporal integration (see also Nakayama & Silverman, 1984). The existence of d_{max} and of integration mechanisms are particularly relevant in the context of tracking moving stimuli, as will be shown in Chapter 7.

Visual stimuli can be quite complex, and hence the visual system is constantly challenged to solve the correspondence problem. In other words, an upper displacement limit is essential, but insufficient, as in many cases several candidate features, or dots, may be matched wrongly without d_{max} being violated. Moreover, even when d_{max} has been exceeded, some visual systems may still recover motion percepts by making certain assumptions, such as inertia and

rigidity, which reflect the physical properties of objects in motion. These assumptions are quite sensible when one considers that, at least within a short time span, changes in direction are generally small, and objects are unlikely to undergo dramatic deformations. However, the fact that some evolved species seem to apply such heuristics indicates that their visual systems may in some cases examine more than two "frames" before making a matching decision (Anstis & Ramachandran, 1987). Also, since small deformations may not be apparent at a coarse scale, it has been suggested that the matching process may be dominated by low spatial frequencies, rather than by the similarities between salient features (Ramachandran *et al.*, 1983). Finally, as a corollary of inertia and rigidity, visual systems also seem to assume that objects continue to exist after being occluded, at least for a short while (Anstis & Ramachandran, 1985).

2.3 Neuronal responses to motion

Directional and speed information are the essential prerequisites for motion perception processes, and the question now is how such information may be extracted and encoded by the visual system. Electrophysiological experiments have revealed the existence in the visual cortex of cells whose reactions differ according to certain aspects of the stimuli. The cells which appear to be the most significant for the perception of motion can be broadly classified into three categories: directionally-selective, velocity-tuned, and motion segregation (or anti-phase) cells (see, for instance, Sekuler *et al.*, 1990).

Directional selectivity

Not surprisingly, in light of the afore-mentioned neuronal cooperativity, the differences between motion-specific cells is not clear-cut. For instance, the responses of directionally-selective cells appear to be partly a function of stimulus velocity, thus, in principle, the distinction between directional and velocity specificity would not be required. Also, the responses may decrease as a result of habituation (Giaschi *et al.*, 1993). However, the responses of a number of cells to stimuli such as stroboscopic light emphasise the distinctive nature of directionally selective mechanisms (Pasternak, 1987, Cremieux *et al.*, 1987). Moreover, measurements of activity in the cat visual cortex have revealed the existence of cells which respond strongly to directional motion along an axis passing through the centre of a given cell's receptive field, and respond feebly, or not at all, to motion in the opposite direction, while other cells do not seem to exhibit such directional selectivity (Orban *et al.*, 1981b).

Directionally selective cells can be characterised by a "direction index", which is the ratio of responses to motion in the preferred direction, to responses to motion in the opposite direction (Warren *et al.*, 1986). However, the index cannot be considered to be a fixed parameter as it may depend on the characteristics of the stimulus, such as duration. For instance, directional selectivity appears to decrease with adaptation following prolonged stimulation (Marlin *et al.*,

1988). By analysing responses to stimulus onset and offset, Ganz (1984) concluded that local asymmetrical inhibitory mechanisms as well as integrative mechanisms might be involved.

The responses of directionally selective cells are also a function of the spatial characteristics of their receptive fields. By suddenly displacing a stimulus which consists of vertical bars of alternating contrast, or sinusoidal gratings, and whose spatial frequency is matched to a given cell's characteristics, it appears that the strongest responses occur for displacements that are slightly smaller than a quarter of the spatial period (Baker *et al.*, 1991). Hence, the cells exhibit an "optimal displacement" characteristic, which decreases for lower contrast stimuli (Boulton & Hess, 1990), and which also seems to be linked to a range of "preferred" velocities (Baker, 1988). These results may be related to the investigation of the "aliasing" phenomenon, which is addressed in Section 4.

Velocity tuning

Velocity-tuned cells belong to a broader class of cells whose velocity sensitivities can also be low-pass, high-pass, or broadband (Orban *et al.*, 1981a). Psychophysical studies, notably by McKee (1981), have shown that human observers can discriminate between velocities which differ by only a few percent. The fact that stimulus duration and the distance travelled by the stimulus do not seem to have much effect on these discriminations, indicates that mechanisms may be implemented locally.

In psychophysical experiments, velocity tuning can be characterised by a "differential threshold", expressed as the Weber fraction of reference velocities, i.e., the ratio of the difference between discriminable velocities, to each reference velocity (Orban *et al.*, 1984). Results show that the threshold increases for low and high velocities, and hence velocity tuning is, in general, sharper for moderate velocities, although cells towards the eye periphery tend to be tuned to higher velocities (Orban *et al.*, 1986). Significantly, cells which appear to be tuned to low velocities, but high spatial frequencies, do not seem to be directionally-selective (Thompson, 1984, and see also Reichardt & Poggio, 1979). Also, tuning to very low or very high velocities seems to be virtually absent at the early stages of visual processing in both cats and monkeys, while velocity selectivity appears to be more prominent at later stages (Maunsell & Van Essen, 1983).

Motion segregation

The responses of a number of motion sensitive cells appear to be influenced by background motion, and in particular their responses may be suppressed if a textured background is moving at the same speed or faster than a stimulus in the foreground (Gulyas *et al.*, 1987). However, the responses of some cells is strongest when stimulus and background are moving in opposite directions, even though these cells are not normally directionally selective (Orban *et*

al., 1987). Instead, these cells are relatively, as opposed to absolutely, sensitive to direction, and appear to play a significant role in estimating depth from motion (Orban & Gulyas, 1988). The existence of mechanisms which are sensitive to relative differences in motion is particularly relevant to the methods presented in Chapter 8.

3. Neuronal modelling

Physiological evidence may be exploited in order to derive models which not only reflect biological reality, but also offer the hope that building efficient hardware systems becomes feasible. Recorded behaviour is therefore analysed with a view to infer the functionality of neuronal components. Consequently, the work of Hubel & Wiesel (1962) was of seminal importance, as it confirmed the predominance of so-called "simple" cells over "complex" cells at the early stages of visual processing. The main difference between these two types of cells is that, for simple cells, responses to stimuli can be predicted by arrangements of excitatory and inhibitory regions. The behaviour of complex cells cannot be predicted in the same way, as the responses do not seem to vary as a function of the stimuli's position within the cells' receptive fields, even though the orientation of the stimulus appears to be critical for both simple and complex cells. Most models thus seem to follow the "simple" cell paradigm, whereby elementary motion detection mechanisms either excite or inhibit large-field units.

Experimental studies by Barlow & Levick (1965) have shown that in the rabbit's visual system, the directional response to successive stimulation of separate receptive fields appears to decline when the separation between the fields increases, even at constant velocity. Consequently, directional selectivity may result from "sequence discrimination", whereby the response of a small receptive field is carried laterally and combined with the response of the neighbouring field to produce a local direction-specific signal. Such signals are then collected by a large cell (ganglion cell in the rabbit retina) whose response to motion is directionally selective, but not position specific. The sequence discrimination scheme also appears to be present in the fly visual system (Riehle & Franceschini, 1984), where "elementary motion detectors" (EMD) excite or inhibit large horizontal "directionally-selective motion detection" (DSMD) neurons present in the optic lobe (Bishop *et al.*, 1968; see also Hausen, 1982a).

An EMD may thus be modelled as two sub-units tuned to opposite directions, and the synaptic connection of one sub-unit to the DSMD neuron is inhibitory, while the other is excitatory. The underlying mechanism by which the output of an individual sub-unit of the EMD is obtained can itself be excitatory or inhibitory, although recent experimental evidence seems to favour the existence of an inhibitory mechanism (Schmid & Bülthoff, 1988). Both mechanisms are depicted in Figure 3.2, where the response from one receptor R is delayed by a time τ with respect to the response from the adjacent receptor, and the operation at the sub-unit level is

represented by an adder. Note that from a functional point of view, logical operators could be used instead, namely, AND for the excitatory case, AND-NOT for the inhibitory case. Alternatively, a multiplication scheme could be employed, such as that suggested by Torre & Poggio (1978). In fact, multiplication appears to match experimental measurements more closely than an additive scheme (Egelhaaf *et al.*, 1989).

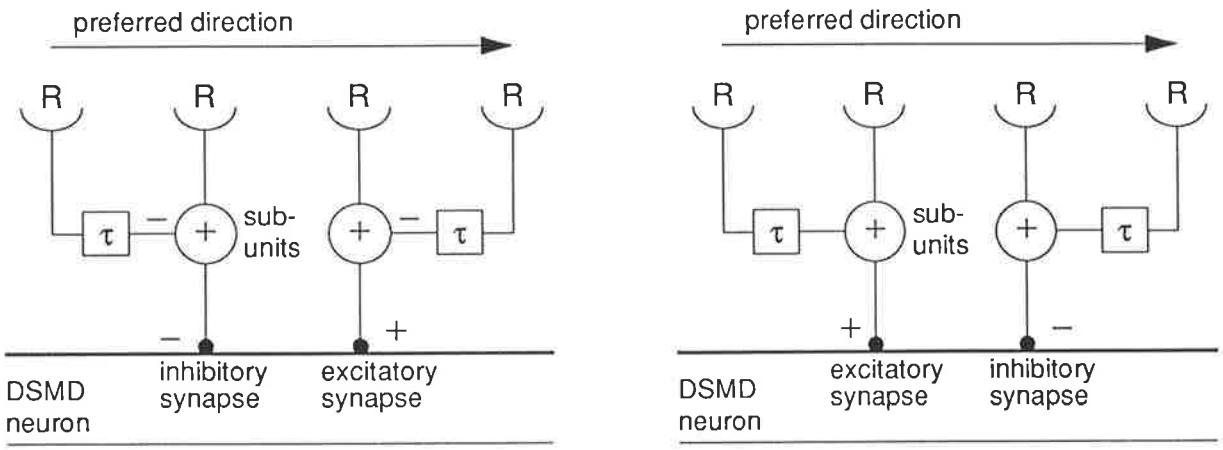


Figure 3.2 Motion detection mechanisms

Inhibition may be viewed as a "veto" mechanism, whereby the response to motion (i.e., a change in luminance) of one receptor disables the response from the adjacent receptor in the non-preferred (null) direction. For motion in the preferred direction, the inhibitory response arrives too late to cancel the output of the sub-unit corresponding to the excitatory synapse (Figure 3.2a). By contrast, in the excitatory mechanism, motion in the preferred direction results in the conjunction of the responses from the adjacent receptors, thus stimulating the sub-unit above its activation threshold (Figure 3.2b). For either mechanism, if motion is in the preferred direction and matches the delay and the separation between the receptors, only the excitatory synaptic input to the DSMD neuron is activated, while motion in the null direction causes the activation of the inhibitory synapse.

While directional selectivity is conserved in both cases, the pattern of responses of the DSMD neuron may differ depending on the type of motion-induced stimuli. In an attempt to resolve the issue of whether electrophysiological data may be accounted for by an inhibitory or an excitatory mechanism, Bouzerdoum (1993) has recently proposed a neural network model of the DSMD neuron which employs lateral inhibition. In the model, shown in Figure 3.3, the τ boxes are pure delays, while, roughly speaking, the L boxes (anatomically corresponding to the *lamina*) are band-pass filters and the M (*medulla*) are multiplicative units. Even though this model is an extremely simplified version of the fly's motion detection system (see Shaw, 1984, for a review of neuro-physiological evidence), the results obtained appear to support the

hypothesis that an inhibitory mechanism is indeed responsible for the observed activity patterns of the DSMD neuron.

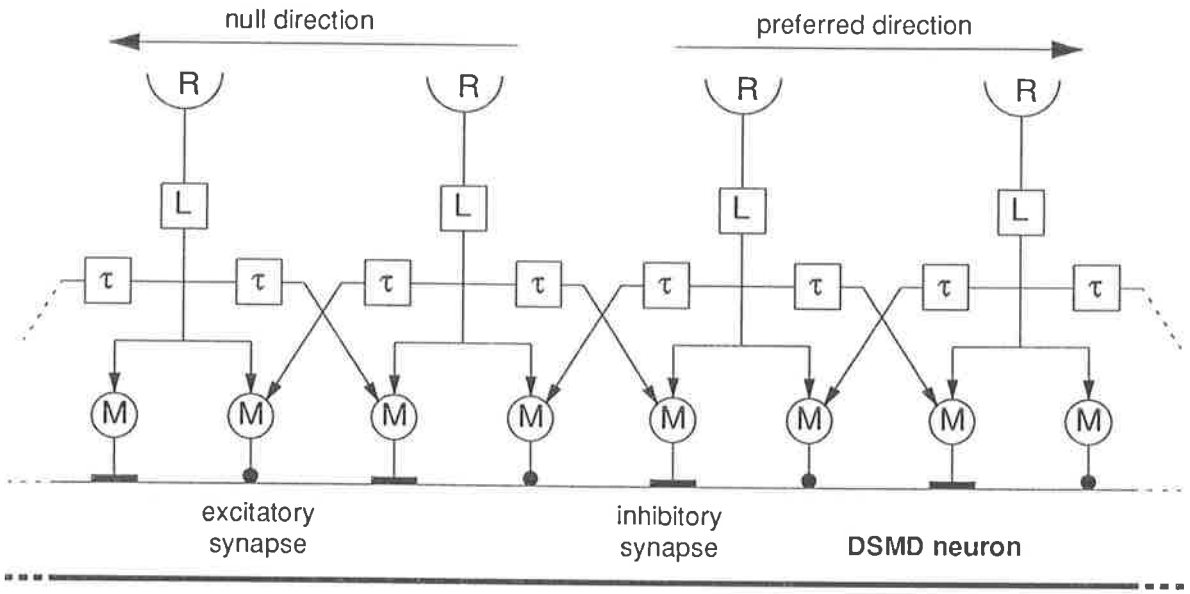


Figure 3.3 Directionally selective motion detection neuron model (Bouzerdoum)

As pointed out in the previous section, the behavioural characteristics of directionally selective cells can be fairly complex, particularly in mammals, and appear to be more dependent on the nature of the stimuli than suggested by the models shown here. In particular, Tolhurst *et al.* (1980) have shown that some cells in the cat visual cortex initially respond with a burst of activity at the onset of a stimulus before settling down. Heeger (1993) suggested that such behaviour could be modelled mathematically by a linear summation followed by non-linear stages, namely, half-wave rectification followed by squaring, and "normalisation", or division. Whether or not the visual cortex performs such operations is a moot point, however, the principle of "mathematical fitting" permits the analysis of experimental data in different domains, as exemplified by some of the models described in the next section.

4. Mathematical motion detection models

Modelling biological systems can be approached with different objectives in mind. In the previous section, the principal motivation was to understand the mechanisms by which visual stimuli are interpreted, and hence the approach mainly consisted of reproducing observed phenomena. A model derived in this way may therefore incorporate mechanisms which are known to exist or have been inferred from experimental studies, and ideally, the model should match the natural system in both structure and performance, including its limitations. The goal of engineers, on the other hand, is to construct a system targeted towards a particular application and within technological constraints, and hence the model may, or may not, end up to being

similar to a natural system. Arguably, the models presented here fall mainly into the latter category, in the sense that, while their underlying structure may be biologically plausible, the internal mechanisms are not necessarily so.

4.1 Directional selectivity

The detection of motion requires that the responses of several receptors be combined. Directional information, in a sense the lowest level of perception, may then be obtained if the interaction between the receptor responses is non-linear and asymmetrical (Borst & Egelhaaf, 1989). In the general case, if high-order non-linearities can be ignored, an n-input system is equivalent to the linear summation of 2-input systems (Poggio & Reichardt, 1973), and hence most motion detection models are based on elementary 2-input units.

The simplest motion detection mechanism was proposed by Reichardt (1961). By studying an insect's reaction to visual stimuli, Reichardt found that the interactions between adjacent (or at most once removed) visual channels could be adequately described by a first-order correlation of the linearly-transformed stimuli. Consequently, Reichardt proposed to model the "opto-motor" response mathematically, by constructing a motion detection stage whose outputs are subtracted to produce a response which is a function of the direction of motion.

The motion detection stage operates along the following principles. Given two detectors, A and B, suppose that the output of detector B is delayed by a time τ with respect to the output of detector A, and that the latter and the delayed signal from B are multiplied (Figure 3.4a). If a light pattern is moving in front of the detectors and in the direction B to A, the multiplier responds strongly if the delayed signal from B coincides with the signal from A, i.e., if the velocity is appropriate. The multiplier responds poorly, if at all, to motion in the opposite direction. An elementary motion detector may thus be built by combining opponent stages, as shown in Figure 3.4b.

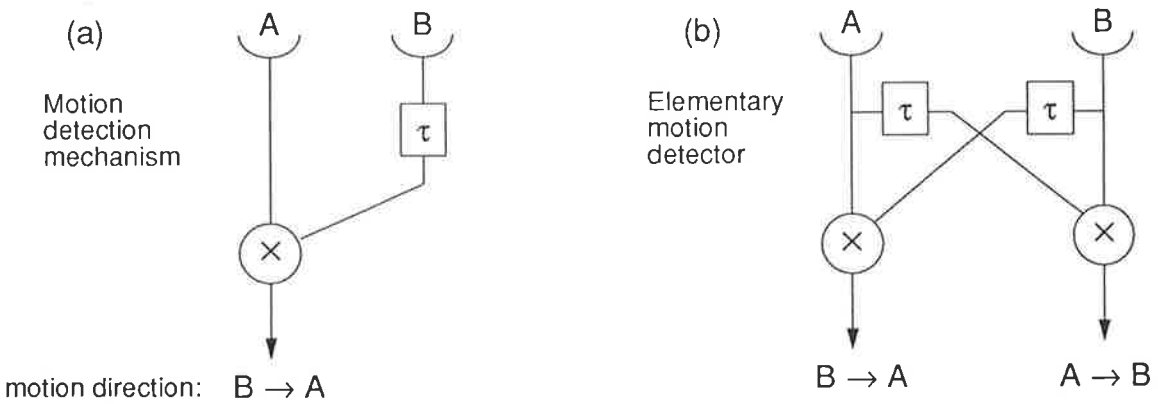


Figure 3.4 Motion detection

The Reichardt model is obtained by low-pass filtering the outputs of the multipliers, which amounts to infinite time-averaging, and finally subtracted, yielding the directional response signal whose sign indicates the direction of motion (Figure 3.5). Here, the precise nature of the boxes D, F and H is not specified, other than they are assumed to be linear filters with different time constants.

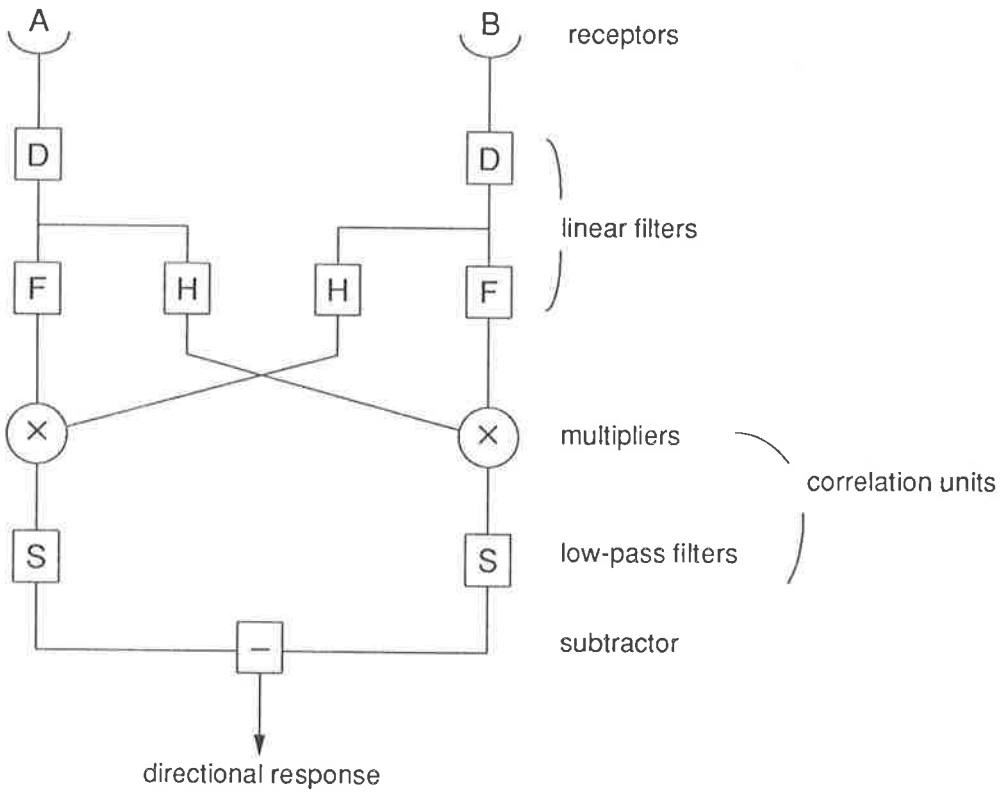
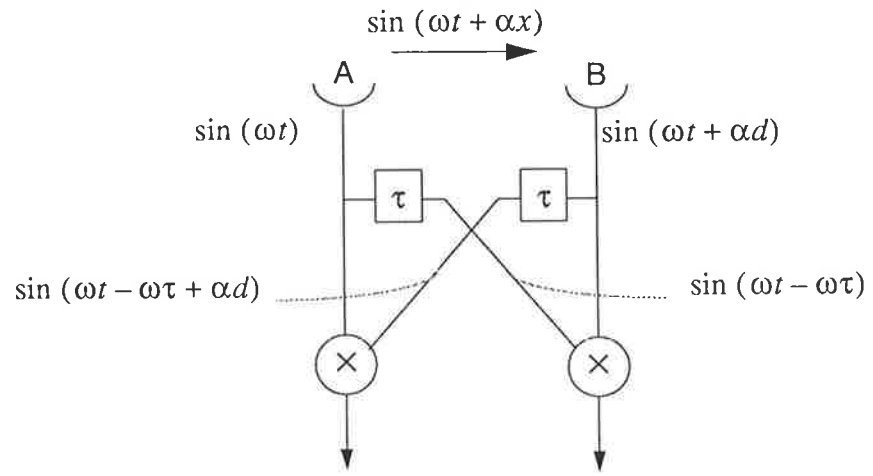


Figure 3.5 Opto-motor response (Reichardt model)

Aliasing

The elementary motion detection mechanism, as it stands in Figure 3.5, may produce the wrong output under certain circumstances. Suppose that the stimulus consists of a pattern of alternating light and dark strips, moving from A to B. In the crudest approximation, the signal produced by receptor B can be viewed as a phase-shifted version of the signal produced by receptor A, and, likewise, the delayed signal from A is also a shifted version of the stimulus. If both phase shifts match, then the directional response will be at its strongest, and correctly indicate motion from A to B. If, however, the difference between the phase shifts is greater than half the period of the pattern, the response may indicate motion in the opposite direction. In standard signal processing parlance, this phenomenon, known as aliasing, occurs when the sampling rate is less than the "Nyquist frequency" (i.e., twice the maximum frequency of the stimulus), and is in accordance with data obtained from psychophysical experiments on the human visual system (Coletta *et al.*, 1990).

To illustrate the phenomenon of aliasing, consider the simple case where the stimulus consists of a sine-wave grating of temporal frequency ω and spatial frequency α. Background illumination, represented by an offset, may be ignored if it remains constant as the term will be cancelled by the subtractor. Assuming a normalised scale, the stimulus at position x can be written as: $\sin(\omega t + \alpha x)$. Taking receptor A as the origin, d as the distance between the receptors, and supposing that the grating is moving in the A to B direction, the outputs of the multipliers m_A and m_B can be determined as follows:



$$m_A = \sin(\omega t) \times \sin(\omega t - \omega\tau + \alpha d) \qquad m_B = \sin(\omega t - \omega\tau) \times \sin(\omega t + \alpha d)$$

$$m_A = \frac{1}{2}\cos(-\omega\tau + \alpha d) - \frac{1}{2}\cos(2\omega t - \omega\tau + \alpha d)$$

$$m_B = \frac{1}{2}\cos(-\omega\tau - \alpha d) - \frac{1}{2}\cos(2\omega t - \omega\tau + \alpha d)$$

The detector response $R(\alpha, \omega)$ is obtained by subtracting m_A from m_B, which yields:

$$R(\alpha, \omega) = \frac{1}{2}\cos(\omega\tau - \alpha d) - \frac{1}{2}\cos(\omega\tau + \alpha d) = \sin(\omega\tau) \times \sin(\alpha d)$$

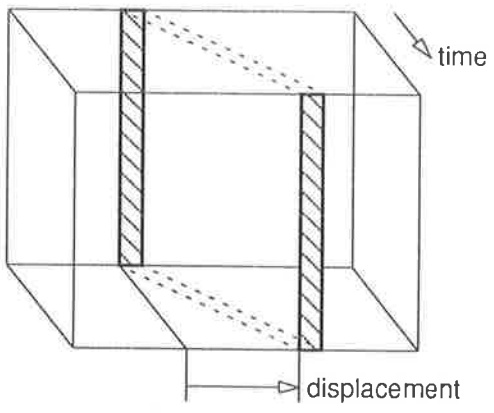
$$R(\alpha, \omega) \text{ is maximum when: } \omega\tau = \alpha d = \frac{\pi}{2}$$

When the response is maximal it follows that $d/\tau = \omega/\alpha$, and hence the detector is tuned to a particular velocity $v = d/\tau$. However, it is readily apparent that the spatial and temporal frequencies α and ω may be varied in such a way that the response reverses, without the grating changing direction. Assuming the grating is constant, the phenomenon of "spatial aliasing" may then be defined as the case where, for a fixed time delay, the separation between the receptors is either too large or too small and causes the apparent direction reversal. In like fashion, "temporal aliasing" occurs when the time delay of the motion detection unit is either too large or too small with respect to the separation between the receptors. The same reasoning may evidently be used when the parameters of the grating vary while the characteristics of the detectors are fixed.

Correlation-based methods

From the above, it is clear that aliasing may be prevented if the spatial and temporal characteristics of the motion detector are kept within certain limits relative to one another. Consequently, Van Santen & Sperling (1984) proposed that the time delay be implemented with a low-pass filter whose phase delay is between 0 and $\pi/2$, and that spatial filters be used at the inputs, thereby extending the receptive field of the receptors. If the receptive fields overlap, spatial aliasing may thus be avoided. Van Santen & Sperling (1985) then completed the model, called the "Elaborated Reichardt Detector" (ERD), by replacing the infinite time-averaging filters (S boxes in Figure 3.5) with linear temporal filters whose characteristics are arbitrary. The rationale is that temporal integration only takes place over an appropriate time interval, for example from the onset of the stimulus until the response becomes negligible, or over an integer number of periods for periodic stimuli.

A different approach was used by Adelson & Bergen (1985), who suggested that time be treated as a spatial dimension, and hence detecting motion amounts to finding orientation (see Figure 3.6). The outputs of filters which are tuned to the same direction of motion but which are 90 degrees out of phase (i.e., in quadrature) are combined and squared, producing a signal corresponding to "motion energy". It has been argued that such a scheme is fairly consistent with measured activity in the visual cortex (Emerson *et al.*, 1992). A characteristic of this model is that the response depends on the direction of motion, but is not affected by the polarity of the stimulus. In other words, a moving black object on a white background elicits the same response as a white object moving against a black background. However, the absolute value of the response is a function of the amount of change in contrast, and hence the responses of several motion selective units tuned to the same spatial frequency must be employed to compute velocity, as only the ratios of the units are contrast-independent. Clearly, though, the velocity estimate would be unreliable at high velocities or low contrasts.



the velocity of the moving vertical bar (hashed boxes) is given by the ratio of displacement to time, which is the slope of the dashed line.

Figure 3.6 Velocity in the spatio-temporal domain

Under certain circumstances, the ERD and the energy model of Adelson and Bergen can be shown to be equivalent, in that they produce the same response to within a scaling factor. The difference lies mainly in the order in which computations are effected, implying that the overall similarity may no longer hold if some of the filters are non-linear (e.g., if thresholding filters are used). Simplified versions of both models are shown in Figure 3.7. The equivalence is shown for the case where the temporal averaging filters TA are absent (or simply ignored), and the temporal filters TF_1 and TF_2 are assumed to be in quadrature in both cases.

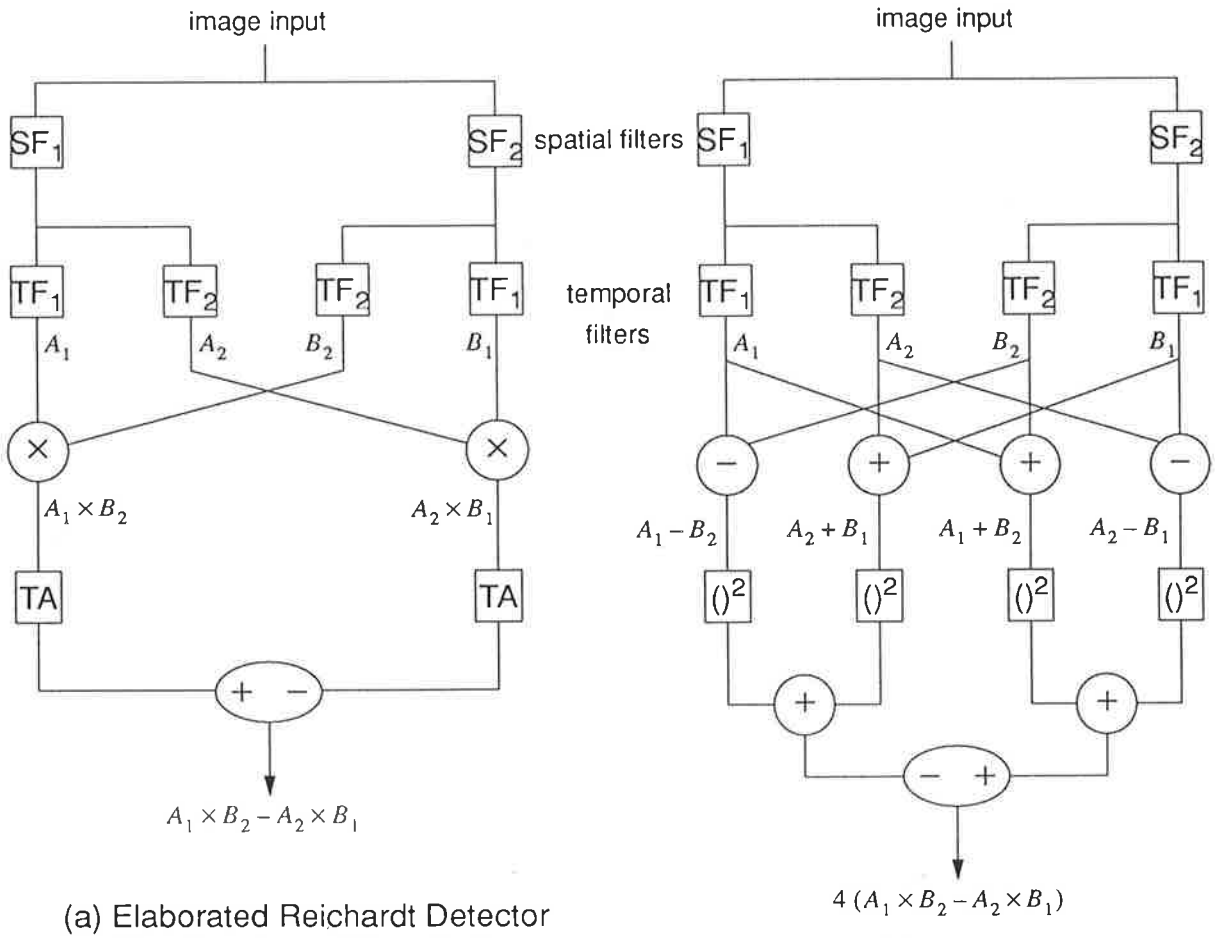


Figure 3.7 Correlation-based models

Temporal frequency modulation

The outputs of both motion energy- and Reichardt-detector-type models can be considered as a form of correlation between the stimulus and the model characteristics. An alternative approach, sometimes referred to as "temporal frequency modulation", has been proposed by Watson & Ahumada (1985), who have utilised psychophysical evidence which suggests the existence of spatial-frequency-tuned elements. Their model comprises two stages, the first consisting of a number of linear sensors tuned to different directions and spatial frequencies

(Figure 3.8a), while the second stage combines the information provided by the individual sensors to produce an estimate of direction and velocity (Figure 3.8b). In the first stage, the response of an individual linear filter to a sinusoidal grating moving in the direction for which the filter is tuned (the "preferred" direction) is also a sine wave, whose frequency is equal to that of the stimulus. If the grating is moving in the opposite, or non-preferred, direction, the response is nil. This behaviour is caused by the Hilbert filters in the quadrature path, which, in effect, invert signals at any frequencies whose phase lags correspond to the non-preferred direction, thus cancelling the signal from the main path.

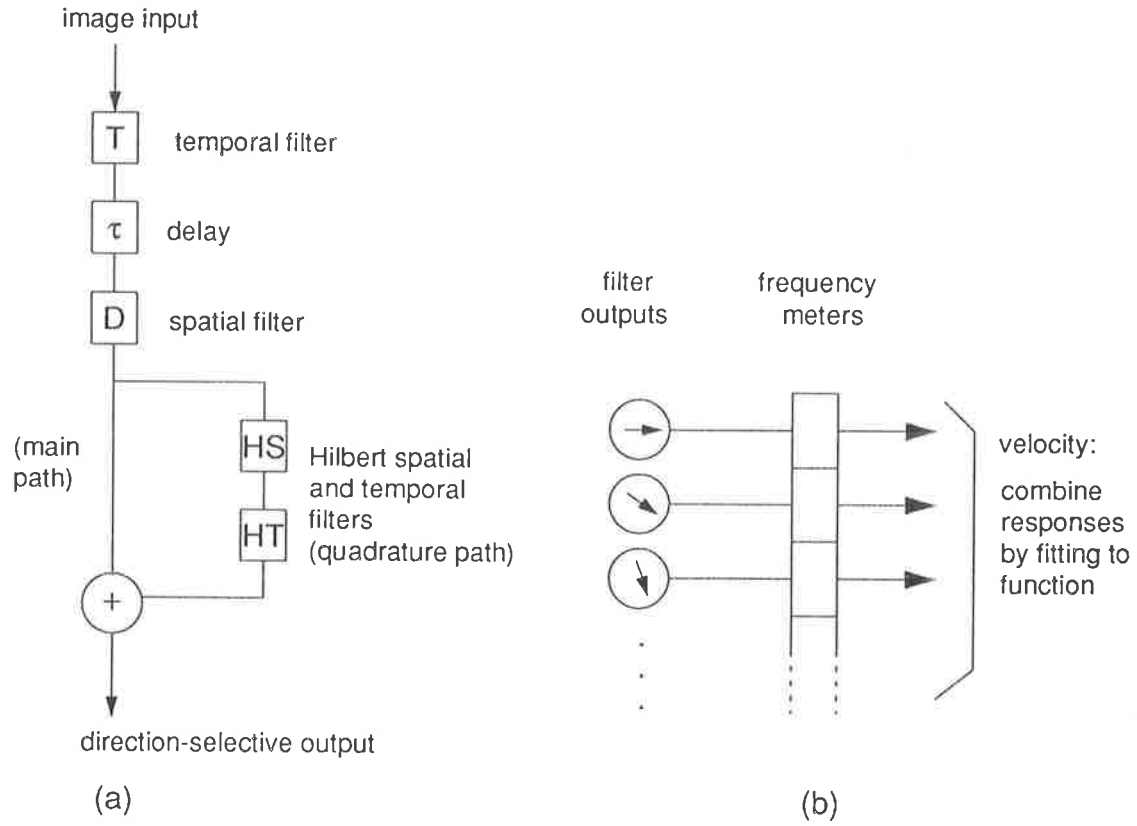


Figure 3.8 (a) Watson and Ahumada sensor, (b) Sensor grouping

For two-dimensional motion, the response of a single sensor corresponds to the velocity component in the preferred direction, and hence, in the ideal case, motion direction and velocity could be recovered from two sensors. In general, though, the sensor is replicated in space, spatial frequency, and orientation. For each location, filters at ten different orientations (i.e., every 36 degrees) and eight spatial frequencies are used in order to adequately cover the visible frequency range and all possible directions. However, the receptive fields of filters tuned to an identical spatial frequency should overlap to avoid spatial aliasing, resulting in there being many more sensors at high than at low spatial frequencies, with the ratio of the highest to the lowest being several orders of magnitude.

4.2 Velocity estimation

The response of motion detection mechanisms is an indication of how close (or how far) the real velocity is to the central velocity of the detector, and therefore is not an absolute measure. Thus, in principle, motion either slower or faster than the central velocity may elicit an identical response. In theory, if motion detector units oriented differently and tuned to all possible spatial frequencies and velocities were available, the velocity would be provided by the unit whose response is strongest. Unless the ranges of expected motions and spatial frequencies are severely constrained, though, the practicality of such a scheme, called "labelling", is dubious. At the other end of the scale, velocity could be encoded by the intensity of a single signal, which would be very economical. However, such a scheme fails when a small receptive field is stimulated by distinct velocities. Hence, since elementary motion detectors are fundamentally geared towards a labelling scheme, some form of compromise is necessary, and indeed physiologically likely, in order to transmit efficiently motion information to higher perceptual levels. In general, it appears to be easier to begin by combining the responses of sensors tuned to the same velocity, but to different spatial frequencies, in order to produce a pattern invariant signal (Figure 3.9).

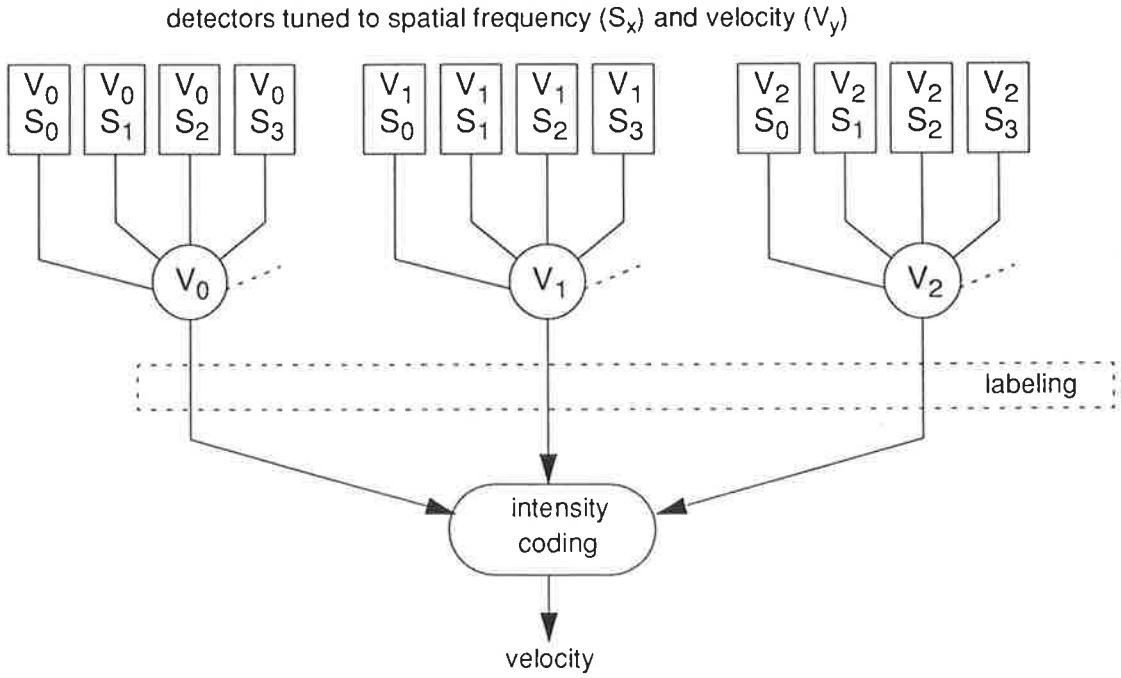


Figure 3.9 Velocity encoding schemes

As shown previously in Figure 3.8b, Watson & Ahumada (1985) combine the sensors at a particular location by measuring their responses to different temporal frequencies. The responses are then fitted to a function of the form $rf_s\cos(\phi - \phi_s)$, where r and ϕ are the velocity and direction, respectively, while f_s and ϕ_s are the spatial and directional tuning characteristics

of the group of sensors. Note that only half of the cosine can be estimated, since only half of the sensors should have responded to the motion. The fitting function in effect encodes the velocity at the particular location in the form of an intensity.

The fitting function is, in fact, the polar form of a function defining the velocity parameters, and is derived by employing some interesting features of image motion in the frequency domain. Consider the case of a pattern such as a sinusoidal grating characterised by a single spatial frequency ω_s. When in continuous, non-accelerated motion, the velocity v of the pattern can be expressed as the ratio of its temporal frequency ω_t to its spatial frequency ω_s, and hence: $v = \omega_t/\omega_s$. A more complex pattern comprising an arbitrary number of spatial frequencies all moving at the same velocity thus defines a straight line in spatio-temporal-frequency space, and passing through the origin. By extension, a translating two-dimensional pattern defines a plane in spatio-temporal-frequency space, which can be written as: $\omega_t = u\omega_x + v\omega_y$, where u and v are the velocity components. The slope of the plane therefore characterises the velocity and the direction of the moving stimulus, and the aperture problem arises because the knowledge of ω_t does not uniquely determine u and v.

As measuring the velocity amounts to finding orientation in time-space, and can be accomplished by using oriented spatio-temporal energy filters (Adelson & Bergen, 1985), the analysis may be carried out in the frequency domain. Heeger (1987) employs sets of quadrature pairs of three-dimensional Gabor filters which are combined to compute the phase-invariant motion energy. The sets are tuned to different orientations and temporal frequencies. The motion energies are convolved with a three-dimensional Gaussian window, and the resulting distribution is then matched with the expected distribution of a stimulus whose power spectrum is flat. The matching is accomplished by minimising the difference between predicted and measured motion energies, yielding the velocity parameters u and v.

Heeger's method works well as long as moving patterns present a flat power spectrum. Also, consider the use of the Gaussian convolution, which approximates the integral of the energy filter responses over the frequency domain. Since the integral can be related to the space-time domain by Parseval's theorem, the convolution produces an estimate of the average velocity within the Gaussian window, which presents difficulties at motion boundaries. Consequently, Grzywacz & Yuille (1990) employ similar filters, tuned to oriented spatio-temporal frequencies, but propose to compute motion locally. Essentially, their method consists of finding the characteristics of hypothetical filters which best fit the available data. Since the fitting process, which is accomplished in the frequency domain, only combines data along the same temporal frequency axis, there are (theoretically) no restrictions on the spatial characteristics of the stimulus.

Snippe & Koenderink (1994) have shown that similar theoretical pattern invariance can be achieved by adding the outputs of Reichardt detectors each tuned to the same velocity, but with different spatial and temporal characteristics. In other words, while the delay τ and the distance d between the detectors vary, the velocity-tuning represented by the ratio $v = d/\tau$ is conserved. Figure 3.10 depicts possible arrangements of detectors and delay units in order to build a multi-input Reichardt detector. The simplest configuration (Figure 3.10b) can easily be extended to incorporate a greater number of receptors.

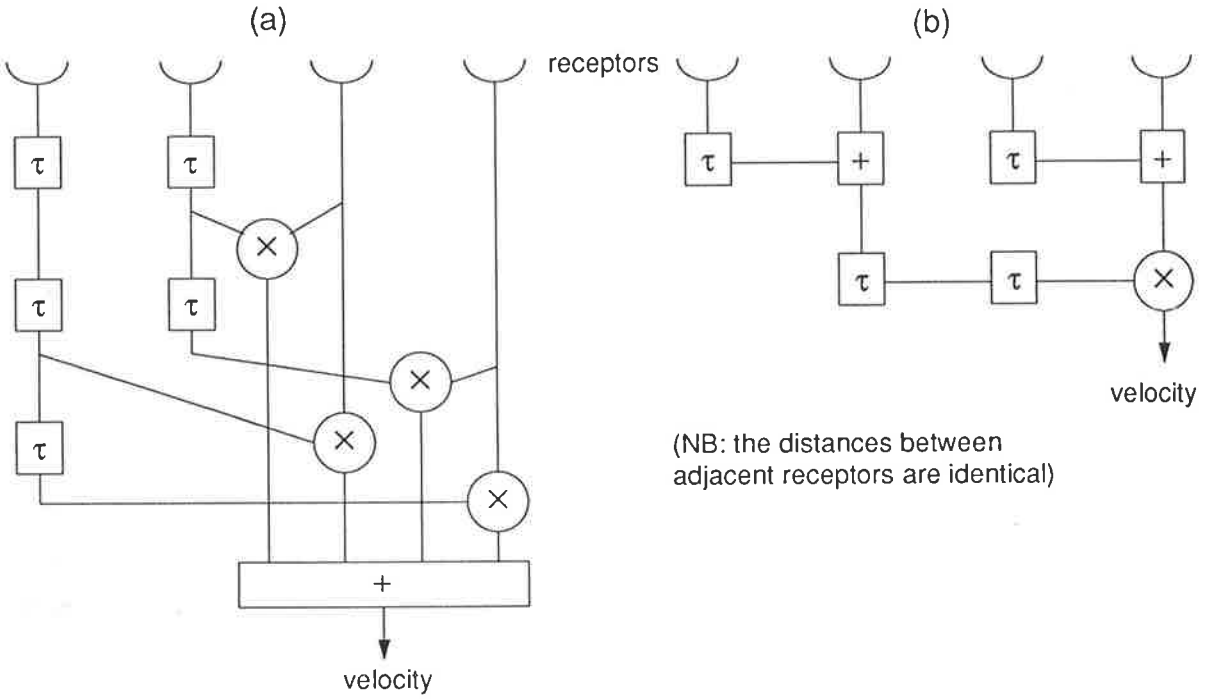


Figure 3.10 Multi-input Reichardt detector configurations

Multi-input Reichardt detectors exhibit sharper tuning than some other detectors, and hence a strong response yields a good approximation of the true velocity of the stimulus. By the same token, though, a greater number of detectors are required for the responses to overlap significantly in order to report velocities that lie in between tuning peaks. Burr *et al.* (1986) have noted that temporal and spatial frequency tuning are coupled, in that fast velocities shift down spatial sensitivity, thus there appears to be a trade-off between the range of detectable velocities and spatial resolution. In other words, since high velocities require that the distance between adjacent receptors be large, or that the time delay be short (or both), the extent to which multi-input Reichardt detectors are pattern invariant is a function of the physical restrictions of receptor combinations. In the scheme suggested by Snippe and Koenderink, groups of 4-input detectors are used (i.e., 8 receptors per detector). A labelling scheme is thus employed for encoding velocity, which is composed of banks of velocity-tuned detectors.

5. Summary

Of the many roles played by motion perception in nature, the detection of relative motion and looming seem to be of particular interest to the low-level control of an autonomous vehicle. However, before addressing the issue of utilising the information (see Chapter 8), some of the relevant characteristics of visual systems should be examined in order to determine how biological systems extract such information from the visual environment.

Firstly, experimental evidence strongly suggests that motion perception results from the hierarchical combinations of a number of elementary processes, and that integration of information may also be involved. Secondly, the existence of a fundamental quantity, d_{max}, indicates that visual systems "prefer" smooth and coherent motion. These two characteristics are the main justifications behind the forward tracking algorithm presented in Chapter 7. Finally, the analysis of neuronal responses to moving stimuli reveals that motion perception processes may be enabled by mechanisms sensitive to direction, both relative and absolute, and velocity.

The mechanisms which provide directional and velocity information can be modelled in different ways. For instance, one approach is to replicate neuronal mechanisms (see Section 3), while another is to analyse desired responses with a view to reproduce them mathematically (Section 4). It is worth noting that the models presented here are both directionally selective and velocity tuned, and require sometimes complex computations.

However, even moderate computational requirements may limit the performance of an autonomous vehicle, which relies primarily on reaction speed, and hence a model which is purely directionally selective is described in the next chapter. The rationale is that the model is simple to implement in hardware, as shown in Chapter 5, and yet permits velocity information to be extracted through integrative methods (see Chapter 6 for preliminary results, and Chapter 7 for the methods).

CHAPTER 4 A model for hardware motion detection

This chapter describes a simple motion perception model which interprets changes in contrast in order to provide directional information. It is argued that, due to hardware constraints, it may be preferable to estimate velocity through integrative methods, rather than relying on the accuracy of motion detectors. The model is investigated by simulation, in order to guide the implementation process and to serve as a reference for testing. The features which make this model attractive to a hardware implementation are summarised at the end of the chapter.

1. Motivations

Some desirable characteristics of a visual sensing system suitable for the lower control levels of an autonomous vehicle are a manageable size, and the ability to provide information rapidly, possibly in a readily usable format so as to facilitate its interpretation by the control system. The format of the information should also be flexible, since, as an experimental system, the overall requirements may not be easily predictable. Hence, the questions are how the sensor should be implemented, and how much processing can be incorporated in the sensor.

Processing of sensory information

Referring to the "criterion for perception" presented in Chapter 2, perception results from the interpretation of detected information, which itself may have been provided by a lower level perception process. A visual "perceptor" should therefore at least provide directional information, from which position, velocity, range, and the sign of the difference in contrast between a moving object and the background (i.e., polarity) may be extracted. Higher levels of perception, such as shape and object classification, may not be essential for simple naviga-

tional tasks, as shown by insects whose visual systems are limited compared to that of humans (Horridge, 1987).

The motion perception models which were presented in Chapter 3 produce a motion sensitive signal inferred from the responses to changes in contrast of a two-input motion detector. The sign of the signal provides the direction of movement, while the magnitude is indicative of velocity (strictly speaking, a motion detector is tuned to a particular velocity, thus the "true" velocity corresponds to the detector whose response is strongest). For example, consider the one-dimensional case of Figure 4.1, where an object moves to the right, starting from detector A at time t_0. By sampling the motion detector outputs, the velocity of the object at time t_i can be estimated either directly from the motion detector whose response is strongest (v_i), or from the average of the strongest responses over a number of sampling instants. The accuracy of the estimated velocity is thus a function of the tuning characteristics of individual motion detectors. It is worth noticing that in insects, motion detectors do not appear to be tuned sharply, and may in some cases respond to a fairly wide range of velocities (Horridge & Marcelja, 1992), implying that velocity should be derived from a combination of responses.

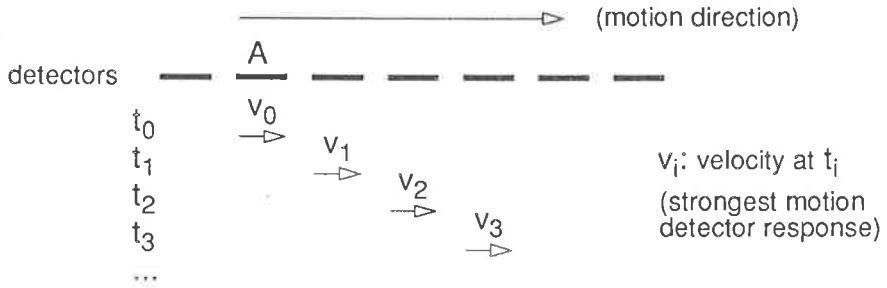


Figure 4.1 Estimating velocity

However, it may not be necessary for the motion detector to provide velocity if directional information can be stored. For the example of Figure 4.1, a simple way of estimating the velocity at a given time t_i would consist of calculating the ratio of the visual angle between, for instance, A and the detector which has responded to the motion at time t_i, to the elapsed time t_i minus t_0. Therefore, the direction of motion is sufficient to estimate velocity, and, as argued next, the estimate is not necessarily less accurate than in the previous case, especially when hardware limitations are taken into account.

Hardware constraints

As size and power requirements may be critical for small autonomous systems, it is proposed to implement a motion detection scheme on a full custom "Very Large Scale Integration" chip (VLSI). Thus the question arises as to which scheme would be most suitable to a VLSI implementation, and to that effect, several aspects of the required processing should be considered. Essentially, light must be detected and converted into an electrical current, which

is sampled and finally processed to the required level of perception. Since a fairly large number of detectors are necessary in order to sample the visual field at a reasonable resolution, the main concerns with motion detection schemes are thus bandwidth and storage requirements.

At the interface between the "outside" and the processing, the sensor must incorporate analog circuitry, and hence a certain amount of processing may be incorporated with the sensing. Analog processing presents the advantage of reducing the bandwidth, since a piece of information can be represented as a single voltage level or current intensity. However, analog implementations tend to require fairly large device sizes in order for the signal-to-noise ratio to be acceptable. In addition, devices at different locations must be carefully matched in order for their responses to identical stimuli to be similar. By contrast, digital implementations result in many more, but much smaller, devices, whose matching is not really critical. Predictably, the device matching problem in VLSI technology becomes more acute as device size decreases, thus large bandwidths and storage space are more easily dealt with when the information is represented digitally. Given a large number of detectors, the trade-off between digital and analog favours a digital representation of information once the direction of motion has been detected, as the accuracy requirements of the motion detectors can be minimised.

To summarise, the motion detectors need only respond qualitatively to motion by indicating its direction using a digital representation, which in turn presents the advantage of being more flexible than an analog representation. Moreover, by reducing the space requirements of the motion detectors, velocity estimation can be accomplished on the same VLSI device by using simple integrative mechanisms (see Chapter 7). Finally, if polarity and positional information are conserved, range estimation and looming detection are facilitated (see Chapter 8).

2. The template model for motion detection

Evidence gathered from electro-physiological experiments effected on some insects and mammals suggests that directional selectivity may result from sequence discrimination (see Chapter 3). Neurologically, the outputs of adjacent light detectors, corresponding to a small portion of the total visual field, either excite or inhibit directionally selective movement-detecting neurons, whose responses are independent of the precise location. Moreover, under certain circumstances the perception of direction does not seem to be affected by brightness polarity reversals of the moving stimulus, thus indicating that motion information may be integrated in time (Shechter & Hochstein, 1990). However, in the case of detectors based on the Reichardt model, the transient responses to moving edges are sometimes ambiguous (Mather, 1984). In fact, the perception of movement appears to agree with the predictions obtained from the model of Marr & Ullman (1981), which uses the spatio-temporal derivative of the luminance at a single position (see Appendix A).

Thus, the so-called "template model", first proposed by Horridge (1990), is of particular interest as it consists of combining, in space and time, the responses to changes in contrast of adjacent light detectors. Unlike the correlation models of Chapter 3, the template model is well suited to detecting object boundaries, or edges, and this characteristic appears to be a function of the fly's "large monopolar cells" (Srinivasan *et al.*, 1990). As will be shown, a moving edge causes some of the combinations of detector responses (hereafter referred to as "templates"[1]) to indicate the direction of movement, and in such a way that the information can be readily utilised by digital systems. Moreover, the templates contain polarity information, i.e., the sign of the change in contrast.

The concept is illustrated in Figure 4.2, where a row of detectors with overlapping receptive fields respond in a continuous fashion to incoming light (the overlap is necessary to provide a continuous coverage of the visual field and to resolve ambiguities, as will be shown later). Each detector output is sampled at regular intervals and compared with the previous response, so that at a given sampling instant the output indicates either an increase (↑), a decrease (↓) or no change (–) in light intensity. Adjacent outputs are then grouped pairwise and at consecutive sampling times, yielding the spatio-temporal templates.

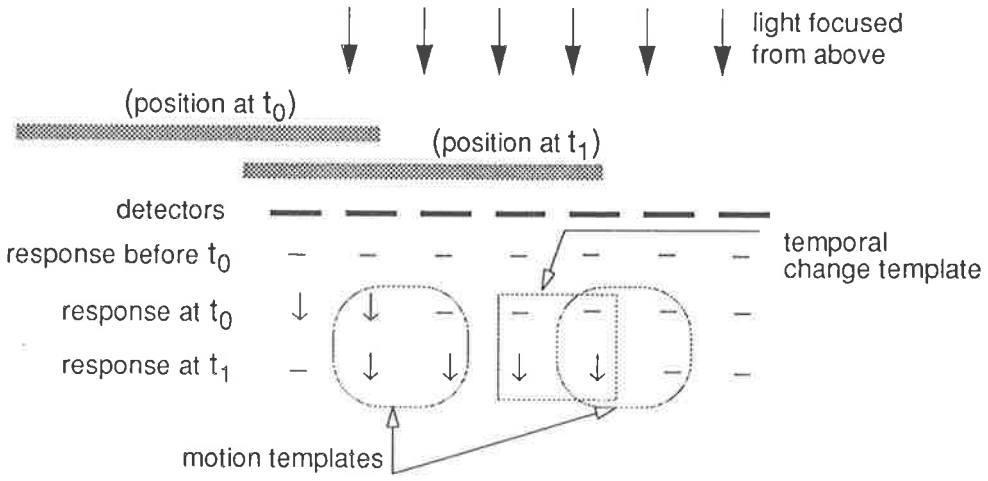


Figure 4.2 Example of template formation

In the example, light is focused onto the row of detectors, all of which before time t_0 indicate no change in light intensity. At t_0 an object has moved in from the left, thereby masking the light and causing the first two detectors to indicate a decrease in contrast, while the other detectors report no change. Between t_0 and t_1 the object moves further across, causing other detectors to report a decrease in contrast. The encircled templates shown in Figure 4.2 convey elementary "digits" of information, such as motion in a given direction, while the "temporal change" template (square box) does not indicate motion.

1. In this thesis, the word "template" refers to a spatio-temporal combination of contrast changes. This meaning should not be confused with its conventional meaning in image processing.

In the next section, a simple simulation is employed to show the manner in which changes in contrast may be combined under a variety of circumstances. The main purpose is to classify the templates in order to assist the VLSI design process (see Chapter 5), and to provide some insight into what to expect under "real-life" conditions (see the results presented in Chapter 6).

3. Template classification

The sampled output of each detector can indicate either an increase ($\uparrow$), a decrease ($\downarrow$) or no change (–) in light intensity. Hence for a pair of detectors there are nine possible combinations: (– –), ($\uparrow$ –), ($\uparrow$ $\uparrow$), (– $\downarrow$), ($\downarrow$ –), ($\downarrow$ $\downarrow$), (– $\downarrow$), ($\downarrow$ $\uparrow$) and (– $\uparrow$). These combinations are formed in the spatial domain. The temporal domain is included by associating the combinations obtained at consecutive sampling times, yielding 81 possible spatio-temporal templates.

Each template conveys some sort of information, such as temporal change and motion, as shown in Figure 4.2. However, a number of other templates do not convey meaningful information in terms of analysing what the detectors have "seen", and indeed would occur under noisy conditions. It is therefore useful to find out which templates are likely to occur and under what conditions, firstly in order to assist in designing the hardware (see Chapter 5), and secondly to assess how closely practical results match the theory (see Chapter 6).

3.1 Step function simulation

One way of characterising the templates consists of moving an edge marking a transition in contrast in front of the row of detectors, one side of the edge being bright, the other dark. Assuming that the edge has been stationary long enough for all the detectors to indicate no change in contrast, their response to the edge starting to move is analogous to a step function.

The step function can be simulated using the method illustrated in Figure 4.3, which consists of plotting the position of the edge with respect to time and superposing a grid where the horizontal lines correspond to sampling instants, and the vertical lines delineate the detectors' receptive fields (due to the overlap, there are two vertical lines per detector). In the event of a dark object moving across a detector's field of view between sampling instants t_i and t_{i+1}, the graph will show a line crossing the square delineated by the vertical receptive field lines and the horizontal lines pertaining to t_i and t_{i+1}. The amount of light received by the detector during that time is a function of the area, within the square, and above the line indicating the edge's movement (e.g., the hashed surface). As the object moves further across the receptive field between t_{i+1} and t_{i+2}, the corresponding area within the next square decreases, and hence the detector should indicate a decrease in contrast.

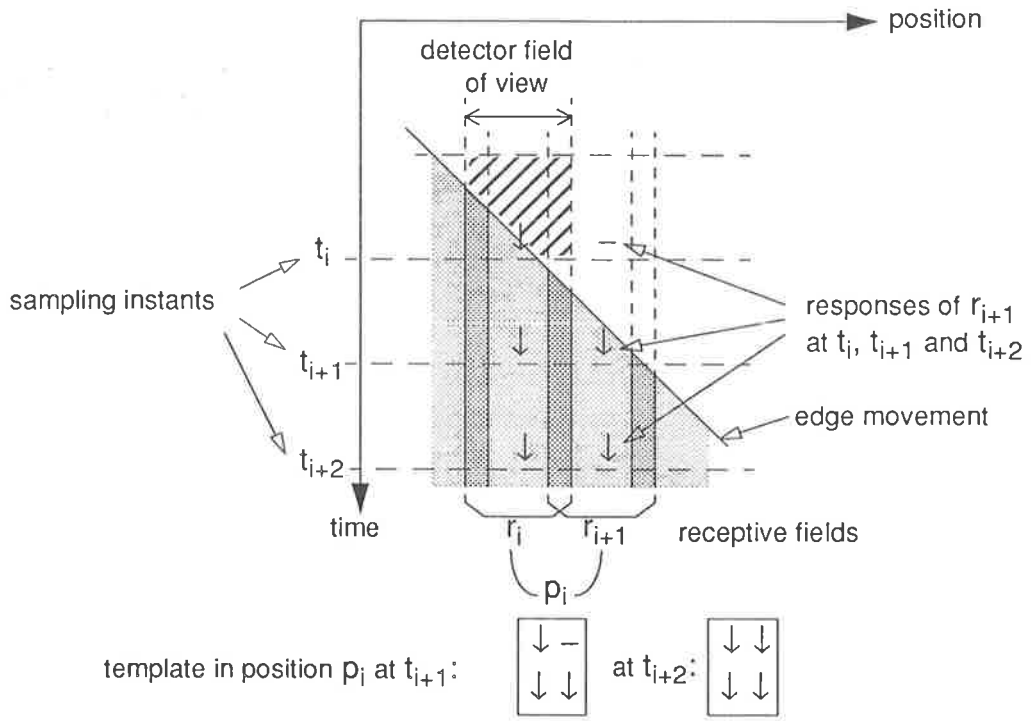


Figure 4.3 Simulation method

In the method described in Figure 4.3, it is assumed that the response of a detector to incoming light is constant within its receptive field, which is a very crude approximation. For a physical implementation, the sensitivity would be of the form shown in Figure 4.4, where it is at its maximum in the middle and decreasing towards either side of the receptive field. However, there does not appear to be major qualitative differences between results obtained by incorporating this variation in a software simulation from the results yielded by the "graphical" simulations which follow (Nguyen, private communication).

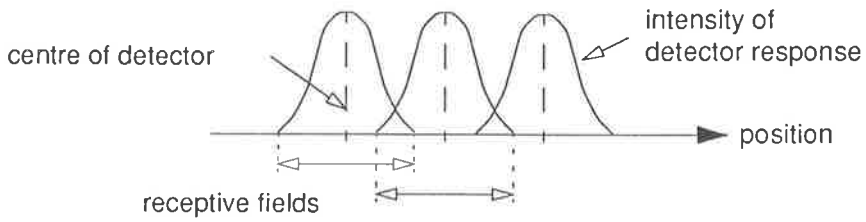
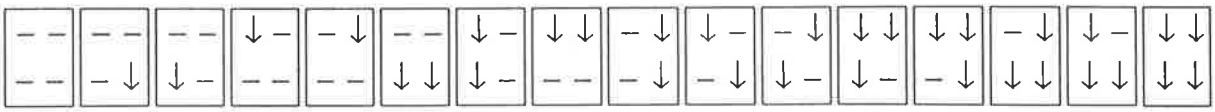


Figure 4.4 Intensity of detector response versus position

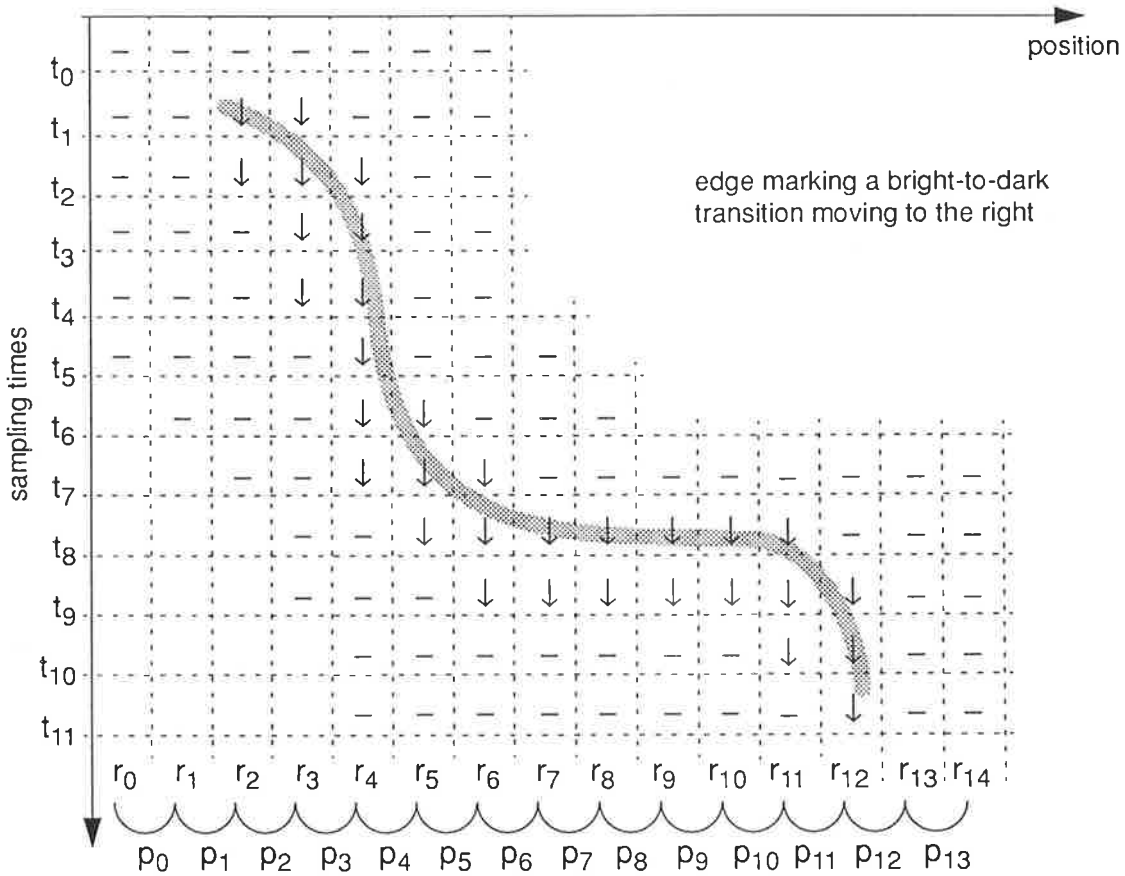
An arbitrary template encoding is used in the examples of Figure 4.5 and Figure 4.6 where a bright-to-dark edge moves to the right and to the left, respectively (the "no motion" template, "0", is represented as a dot). Similar responses can be obtained with the other polarity, i.e. dark-to-bright, by replacing the "down arrow" (↓) with the "up arrow" (↑). In the examples, in order not to crowd the picture, the edge's position is plotted as a line whose width corresponds to the detectors' overlapping receptive fields.

Arbitrary template encoding:



0 (.) 1 2 3 4 5 6 7 8 9 A B C D E F

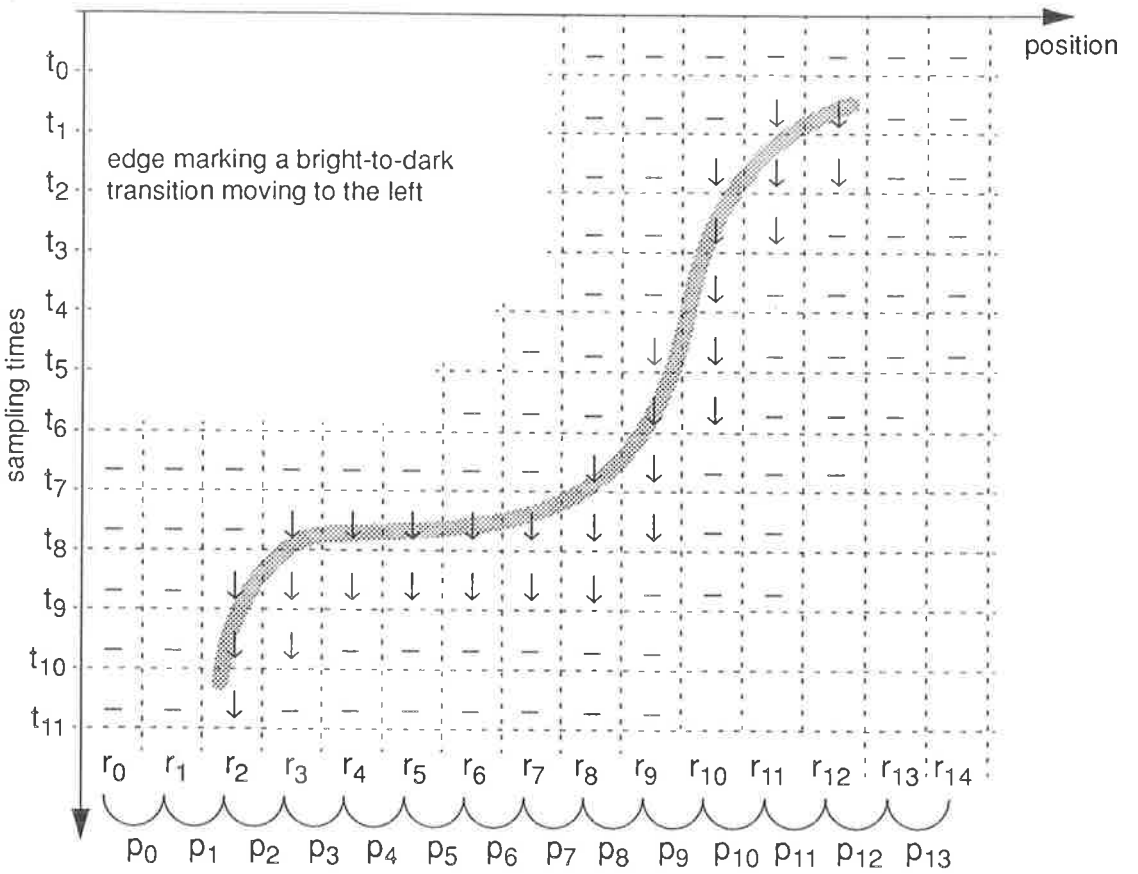
(NB: a different encoding will be used experimentally - see Chapter 6)



	p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}
t_0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
t_1	.	1	5	2	.	.	.	.	.	.	.	.	.	.
t_2	.	8	F	E	2	.	.	.	.	.	.	.	.	.
t_3	.	4	C	F	6	.	.	.	.	.	.	.	.	.
t_4	.	.	8	F	6	.	.	.	.	.	.	.	.	.
t_5	.	.	4	C	6	.	.	.	.	.	.	.	.	.
t_6	.	.	.	8	E	2	.	.	.	.	.	.	.	.
t_7	.	.	.	8	F	E	2	.	.	.	.	.	.	.
t_8	.	.	.	4	C	F	E	5	5	5	5	2	.	.
t_9	.	.	.	.	4	C	F	F	F	F	F	E	2	.
t_{10}	.	.	.	.	.	4	7	7	7	7	C	F	6	.
t_{11}	.	.	.	.	.	.	.	.	.	.	4	C	6	.

template response (position versus time)

Figure 4.5 Step function simulation - rightward motion, bright-to-dark



	p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}
t_0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
t_1	.	.	.	.	.	.	.	.	.	.	1	5	2	.
t_2	.	.	.	.	.	.	.	.	.	1	D	F	6	.
t_3	.	.	.	.	.	.	.	.	.	8	F	B	3	.
t_4	.	.	.	.	.	.	.	.	.	8	B	3	.	.
t_5	.	.	.	.	.	.	.	.	1	D	6	.	.	.
t_6	.	.	.	.	.	.	.	.	8	F	6	.	.	.
t_7	.	.	.	.	.	.	.	1	D	B	3	.	.	.
t_8	.	.	1	5	5	5	5	D	F	6	.	.	.	.
t_9	.	1	D	F	F	F	F	F	B	3	.	.	.	.
t_{10}	.	8	F	B	7	7	7	7	3	.	.	.	.	.
t_{11}	.	8	B	3	.	.	.	.	.	.	.	.	.	.

template response (position versus time)

Figure 4.6 Step function simulation - leftward motion, bright-to-dark

The simulations of Figure 4.5 and Figure 4.6 each illustrate the case where a single transition has occurred, from dark to bright or from bright to dark. However, if an edge is followed by another edge of opposite polarity, the response to the second edge can only be fully determined by the step function response if there is enough time in between for the detectors to

reach the steady state. Figure 4.7 illustrates the case where a bright-to-dark edge moving to the right is followed shortly afterwards by a dark-to-bright edge, also moving to the right. The encircled template indicates that one detector did not reach the steady state before the onset of the second edge.

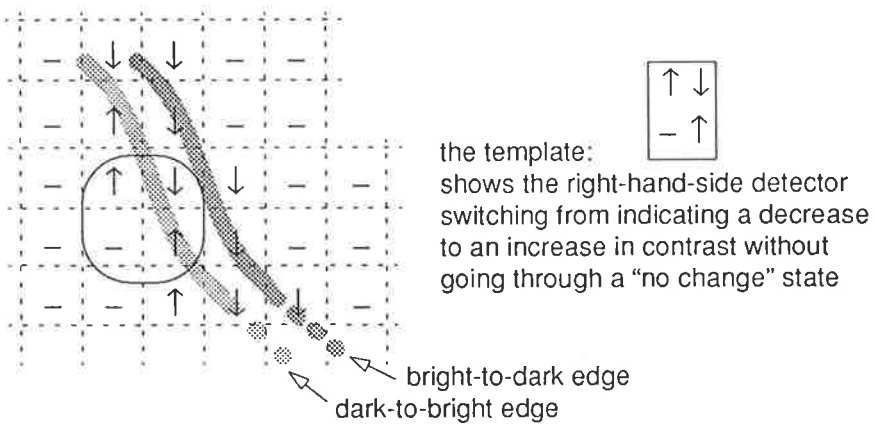


Figure 4.7 Bright-to-dark followed by dark-to-bright edges, moving to the right

Obviously, the conditions can be varied in many ways: edges following each other at different and varying velocities, changes in direction, edge crossovers, dynamically varying backgrounds, noise, and so on. However, it is questionable whether more useful insight into how templates occur can be gained through further simulations, as the purpose is to have some idea of what to expect in order to design the hardware. One aspect of hardware design is that the expected behaviour is rarely, if ever, exactly matched in practice, due to a certain degree of unpredictability inherent to fabrication processes. In fact, the inaccuracies of the simulation models, particularly in VLSI, may be the source of misleading interpretations, ultimately resulting in a hardware implementation which leaves only a small margin for errors.

3.2 Template characterisation

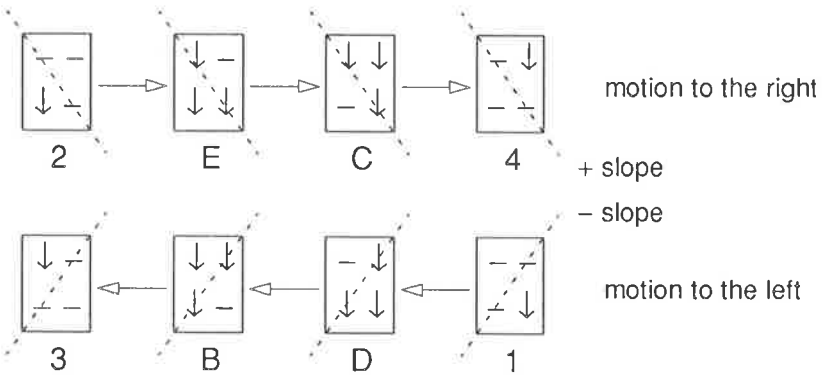
The following observations can be made concerning the step function simulations:

- The template response is spread over space and time, which is to be expected due to the fact that templates are associated with pairs of detectors which mark the transition from the "no change" state to the "decrease" state, and then from "decrease" to "no change". In some cases, a "decrease" state is followed by another one as the edge gradually moves across the detector. Hence the steady state is reached some time after the onset of the transition.
- The edge's motion is reflected by the occurrences of certain templates which appear to be directionally sensitive. Furthermore, these templates occur in pairs which mark both the onset of the edge and the establishment of the steady state.

- Some templates occur irrespective of the direction of motion, but indicate either spatial or temporal change.
- Two templates have not occurred (9 and A).
- Lastly, note that in both examples, motion templates 1 and 2 occur simultaneously when the edge starts to move. This is unavoidable due to the fact that one detector responds to a change in contrast while the lighting conditions above the adjacent detector in the opposite direction of motion do not change.

Motion sensitive templates

Provided the edge's motion is continuous, the initial response of a pair of adjacent detectors to a bright-to-dark edge moving to the right is from (– –) to (↓ –), followed some time later by (↓ ↓), (– ↓) and again (– –). The responses are reversed if the edge is moving to the left. The templates obtained by combining these responses are, using the arbitrary encoding:



The common denominator between these templates is that three receptor responses in each are identical, forming triangles whose diagonals have the same slope, arbitrarily called "positive" when motion is to the right and "negative" when motion is to the left. In both examples, the onset of the edge is marked by the pairs (2,E) or (1,D) for motion to the right and left respectively, while the steady state is established after (C,4) or (B,3) have occurred.

The templates marking the onset of an edge (i.e., (2,E) and (1,D) pairs) seem to convey the most useful information as they are indicative of the edge's position in time (therefore its velocity), and will be referred to as *primary* motion templates. Moreover, since two primary templates occur for each type of edge, their association forms what will be called a *conjugate* pair, which is particularly relevant to one of the tracking algorithms described in Chapter 7.

The template pairs marking the establishment of the steady state, i.e., (C, 4) and (B, 3), match their primary counterparts in terms of positions (except in the case where temporal change templates cause them to be translated - see further), but with a time lag. Their usefulness is therefore somewhat dubious, especially if edges follow one another closely, thus caus-

ing some of the intermediate templates in the "expected" sequences not to occur. An example of what might happen under these circumstances is given in Figure 4.8, which depicts possible sequences of templates caused by a bright-to-dark edge followed by a dark-to-bright edge, both moving to the right. Each case corresponds to a fixed distance between the edges, and, as the distance decreases, shorter sequences occur. The sequences are inferred from one another by "merging" detector responses.

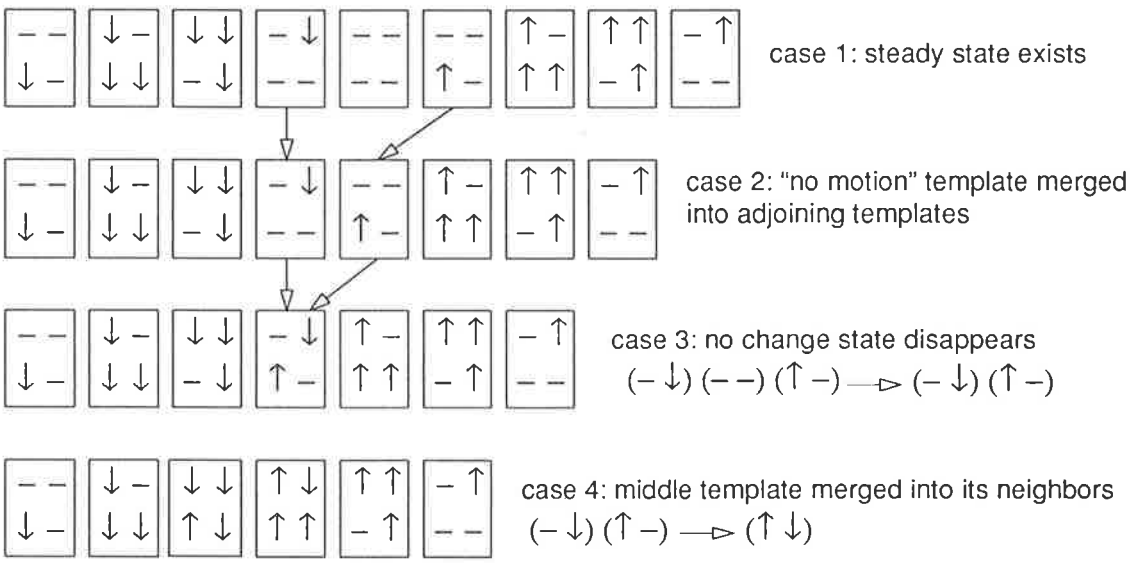


Figure 4.8 Motion template sequences (motion to the right)

The purpose of the simulation of Figure 4.8 is to show that "noisy" templates, caused by the amalgamation of detector responses, may still convey motion information. In fact, the templates all have in common that the detector responses along the positive slope diagonal are identical, while the responses along the negative slope are always different. Hence the definition of a motion template can be extended to include all the templates which show identical responses along one diagonal, and different responses along the other.

Temporal and spatial change templates

Templates 5 and 7 in the examples of Figure 4.5 and Figure 4.6 occur irrespectively of the direction of motion, but are nonetheless significant in that in both cases the edge has accelerated. It was previously noted that the motion template pairs indicating the establishment of the steady state do not always occur in the same positions as the corresponding pairs marking the onset of the edge. For instance, the pair (2, E) in Figure 4.5 occurs in position p_6, but not the pair (C, 4), whose next occurrence in the direction of motion is at p_{10}. In between, from p_7 to p_9, templates 5 and 7 are present instead (at times t_8 and t_{10} respectively). The same kind of event can be noted in the example of Figure 4.6

Templates 6 and 8 also occur in both cases, but only when the edge is moving at less than one position per sampling period. In Figure 4.5 for instance, template 6 occurs with the onset of the edge between the pair (2, E) in position p_4 (at times t_3, t_4 and t_5), while 8 marks the transition to the new steady state, between the pair (C, 4) in position p_3 (at t_6 and t_7). Again, a similar situation occurs in Figure 4.6

Other templates

Templates 9 and A do not occur, which is not surprising given the overlap in the receptive fields of the detectors. Intuitively, these templates should only appear if an edge has moved in from one side before a sampling instant t_0, stops moving between t_0 and the next step t_1, while another edge moves in from the other side between t_0 and t_1.

3.3 Summary of template characterisation

$t_0 \setminus t_1$	– –	↓ –	↓ ↓	↓ ↑	– ↑	↑ ↑	↑ –	↑ ↓	– ↓
– –	nc	→ D	T		← B	T	→ B		← D
↓ –	← d	S	→ D				T	→	
↓ ↓	T	← d		←		T		→	→ d
↓ ↑			→	S		←	←		→
– ↑	→ b				S	← B		←	T
↑ ↑	T		T	→	→ b		← b	←	
↑ –	← b	T		→		→ B	S		
↑ ↓		←	←		→	→		S	
– ↓	→ d		← D	←	T				S

Movement:
→ : to the right
← : to the left

Edge onset polarity:
D: bright-to-dark
B: dark-to-bright

Transition to steady state
d: bright-to-dark
b: dark-to-bright

T: temporal change
S: spatial change

nc: no change template

Table 4.1 Template table

In Table 4.1, the shaded boxes highlight the most relevant templates, i.e., those corresponding to the onset of an edge. The other templates containing directional information are those marking the transition to steady state and "noisy" templates, which may occur under certain circumstances.

4. Discussion

The template model provides a simple mechanism by which the spatio-temporal derivative of an image may be obtained in a manner reminiscent of some common image analysis techniques (see Appendix A). However, extracting templates consists of a mapping from the spatio-temporal image space into "direction-time" space by finding where changes in contrast are locally consistent. In other words, a new image is obtained by forming a logical association of contrast change pixels adjacent in space and time.

In order to understand the scope of the template model, it may be useful to draw a comparison between the interpretation of contrast changes and a morphological operation. Morphological filtering (see Serra & Vincent, 1992, and Sternberg, 1986, for reviews) may be employed to more clearly define two-dimensional shapes by, for instance, filtering out noisy pixels on the basis of their being inconsistent with their neighbours (erosion), or by restoring the value of pixels to that of neighbouring pixels (dilation). Such operations could thus be applied either to a "contrast change" image, or, maybe more effectively, to the "template" image. In the first case, spurious contrast changes could be filtered out or "missing" changes restored, while in the second case, different operators could be used to eliminate noisy templates or to restore motion sensitive templates which should normally occur in conjugate pairs.

The experimental results of Chapter 6 show that such post-processing may not normally be required in order to obtain velocity estimates, using the methods described in Chapter 7. However, the results were obtained under fairly well controlled conditions, and hence it may be useful to perform some sort of filtering if warranted by extreme circumstances.

Another aspect concerns the fact that the template model detects motion in one dimension. While desirable in principle, an extension of the model to two-dimensional motion detection may not be necessary for the lower control levels of an autonomous vehicle, and it may be sufficient to simply use another detector oriented perpendicularly to the first (see Chapter 9). Also, if the vehicle's movements are restricted to a horizontal plane, obstacle avoidance only implies a one-dimensional change in direction, i.e., to the right or to the left. Therefore, to a first approximation, knowledge of the relative motion of the vertical components of objects may be sufficient.

Finally, the so-called "aperture problem" arises in two-dimensional motion, as described at some length in Appendix A. Briefly, if the extremities of a moving edge are outside the field of view of a visual receptor array, the edge's motion can only be determined to within 180 degrees. Since this condition may occur when the sensor is close to an object, two-dimensional motion information is almost useless when the vehicle is in danger of colliding with an obstacle.

5. Summary

An object in motion may alter the pattern of contrast levels in the field of view of an observer. The template model provides an efficient mechanism by which detected changes in contrast, separated in space and time, can be combined in order to infer directional information. The model is particularly suited to being implemented in hardware as it fulfils the requirements expressed at the beginning of this chapter, namely, that at least motion direction be provided, and in a readily usable format.

For the hardware implementation, which will be described in Chapter 5, the most attractive features of the template model are the following. Firstly, the front end of the detection stage, which necessitates analog circuitry, can be fairly easily implemented since the detectors need only respond qualitatively to local changes in contrast, and hence high accuracy is not required. Secondly, the interpretation of the changes in contrast is implicitly contained by the templates, which are simple combinations of detector responses. Importantly, no computations, in the arithmetic sense, are required in order to extract the direction of movement. Thirdly, motion information is efficiently conveyed in a readily encoded form containing directional as well as polarity information. This last point is most relevant to the algorithms of Chapters 7 and 8.

CHAPTER 5 Implementation: the "bugeye"

This chapter describes the implementation of the template model on a VLSI chip, dubbed the "bugeye". The chip is composed of three main architectural blocks, each corresponding to a distinct stage of information processing: detection, interpretation, and processing. The detection stage is analog, and comprises a single row of photo-sensitive elements, therefore the direction of motion is detected in one dimension only. Both the interpretation and processing stages are digital. Interpretation consists of combining the outputs of the detection stage to form the templates, which may then be analysed by the processing stage.

1. Architectural and technological choices

The template model, which is described in Chapter 4, combines a simple concept with a very efficient representation of directional motion information, and is particularly suited to being implemented in VLSI technology for the following reasons. Firstly, some fairly well-behaved characteristics of semiconductor materials can be utilised to build light detectors. Secondly, the detection of changes in contrast does not require high accuracy or precise device matching, thus simplifying the design of the front-end analog circuitry. Lastly, the representation of directional information is compact and can be readily processed digitally.

However, the design presents a "mechanical" challenge in that the input information consists of light which must be detected and transduced into a current or voltage in order to be processed. Thus the issues which arise concern the choices of technology and of the optical interface, and the manner in which these choices affect the general concept of the processing architecture. Another important consideration, mentioned in Chapter 1, is that for practical reasons the first version of the system incorporates as much functionality as possible while allow-

ing for extensive testing, and hence the options that are the most likely to work are usually chosen. To a large extent, the contents of this chapter have been described previously by Yakovleff *et al.* (1993), and Nguyen *et al.* (1993a).

1.1 Functional architecture

In Chapter 4, it was suggested that one-dimensional motion information may be sufficient for the low-level control of an autonomous vehicle. Therefore, a single row of photo-receptors has been implemented, and each photo-receptor responds to light focused onto it from a restricted field of view. Local changes in contrast are then detected, sampled and stored. The templates are formed by combining the responses of adjacent change detectors at consecutive sampling instants. Finally, velocity information may be inferred from the displacements of directionally sensitive templates. Thus, there are essentially three tasks to be performed, each translated into a distinct component of the architecture (Figure 5.1). The first, contrast change detection, necessitates an analog implementation. However, the sampled outputs can be thresholded and encoded digitally, thus facilitating subsequent processing. The contrast change information is then interpreted, resulting in the formation of templates which are stored in order to provide easy access to intermediate results. Finally, the outputs of the processing stage are also stored.

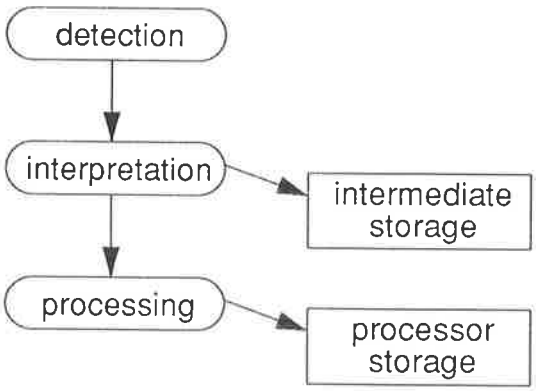


Figure 5.1 Functional architecture

The sampling rate of the detection stage should not be too high, otherwise changes in contrast may be too small to be noticeable. Hence it is proposed that the photo-receptors be sampled in parallel about every 10 milliseconds, which is adequate for analog VLSI, and also comparable with the fastest time constant of the insect motion detection neurons (Hausen, 1982b). However, as digital circuitry can operate at frequencies in the mega-Hertz range, processing speed is not much of an issue, and hence the templates can be formed by combining the detection stage outputs sequentially. Interpretation and processing can thus be accomplished serially, which, as will be shown in Section 2, allows the system to be more flexible than if the implementation were to be fully parallel.

1.2 Technology

Since processing speed is not an overriding concern, the choice of a VLSI technology is largely dictated by the requirements of the detection stage, which comprises light detectors and some analog circuitry. An important consideration for implementing the light detectors is the response of semiconductor materials to visible wavelengths. In this respect, Gallium Arsenide (GaAs) technology may be optimal (Abbott *et al.*, 1991), as the responsivity of a GaAs photo-receptor peaks for wavelengths which are more or less in the middle of the visible range. In comparison, CMOS photo-receptors, although adequate, are better suited to wavelengths which are towards the infrared end of the spectrum, and overall, the responsivity is lower than for GaAs. Another consideration concerns the stability characteristics of the analog circuitry. CMOS devices appear to be better behaved than GaAs devices, whose stability problems may be critical. Hence the safest technology for the implementation is CMOS, in spite of the desirable characteristics of GaAs photo-receptors.

The main characteristics of the chosen CMOS process are a 2 micron minimum feature size, and the availability of second polysilicon and metal layers, which presents the following advantages. Firstly, the relatively slow sampling rate requires a high time constant (i.e., large resistor and capacitor values), and fairly large capacitors can be implemented by superposing polysilicon layers. Secondly, the illumination of the photo-receptors imply that the surface of the chip is at least partly exposed. Thus it may be advisable to mask the digital circuitry with the second metal layer, in order to reduce the effects of spurious currents caused by photons hitting the surface outside the photo-detection circuits. Finally, the chosen process is reliable and accurate simulation models are available, which is of critical importance to analog design.

1.3 Optical interface

The maximum chip surface limitations, approximately 25 square millimetres, imply that the row of light detectors should not exceed a few millimetres in length, thus the optical device should be able to focus light onto a small surface. Also, for the sensor to be of practical use, the optical device should not be too mechanically cumbersome, and should present a reasonable aperture. A device which fulfils these requirements is a "gradient index", or GRIN, lens. The refractory index of a GRIN lens decreases radially, and its shape may be cylindrical, with a flat surface at each end. As its length may be chosen in such a way that the focal plane coincides with a flat surface, the lens can be positioned directly on the chip surface and in front of the photo-receptors. In addition, focusing by mechanical means is not required, as the lens has a small focal distance (1.4 mm), which implies that an object a few centimetres away from the lens is in focus. Finally, the lens chosen for the implementation (see Figure 5.2) presents a good compromise between an adequate aperture (72 degrees) and a size suitable for VLSI photo-receptors (the dimensions of the lens are 3.8 mm in length, and 1.8 mm in diameter).

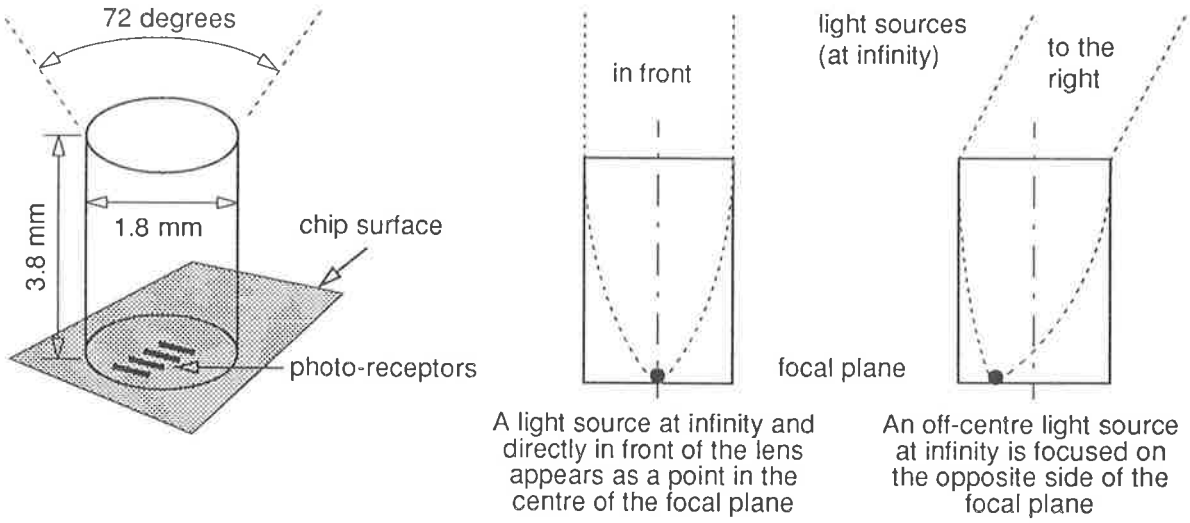


Figure 5.2 Gradient index lens characteristics[1]

The visual angle subtended by an individual photo-receptor and the separation between adjacent receptors are determined by a trade-off between the characteristics of the lens, the technological constraints of VLSI, and the imperatives of motion detection. In nature, the angular resolution of motion detection systems is in the order of one to two degrees (for the human visual system see, for instance, Westheimer & Wehrhahn, 1994). The GRIN lens has an aperture of 72 degrees and a diameter of 1.8 millimetres, and hence, on average, one degree of visual angle corresponds to 25 microns (i.e., 1.8 mm ÷ 72). However, due to the optical aberrations of the lens, a photo-receptor width of about 7 microns corresponds to approximately 1.2 degrees. Therefore, since adjacent visual receptive fields should overlap (see Chapter 4), a 25-micron pitch between receptors 7 microns wide, which are appropriate dimensions for a VLSI implementation, seems to be a reasonable compromise. The photo-receptors are depicted in Figure 5.3. Note that the length of a receptor (40 microns) is such that the angular field in the direction perpendicular to the axis of the row corresponds to approximately two degrees, which, it is assumed, may facilitate the detection of slanted edges.

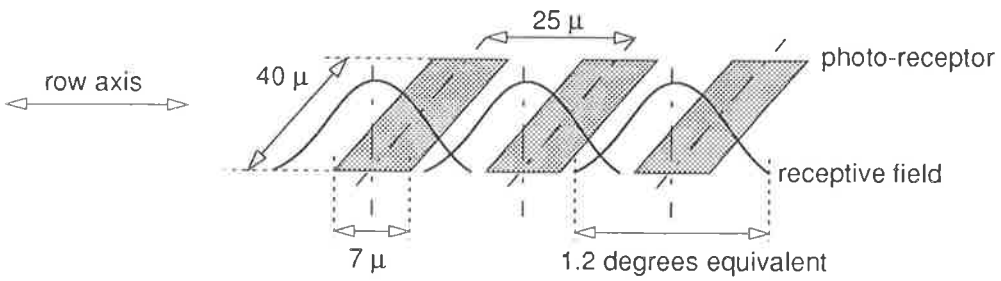


Figure 5.3 Photo-receptor dimensions

1. see Melles-Griot Catalog, Optics Guide 5, 1990 (pp 20-58 to 20-63) for a comprehensive description of GRIN lens characteristics

Lastly, as the optical aberrations increase sharply towards the periphery, it is proposed to restrict the visual field to a total of 60 degrees, implying that 61 photo-detectors be implemented. It should be pointed out that measurements (see Chapter 6) indicate that the distance-per-degree ratio is higher in the centre of the lens and decreases with eccentricity, although the variations are comparatively small and almost linear. Hence, it may be easier to introduce a correction factor when computing velocities rather than attempting to match precisely the layout to the characteristics of the lens.

2. VLSI implementation of the template model

2.1 Hardware architecture

The architecture and the floorplan of the chip (dubbed the "bugeye") are shown in Figure 5.4. The detectors comprise the photo-receptors and the contrast change detection circuitry, the outputs of which are sampled and stored, and, together with the multiplexer, form the detection stage. Thus, in between sampling pulses, the change detectors are sequentially accessed by digitally controlling the multiplexer select signals. Each combination, essentially an un-encoded template, is then used as an address to a memory constituting the interpretation stage, which is initially loaded with a pre-defined template encoding. The interpretation stage outputs are stored sequentially in the intermediate result memory, which can be read in between sampling pulses.

The templates are encoded for three main reasons. Firstly, a particular encoding of motion information may greatly facilitate further processing, by specifying, for instance, one bit for the direction of motion, another for the type of polarity change, and so on. Secondly, since only a minority of the possible templates contain relevant directional information (see Chapter 4), subsequent storage requirements may be reduced, as the template memory encodes 8 bits into 4. The last reason is flexibility, and is a corollary of the first two. Since, by definition, real-time data was unavailable while the chip was being designed, the analog circuitry could only be simulated, implying some uncertainty, particularly concerning stimulus data for the photo-receptors. Therefore, while there existed a fairly high degree of confidence that directional templates would occur as predicted, it was deemed preferable to incorporate enough flexibility in order to be able to interpret the "noisy" templates, revealed by the simulation of Chapter 4, as genuine directional templates, if required. A desirable side effect of such a "programmable" interpretation of motion information is filtering, as it may be advantageous in some cases to encode any template, other than primary directional templates, as indicating no motion.

Finally, template storage and tracking take place concurrently. As described further in Section 3, the tracking processors are initially set to track specific templates, and their outputs are stored in another memory. The various memories, all implemented as static "random access memories" (RAM), are accessible in between sampling pulses, via input-output (I/O) pins connected to internal data and address buses.

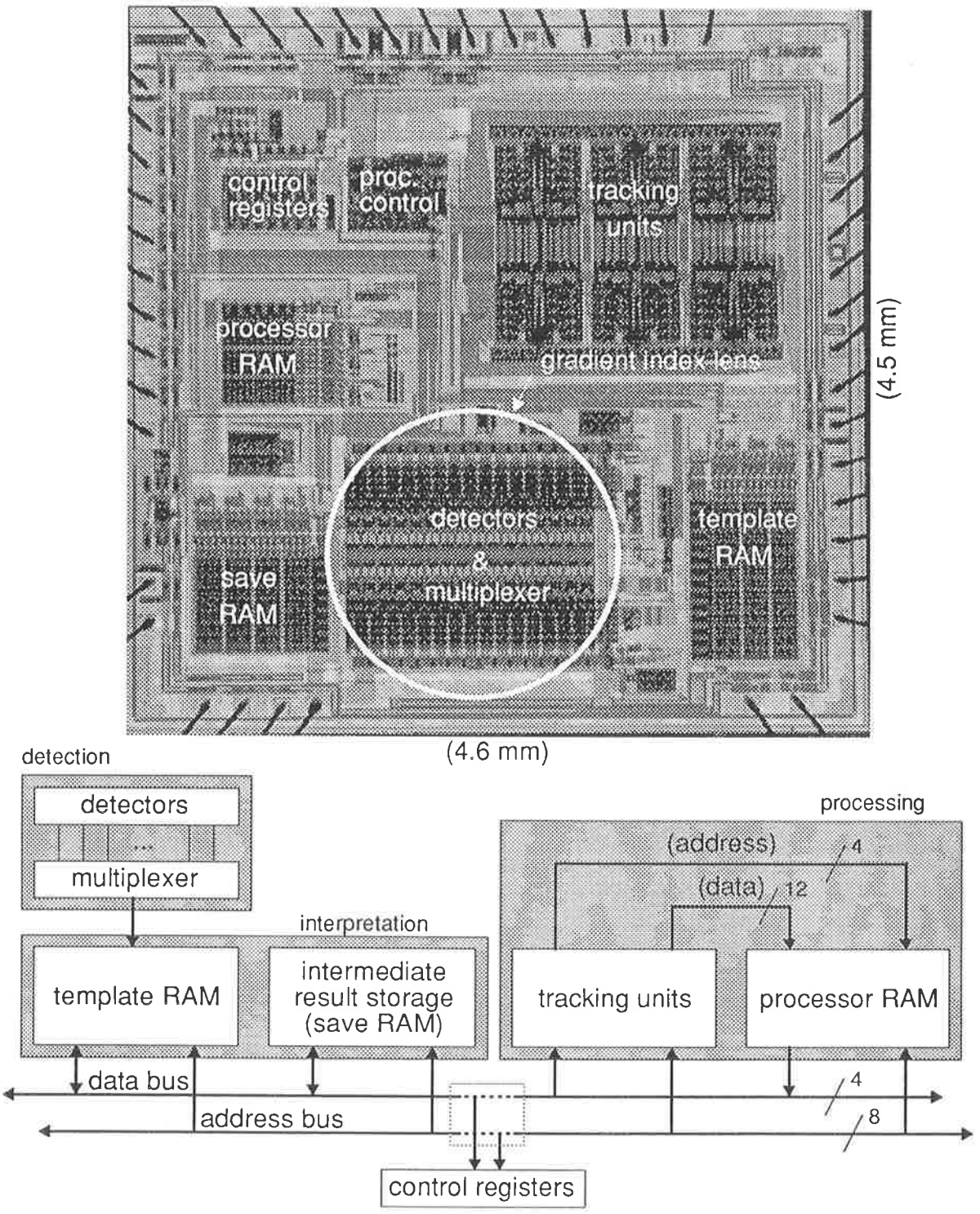


Figure 5.4 Bugeye chip floorplan (microphotograph) and architecture

2.2 Timing

Two clock signals are to be provided from outside the bugeye, one to control the sampling of the contrast change detectors, and the other to drive the digital circuitry. The timing of the different operations is shown in Figure 5.5. The processor and template clocks are derived from the digital clock.

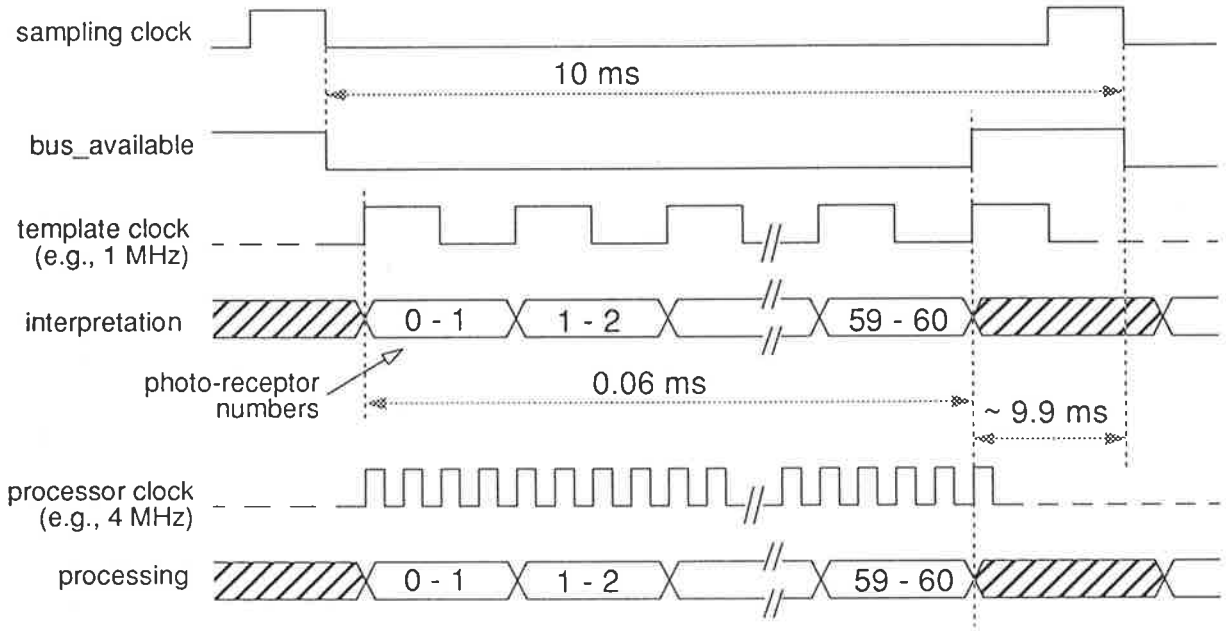


Figure 5.5 Timing diagram

Following the falling edge of the sampling clock, the templates are formed sequentially and stored in the intermediate result RAM, during which time the chip's internal buses are inaccessible from the outside, as indicated by the bus_available signal. While the latter is low, the template clock serves both to control the generation of the multiplexer select signals, and as a "write enable" signal for the result RAM. Thus with each template clock pulse, adjacent contrast change detector responses are selected and combined to form a template, which is stored at an address provided by a counter incremented by the clock.

While being stored, a template is present on the data bus, alongside an address which indicates its angular position. Therefore, the template can be processed concurrently, although, since the processor requires three steps to process each template and to store the result (see Section 3), the processor clock rate is four times the template clock rate.

The digital clock provided externally is divided by two, producing a two-phase, non-overlapping, processor clock, which is in turn divided by four to provide the template clock. Therefore, if the frequency of the digital clock is 8 MHz, the buses become available for read and write operations approximately 60 micro-seconds after the falling edge of the sampling clock.

2.3 Control registers and chip operation

A detailed description of the chip's operation is provided in Appendix B, which also includes the testing procedures used in Chapter 6. Briefly, there are five operating modes, determined by the contents of the control registers. In the "normal mode", a sampling pulse is expected, and, following its occurrence, the templates are formed, stored, and processed, without any external intervention. In the next three modes, the chip behaves like a memory, and either template, result, or processor RAM can be read or written into, as specified by externally provided "read-write" and "chip select" signals. Finally, the processor can be tested independently, by locally disconnecting the buses from the rest of the chip, and by feeding the processor data and addresses externally, along with a synchronisation signal.

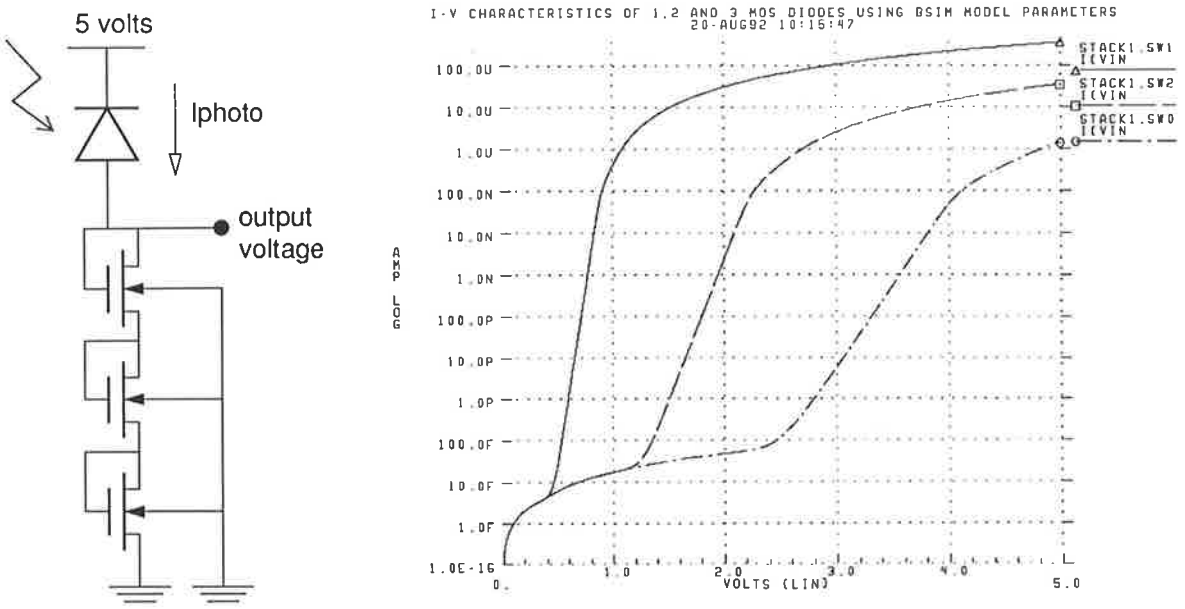
The control registers contain the processor parameters and the control mode. The contents of the registers can be changed by writing to specific locations, which are different from the addresses used to access the various memories. Thus, for instance, if it is desired to analyse the contents of the result RAM at each sampling step, the template RAM is first initialised and the following procedure is used iteratively: (i) set "normal mode", (ii) apply sampling signal, (iii) set "read result RAM mode", (iv) read the contents of the memory.

3. Main components

3.1 Detection stage

Photo-receptors

A well-substrate parasitic diode is used as a photo-receptor, and generates a current varying from pico- to nano-amperes, depending on the amount of light it receives. The circuit shown in Figure 5.6a, which is derived from an earlier design by Tanner & Mead (1984), converts the current into a voltage, and has the advantage of acting as a wide range automatic gain control, or AGC. Due to the very low photo-currents, the MOS transistors, connected as diodes, operate in the subthreshold region, resulting in a logarithmic function (Figure 5.6b). The output voltage of the circuit is proportional to the number of diodes, which is thus chosen according to the required sensitivity, while taking into account the fact that the voltage drop across the receptor element should be kept within a reasonable range in order to have sufficient reverse bias. In this design, three diodes are used, so as to compensate for the voltage drop of the next stage.



(a) circuitry (b) circuit characteristics

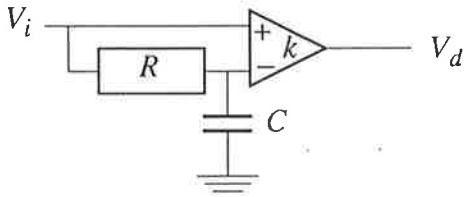
Figure 5.6 Logarithmic current to voltage converter

Differentiation stage

Time differentiation of a signal can be approximated using either feedback or feedforward methods, as shown in the examples of Figure 5.7. Due to the low frequency nature of intensity changes, the corner frequency of the differentiator circuit should be in the kilo-Hertz range. As the highest capacitance available in CMOS VLSI is in the pico-Farad range, even when using a double-poly process, the resistor should be in the order of a few giga-Ohm. Since it is infeasible to implement such a large resistor value in VLSI with a passive element, an active resistor with a large dynamic range is used instead.

(a) feedforward circuit

$$H(s) = \frac{kRCs}{1+RCs} \approx kRCs$$



(b) feedback circuit

$$H(s) = \frac{-kRCs}{1+k+RCs} \approx -RCs$$

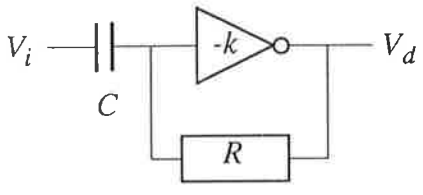


Figure 5.7 Time differentiation circuits

The active resistor includes a current mirror (see Mead, 1989) which necessitates precise matching between transistor elements in order for the circuit's characteristics to be those of an ideal resistor. Current fabrication processes, however, do not guarantee such precise matching,

and hence a voltage drop across the resistive circuit may occur under DC conditions (i.e., in the absence of motion). The feedforward circuit of Figure 5.7a presents the disadvantage that the voltage drop is amplified, and hence it may be preferable to use the feedback circuit shown in Figure 5.7b, which minimises the effect of mismatches between components of the resistive element. The time constant $RC/(1 + k)$ of the feedback circuit is in the order of microseconds, therefore the 10 millisecond sampling time is adequate.

Variations around the DC output voltage of the differentiator circuit reflect changes in light intensity, which are to be encoded digitally by a thresholding circuit, implemented with two high-gain inverters (Figure 5.8). However, due to the high compression of the current in the AGC, the output V_d of the differentiator circuit varies within a range of only 20 to 30 millivolts around the DC value $V_{d(dc)}$. Hence V_d is amplified in order to facilitate the implementation of the thresholding circuit.

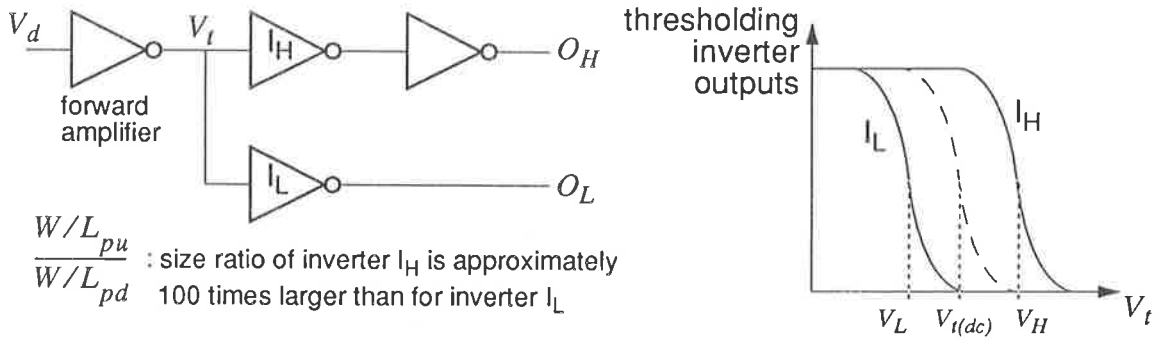


Figure 5.8 Thresholding circuit and voltage characteristics

The threshold of an inverter is a function of the ratio of pull-up to pull-down resistances, which are in turn determined by the aspect ratios of the transistors (i.e., width divided by length: W/L_{pu} and W/L_{pd} for the pull-up and pull-down transistors respectively). The transistor dimensions are chosen in such a way that the threshold V_H of inverter I_H is greater than the DC voltage $V_{t(dc)}$, while the threshold V_L of inverter I_L is smaller than $V_{t(dc)}$. Typically, due to variations in fabrication processes, the difference between the threshold values V_H and V_L should be at least 1 volt in order to ensure a reliable operation of the circuit, and therefore the gain of the forward amplifier of Figure 5.8 is in the order of 50 to 100.

When there is no change in light intensity, the input voltage of the thresholding circuit is $V_{t(dc)}$, and hence both digital outputs O_H and O_L are low. When V_t is greater than the threshold value V_H, the output O_H goes high, and conversely, when V_t is lower than V_L, O_L goes high. The digital bits O_H and O_L thus encode the three possible states: no change, increase, and decrease in light intensity. The bits are then stored in order to provide the change information pertaining to the previous sampling time. The complete circuit is shown in Figure 5.9 (see Moini *et al.*, 1993, for a comprehensive description).

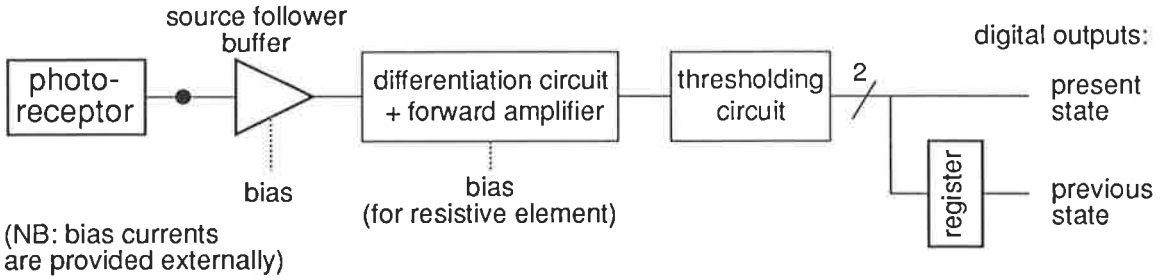


Figure 5.9 Contrast change detector circuitry

3.2 Interpretation stage

The contrast change detector outputs are multiplexed pairwise, and together with the pair's states at the previous sampling time, form the template described in Chapter 4. The template serves as an address to a memory containing an interpretation of the changes in contrast, such as an indication of the presence of movement, the direction of motion, the type of contrast change, i.e., bright-to-dark or dark-to-bright, and so on. The output of the memory (the template RAM) is then stored in the intermediate result memory, or save RAM. The interpretation stage is depicted in Figure 5.10.

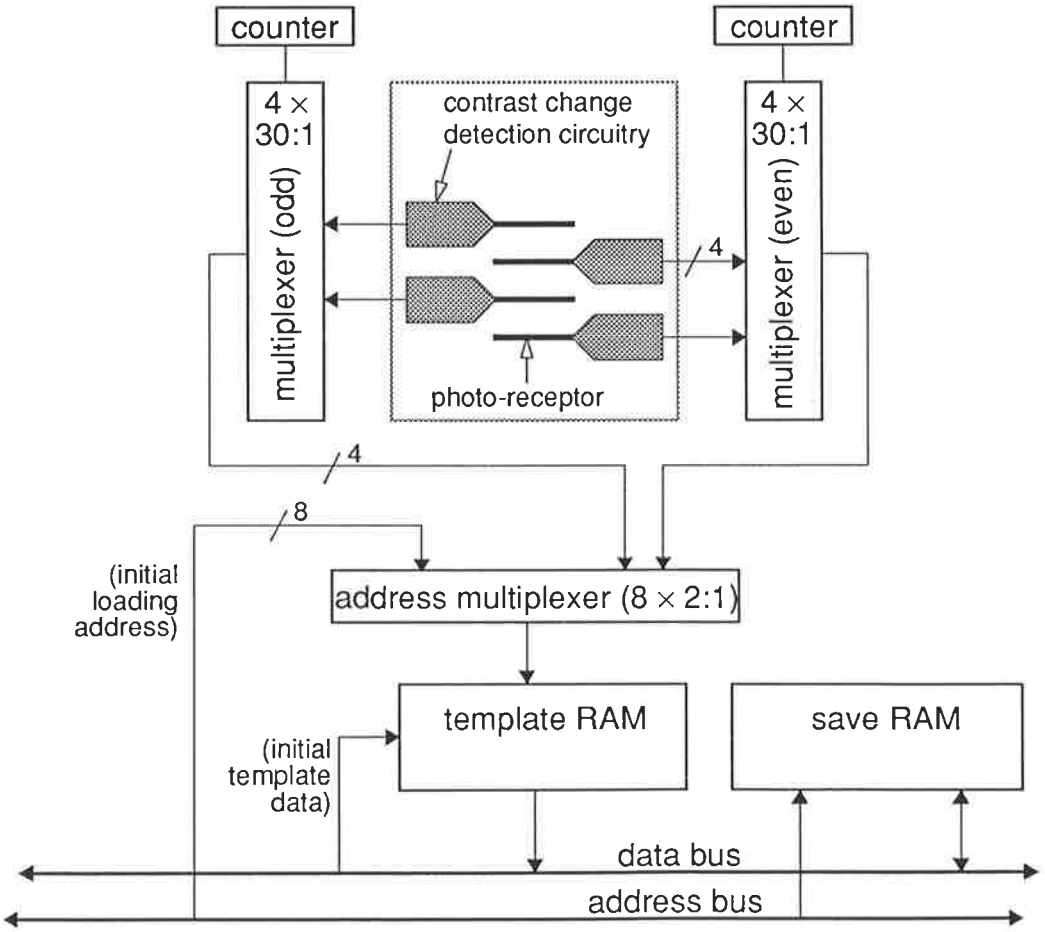


Figure 5.10 Interpretation stage

Note that the template RAM should only be written into during initialisation, at which time the address is provided externally, and the address multiplexer is switched accordingly by a control register bit.

Due to physical layout limitations, the change detectors are situated on either side of the row of photo-receptors, with one side containing the odd-numbered detectors and the other side the even-numbered. A template is formed by combining one change detector output from each side via two multiplexers, whose select signals are provided by counters which are incremented alternately. Hence, the template stored in the first location of the save RAM corresponds to the first two detectors, the template in the second location to the second and third detectors, and so on. As a result, the address of a template indicates its visual angle.

3.3 Processing stage

The processing stage comprises six individual tracking units, each capable of tracking two target templates within a pre-set area of the visual field. It is worth noting that since adjacent areas may overlap, two tracking units may be active simultaneously (Figure 5.11).

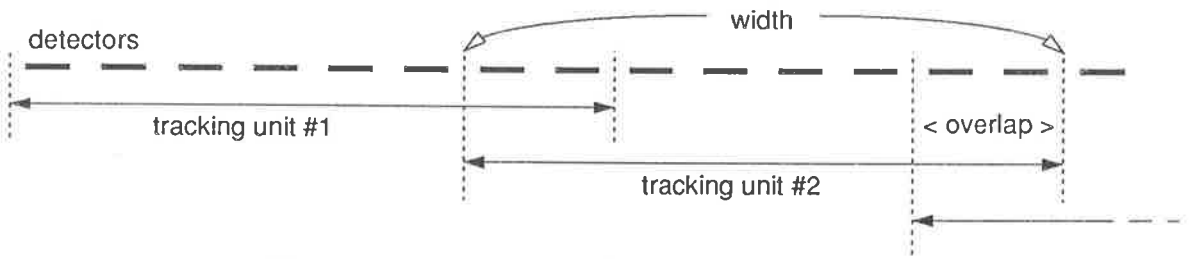


Figure 5.11 Example of tracking unit activation

The architecture of the processing stage is depicted in Figure 5.12. The "width", "overlap" and "target" parameters are provided by the chip's control registers, and the processor control uses the width and overlap values to generate the signals which enable individual tracking units (strictly speaking, the overlap value is the width minus the "real" overlap - see Appendix B). The visual area within which a tracking unit is active corresponds to a range of addresses, and tracking units are activated one after the other, concurrently with the storage of the interpretation stage outputs.

When active, a tracking unit compares the template present on the data bus with the target parameters, and, if there is a match, the current address is subtracted from the position of the template's previous occurrence, and the position is then updated with the current address. The outputs of the tracking units thus consist of the current positions of target templates, along with the displacements since their last occurrence. These results are stored in the processor RAM at an internally generated address.

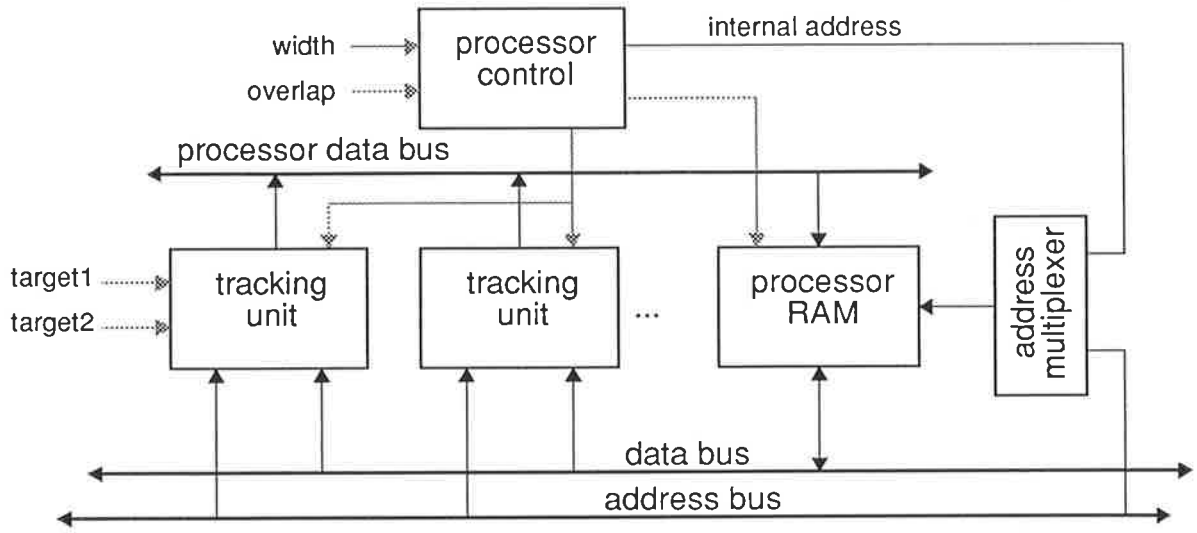


Figure 5.12 Processing stage architecture

Partitioning the total visual field into smaller areas reflects the finding that in many natural systems, directionally sensitive neurons collect signals from a limited number of small-field motion detectors (see Chapter 3), and hence each neuron would only respond to motion occurring in a restricted field of view. Therefore, the processor is meant to collect motion information from distinct, albeit overlapping, receptive fields. The information should then be further analysed with a view to producing global motion percepts. It should be pointed out, however, that the biological plausibility of such a scheme may be undermined in the case of pre-attentive vision mechanisms, as the receptive field of a directionally sensitive neuron is not always restricted to the receptive fields of the motion detectors from which the neuron gathers information (Allman *et al.*, 1985).

The current implementation of the processor suffers from a number of disadvantages. Firstly, an object may, in time, move across several areas, and hence the problem of associating motion reported by different tracking units should be addressed. Secondly, the velocity estimate can be inaccurate, as only the latest displacement is calculated, irrespective of the number of sampling pulses in between consecutive detections. And thirdly, the directional information which some templates convey is not utilised, which may cause some ambiguities when several objects are moving in the same direction.

The building and evaluation of the first chip was a learning exercise for uncovering the significance of such design compromises. Since then, new tracking and velocity estimation algorithms have been devised, and are based on data obtained experimentally (see Chapters 6 and 7 for results and tracking algorithms, respectively). Therefore, while the processing stage of the chip performs as expected, the data it provides will not be used subsequently in this thesis, and hence, further details concerning the original processor are provided in Appendix B instead of

being including here. In spite of its shortcomings, however, the successful implementation of the processing stage indicates that tracking templates in real-time is feasible, and some hardware features of the current design may indeed be used in the implementation of the tracking algorithms described in Chapter 7.

4. Comparisons with other devices and systems

In the last few years, advances in VLSI technology have made possible the design of chips which incorporate analog detectors and associated circuitry. Such devices usually perform both data acquisition and pre-processing tasks, and reduce the bottleneck between the camera and a processor, thus providing the capability to characterise images in real time (Zavidovique & Bernard, 1992). In most cases, a sensor array measures luminance, which the processor then uses to perform operations such as Fourier transforms, edge detection, and so on (see Appendix A for a description of edge detection methods). Processing may be accomplished off-chip (e.g., Tremblay *et al.*, 1993), or on-chip, in which case analog and digital processing may be combined to various degrees (e.g., Wyatt Jr *et al.*, 1992, Tanner & Luo, 1991, Forchheimer *et al.*, 1993). Many of these devices are also programmable, and hence their usage can be extended to a wide range of applications, which is not really the case for the chip described here.

A number of devices have been designed specifically to detect motion, and are thus closer to the bugeye, although the methods employed may differ. For instance, both Etienne-Cummings *et al.* (1992) and Bair & Koch (1991) use the properties of resistive networks to implement functions derived from "Laplacian-of-Gaussian" methods (see Appendix A). Motion may also be characterised by dividing or multiplying the temporal and spatial derivatives of the image intensity (see Tanner & Mead, 1989, and Moore & Koch, 1991). Yet another approach consists of reproducing the velocity tuning characteristics of some biological motion detection mechanisms, such as those described in Chapter 3. For example, Delbrück (1993) has designed a device containing photo-detectors which are connected is such a way that the response to a moving object is re-enforced when its velocity coincides with the built-in time lag between detectors.

Two devices are, arguably, close in spirit to the bugeye. The first, designed by Delbrück & Mead (1991), comprises a two-dimensional array of photo-detectors whose outputs are temporally high-pass filtered, or time-differentiated, and can be viewed directly and in real-time on a display terminal (see Mead & Delbrück, 1991). In effect, the image on the screen represents what an insect would "see". However, further processing would be necessary to extract directional motion information, and hence, according to the criterion of Chapter 2, this device does not "perceive" motion in the same manner as the bugeye.

Another chip, designed by Horiuchi *et al.* (1992), implements a velocity estimation scheme by establishing the time coincidences between the responses to changes in light intensity of photo-receptors. The device comprises a one-dimensional array of receptors with a delay line connecting each receptor pair. Therefore, the delayed response of a receptor to a change in light intensity may eventually "meet" the response of the adjacent receptor along the delay line, and at a location which is indicative of the velocity between the pair of receptors. The signals travelling along the delay lines are then collected, and the velocity is estimated overall via a winner-take-all mechanism. This device can measure velocity fairly accurately over a wide range and in both directions, although it does not provide spatial information. By contrast, the bugeye indicates the angular position of the occurrence of motion, and this is necessary to measure the distance between the sensor and objects in the environment.

Finally, the template model has been implemented by Sobey & Horridge (1990), using a CCD ("charge-coupled device") camera for data acquisition. Along a visual axis, the one-dimensional image is convolved with a spatial filter of the form [-1;3;-1], thus enhancing edges by effecting a form of lateral inhibition. Each new pixel is then subtracted from the previous measurement, and the result is thresholded. The results obtained are most encouraging, and are consistent with predictions (Horridge & Sobey, 1991).

The Sobey and Horridge implementation of the template model differs from that described in this chapter in three important respects. Firstly, instead of using a camera to feed data into a processor, the optical interface of the present device can be considered as forming an integral part of the system. Secondly, measurements of changes in contrast and the formation of the templates are accomplished on the chip itself. And thirdly, a lateral inhibition mechanism has not been incorporated, mainly because the analog detection stage would have been more complex, thus increasing design risks. However, it may be advantageous to implement lateral inhibition in a future generation of the device in order to increase noise immunity.

5. Summary

The bugeye is a novel implementation of a motion detection scheme that was described in Chapter 4. It draws upon biological principles to extract motion information from visual stimuli. The design exploits the combined advantages of analog and digital VLSI techniques, and is geared towards conveying readily usable information in real-time. The optical interface consists of a small "gradient index" (GRIN) lens glued onto the chip surface, and no focusing mechanism is required. The GRIN lens is used primarily to make the sensor mechanically compact, and thus well suited to being utilised by an autonomous control system. It should be pointed out that, in principle, any lens presenting similar optical characteristics could be utilised instead.

The main difference between the bugeye and other motion sensors lies in the manner in which the information is conveyed. Specifically, the directions and contrasts of moving edges are encoded digitally, immediately following the detection of the stimulus, and the implementation allows that information to be efficiently encoded (e.g., one bit for the direction of motion, another bit for the sign of the contrast change, and so on). Moreover, the information is stored in memory locations which correspond to the angular positions of moving stimuli, and hence, as will be shown in Chapters 7 and 8, the computation of velocity and the detection of looming may also be accomplished in real time.

CHAPTER 6 Experimental results

Hardware testing is utilised to explore the consequences of the design decisions of the first implementation of the bugeye. The testing apparatus and procedures are briefly described, followed by a presentation of preliminary results obtained with various stimuli, and recorded in real time. These results give guidance for improvement in the design of the analog detection stage. The data confirms the simulations of Chapter 4, albeit in an unexpected manner, and velocity estimates obtained from real-time data are reasonably consistent with stimulus velocity.

1. Experimental set-up and procedures

This section presents the manner in which data obtained from the bugeye is recorded. A more detailed description of the chip's interfacing requirements and of the experimental apparatus can be found in Appendix B.

1.1 Communicating with the bugeye

As described in Chapter 5, the accessible components of the chip are the memories and the processor. The template and save memories can be directly written into and read from without interfering with other components, while the change detection circuitry can only be tested indirectly by reading the save RAM after sampling. The processor cannot be tested without its own memory, as the processor outputs are not connected to the data bus.

The chip communicates with the outside world via a bidirectional data bus (4 bits) and an address bus (8 bits). The timing and control requirements are those of a standard static mem-

ory, i.e., a "chip select" signal activates the chip's input-output (I/O) functions, while a "read-write" signal indicates the direction of the data transfer. For practical purposes, the internal control registers correspond to a specific section of the address space, and contain the processor parameters and the mode of operation (Figure 6.1). The various memories correspond to another area, and are individually selected according to the contents of the mode register.

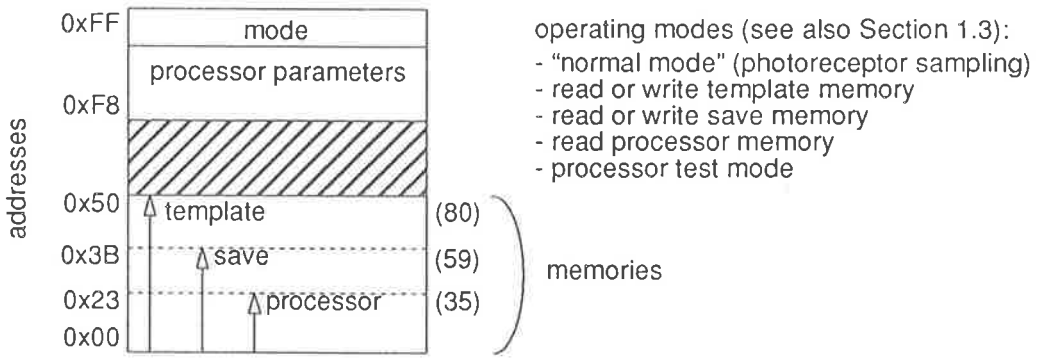


Figure 6.1 Bugeye memory map

The bugeye operating mode is set by writing a 4-bit datum to the mode register, and the device's memories can only be read from or written into after the operating mode has been set. The internal control ignores external requests when in "detection mode", and for a number of cycles following the falling edge of the sampling clock, i.e., while the templates are being formed and stored into the save RAM (see Section 2 of Chapter 5). The chip provides a status signal ("bus available") to indicate whether its memories can be accessed.

Other signals to be provided to the chip are the sampling signal, digital clock, reset, and biasing currents for the contrast change detectors. The signals used for testing the processor and various analog test structures are described in Appendix B (see also Moini, 1993).

1.2 Testing apparatus

For experimental purposes, the bugeye is mounted on a test board which is connected to a computer (Figure 6.2). The test board contains adjustable current generators for biasing the chip's analog circuitry, a digital clock generator, an asynchronous reset, and an interface to the parallel input-output port (I/O) of the computer.

The computer controls all the accesses to the chip, as well as the generation of the sampling signal. Therefore, by reading the memories in between sampling pulses, the results can be displayed in real-time and stored for further analysis.

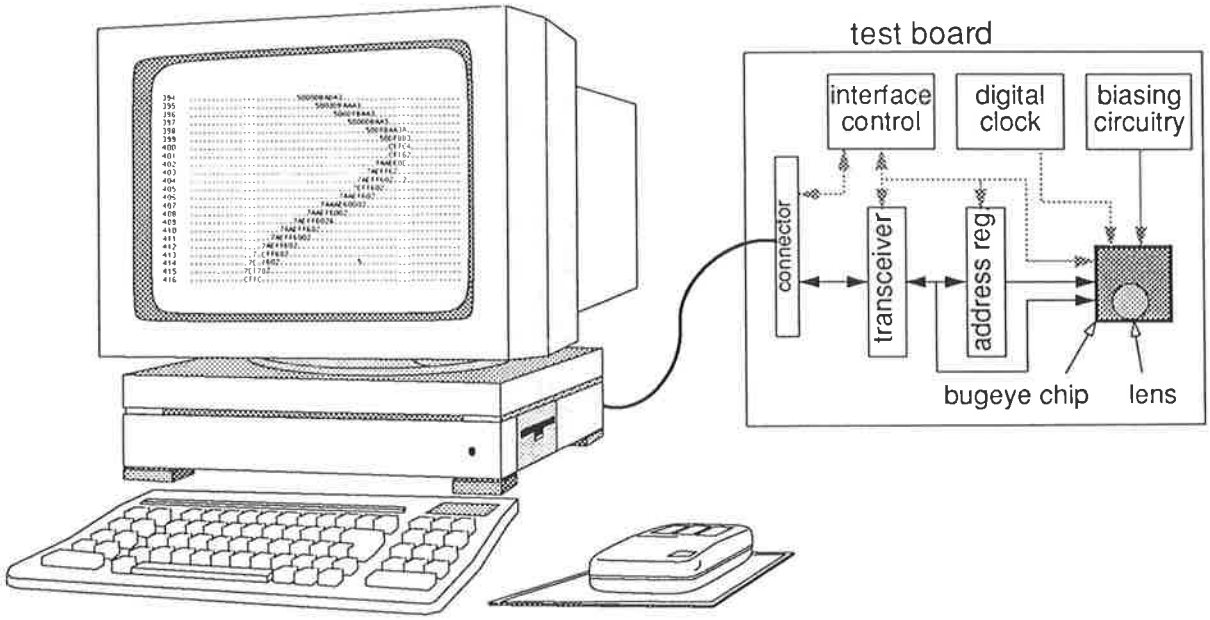


Figure 6.2 Experimental set-up

1.3 Interface

The interface comprises a bidirectional registered transceiver for data transfers and an address register, both controlled by a state machine implemented with standard programmable logic devices, or PLDs (Figure 6.2). While the control state machine is idle, the interface monitors a "request" signal from the computer's parallel port. If the request becomes active, the operation specified by control signals, which are also generated by the computer, is carried out. Upon completion of the operation, the interface generates an "acknowledge" signal, drives data to the computer if appropriate, and the state machine reverts to its idle state when the request has been de-activated. The handshaking protocol between the computer and the interface is therefore asynchronous (i.e., control signals remain active until being acknowledged) as different clocks are used at either end.

Four types of operations can be initiated from the computer:

- **read** (from either the template, save, or processor RAM)
- **write** (to the control registers, the template or save RAM)
- **processor test** (see Appendix B)
- **sampling signal** (set high or low)

The processor RAM can only be written into through the processor itself. Since in normal mode the processor control is reset by the sampling signal, and the data and addresses are synchronised with the template clock, **processor test** consists of feeding data and addresses to the processor externally, along with a test signal which replaces the sampling signal.

1.4 Software procedures

The computer communicates to its parallel port via two 8-bit registers, namely, data register and control register. Therefore, a typical operation involves loading data into the data register and setting the appropriate request and control bits in the status register. The latter is then continuously polled until a specific "acknowledge" bit is set, indicating that the test board has completed the operation, and in case of a "read" operation, a 4-bit datum is present in the data register. Note that "write" operations require two requests, as the data register is not wide enough to contain simultaneously an address and some data.

Details concerning initialisation of the bugeye, addressing, data transfers, and so on, are included inside function calls. The software allows the chip to be tested in different ways, simply by entering a single command. For instance, a particular memory can be initialised with a user-defined data table and read in its entirety, while individual memory locations can also be written into and read from. The test used in section 2 consists of the following steps, each comprising several elementary operations:

- **Initialise template memory** (with user-defined table)
  - set template RAM mode by writing to the mode register
  - write 81 times to load the template RAM with all possible templates
- **Standard test** (infinite loop)
  - set detection mode
  - set sample signal high, wait for preset delay, and set sample signal low
  - set save RAM mode
  - read 60 times, store and display

In the standard test, the results are displayed on the screen and stored in real-time in the same format as they are being acquired from the chip. Therefore the templates, which are 4-bit quantities, are printed as hexadecimal numbers, except for the "no motion" template which is printed as a dot in order to be clearly distinguished from other templates. The 60 templates are read sequentially from the save RAM in between sampling, and are printed as a single line of characters, thus the position in the line corresponds to the angular position. A counter is incremented at each sampling instant and is printed at the start of each line, as shown in Figure 6.3.

Figure 6.3 Result format and display

2. Preliminary testing results

2.1 Objectives and methods

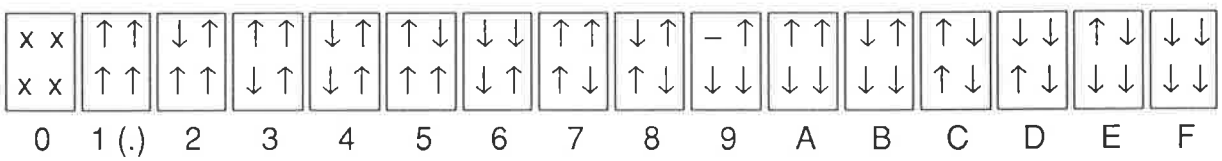
Due to the variations inherent in the VLSI fabrication process, it is difficult to predict accurately the behaviour of the analog circuitry, and to assess the influence of process parameters over the sensitivity of the detection stage. However, in order to test the detection stage, both the template and save memories must be fully operational, and hence each memory location is written into and read from with a variety of test patterns. If the data appears to be uncorrupted, the lens is placed on top of the row of photo-detectors, and the bias circuits are adjusted until responses to a moving test pattern seem to be consistent, and the occurrences of spurious templates are only occasional. The standard test described in the previous section is then carried out with different moving stimuli (see Appendix B for a description of the processor test).

The criteria used to establish whether a chip functions properly under normal lighting conditions are the following: (i) the responses read from the save RAM are consistent with the motion presented to the chip, (ii) motion is detected up to at least three metres away from the lens, and (iii) motion is detected within an aperture of approximately 60 degrees, allowing for the lens to be slightly off-centre.

2.2 Initial results and interpretation

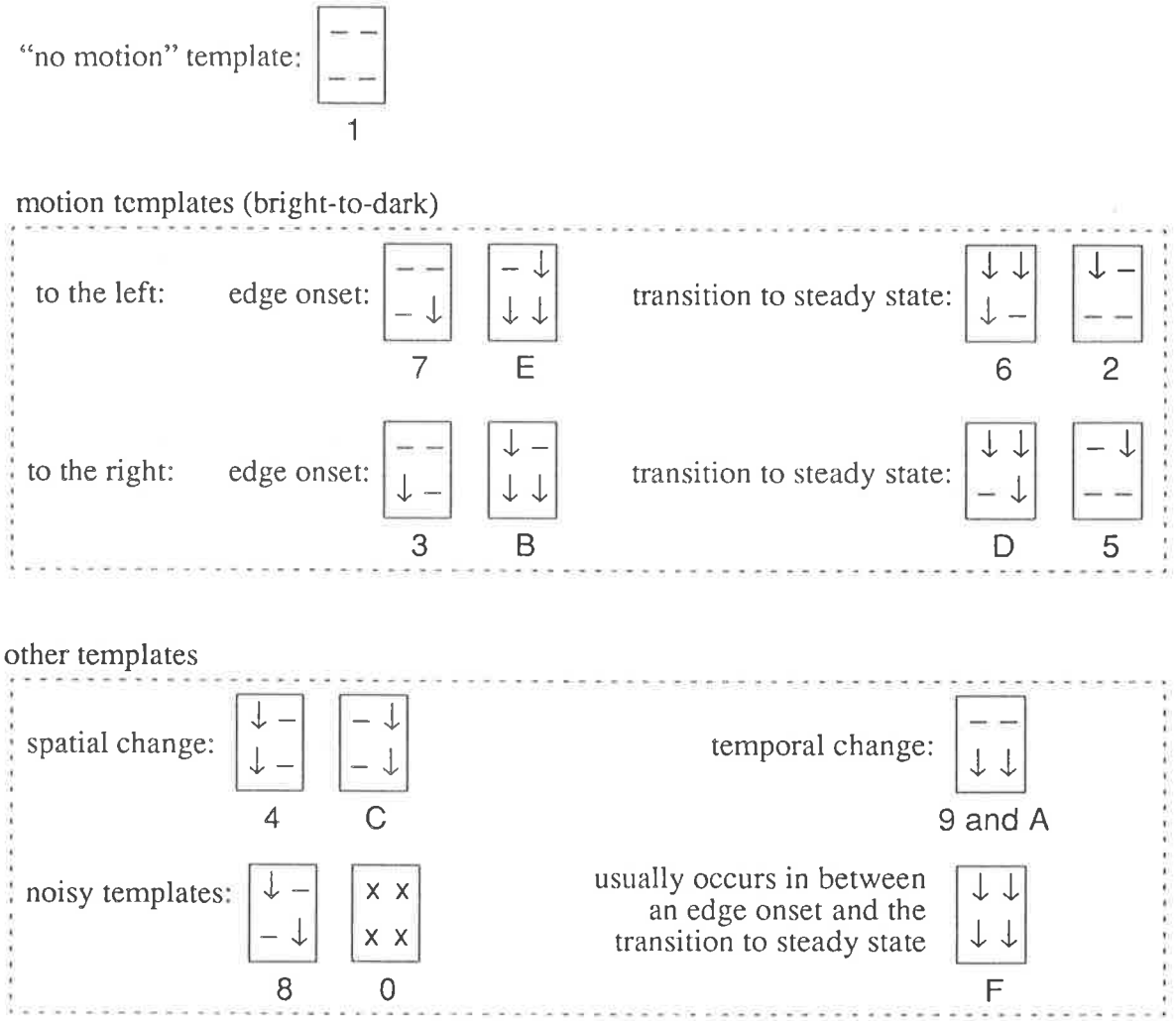
Provided that both the template and save memories appear to be operational, the differences between successful chips reside mainly in the levels at which the biasing should be set for optimal performance. In other words, once the biasing levels have been adjusted, sensitivity to motion and light conditions does not seem to vary significantly from one chip to another. Therefore, the fabrication process should be considered to be adequate, at least for prototyping.

Testing did, however, reveal a problem with the analog circuitry, as an unexpected template occurred in the absence of a moving stimulus, and dark to bright changes in contrast elicited virtually no response (see Moini, 1993). Therefore, the "ideal" template encoding employed for the simulations of Chapter 4 could not be utilised experimentally. Instead, through a process of elimination consisting of determining which template memory locations appeared to be the most frequently addressed by the detection stage, the following encoding was devised:



NB: encoding from 1 to F is arbitrary - 0 encodes all the other templates
template 1 occurs when no motion is present, and is printed as a dot

Compared with the expected templates of Chapter 4, those that are obtained experimentally could be interpreted differently by replacing increases in contrast with no change, yielding:



The occurrences of the templates is illustrated in Figure 6.4, which is an example of the chip's output when two dark strips are moving in opposite directions.

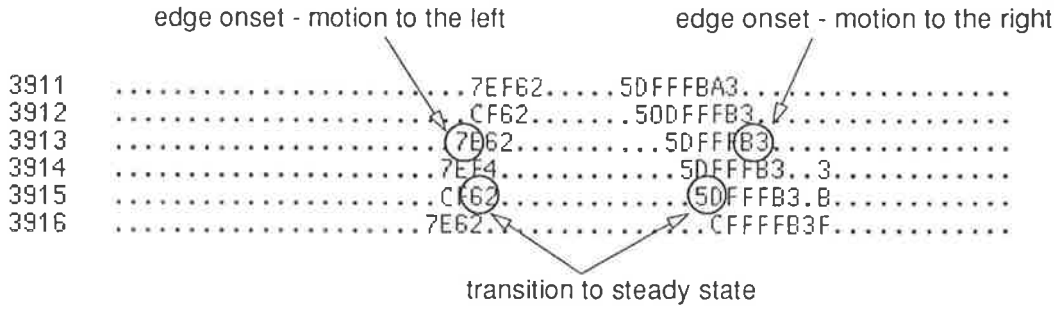


Figure 6.4 Left and right motion (bright-to-dark contrast changes)

Hence, the bugeye responds to motion in both directions, but only when bright-to-dark contrast changes occur. Based on the experimental evidence, Moini (private communication) concluded that mismatches between components of the active resistor in the feedback path of the contrast change detectors were responsible. Figure 6.5 depicts the relevant portion of the circuit (see also Chapter 5), along with the amplifier's voltage characteristics.

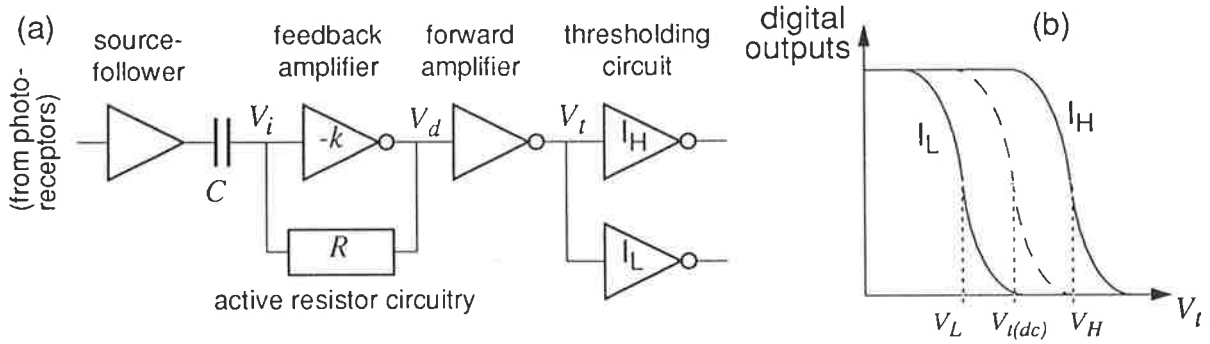


Figure 6.5 (a) Section of contrast change detector circuitry, (b) Voltage characteristics

Recall from Chapter 5 that under DC conditions (i.e., in the absence of motion), the input and output voltages (V_i and V_d) of the feedback amplifier should be equal, provided that the resistive element is ideal. However, mismatches due to the fabrication process may cause a voltage drop of a few millivolts across the active resistor circuitry. As the gain of the forward amplifier is large (greater than 50), the voltage drop shifts the DC voltage $V_{t(dc)}$ towards V_H, as if an increase in contrast brought about by a dark to bright transition had occurred. As a consequence, the change detector responds well to a decrease in contrast, but little or not at all to an increase.

A possible solution is to replace the source-follower buffer, currently located before the differentiator circuit, with an adaptive amplifier such as that used by Delbrück & Mead (1991). The voltage variations due to changes in light intensity are then amplified before being differentiated, while the voltage drop across the active resistor remains in the order of a few millivolts. Moreover, if the gain of the adaptive amplifier is sufficient, the large-gain forward amplifier is no longer required, and hence the effect of mismatches would be greatly reduced. This scheme has been described by Moini (1994), who also suggested ways in which the characteristics of the thresholding inverters could be electrically adjusted in order to vary the sensitivity of the circuit.

Another problem is aliasing under artificial lighting conditions, particularly with neon lamps, and is most likely due to the cut-off frequency of the analog circuitry. With natural light, however, most of the chips respond well for a reasonable range of background intensities, roughly speaking from dimly lit conditions to normal daylight, which translates to photo-currents in the order of 10 pico-amperes to 100 nano-amperes.

Finally, the sampling rate of the photo-detectors was found to be slightly irregular and slower than the 100 Hz for which the circuitry is designed. Software control of the sampling signal facilitates synchronisation between the computer and the bugeye, but at the expense of reduced control over the sampling rate. The irregularity seems to be due mainly to software overheads incurred by displaying data and printing data into files, and to the fairly slow transfer rate of the computer's parallel port. In addition, the sampling rate appears to drop in the course of experiments during which data is stored (see velocity measurements in Section 3.3).

2.3 Examples

Following are examples of data generated in real time by the bugeye for different types of stimuli, and under normal daylight conditions. In Figure 6.6, a person is walking in front of the sensor, a few metres away. Motion up to 8 metres can be detected under ideal lighting conditions, and if the difference in contrast between the person's clothes and the background is significant. Figure 6.7 shows the response of the sensor to the motion of a small object, such as a pen, which is being waved in front of the detector array, first slowly and then rapidly. The object and person motions are combined in Figure 6.8, and finally, the looming experiment of Figure 6.9 consists of approaching the lens with an object in front of a background of opposite contrast.

```
1013  ..........................................7EF62.............
1014  .........................................CF62..............
1015  ........................................7EF4...............
1016  .......................................CFF4...............
1017  ......................................7EF62...............
1018  .....................................7EF62................
1019  ....................................7EF62.................
1020  ...................................7EF62..................
1021  ..................................CFF4....................
1022  .................................7EF62....................
1023  ................................7EF62.....................
1024  ...............................CFF4.......................
1025  ..............................7EF62.......................
1026  .............................EF62........................
1027  ............................7.FF4........................
1028  ...........................CFF62........................
1029  ..........................7E.62........................
1030  .........................7EFF4........................
1031  ........................CFFC2........................
1032  .......................CFF4F........................
1033  ......................7EF62F........................
1034  .....................7EF62.6........................
1035  ....................7EF62..2........................
1036  ...................7EF62.............................
1037  ..................CFF4...............................
1038  .................7EF62...............................
1039  ................CF62.................................
```

Figure 6.6 Person walking from right to left approximately 3 metres away

```
1727  ...........................7EF62.............................
1728  .........................7AEF62F.............................
1729  .......................7AEF602.0.............................
1730  .....................7AEF602.................................
1731  ....................7EF602...................................
1732  ...................7EF62.....................................
1733  ...................CF62......................................
1734  ...................5D4.......................................
1735  ....................52.7A3...................................
1736  .......................CFB3..................................
1737  .......................5DFBA3................................
1738  ........................500DBAAB.............................
1739  ...........................50DFBA3...........................
1740  .............................500DBAA3........................
1741  .................................500027AAAA3.................
1742  ......................................500DFBA3...............
1743  .........................................50DFBA3.............
1744  ...........................................5DFFBF3...........
1745  ............................................CFFFF4...........
1746  ...........................................7EFFF4............
1747  ..........................................7EF6006............
1748  ......................................7AAE602................
1749  ....................................7AEF602..................
1750  .................................7AAE6002....................
1751  ..............................7.FF602........................
1752  ............................7AEF602..........................
1753  .........................7AAE60E2............................
1754  .......................7AEF602.0.............................
1755  .....................7AEF602.................................
1756  ...................7AEF602...................................
1757  ...................CFF62.....................................
1758  ...................5DFB3.....................................
1759  ....................50DBA3...................................
1760  ......................5DFBA3.................................
```

faster motion of the same object:

```
1940  ............500027AAAA3......................................
1941  .................5000027AAAA3................................
1942  .......................5000027A2AAA3.........................
1943  .............................50000027AAAA3...................
1944  .......................................5000027AAAAA3.........
1945  ...............................................C.FFFF4.......
1946  ..............................................7EFFFF62.......
1947  ...........................................7AAEF60002........
1948  .......................................7AAA000000............
1949  .................................7AAA350002..................
1950  ............................7AAA350002.......................
1951  .......................7AAA350052............................
1952  ...................7AAA00002.................................
1953  ...................50DFBA3...................................
1954  .....................50008AAAA3A.............................
1955  ...........................500D0BAA3.........................
1956  ..............................5000027AAA3....................
1957  ....................................50008AAAAA3..............
1958  ........................................500DFFBFA3...........
1959  ..........................................7EFF6F02...........
1960  ......................................7AAA00002..............
1961  ................................7AEF602......................
1962  ...........................7AAA350002........................
```

Figure 6.7 Small object (pen) waved in front of the bugeye, approximately 30 cm away

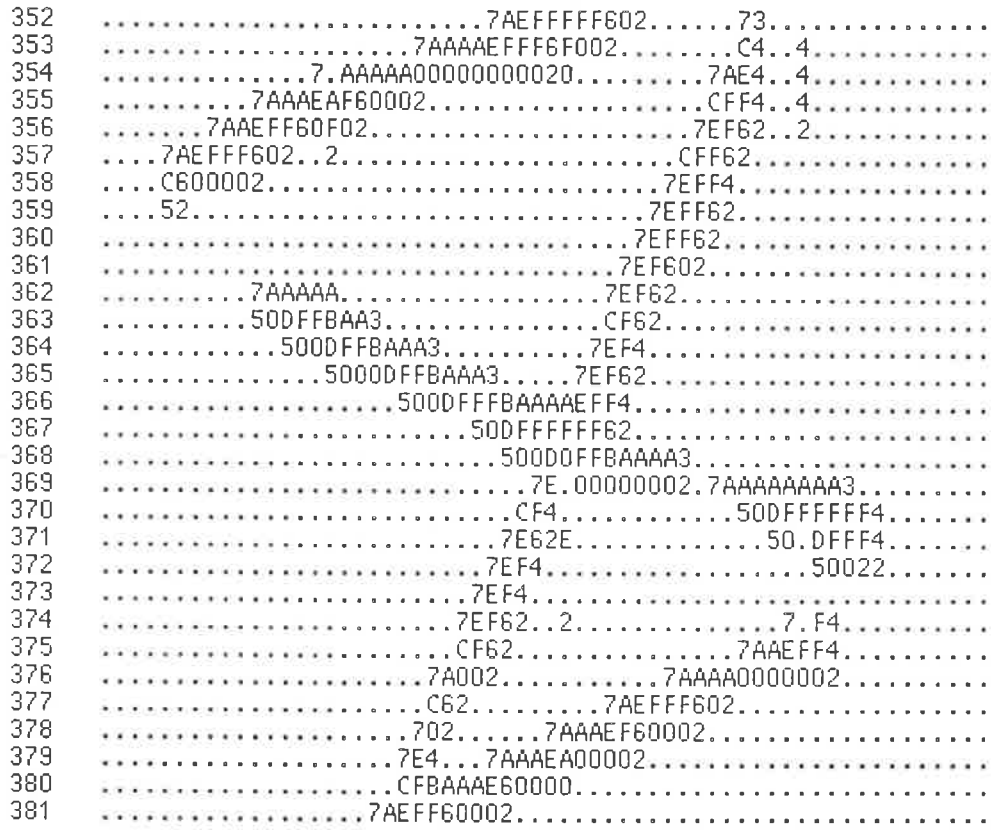


Figure 6.8 Person walking while the pen is waved in front of the bugeye

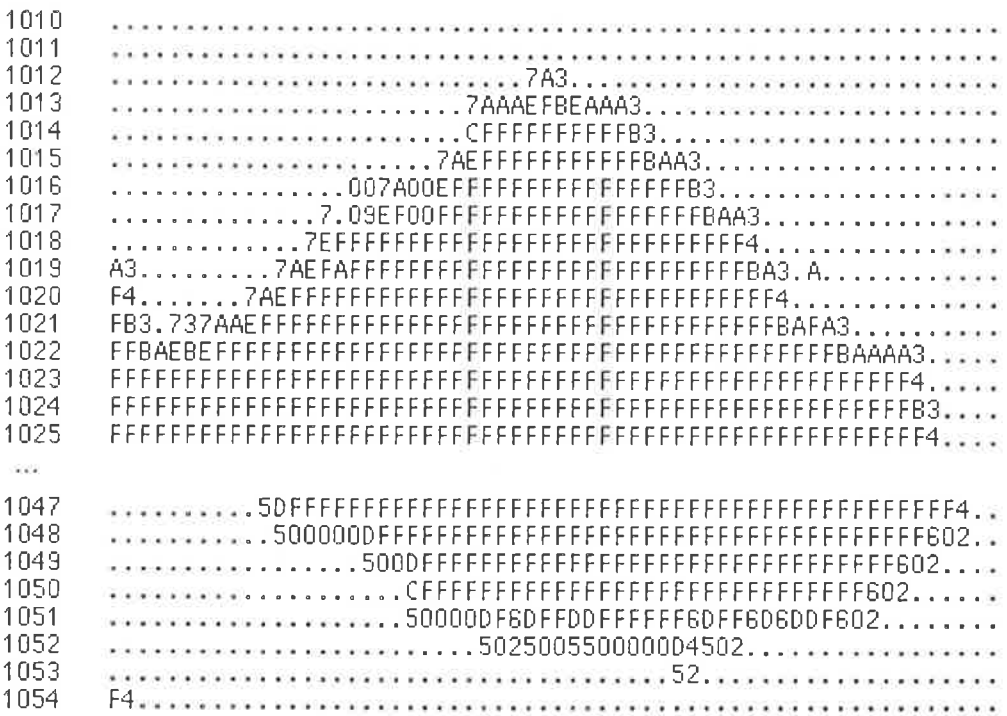


Figure 6.9 Looming (dark object approaching the bugeye)

For the looming experiment (Figure 6.9), a dark object approaches the bugeye almost until contact, and stops. In principle, two distinct edges separated by "no motion" templates should have been detected, as shown in the theoretical example of Figure 6.10. At each sampling time the indication of motion to the left (templates 7 and E) is accompanied by the corresponding "transition to steady state" templates (6 and 2), and likewise for rightward motion (3 and B followed by the "transition to steady state" templates D and 5).

```
1010  ..........................................................
1011  ..........................................................
1012  ...........................7EB3...........................
1013  ........................7EF625DFB3........................
1014  ......................7EF62....5DFB3......................
1015  ...................7EF62..........5DFB3...................
1016  .................7EF62...............5DFB3................
```

Figure 6.10 Theoretical response to looming object

In the experiment of Figure 6.9, however, the "transition to steady state" templates are only produced once the looming object has ceased moving, and are eventually followed by the occurrence of "no motion" templates (bottom part of Figure 6.9). Thus, it appears that the bugeye requires some time to accommodate from normal illumination to dimly lit conditions, as the dark object ends up covering the entire visual field of the lens.

A possible explanation for this phenomenon lies with the problem mentioned earlier, namely, that the resting state of the contrast change detection circuitry is at saturation, and is encoded as such by the "no motion" template. Thus, following a sudden decrease in contrast, the forward amplifiers of the detectors (see Figure 6.5a) require a longer time to reach saturation than if the resting point were in the middle of the linear range of the amplifiers. The recovery time of the detectors is reflected by the occurrences of template F, which indicates decreases in contrast. The accommodation phenomenon can also be illustrated by simultaneously moving two dark strips of different widths in front of the bugeye, and at the same velocity (Figure 6.11).

```
135   ..........................7EFFFFFF62..7E62..................
136   .........................7EFFFFFF62..7E62...................
137   .......................7AEFFFF6002.7A002....................
138   ......................7EFFFFF62...7E62......................
139   .......... (wide strip)...7EFFFFF62...7E62......................
140   ....................7AEFFFF602.07AE62.......................
141   ...................7EFFFF602....EF62........................
142   ...................7EFFFF602....EF62........................
143   .................7AEFFF602....7.602.........................
```

Figure 6.11 Response to two dark strips

3. Velocity measurements

3.1 Method

As discussed in Chapter 4, all the possible spatio-temporal combinations of contrast changes, or templates, can be grouped into four categories: (i) un-interpretable or useless, (ii) no change [one template only], (iii) indicating temporal or spatial change, and (iv) indicating motion, i.e., temporal *and* spatial change. Templates belonging to the fourth category, referred to as "motion templates", provide directional information, and their displacements in time are indicative of velocity.

The experimental results of the previous section show that a moving edge fairly consistently causes the same motion templates to occur at subsequent time steps, and at positions corresponding to the displacement of the edge relative to the detector, as illustrated in Figure 6.12.

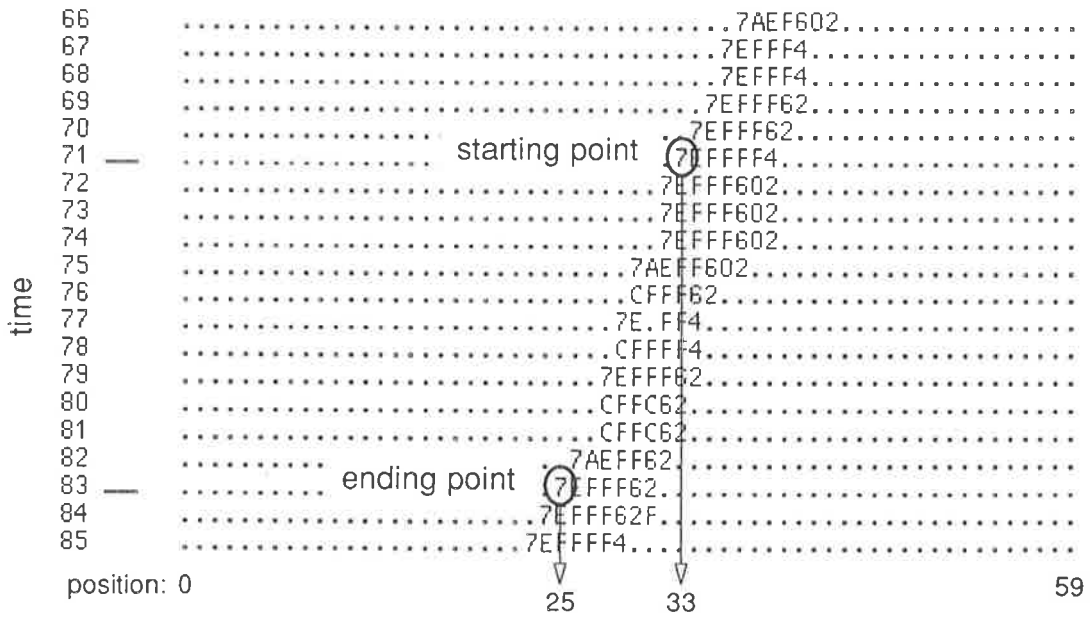


Figure 6.12 Example of velocity estimation

The slope of the line which most closely fits the occurrences of template 7 between time steps 71 and 83 in Figure 6.12 yields the average angular velocity during that time. Assuming that a displacement of one position corresponds to one degree of visual angle, the angular velocity can be approximated by calculating the ratio of the ending point position (25) minus the starting point position (33), to the number of time steps (12). This results in –0.67 degrees per sampling period, where the minus sign indicates motion to the left. Note that while the templates 7 and E on the left-hand-side of the track indicate a bright-to-dark edge, those on the right-hand side (6 and 2) indicate the transition to steady state which occurs after the

bright-to-dark edge. In principle, "transition to steady state" templates should indicate the same velocity as those marking the onset of the edge, although, if the spatial separation between edges following one another is small, the formation of "transition to steady state" templates may be compromised as successive contrast changes could overlap (see Chapter 4).

3.2 Measurement uncertainty and optical correction factor

A template is composed of the responses of two adjacent detectors whose receptive fields overlap, and hence the instantaneous position of a moving stimulus may correspond to either detector. Therefore, the velocity estimate defined previously can be expressed as:

$$\text{velocity} = \frac{(\text{end position} \pm 1) - (\text{start position} \pm 1)}{\text{number of time steps in between}}$$

Thus the relative template position error increases as the displacement decreases. Referring to the example of Figure 6.12, the calculated minimum velocity would be –0.5 degrees per sampling period and the maximum –0.83, and hence the error of the median velocity (–0.67) is approximately ± 25 percent.

Some adjustment may be required due to the optical characteristics of the lens, which has an aperture of 72 degrees and a diameter of 1.8 millimetres (see Chapter 5). On average, one degree would therefore correspond to 25 microns, which is the measure used for the separation between adjacent photo-receptors on the chip surface. However, the distance-per-degree ratio is a function of the distance from the centre of the lens, and decreases with eccentricity. Measurements effected by Andrew Blanksby (private communication) show that one degree corresponds to approximately 28 microns in the centre of the lens, and to 21 microns towards the periphery.

Therefore, in principle, a correction factor should be introduced when measuring velocity. Absolute template positions can be converted into distances relative to the centre of the lens, and then mapped into angular positions. The velocity originally estimated from the ratio of template displacement to the number of time steps is then multiplied by the correction factor, which is defined as the ratio of the angular displacement, to the difference between the template's recorded positions. As the distance-per-degree ratio decreases with eccentricity, the correction factor would vary between 0.9 ($\approx 25 \div 28$) and 1.2 ($\approx 25 \div 21$) approximately. The angle to distance mapping is shown in Figure 6.13.

As an example, consider the case where velocity is measured over 10 template positions and towards the centre of the visual field, between absolute positions 30 and 40. The mapping from the corresponding portion of the receptor array (i.e., 250 microns for 10 templates) to the visual angle yields approximately 9 degrees, and hence a correction factor of 0.9.

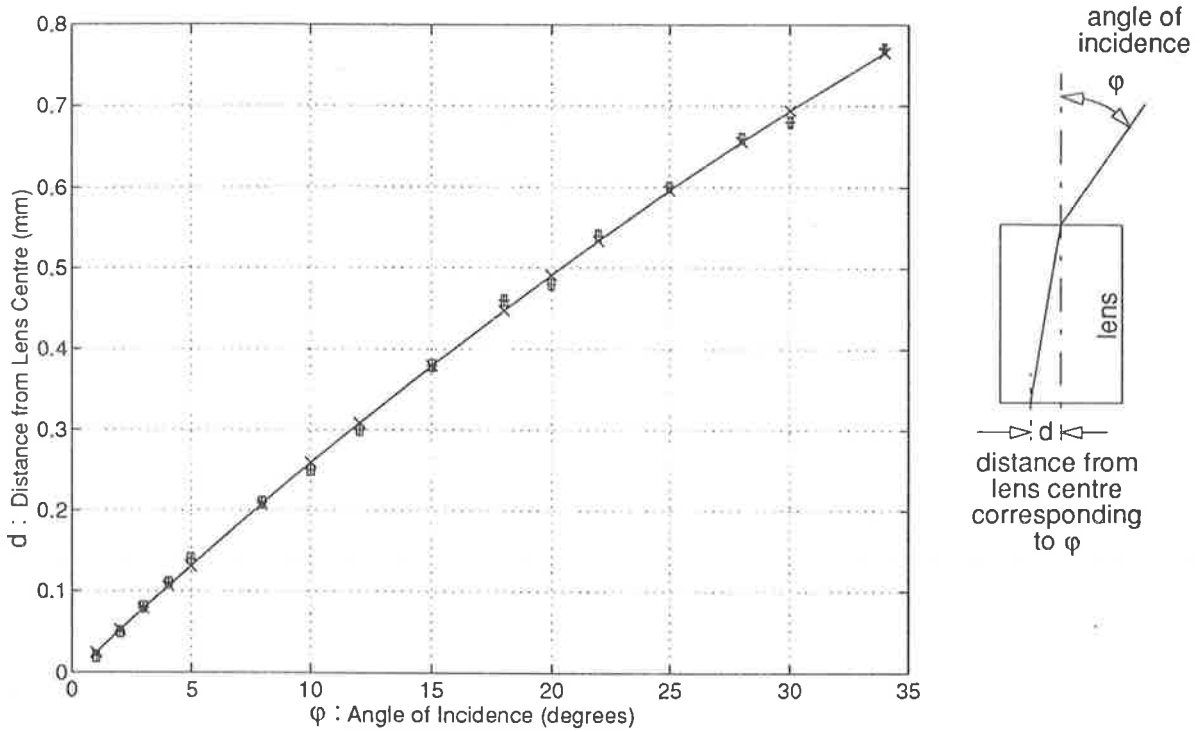


Figure 6.13 Lens angle to distance mapping

3.3 Velocity measurements

The response of the bugeye to moving objects can be measured under controlled conditions using the apparatus shown in Figure 6.14. The apparatus comprises a cylinder (or drum) whose rotation is controlled by a step motor, and the angular velocity of the drum can be measured fairly accurately with a frequency meter. The inside of the drum is white, except for a black strip placed perpendicularly to the drum's diameter. The test board, which comprises the bugeye and its interface to the computer, is placed in such a way that the lens is situated in the centre of the drum, whose dimensions are 60 cm in diameter and 20 cm deep.

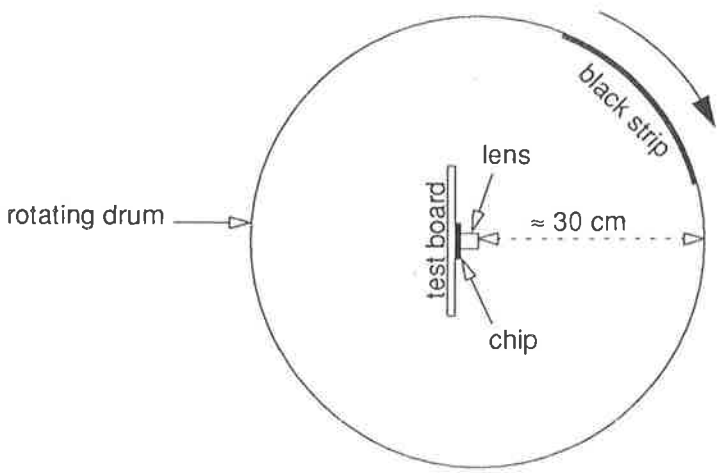


Figure 6.14 Velocity measurement apparatus (top view)

The outputs of the interpretation stage of the bugeye are recorded in a file on the computer for a number of rotations of the drum at constant angular velocity. The velocity is then estimated from the recording as in the example of Figure 6.12, i.e., by selecting two of the motion sensitive templates which are produced each time the dark strip passes in front of the chip. For consistency, templates 20 positions apart are chosen, and on either side of the centre of the receptor array. According to the mapping of Figure 6.13, the template position error is ±10 percent, and the corrected visual angle is 18 degrees (correction factor = 0.9).

Hence, for a sampling period of 15.5 milliseconds, and if n_s is the number of time steps in between the selected templates, the measured velocity is given by:

$$\text{velocity} = \frac{18}{n_s \times 0.0155} \quad \text{degrees per second}$$

The results for various angular velocities of the drum are summarised in Table 6.1.

real velocity (degrees per second)	measured velocity (degrees per second)		
	minimum	average	maximum
60	61 (+ 2%)	**68 (+ 13%)**	77 (+ 28%)
80	83 (+ 4%)	**95 (+ 19%)**	105 (+ 31%)
85	89 (+ 5%)	**104 (+ 22%)**	116 (+ 25%)
115	116 (+ 1%)	**139 (+ 21%)**	166 (+ 44%)
155	145 (- 6%)	**178 (+ 15%)**	232 (+ 50%)

(percentages in brackets show the differences between measured and real velocities)

Table 6.1 Comparison between real and measured velocities

It is worth noting that the measured angular velocities are usually greater than the real velocities, and the average error varies from 13 to 22 percent, which is higher than the 10 percent template position error. This fairly consistent discrepancy may be attributed to a decreasing sampling rate (i.e., an increasing sampling period), which is software-controlled. The sampling rate is approximately 64.5 Hz when a recording begins (i.e., a period of 15.5 milliseconds). In the course of an experiment, the template responses are read from the bugeye and stored by the computer for further analysis. As the size of the recording increases, dynamic storage allocation overheads occur as the computer transfers blocks of data into its long-term memory. The template position error is thus compounded by a decrease in the sampling rate, which would result in measured velocities being higher than expected.

Note that the decreasing sampling rate tends to have a greater effect on the measurements as the velocity increases, since the number of time steps between the templates selected for the measurements is smaller than at low velocities.

4. Discussion and summary

Variations of the VLSI process which was used to fabricate the chip have resulted in a major difficulty, namely that dark to bright changes in contrast are not reliably detected. The cause of this difficulty, however, appears to be confined to mismatches between components of the active resistor circuitry. It is believed that incorporating an adaptive amplifier before the change detector circuit should be sufficient to decrease the chip's sensitivity to process parameters, to the point where both increases and decreases in contrast may be detected equally.

Operating the chip under software control greatly facilitates testing procedures, but at the cost of introducing software overheads which adversely affect the performance of the bugeye. A solution would be to generate the sampling signal and to perform data acquisition operations on the test board itself, in order to ensure a constant sampling rate. If the bugeye were to be used as a sensor by an autonomous control system, it is worth pointing out that this solution should preferably be adopted.

In spite of the afore-mentioned difficulties, there remain several very encouraging aspects revealed by real-time testing. Firstly, the velocity of objects presenting a bright-to-dark contrast with their background can be measured fairly reliably, thus lending support to the tracking algorithms presented in Chapter 7. Secondly, even though only a dozen chips were fabricated, the testing results do not seem to vary significantly. In other words, the variations in performance between chips, in terms of detection range and light sensitivity, appear to be quite small, in spite of inconsistencies in the analog circuits, thus underlining the robustness of the template model.

CHAPTER 7 Tracking algorithms

The experimental results presented in the previous chapter show that motion detected by the bugeye is indicative of velocity. This chapter presents two tracking algorithms and their hardware implementations for a future generation of the device. The first algorithm back-tracks motion paths by examining a time-space image, and the second predicts future motion on the basis of past events.

1. Underlying principles

This chapter is concerned with the first level of processing of the bugeye's output, namely estimating velocities of moving edges. Further processing methods are presented in Chapter 8. Processing is done in software, using the experimental set-up described in Chapter 6.

The tracking algorithms implement the velocity estimation method presented in Chapter 6, which consists of tracking the occurrences of motion templates, i.e., those indicating spatial and temporal change, and noting their positions in space and time. The average angular velocity can then be estimated as the ratio of the displacement between the first and last occurrences, to the number of sampling steps in between. The accuracy of the measurement depends on the following:

- ***The choice of target templates***
  In principle, any motion sensitive template, indicating the onset of a moving edge or the transition to steady state, can be used as a target. Those which indicate the onset of an edge should be the preferred candidates, as they are also indicative of the edge's position and may therefore be used for range estimation (see chapters 3 and 8).

- *The "memory" of the tracking*
  Memory capacity is a function of hardware constraints, and hence a small size would be desirable. More importantly, variations in angular velocity, which may be critical for collision avoidance (see Chapter 8), may not be noticeable if velocity is estimated by averaging displacements over a relatively long period.
- *Handling of track cross-overs*
  The motion of one object can mask that of another object, thus some provisions should be made for recovering tracks that are temporarily interrupted. It is worth noting that there is a trade-off between track recovery and the tracking memory.

In this chapter, two tracking methods are described. The first, proposed by Nguyen *et al.* (1993b), consists of back-tracking so-called "motion paths", and therefore requires full storage of past history, while the second (Yakovleff *et al.*, 1994) attempts to predict future occurrences of motion templates based on past events, or forward tracking. Both algorithms have been implemented in software on the computer, using the current experimental set-up with the front-end providing data in real time (see Chapter 6). Operating parameters can be varied, with a view to mapping the optimal versions of the algorithms in hardware in a future version of the chip.

2. Stair-step tracking

The stair-step tracking algorithm (Nguyen *et al.*, 1993b) consists of analysing a stored time-space image ("image memory"), where each row is composed of the template responses obtained at a given sampling time, and the position within the row corresponds to the visual angle. The number of rows determines the tracking depth in time.

2.1 Time-space image analysis

The analysis produces motion paths which are defined by the sequences of positions in time and space of directionally sensitive templates. Each sequence reflects the detected angular motion of an edge, whose velocity is provided by the slope of the line connecting the beginning to the end of the motion path.

At a given sampling time, each motion sensitive template may indicate the presence of a motion path. The image memory is searched in an attempt to reconstruct the positional changes of the motion sensitive template's previous occurrences. Figure 7.1 shows three paths starting from points A_1, A_2, A_3, and finishing at points B_1, B_2, B_3 respectively. All the paths end at the earliest template, i.e., the last template detected while back-tracking, usually before the boundaries of the memory have been reached.

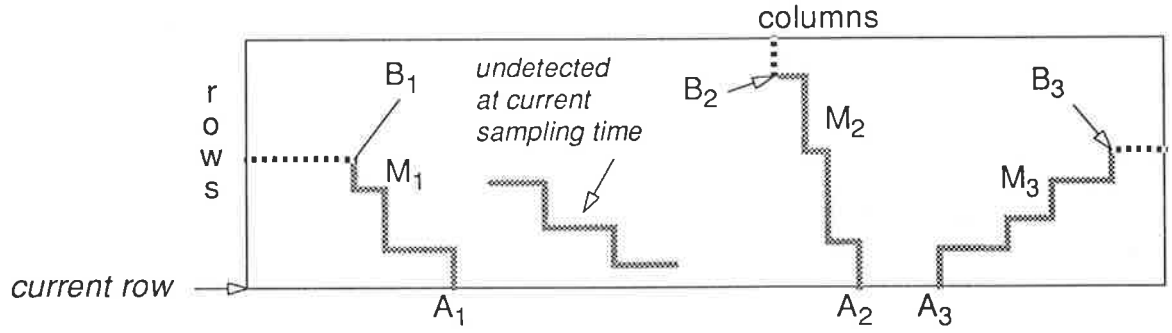


Figure 7.1 Motion paths in the time-space image memory

If the time step corresponds to a row of the image memory and the position within the row to a column of the memory, a motion path can be formally described as the sequence:

$$M = R_0, C_0, R_1, C_1, R_2, C_2, \ldots, R_n, C_n$$

The addresses of the alternating motion sensitive and conjugate templates are, starting from the first motion sensitive template: (R_0, C_0), (R_1, C_0), (R_1, C_1), (R_2, C_1) and so on. The average velocity, which may not be the precise value, is given by the ratio: $(R_0 - R_n)/(C_0 - C_n)$.

The search algorithm is based on the observation that motion sensitive templates occur in pairs which can be classified as *motion conjugate* (see Chapter 4). These templates should, in principle, occur simultaneously as illustrated in Figure 7.2, where a bright edge is moving to the right. The motion path shown in the example contains the points: (t_6, p_4), (t_3, p_4), (t_3, p_1).

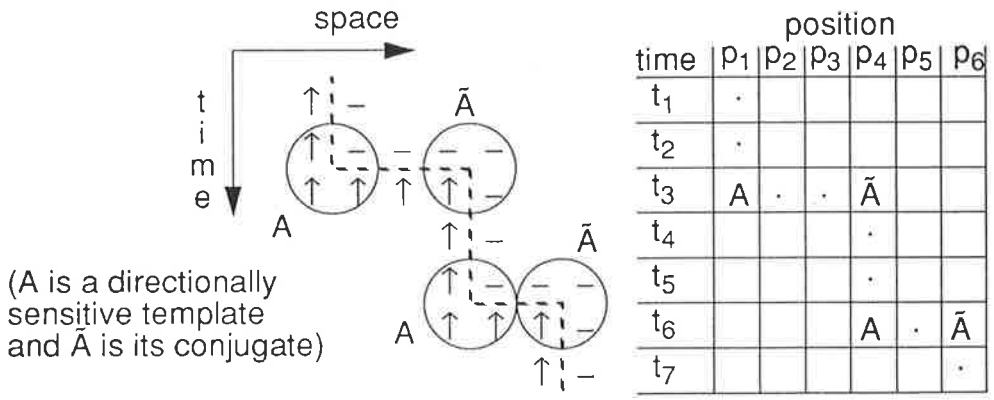


Figure 7.2 Template occurrences on a motion path (bright edge moving to the right)

When a directional template is detected, the sequence of templates can be examined to find occurrences of that template and its conjugate at earlier times, yielding the motion path. In Figure 7.2, when template A is detected at time t_6 and in position p_4, the motion path is determined by searching first vertically (i.e., temporally) for the conjugate template Ã which occurred in the same position but at time t_3, then horizontally (i.e., spatially) and to the left since A indicates motion to the right, until A is found at position p_1, and so on.

The addressing modes of the memory correspond to sections of the algorithm:

- ***Initial loading:*** following the falling edge of the sampling signal, the interpretation stage outputs are stored in consecutive locations of the image memory at the current row R_c.
- ***Search mode:*** after ***initial loading***, the current row is read sequentially until a motion sensitive template is found. Its column address is then stored (C_{start}), the conjugate template is fetched, and ***tracking mode*** begins.
- ***Tracking mode:*** the path is followed by searching for the previous occurrences of alternating motion sensitive and conjugate templates. Two end-of-track registers, R_e and C_e, are initialised with the row and column addresses R_c and C_{start}. In column C_{start} , the previous rows are read consecutively until the conjugate template is found. The row address is stored (R_e), and the memory is then searched in row R_e by either incrementing or decrementing the column address, depending on the direction of motion. If the motion sensitive template is found, the column address is stored (C_e), and the conjugate template is then searched in column C_e, at previous rows, and so on. This procedure is iterated until either all the rows have been accessed, or the first or last columns of the image memory have been reached. ***Search mode*** then resumes at address (R_c , C_{start} +1).

Figure 7.3 depicts the search and tracking modes flow diagram. When the search concludes, registers R_e and C_e contain the row and column addresses of the end of the motion path, whose slope is given by:

$$\frac{R_c - R_e}{C_{start} - C_e}$$

2.2 Experimental results

Examples of motion paths produced by the stair-step algorithm are represented in Figure 7.4 by the broken lines, and the tracking data is shown from time step 75 onwards. The template response is the same as in Figure 6.12 (Chapter 6). The path on the left uses the conjugate pair (E,7) of templates indicating the onset of an edge, while the path on the right uses the pair (2,6), which indicates the transition to steady state.

As mentioned at the beginning of this chapter, templates indicating the onset of an edge should be used in preference to templates indicating the transition to steady state. However, provided that steady state templates are not corrupted by changes in contrast following one another too closely, both types of templates should, nonetheless, yield more or less the same velocity. Accordingly, in the example of Figure 7.4, the "time" and "displacement" values computed for the motion paths of each target template pair vary within similar ranges.

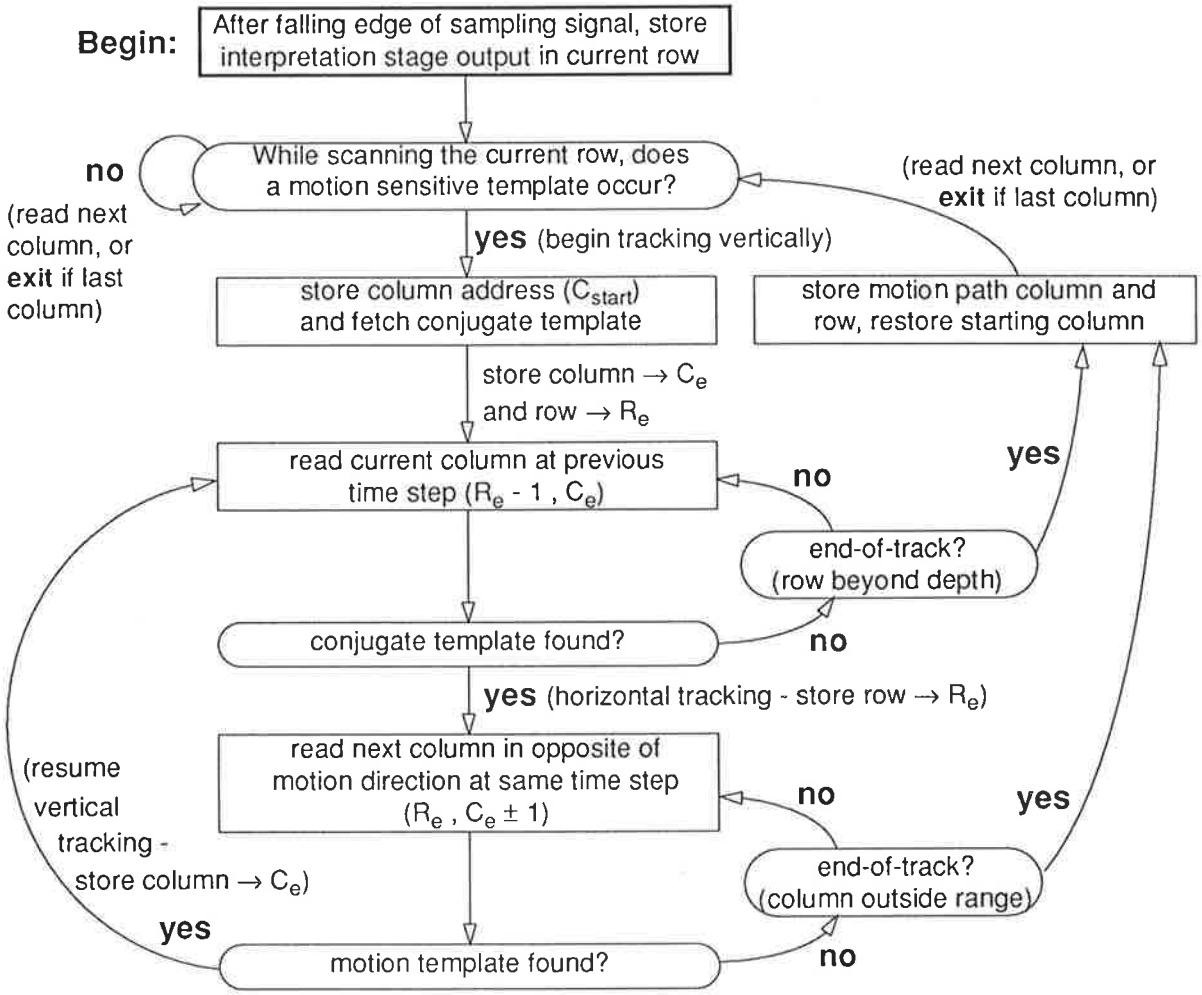


Figure 7.3 Search and tracking modes flow diagram

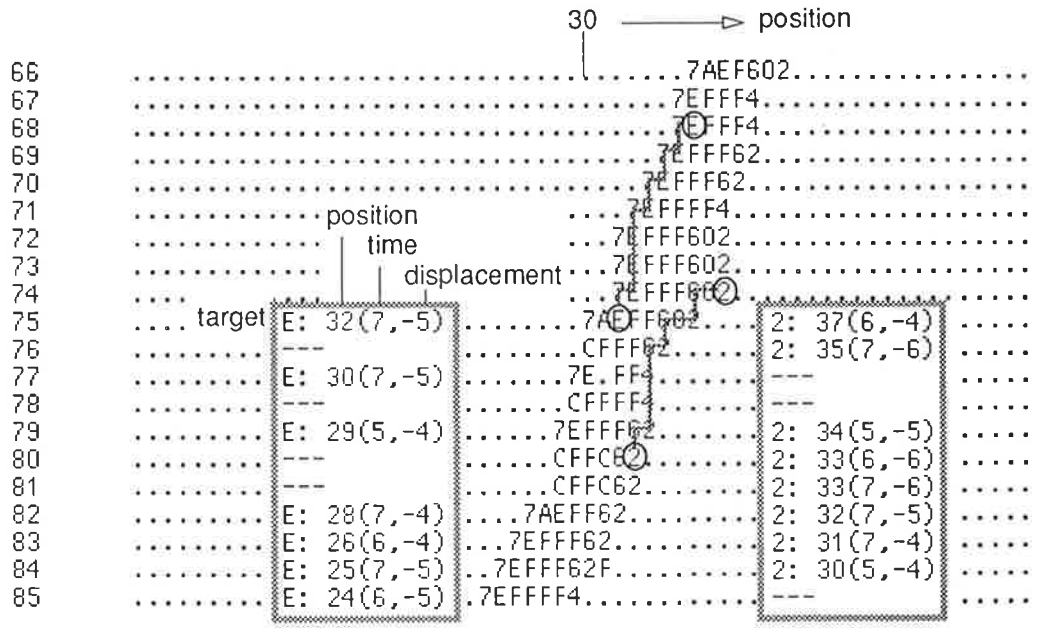


Figure 7.4 Back-tracking motion paths ("time" is the number of time steps from the beginning to the end of the motion path, and the image memory depth is 8)

2.3 VLSI architecture

The algorithm could be implemented in hardware in four main blocks (Nguyen *et al.*, 1994), shown in Figure 7.5, namely, the image memory, the address generator, and the search and track modules. The control block directs the address generator and activates in turn the search and track modules.

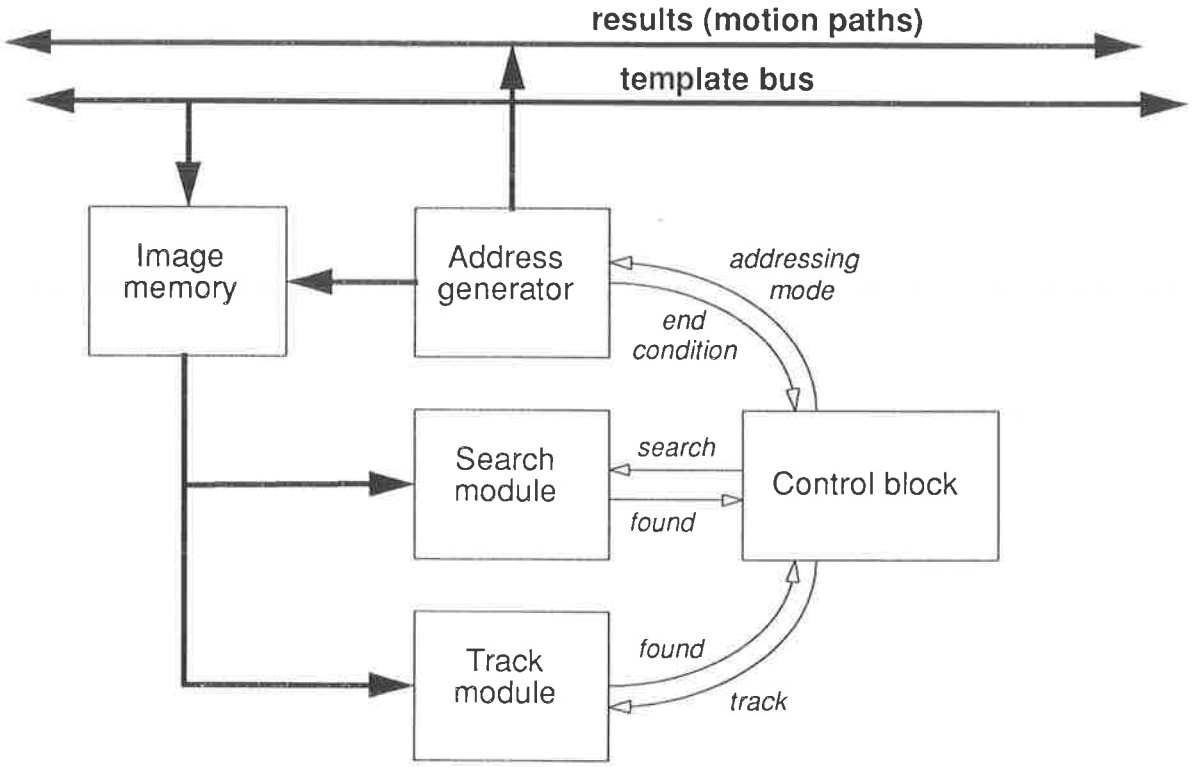


Figure 7.5 Stair-step tracker organisation

Image memory and address generator

As specified earlier, the image memory is organised in rows and columns, where at each sampling instant the templates produced by the interpretation stage are stored in a single row, and at successive columns. Therefore, a row contains a copy of the save RAM.

Figure 7.6 depicts the row and column address generation. The addressing mode determines how the incrementer-decrementers (inc/dec) and multiplexers are controlled, as well as which registers are used. The row inc/dec is designed to "wrap around" in either of the two following cases: (i) the row address reaches its maximum value, which is equal to the depth of the image, or (ii) the row address is equal to zero, which may occur while a motion path is being back-tracked.

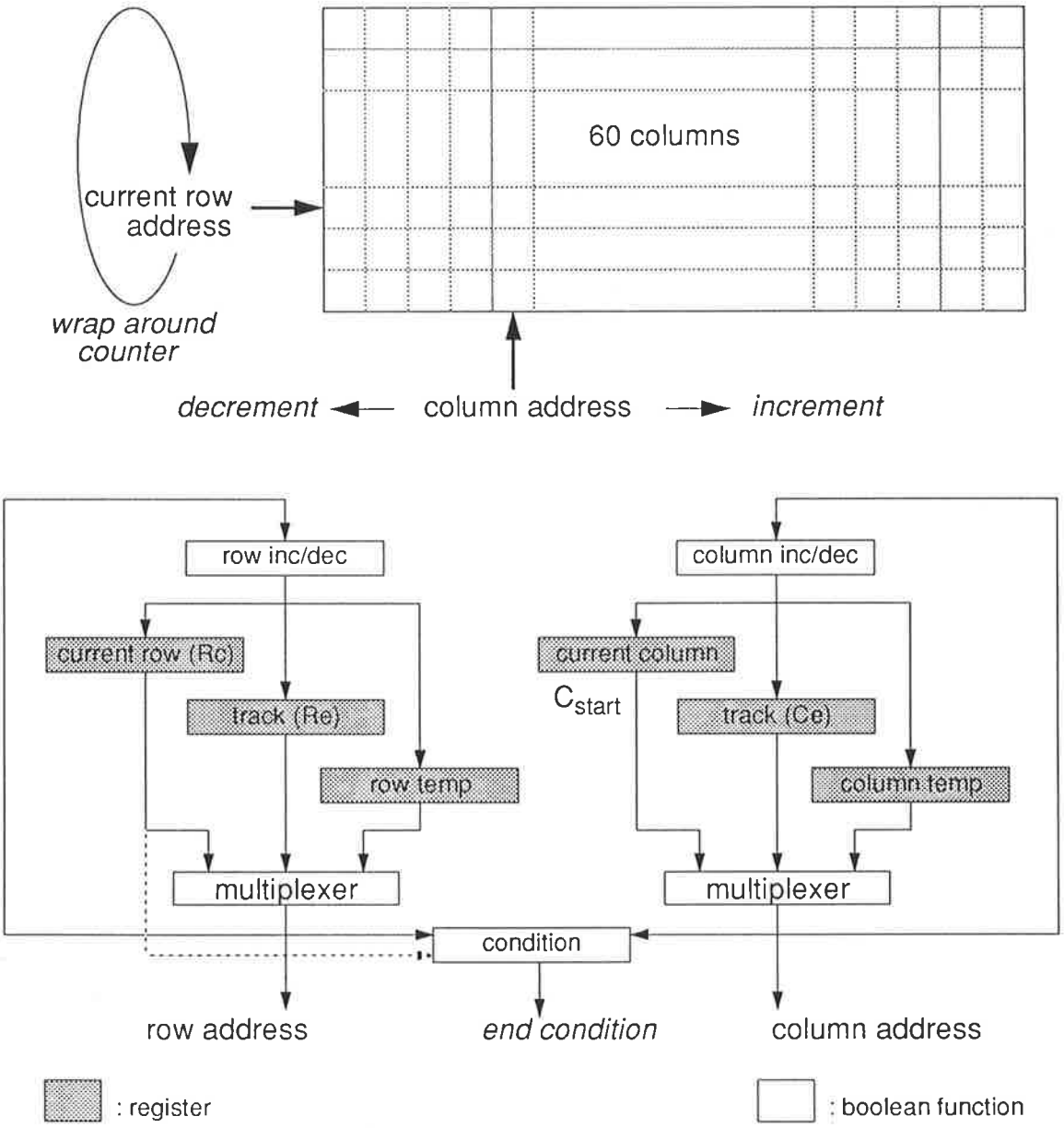


Figure 7.6 Image memory and address generator

As mentioned earlier, there are three modes of operations:

- **Initial loading.** The current row and current column registers are selected. Initially, the row is incremented once and the column is set to 0. The row is also stored in the track (R_e) and row temp registers, in preparation for the tracking mode. The column is then incremented each time a template is available for storage. The *end condition* is generated when the column reaches the end of the row (i.e., column 59).
- **Search mode.** As previously, the current row and current column registers are initially selected. The column is set to 0 and then incremented until a motion template is found. The column is then stored in the track (C_e) and column temp registers, and **tracking mode** begins. Upon resumption of **search mode**, the current row and current column

registers are again selected and the column is incremented. The *end condition* is generated when the column reaches the end of the row (i.e., column 59).

- **Tracking mode.** The row temp and column temp registers are selected. Starting with the row, the row and column sections of the address are alternately produced as follows: the row is decremented until a conjugate template is found, and the row is then stored into the track (R_e) register. Depending on the search direction, the column is then either incremented or decremented until the motion template is found, at which point the track (C_e) register is updated. This procedure is iterated until either row temp is equal to current row (i.e., the row address counter has wrapped around), or column temp has reached its lower or upper boundary. The *end condition* is then generated, resulting in the start and end points of the track being sent on the **result** bus (current row, current column and track (R_e), track (C_e) respectively). **Search mode** then resumes.

Search and track modules

The search and track modules share some of the same hardware, as shown in Figure 7.7. The stack containing the target motion sensitive templates and the look-up table for the corresponding conjugate templates are initially loaded through the template bus, although it should be pointed out that a hardware implementation would be facilitated if the templates were efficiently encoded by the interpretation stage. For instance, a particular bit can be used to signify that a template is motion sensitive, and another bit used to encode that template's conjugate (see the forward tracking algorithm further). Hence the register stack can be replaced by a mask while an inverter replaces the look-up table, which makes the conjugate template register unnecessary. Another advantage of such an encoding is that in search mode only one comparison is necessary each time a template is read from the image memory.

The multiplexer, register loading, and condition logic, are controlled according to the mode of operation:

- **Search mode.** The mask (or register stack) is selected. If the comparator output indicates a match between the target and the current template from the image memory, the *track found* condition is generated, resulting in the (unmasked) target being stored in the motion template register, and its conjugate in the conjugate template register (if applicable). **Tracking mode** begins.
- **Tracking mode.** The multiplexer alternately selects the motion template and conjugate template registers, according to the current status of the tracking. If the comparator output indicates a match between the selected register and the template from the image memory, the *template found* condition is generated, resulting in the multiplexer being switched from one register to the other.

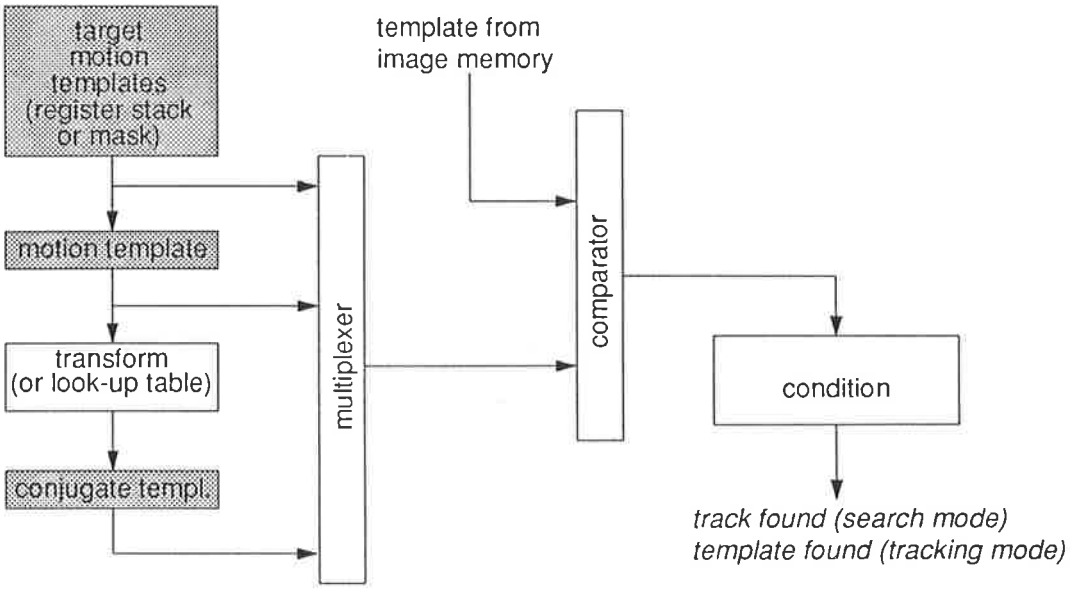


Figure 7.7 Search and track modules

3. Forward tracking

The forward tracking algorithm, first mentioned by Yakovleff *et al.*(1994), is based on the premise that if a motion template has occurred at a certain position, indicating movement in a given direction, then the same moving edge should cause an identical template to eventually occur nearby, i.e., "a few" degrees further in the direction of motion. It is therefore a probabilistic method where assumptions are made concerning the velocities of moving objects. While motion direction is presumed, it is relatively simple to extend the method to handle changes in direction, as shown at the end of this section. For the time being, however, it is preferable to describe the underlying principles of the algorithm.

The algorithm is designed to track the movements of a single target and to estimate its current velocity. For a hardware implementation, multi-target tracking thus requires several tracking *engines*. Given the hardware constraints, and storage space limitations in particular, this algorithm has been devised with two objectives in mind: firstly, minimise past history storage, and secondly, provide a cheap way, in terms of hardware, to update velocity information.

3.1 Single target tracking

A convenient illustration of the forward tracking algorithm is to view a sliding *tracking window*, defined in time-space by the *current target template position* and the *base position*, which is the target template's position a number of time steps previously, as specified by a *window depth* parameter. Tracking windows are depicted in Figure 7.8, where template 7 is the target and the window depth is 6.

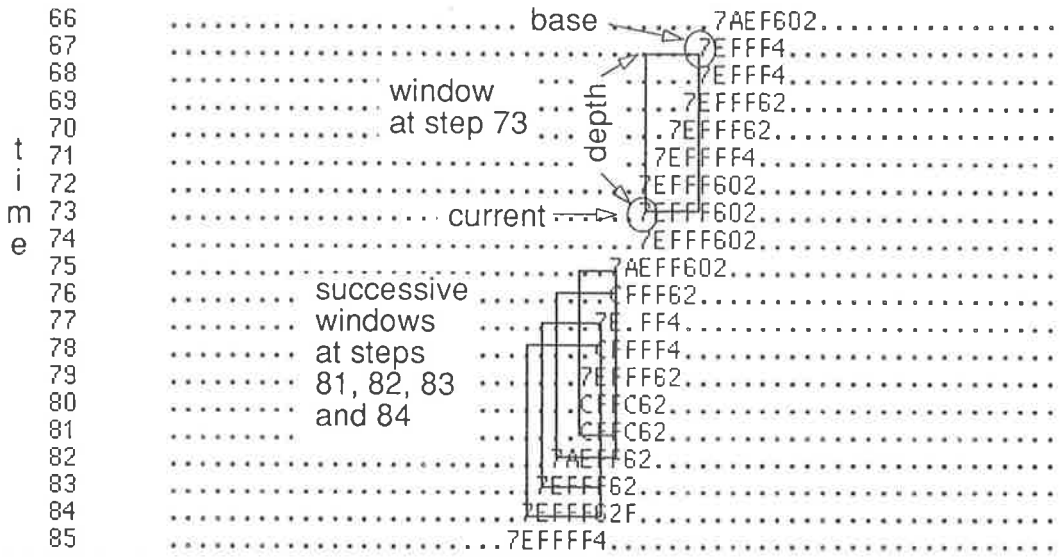


Figure 7.8 Tracking windows

At each time step, the current position is updated only if the target template has been detected, while the base is updated according to the past history of the template's occurrences. The velocity is then given by the slope of the line joining the current position to the base. However, instead of storing the sequence of current positions, only the displacements between one template occurrence to the next are needed, as the base can be recovered by adding or subtracting the sum of the displacements to the current position, depending on the direction of motion.

The sequence of displacements is stored in a "first-in, first-out" (FIFO) register whose size specifies the depth parameter. The FIFO register is shifted by one position at each time step, and the output value which is shifted out is discarded. In addition, so-called "detect" bits are stored alongside the displacements, and serve to indicate that detection has occurred within the time specified by the depth parameter, in order to determine whether tracking is in progress.

An engine's operation is controlled by the following parameters:

- ***Target template***, which also specifies the ***motion direction***
- ***Catch range*** (assumed proximity of the target's next occurrence)
- ***Tracking window depth*** (maximum duration of the recording of the target's occurrences)

A flow diagram of the algorithm is shown in Figure 7.9. Recall that at each time step, the encoded templates are stored in the save memory. Tracking begins when, while the save memory is being written into, a template matches the engine's *target* parameter, resulting in an "initial hit". The template's position, indicated by its address in the save memory, is stored in a register as the "current position", and a detect bit is shifted into the FIFO. The engine is then said to have "locked" onto a target, and thus it becomes dedicated to a single edge.

A "tracking hit" occurs if the template is detected again at a subsequent time step, before all detect bits have been shifted out of the FIFO, and in a position nearby the current position in the direction of motion (as defined by the *range* and *direction* parameters). The displacement between the new and the previous "current positions" is shifted into the FIFO, as well as another detect bit. The current position register is then updated with the new position. For the time steps at which the target template is not detected within the specified range, zero displacements and zero detect bits are shifted in. The velocity can be updated at each time step by evaluating the ratio of the sum of the displacements to the tracking window depth.

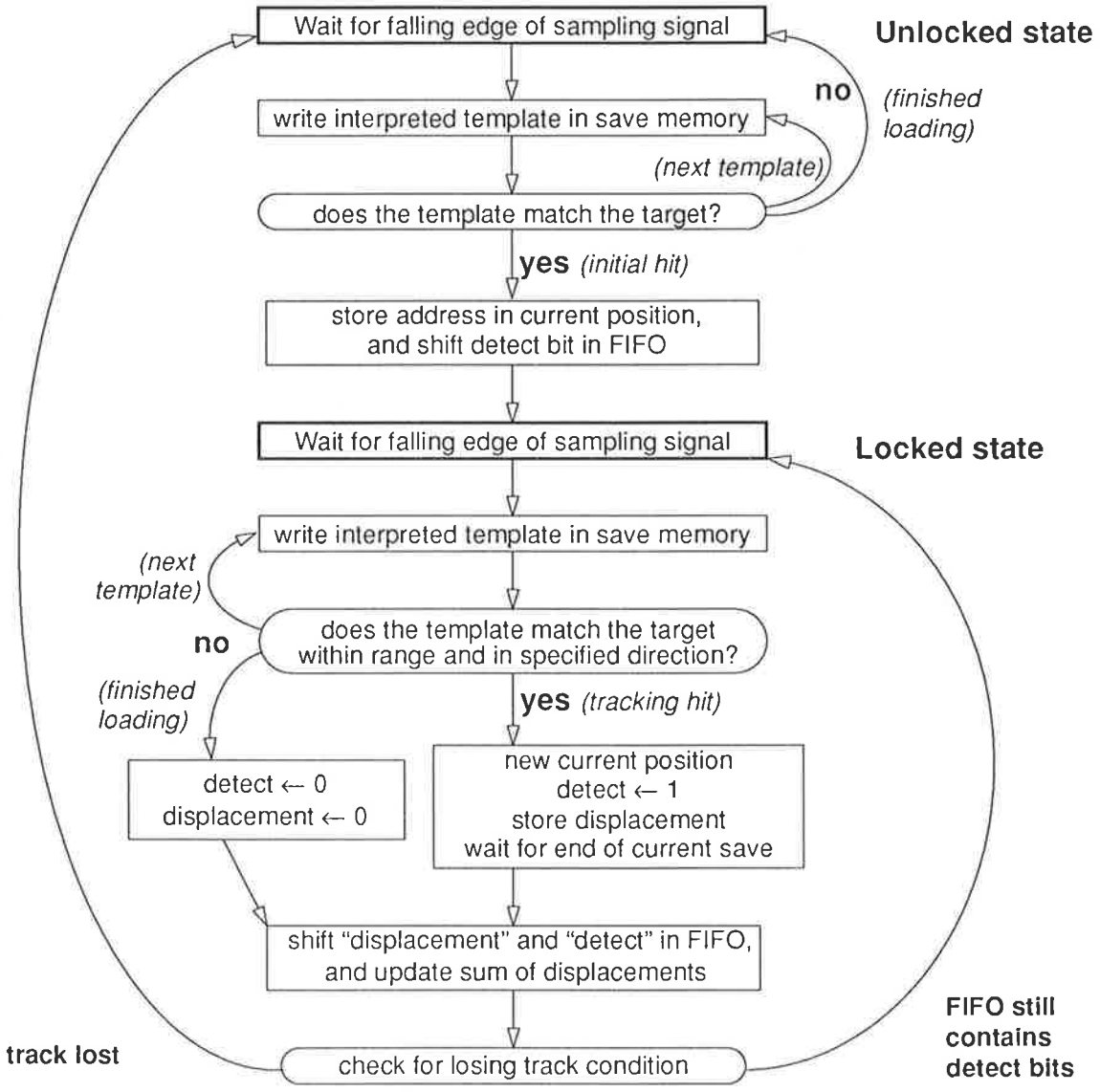


Figure 7.9 Forward tracking flow diagram

The presence of detect bits in the FIFO is constantly monitored. The engine is reset when none are present, and may then resume tracking at any position. Note that the track may be lost under any of the following conditions: (i) the edge has accelerated, causing the target template to occur beyond the range, or (ii) the edge has stopped moving, or (iii) changed direction.

3.2 Multiple target tracking and experimental results

Multiple target tracking requires the presence of several engines, and a small controller ensures that engines do not lock onto the same target. Figure 7.10 shows an example where a bright-to-dark edge moving to the right has been detected. The template occurrences are shown on the left hand side while the tracking output data is shown on the right. Two engines become active, one tracking template 3 (edge onset), the other template 5 (steady state).

```
          0         10  ——▷ position   target: position (displacement)
          |         |
     151  ................................  5: --(--)      3: --(--)
     152  ................................  5: --(--)      3: --(--)
     153  .......73.......................  5: --(--)      3:  8(0)
t    154  .......CB3......................  5: --(--)      3:  9(+1)
i    155  .......CF4......................  5: --(--)      3:  9(+1)
m    156  .......5DB3.....................  5:  7(0)       3: 10(+2)
e    157  ........CF4.....................  5:  7(0)       3: 10(+2)
     158  ........CFB3....................  5:  7(0)       3: 11(+3)
     159  ........CFF00..0................  5:  7(0)       3: 11(+3)
     160  ........5DF00..0................  5:  8(+1)      3: 11(+3)
     161  .........5DFB3.B................  5:  9(+2)      3: 13(+5)
     162  .........5DFB3.B................  5:  9(+2)      3: 13(+4)
     163  ..........CFFB3.................  5:  9(+2)      3: 14(+5)
     164  ..........5DFFB3................  5: 10(+3)      3: 15(+5)
     165  ...........5DFFD3...............  5: 11(+4)      3: 16(+6)
     166  ............5DF54...............  5: 12(+5)      3: 16(+5)
     167  .............CFF4...............  5: 12(+5)      3: 16(+5)
     168  .............CFFB3..............  5: 12(+4)      3: 17(+6)
     169  .............5D.FB3.............  5: 13(+4)      3: 18(+5)
     170  ..............5DFF4.............  5: 14(+5)      3: 18(+5)
     171  ..............5DFF4.............  5: 14(+5)      3: 18(+4)
```

	FIFO #1 - target: 5	posit.	FIFO #2 - target: 3	posit.
time	detect displacement	(sum)	detect displacement	(sum)
151	0 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	-	0 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	-
152	0 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	-	1 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	8 (0)
153	0 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	-	1 1 0 0 0 0 0 0 - 1 0 0 0 0 0 0 0	9 (1)
154	0 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	-	0 1 1 0 0 0 0 0 - 0 1 0 0 0 0 0 0	9 (1)
156	1 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	7 (0)	1 0 1 1 0 0 0 0 - 1 0 1 0 0 0 0 0	10 (2)
157	0 1 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0	7 (0)	0 1 0 1 1 0 0 0 - 0 1 0 1 0 0 0 0	10 (2)
158	0 0 1 0 0 0 0 0 - 0 0 0 0 0 0 0 0	7 (0)	1 0 1 0 1 1 0 0 - 1 0 1 0 1 0 0 0	11 (3)
159	0 0 0 1 0 0 0 0 - 0 0 0 0 0 0 0 0	7 (0)	0 1 0 1 0 1 1 0 - 0 1 0 1 0 1 0 0	11 (3)
160	1 0 0 0 1 0 0 0 - 1 0 0 0 0 0 0 0	8 (1)	0 0 1 0 1 0 1 1 - 0 0 1 0 1 0 1 0	11 (3)
161	1 1 0 0 0 1 0 0 - 1 1 0 0 0 0 0 0	9 (2)	1 0 0 1 0 1 0 1 - 2 0 0 1 0 1 0 1	13 (5)
162	0 1 1 0 0 0 1 0 - 0 1 1 0 0 0 0 0	9 (2)	0 1 0 0 1 0 1 0 - 0 2 0 0 1 0 1 0	13 (4)
163	0 0 1 1 0 0 0 1 - 0 0 1 1 0 0 0 0	9 (2)	1 0 1 0 0 1 0 1 - 1 0 2 0 0 1 0 1	14 (5)
164	1 0 0 1 1 0 0 0 - 1 0 0 1 1 0 0 0	10 (3)	1 1 0 1 0 0 1 0 - 1 1 0 2 0 0 1 0	15 (5)
165	1 1 0 0 1 1 0 0 - 1 1 0 0 1 1 0 0	11 (4)	1 1 1 0 1 0 0 1 - 1 1 1 0 2 0 0 1	16 (6)
166	1 1 1 0 0 1 1 0 - 1 1 1 0 0 1 1 0	12 (5)	0 1 1 1 0 1 0 0 - 0 1 1 1 0 2 0 0	16 (5)
167	0 1 1 1 0 0 1 1 - 0 1 1 1 0 0 1 1	12 (5)	0 0 1 1 1 0 1 0 - 0 0 1 1 1 0 2 0	16 (5)
168	0 0 1 1 1 0 0 1 - 0 0 1 1 1 0 0 1	12 (4)	1 0 0 1 1 1 0 1 - 1 0 0 1 1 1 0 2	17 (6)
169	1 0 0 1 1 1 0 0 - 1 0 0 1 1 1 0 0	13 (4)	1 1 0 0 1 1 1 0 - 1 1 0 0 1 1 1 0	18 (5)
170	1 1 0 0 1 1 1 0 - 1 1 0 0 1 1 1 0	14 (5)	0 1 1 0 0 1 1 1 - 0 1 1 0 0 1 1 1	18 (5)
171	0 1 1 0 0 1 1 1 - 0 1 1 0 0 1 1 1	14 (5)	0 0 1 1 0 0 1 1 - 0 0 1 1 0 0 1 1	18 (4)

Figure 7.10 Motion tracking - Targets: 3 and 5, right (+) direction, catch range: 2, tracking window depth: 8

The engine outputs are given in the following format: target template: current position (sum of displacements). As can be seen, the engine tracking template 3 locks on first at time step 153, while template 5 is first detected at time step 156.

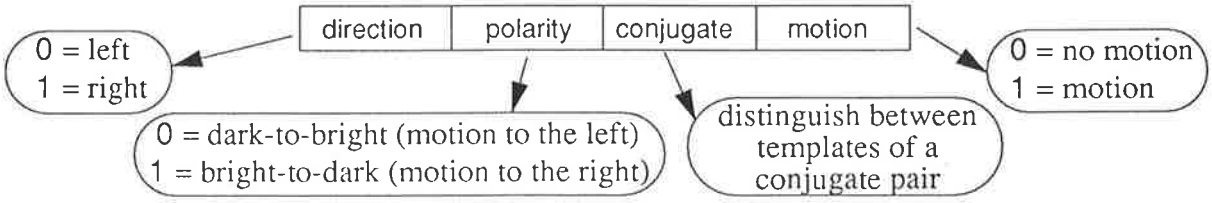
3.3 Changes in motion direction

Handling changes in direction can be accomplished by a relatively simple extension to the forward tracking algorithm, and consists of allowing a tracking engine to switch to another target if the expected motion template is not detected for a significant number of time steps.

The new target should be selected according to the information provided by the original target template, which characterises both the direction and polarity (the sign of the change in contrast) induced by a moving edge. Thus, by observing that a change in the direction of motion also causes a change in polarity, provided that the edge's contrast with respect to the background remains constant, the tracking engine may switch to a new target which differs from the first in both direction and polarity. The new target would then be expected within the specified catch range, but in the direction opposite to that indicated by the original target template.

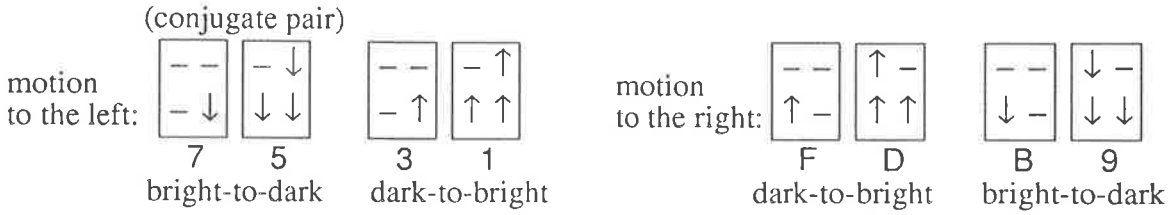
Another aspect of changing the tracking direction concerns the mechanism which triggers the target switch. Recall that if a targeted template has not been detected for several consecutive sampling steps, the corresponding portion of the FIFO contains only zeros, and this may be used as an indication that the engine is about to lose the track on which it is currently locked. In a practical sense, an "early warning" signal may be generated by OR-ing some of the FIFO's detect bits. Upon activation of the warning signal, tracking is extended to different templates in opposite directions, and a "tracking hit" signal is produced if either target is detected within the corresponding negative or positive range. If such is the case, the engine updates its target parameter, and tracking proceeds as if an "initial hit" had occurred, i.e., by ignoring the stored displacements corresponding to the original target.

In order to facilitate a hardware implementation, it may be desirable to encode the templates in such a way that switching from one target to another only requires changing a single bit. As an example, the following 4-bit format may be used:



The "no motion" template can be arbitrarily encoded as 0, and all the templates other than those indicating the onset of an edge, may be encoded as 2 (i.e., 0010 binary).

Following a change in direction, the target switch may be accomplished by inverting the higher order bit. Therefore, if the motion templates indicating the onset of an edge are encoded as shown below, template 7 switches to F or vice-versa, 5 switches to D, and so on.



In spite of the difficulty mentioned in Chapter 6, namely that the bugeye fails to detect dark-to-bright changes in contrast, the available experimental results can nonetheless be utilised to illustrate the manner in which changes in direction may be tracked. Recall that the motion of a dark strip produces templates indicating the onset of the bright-to-dark edge, followed by the transition to steady state. As pointed out in Chapter 6, one of the consequences of the problem with the analog circuitry is that the positions in which the steady state templates occur are a function of the width of the strip, provided that the latter does not exceed a few centimetres. Therefore, these templates could be interpreted as indicating the onset of the dark-to-bright edge, i.e., the other side of the strip, instead of the transition to steady state.

Figure 7.11 shows an example of tracking a change in motion direction, where steady state templates have been encoded as indicating the onset of a bright-to-dark edge. The tracking window depth is 8, and the "early warning" signal is generated if a target has not been detected for at least 4 consecutive time steps.

Both engines lock onto their respective targets at time 3827, one tracking template 5 (or D) and the other template 1 (or 9). Motion is to the left until time step 3835, when the direction appears to reverse. The engine tracking template 5 finds the opposite motion direction template (D) at time 3841. The engine tracking template 1 loses the track, as the occurrence of the target at time 3838, possibly due to noise, delays the generation of the early warning signal. Thus, when the warning signal is eventually produced, target 9 is out of range. However, the engine does pick up the track later, when template 9 is detected at time 3848.

3.4 VLSI architecture

Figure 7.12 depicts the organisation of the tracking engines. Each of them receive data from the address and template buses and an engine produces its output on a "result" bus which can be read externally. A central controller (or request handler) is required to ensure that different engines do not lock onto the same track. The controller receives the "initial hit" and "tracking hit" signals from all the engines, and generates *acknowledge* signals to each engine.

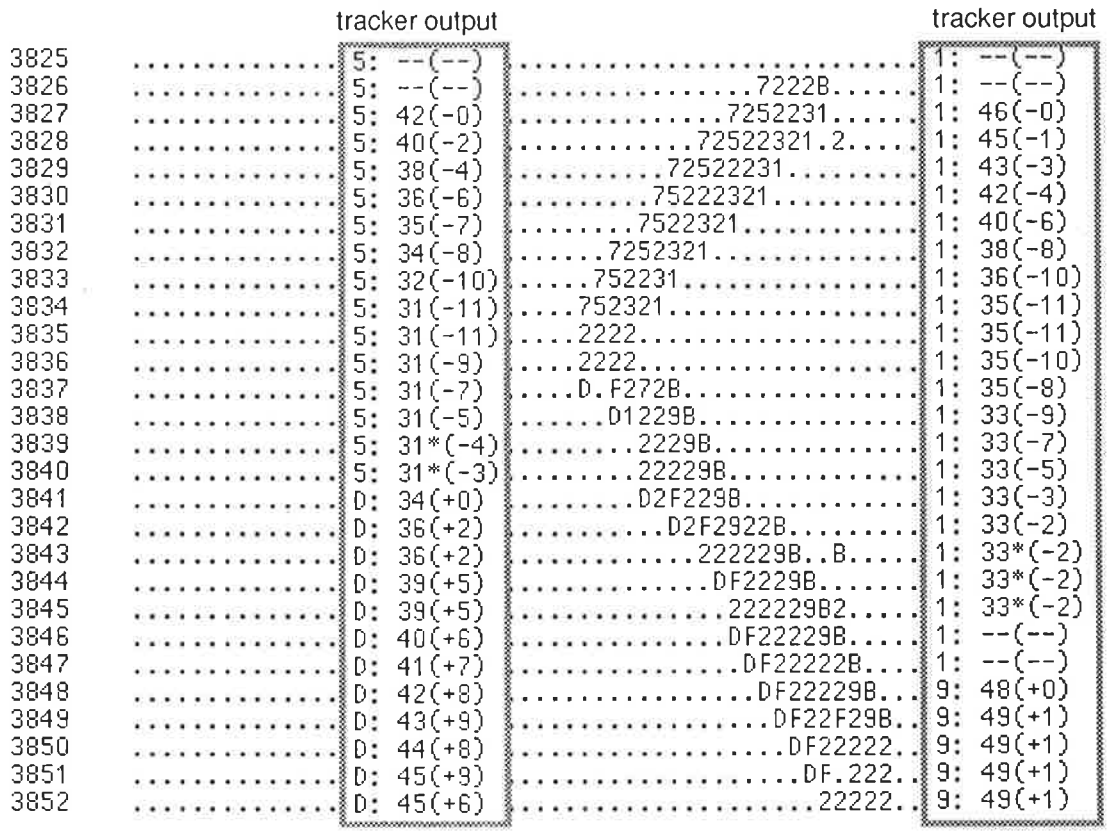


Figure 7.11 Tracking of changes in motion direction (the tracker output shows a "*" while the engine searches for targets in either direction)

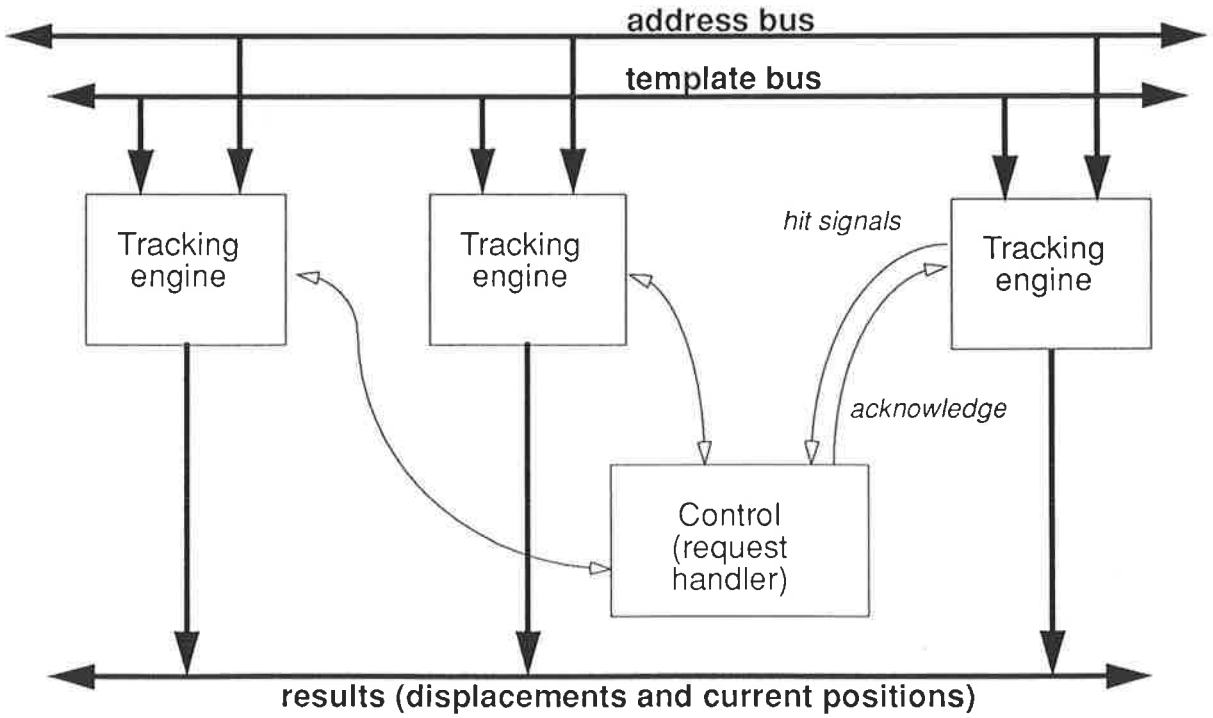


Figure 7.12 Forward tracking organisation

The controller (or request handler) operates as follows:

1. A single "initial hit" results in the requester being acknowledged.
2. Simultaneous "initial hits", which correspond to the same template and the same address (i.e., angular position) result in only one requester, chosen arbitrarily, to be acknowledged.
3. Simultaneous "tracking" and "initial" hits do not result in an *acknowledge* to either requester, as the engine that produced the "tracking hit" operates independently until it loses the track on which it is currently locked. Note that rules 1 and 2 ensure that simultaneous "tracking hits" cannot occur.

A tracking engine is shown in Figure 7.13, where the dark boxes represent registers and the clear boxes the specified boolean functions. The target template and range registers are loaded during initialisation and the FIFO is shifted by one position at the end of each sampling step, after the current interpretation stage output has been examined and the save RAM has been loaded.

Each time a template is written into the save memory, it is stored in the new template register and its address in the new position register. The template is compared with the target, producing a *template hit* signal if there is a match. The new and current position registers are subtracted according to the direction parameter, which is provided by the target template. The result of the subtraction is then compared with the range parameter and an *address hit* signal is produced if the distance between the new and current positions is within the range. In order to facilitate the hardware implementation, the new and current positions may be switched so that the subtraction is always positive in case of an *address hit* corresponding to the original target template, and always negative for the opposite motion template.

The condition logic produces an *initial hit* signal if a *template hit* occurs and if the engine is not currently locked on a track. If an *acknowledge* signal ensues, a *detect* bit is prepared for being shifted into the FIFO (note that there is no displacement as yet). If the engine is locked on a track, a *tracking hit* signal is produced if both the *template hit* and *address hit* signals occur. The subtractor output is then stored in the displacement register (*store* signal), a *detect* bit is enabled, and the new position is copied into the current position register. If no hit occurs at a sampling step, a zero displacement is stored and the current position is not updated. Lastly the total displacement is available after each sampling step and the *end-of-track* condition is checked, resulting in the engine being reset (i.e., unlocked) if no *detect bits* are present.

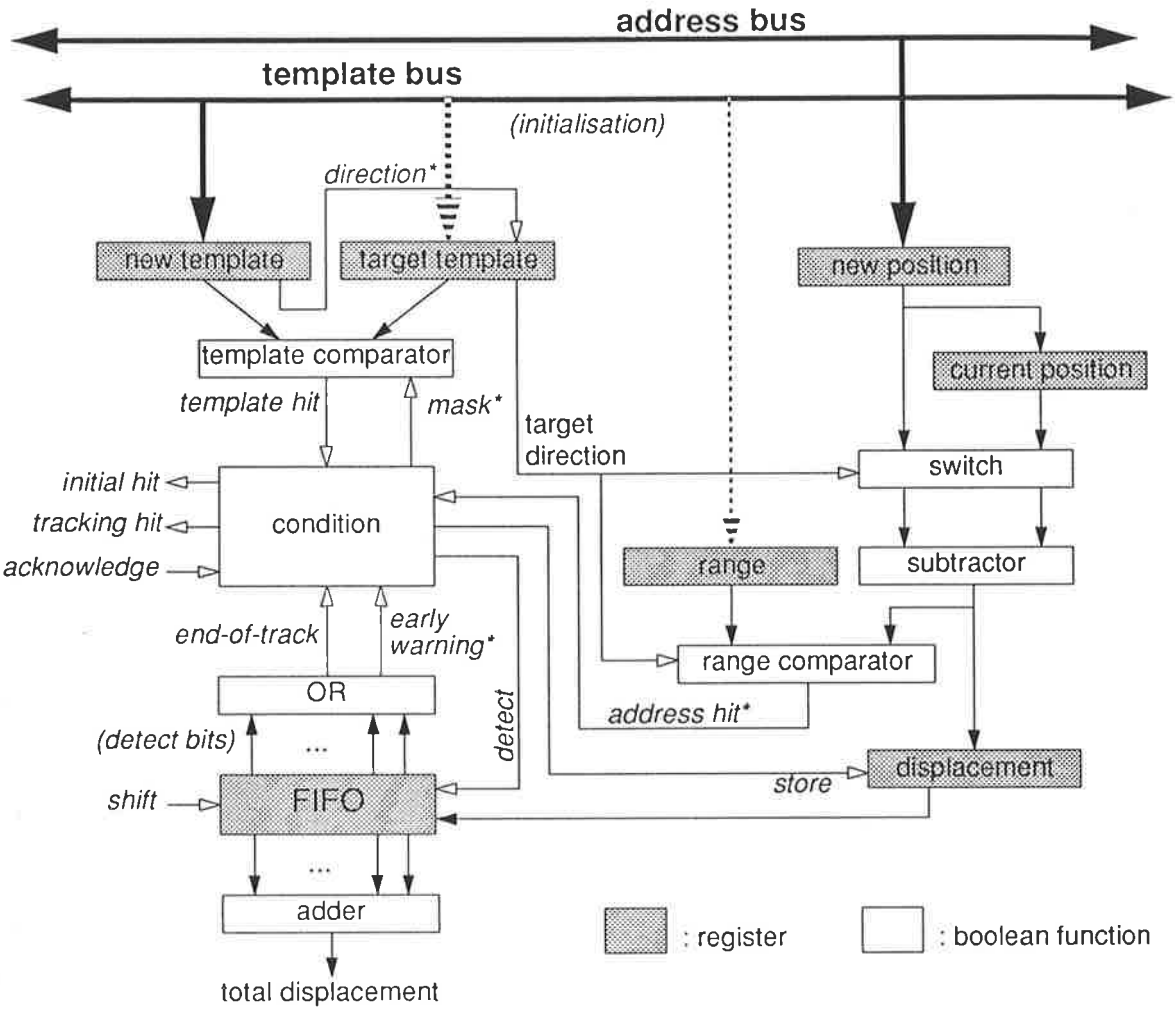


Figure 7.13 Forward tracking engine block diagram

The signals which concern the capability to handle changes in direction are shown with an asterisk (*), and are generated, or used, as follows:

- The *early warning** signal is generated by OR-ing the detect bits corresponding to the latest detections.
- When *mask** is active, the direction bit is not taken into account by the template comparator. The condition logic generates *mask** if the engine is not currently locked on a track or if the *early warning** signal is produced.
- The *direction** bit of the target template register is updated following an *initial hit* (i.e., a "masked" *template hit* occurring when the engine is unlocked) or if a *tracking hit* occurs when *early warning** is active.
- The *address hit** signal is generated by the range comparator and comprises two bits, one indicating that the subtractor output is positive and within range for the current target direction, while the other bit indicates that the output is negative and within range (i.e., ≥ – range), but for the opposite direction.

4. Comparative results

The algorithms estimate velocity in somewhat different ways. At a given sampling instant, the back-tracking algorithm provides a velocity estimate only if the target has been detected. The velocity is then evaluated as the difference between the current position and the earliest occurrence of the target (or its conjugate) on the motion path and inside the image memory, divided by the number of intermediate time steps. For the forward tracking algorithm, the velocity is available whether or not the target has been detected at the considered instant, and is given by the ratio of the sum of the stored displacements to the tracking window depth.

Using the same data as in Chapter 6, the tracking results are shown in Table 7.1. The tracking outputs are chosen arbitrarily near the centre of the receptive field, and hence by applying the optical correction factor, the performance of the algorithms can be compared with the results obtained "manually" in Chapter 6. Note that estimated velocities tend to be higher than real velocities, as the computer-controlled sampling rate decreases (see Chapter 6).

The duration of the recordings, i.e., the depth of the back-tracking image memory and the forward tracking window depth parameter, is set to 8 frames for both algorithms. No significant differences between the results yielded by the algorithms were noted with larger values, while a smaller depth tended to produce erratic measurements. Notice that for the highest velocity the forward tracking algorithm does not produce reliable results, due to the interframe displacement exceeding the pre-set range parameter, thus causing the track to be lost. The biological plausibility of this characteristic of the algorithm is discussed in Section 5.

real velocity (degrees per second)	tracking results (mean values)		measured velocity (from Chapter 6)
	stair-step (back-track)	forward	
60	57 (– 5%)	85 (+ 42%)	**68 (+ 13%)**
80	92 (+ 15%)	99 (+ 24%)	**95 (+ 19%)**
85	106 (+ 25%)	111 (+ 31%)	**104 (+ 22%)**
115	132 (+ 15%)	131 (+ 14%)	**139 (+ 21%)**
155	197 (+ 27%)	(N/A)	**178 (+ 15%)**

(percentages in brackets show the differences between measured and real velocities)

Table 7.1 Velocity estimation with the tracking algorithms

The interframe variations for forward tracking do not exceed the ratio of the range to the window depth, while in the case of the back-tracking algorithm, variations around the mean values can be significant, as a misplaced or missing template may greatly shorten the recorded track. However, in spite of these variations, which may reach 50 percent of the real velocity, the results indicate that it is possible to discriminate between "slow" and "fast" motion.

5. Discussion

Conventional image acquisition systems tend to be fairly slow, due mainly to high bandwidth requirements, and hence tracking a target in real time requires that its motion be predicted in order to restrict the search space (e.g., Frau *et al.*, 1990). For instance, the future position of a target may be extrapolated from previous positions using Kalman filters or least-mean-square fit methods, and may even be estimated with no latency provided that the motion is fully predictable (see for instance McDonald & Bahill, 1983, and Wavering & Lumia, 1993). By comparison, the bugeye requires a very low bandwidth, due to motion information being encoded immediately following detection, and hence the implementation of either of the algorithms presented here is feasible in real time. Moreover, no restrictions are placed on the search space, as the entire template response produced by the chip is examined at each sampling instant.

Both algorithms require that data corresponding to previous sampling times be stored, and therefore the depth of the tracking is limited by the hardware. However, estimating velocity by averaging over a large number of sampling times only increases the accuracy if motion is smooth. The experimental results presented in the previous section suggest that an integration time, or tracking depth, of about 8 frames is optimal, and should not exceed, say, 12 to 16 frames. This duration should be short enough to detect reliably acceleration and deceleration, and yet long enough for velocity to be discriminated, as indicated by experiments on biological visual systems (Watamaniuk & Sekuler, 1992).

The forward and back-tracking algorithms illustrate two different approaches to motion analysis. The stair-step back-tracking algorithm consists of an iterative search through a number of stored frames corresponding to the tracking depth. However, since an object in motion does not necessarily elicit motion templates at each time step, it might be advisable to temporarily store information concerning tracks that could be in progress. Arguably, the back-tracking algorithm is close to conventional methods as the motion analysis is systematic, and the full recent history of template occurrences is kept.

In contrast, the forward tracking algorithm may be justified biologically in the sense that some assumptions are made concerning the motion of objects, as appears to be the case in natural visual systems (see Chapter 3). Firstly an interframe upper displacement limit akin to d_{max} is implemented in the form of the range parameter, and secondly "motion inertia" is assumed, as target templates are expected in the current direction of motion. As a result, the storage requirements are minimised, and may be roughly compared with psychophysical measurements (Verghese & Pelli, 1992). Moreover, the manner in which the algorithm operates reflects other characteristics observed in biological systems. For instance, a sudden acceleration causes

one engine to lose the track, which can then be picked up by another engine, and hence acceleration may be assimilated to a "new" motion onset (Dzhafarov *et al.*, 1990, Sekuler & Sekuler, 1993), while changes in motion direction can only be detected once the motion is well established (Sekuler *et al.*, 1990; see also Boman & Hotson, 1992).

Finally, experiments on human visual tracking, or "oculomotor pursuit", indicate that eye movements are smoother when the subject does not exercise attentional control, i.e., in passive conditions (Wyatt & Pola, 1987), and that expectations concerning future motion do not seem to be greatly affected by habituation (Kowler, 1989). These results suggest that it may be sufficient to establish the optimal tracking parameters through preliminary experimentation. In other words, it is debatable whether the performance of the forward tracking algorithm would be significantly improved by implementing complex mechanisms whereby the tracking parameters are continuously adjusted "on line" as a function of recent motion history.

6. Summary

The two algorithms for estimating velocity from the occurrences of motion sensitive templates produce fairly similar results, even though their underlying principles differ in some important respects. The back-tracking algorithm is the closest in spirit to conventional information analysis, and consists of searching a stored time-space image containing the full recent history of template responses. Thus the rationale is that as much as possible of the available data should be examined, lest potentially relevant events be overlooked. However, there are in principle no limitations on the number of objects which may be tracked.

In contrast, the forward tracking algorithm consists of locking onto a single motion template, and expecting that target to occur subsequently within a pre-determined range, and in the direction specified by the target. The assumptions concerning an upper interframe displacement limit and motion inertia are biologically plausible, and result in a significant reduction in storage requirements, as only the latest displacements are needed in order to estimate velocity.

The forward tracking hardware consists of small "engines" which operate independently once locked onto a target. The system is therefore distributed, with the exception of a simple arbitration logic which ensures that distinct engines do not lock simultaneously onto the same target. In comparison, the back-tracking hardware consists of a central processing and control unit which manages a fairly large image memory. Preliminary investigations indicate that the costs in terms of respective areas occupied by a VLSI implementation of both algorithms should be roughly equivalent, provided that a reasonable number of tracking engines are to be used.

CHAPTER 8 Track association and control dynamics

The velocity information provided by the tracking algorithms presented in the previous chapter can be further utilised by a control system to enhance the system's knowledge about the immediate surroundings. In particular, the motions of edges can be associated on the basis of their having similar velocities. Then, by taking into account the directions and signs of the contrast changes, it may be possible to infer that distinct edges belong to the same object and to detect looming, which is of particular importance for collision avoidance and localised path planning. Possible ways in which sensing information obtained from the bugeye could be utilised by an autonomous control system are discussed. Finally, it is argued that the limitations of the bugeye may not adversely affect reliability.

1. Track association

1.1 Underlying principles

Velocity is not the only kind of information which can be extracted from tracking motion sensitive templates. Recall from Chapter 4 that templates also carry polarity information, i.e., the sign of the difference in contrast between a moving edge and the background. Therefore, if an object's contrast with respect to the background is reasonably constant, and if the object is wide enough or close enough to the bugeye, its relative motion should elicit the detection of at least two edges moving at similar velocities.

Consider the example where the relative heading direction is in between the two outside edges of an object. The detected motions are of opposite directions but of identical contrast, and if the edges are seen to be moving away from each other, the bugeye and the object could

be on a collision course (see the looming experiment in Chapter 6). Conversely, if the directions are identical but the polarities are opposite, the relative heading direction of the bugeye would be towards either side of the object. The simplest cases are depicted in Figure 8.1.

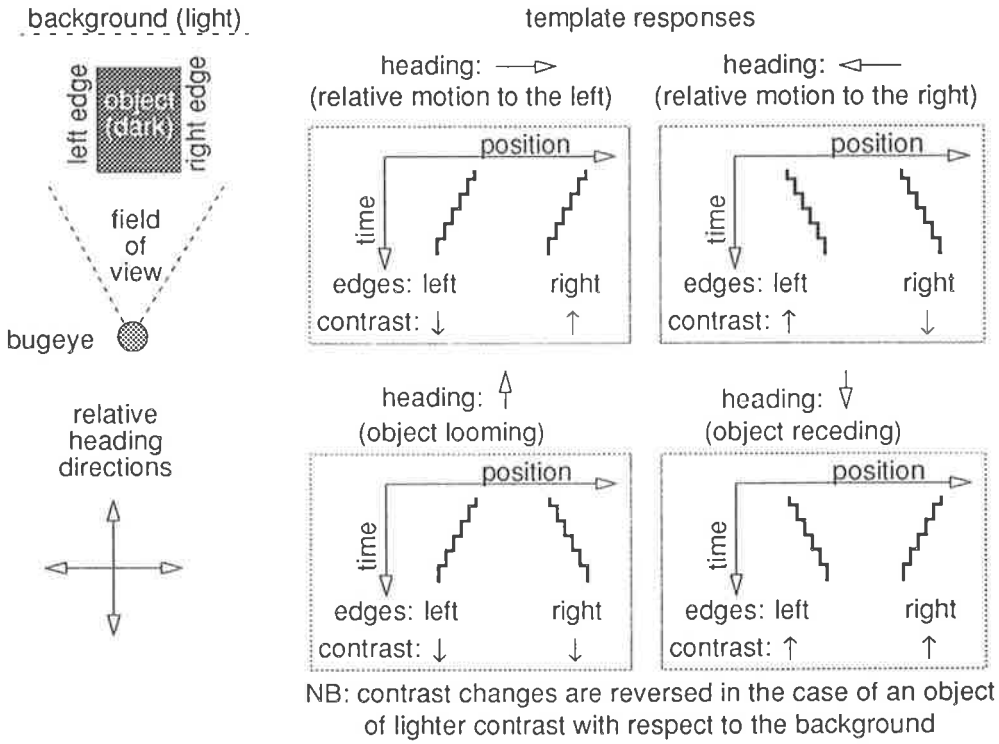


Figure 8.1 Tracks produced for different heading directions

These conditions, which are ideal in the sense that there are no ambiguities, may possibly be recovered in more general cases by controlling the rotational motion of the bugeye. Thus, by inducing relative motion in addition to the existing motion, one of the ideal cases of Figure 8.1 could arise. Then, since the induced motion is known to the control system of the bugeye, the real relative heading direction may be recovered by compensating for the added motion. In fact, "panning" and "peering", or inducing motion in order to resolve ambiguities, is a form of behaviour common to many insects (see Chapter 3), and even comes naturally to humans to enhance depth perception. Interpreting the induced motion, however, can be quite complex and will be speculated upon later in this chapter. Firstly, the algorithm for associating velocities is described.

1.2 Velocity association

The motions of distinct edges may be associated on the basis of their velocities being similar for a reasonable number of sampling instants. The respective motion directions, polarities, and relative positions, are taken into account at a latter stage in order to differentiate between the possible interpretations, such as edges belonging to the same object, looming, and so forth.

The mechanism by which velocities can be compared consists of subtracting the velocities at each time step (i.e., irrespective of motion direction), and generating a signal whose value reflects the closeness of the velocities. In practical terms, the signal is maximum if the absolute velocities are matched, and decreases as the difference between the velocities increases. The signal is zero if the difference is greater than a "matching limit" parameter. Signals from the comparator are accumulated in a storage element as follows. A positive signal increases the content of the storage element proportionately to the value of the signal, and when the signal is zero, the content is decreased according to a time constant, in order to "forget" earlier values. The storage element thus reflects the past history of velocity matching, and controls the input to a thresholding device. When the device input reaches a pre-set threshold (not to be confused with the "matching limit" parameter), its output "fires", thus indicating that velocities have been reasonably close for some time, as defined by the matching limit and threshold parameters. Each time the device fires, the content of the storage element is decreased, and the device ceases to fire when its input has decreased below the threshold parameter value.

The mechanism may be viewed as a crude "charge build-up" neuronal mechanism, modelled by a matching comparator which feeds electrical charges into a thresholding device with a leaking capacitive element at its input. In Figure 8.2, which shows an analog version of the model, the capacitor loses charges through the leakage resistor when none are added, and if the thresholding amplifier output is high. Notice that charges may still be added to the capacitor while the device is firing, depending on the closeness of the velocities. The dynamics of the model are inspired from a biologically plausible model proposed by Abbott & Kepler (1990).

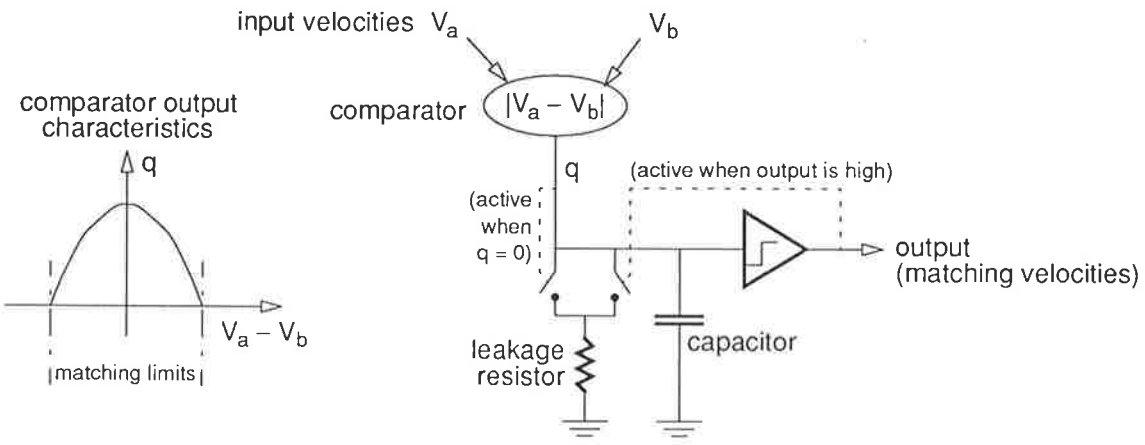


Figure 8.2 Charge build-up model for matching velocities

The reason for decreasing the charges held by the capacitor while the amplifier is "firing" is to ensure that velocities are no longer considered to be matched as soon as possible after they cease to be similar. Ideally, similar velocities should therefore cause the charges to hover around the threshold, without building up. An alternative scheme would be to place a limit on the amount of charge that the capacitor may hold.

The characteristics of the comparator and the value of the threshold should be set in such a way that velocities are matched if their values are found to be similar for a relatively small number of time steps. The inertia of the system reduces the probability of false matches, albeit at the cost of deferring matching if the velocity measurements vary beyond the matching limits, as may well be the case due to the nature of the tracking algorithms of Chapter 7.

1.3 Interpretation

Having determined whether the relative motions of distinct edges are similar, the next step consists of utilising the information provided by the templates (i.e., polarity and direction) in order to infer the presence of objects and, hopefully, the relative heading direction. Note that it may be necessary to take into account the current edge positions, as velocities should only be associated if the corresponding edges immediately follow one another (i.e., there is no edge in between the pair under consideration).

The velocity association mechanism can thus be used to enable the different combinations of direction, polarity, and relative edge positions, in order to generate a signal corresponding to the interpretation, such as "looming", "receding", object motion (as distinct from edge motion), and so forth. The conditions under which the signals are generated are summarised in Figure 8.3, where it is assumed, arbitrarily, that template A occurs to the left of template B.

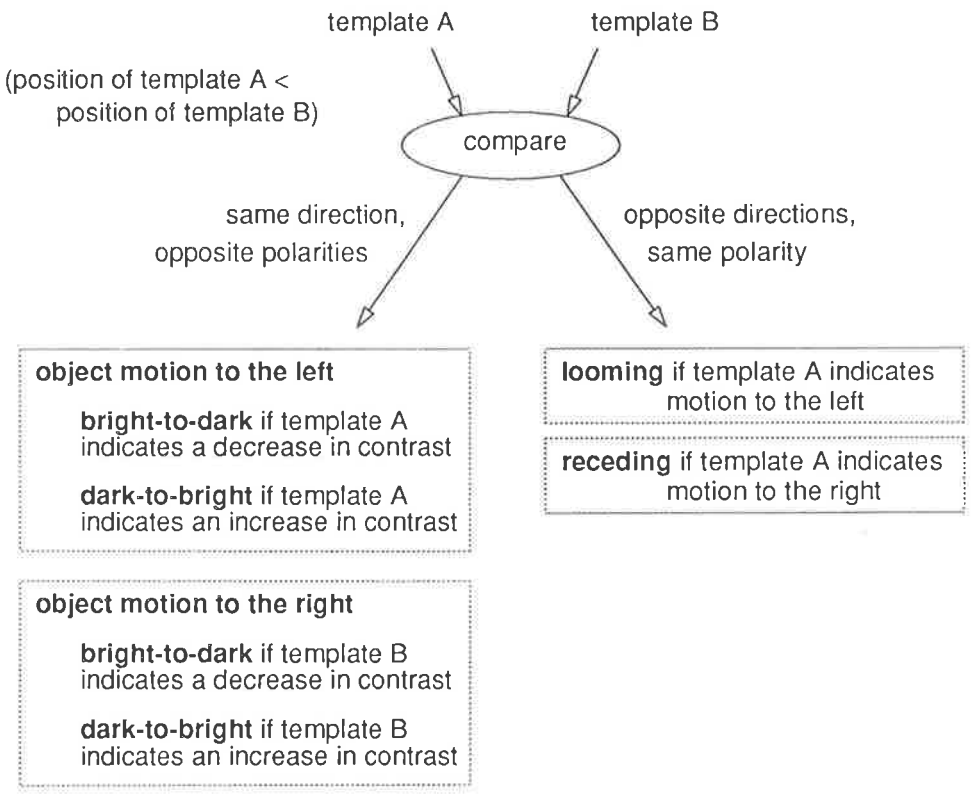


Figure 8.3 Generation of interpretation signals for track association

2. Algorithm implementation

Since there is little way of knowing beforehand how many objects may be in motion simultaneously relative to one another and to the bugeye, provisions should be made for associating every pair of detected tracks. Implementing such a scheme using current VLSI technology would be feasible but hardly optimal as the hardware may be under-utilised. Therefore, it is proposed to implement the track association algorithm in software, where every possible pair of detected tracks can be examined easily before determining whether tracks should be associated according to their respective polarities, directions, and positions.

As a consequence of the digital nature of the tracking algorithms of Chapter 7, the analog matching mechanism described previously is approximated digitally. The capacitor of Figure 8.2 can be modelled as a register whose contents, representing the charges, are increased according to the closeness of the measured velocities, or decreased by a fixed decrement if the velocities are too far apart. Thus, the velocities are considered to be matched if the difference between the register value and the pre-set threshold is positive, in which case the register contents are also decreased (Figure 8.4).

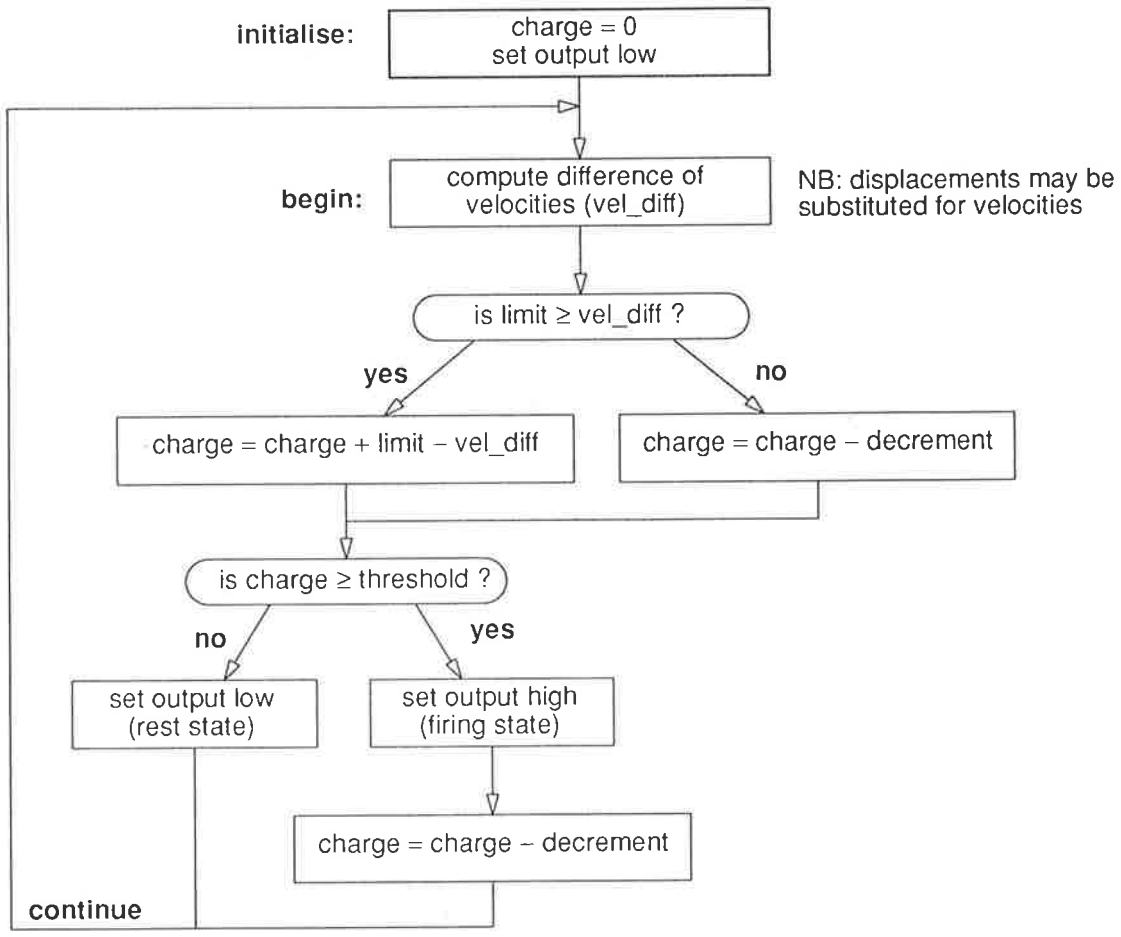


Figure 8.4 Velocity association algorithm for forward tracking

In the algorithm depicted in Figure 8.4, the amount by which the charges are increased if the velocities are similar is evaluated linearly as the difference between the matching limit parameter, and the absolute difference in velocities. It is suggested that the use of a more complex "similarity measure" would only be worthwhile if the velocity measurements were more accurate. Moreover, a linear measure enables the outputs of the forward tracking engines of Chapter 7 to be utilised by the velocity association algorithm directly, without having to compute the ratio of displacement to tracking window depth.

Recall from Chapter 7 that the outputs of a forward tracking engine are the current position and the sum of the recorded displacements. If the engine has been locked on the target template for a time greater than the maximum duration of the recording (i.e., the tracking window is "established"), the velocity can be estimated by evaluating the ratio of the total displacement to the tracking window depth parameter. Therefore, if the depth parameter is identical for all the tracking engines, the differences in velocities are proportional to the respective differences between the sums of the displacements, provided that the windows are established.

Finally, by setting the "limit", "threshold", and "decrement" parameters appropriately, the only operations required are subtractions and additions involving integer numbers. Such simplicity could be an advantage for small autonomous systems whose processing capabilities may be limited in terms of floating point computations.

3. Experimental results

3.1 Object motion

Figure 8.5 shows an example of track association based on the template response to a dark object moving to the right in front of a lighter background. The template encoding is the same as that used in the example of Chapter 7, concerning the detection of changes in direction. For instance, template D is interpreted as a motion sensitive template indicating a dark-to-bright change in contrast, although, in reality, it indicates the establishment of the steady state following the bright-to-dark change in contrast (template 9).

The matching process begins once both tracking windows are established, at time step 3982. The velocities are found to be similar between times 3989 and 3992 when the matching limit parameter is set to 3, which appears to be appropriate. When the limit is set to 4, the similarity is discovered earlier, but the output tends to continue "firing" even after the velocities are no longer similar. As suggested earlier, the interpretation would be that both edges belong to the same dark object.

template response to object moving to the right

encoding: motion to the left, bright-to-dark: **5** [conjugate: **7**], dark-to-bright: **3** [**1**]
motion to the right, bright-to-dark: **9** [**B**], dark-to-bright: **D** [**F**]
all other templates: **2**

```
3971  ..........................................................
3972  ........................7B................................
3973  ........................29B...............................
3974  ........................229B..............................
3975  ......................DF292B..B..........................
3976  ......................DF229B.9...........................
3977  ......................D2F2922............................
3978  ....................DF22F2B..............................
3979  ....................DF22292B............................
3980  ....................D2F2229B............................
3981  ..................F22292B..............................
3982  ..................D2F2229B............................
3983  ..................DF2229B..............................
3984  ..................DF22292B............................
3985  ..................D2F22292B..........................
3986  ..................DF222292B..........................
3987  ..................DF2222292.........................
3988  ..................D2F22222B.........................
3989  ..................DF222292B.....................
3990  ..................DF222229B.....................
3991  ..................D2FD2222.....................
3992  ..................D2F3F2.....................
3993  ..................D1D1.....................
3994  ..........................................................
```

forward tracking data (depth = 8)

```
3971    D: --(--)       9: --(--)
3972    D: --(--)       9: --(--)
3973    D: --(--)       9: 24(+0)
3974    D: --(--)       9: 25(+1)
3975    D: 23(+0)       9: 26(+2)
3976    D: 24(+1)       9: 28(+4)
3977    D: 25(+2)       9: 29(+5)
3978    D: 27(+4)       9: 29(+5)
3979    D: 28(+5)       9: 33(+9)
3980    D: 29(+6)       9: 35(+11)
3981    D: 29(+6)       9: 36(+12)
3982    D: 32(+9)       9: 38(+13)
3983    D: 34(+11)      9: 39(+13)
3984    D: 35(+11)      9: 40(+12)
3985    D: 36(+11)      9: 42(+13)
3986    D: 38(+11)      9: 44(+15)
3987    D: 39(+11)      9: 46(+13)
3988    D: 40(+11)      9: 46(+11)
3989    D: 42(+13)      9: 48(+12)
3990    D: 43(+11)      9: 50(+12)
3991    D: 44(+10)      9: 50(+11)
3992    D: 46(+11)      9: 50(+10)
3993    D: 48(+12)      9: 50(+8)
3994    D: 48(+10)      9: 50(+6)
```

track association

```
D /9
D /9        threshold = 8
D /9        decrement = 2
D /9
D /9          charge register value
D /9          (arbitrary units)
D /9          limit = 3 and 4
D /9
D /9
D /9#
D /9#
D#/9# 4  0    0      output "firing"
D#/9# 2  1    2      (velocities are
D#/9# 1  3    5      matched)
D#/9# 2  4    7
D#/9# 4  2    7
D#/9# 2  3    9*
D#/9# 0  6    11*
D#/9# 1  8*   12*
D#/9# 1  8*   13*
D#/9# 1  8*   14*
D#/9# 1  8*   15*
D#/9# 4  4    13*
D#/9# 4  2    11*
```

displacement difference

window established

Figure 8.5 Example of track association

3.2 Looming

The detection of looming can be time-critical for collision avoidance, and may require that tracks indicating motion in opposite directions be associated as soon as possible after being detected. In this respect, the mechanism used previously, with its built-in inertia, could be too slow, as the comparison between the outputs of forward tracking engines only takes place after both tracking windows are established.

The problem is illustrated in Figure 8.6, which shows the response to a dark object on a collision course with the bugeye, along with the data obtained from the tracking algorithms of Chapter 7. The data is the same as that shown in the looming experiment of Chapter 6, except that the templates have been swapped according to the encoding used in Chapter 7 (see Figure 8.5). Notice that several forward tracking engines with identical targets (template 9) are employed, as the first displacement of the target exceeds the range parameter, and hence one engine loses the track. While increasing the range may not be an appropriate solution, particularly for cases other than looming, setting up tracking engines with identical parameters is justified, as several objects of similar contrast may be in motion relative to the sensor.

The results illustrate the difficulty of associating high velocities, due to large interframe variations of the displacements, which in turn are caused by measurement errors (see Chapter 6). Also, it appears that the detected motion to the left is, on average, faster than its counterpart to the right, implying that the relative heading direction is not exactly perpendicular to the plane between the two edges, which could correspond to the surface of an object. Therefore, in order to detect that the velocities are similar, the track association criteria would have to be relaxed, maybe significantly, by decreasing the threshold and increasing the matching limit. In doing so, however, the probability of wrongly associating the motions of edges belonging to distinct objects would increase.

Thus, while in principle the tracking engine and track association parameters may be altered in order to suit current circumstances, it may not be advisable to do so in practice. Firstly, information concerning the environment should be known beforehand in order for the control system to be able to set the appropriate parameters, and secondly, it is unlikely that those circumstances will remain stable. Therefore, the parameters would have to be adjusted continuously, to the detriment of the overall performance of the system.

Since the results indicate that the track association mechanism used previously yields inconclusive results, a different association mechanism should be employed. Quite simply, it may be sufficient to decide that an object is looming if it is discovered that edges of identical polarity are moving "fast" in opposite directions, and towards the limits of the aperture of the optical system. For this purpose, "fast" may be defined as a rate of at least one position per

time step, and a looming condition is signalled if such velocities are detected far enough from the centre of the aperture (e.g., 15 positions). In Figure 8.6, looming is thus detected at time steps 1021 and 1022 if the back-tracking data is used, while the forward tracking data indicates looming from time step 1021 onwards.

template response to looming object (data from chapter 6 re-encoded)

```
1010  ..........................................................
1011  ..........................................................
1012  ..............................72B.........................
1013  ........................72225295222B......................
1014  ........................222222222229B.....................
1015  ......................7252222222222922B...................
1016  ..............2272225222222222222229B...................
1017  ............7.225222222222222222222922B.................
1018  ............752222222222222222222222222.................
1019  2B.........7252222222222222222222222222292B.2...........
1020  22.......725222222222222222222222222222222.............
1021  29B.7B72252222222222222222222222222222222229222B.......
1022  2292595222222222222222222222222222222222222222292222B.....
1023  22222222222222222222222222222222222222222222222222222.....
1024  222222222222222222222222222222222222222222222222222229B....
1025  2222222222222222222222222222222222222222222222222222....
```

forward tracking data (depth = 8, range = 4)

```
1010  5: --(--)    9: --(--)    9: --(--)
1011  5: --(--)    9: --(--)    9: --(--)
1012  5: --(--)    9: --(--)    9: --(--)
1013  5: 28(-0)    9: 30(+0)    9: --(--)
1014  5: 28(-0)    9: 30(+0)    9: 35(+0)
1015  5: 24(-4)    9: 30(+0)    9: 36(+1)
1016  5: 22(-6)    9: 30(+0)    9: 39(+4)
1017  5: 18(-10)   9: 30(+0)    9: 40(+5)
1018  5: 14(-14)   9: 30(+0)    9: 40(+5)
1019  5: 13(-15)   9: 30(+0)    9: 43(+8)
1020  5: 11(-17)   9: 30(+0)    9: 43(+8)
1021  5:  9(-19)   9: --(--)    9: 45(+10)
1022  5:  6(-22)   9:  2(+0)    9: 49(+14)
1023  5:  6(-18)   9:  2(+0)    9: 49(+13)
1024  5:  6(-16)   9:  2(+0)    9: 49(+10)
1025  5:  6(-12)   9:  2(+0)    9: 49(+9)
```

looming detected

back-tracking data (image depth = 8)

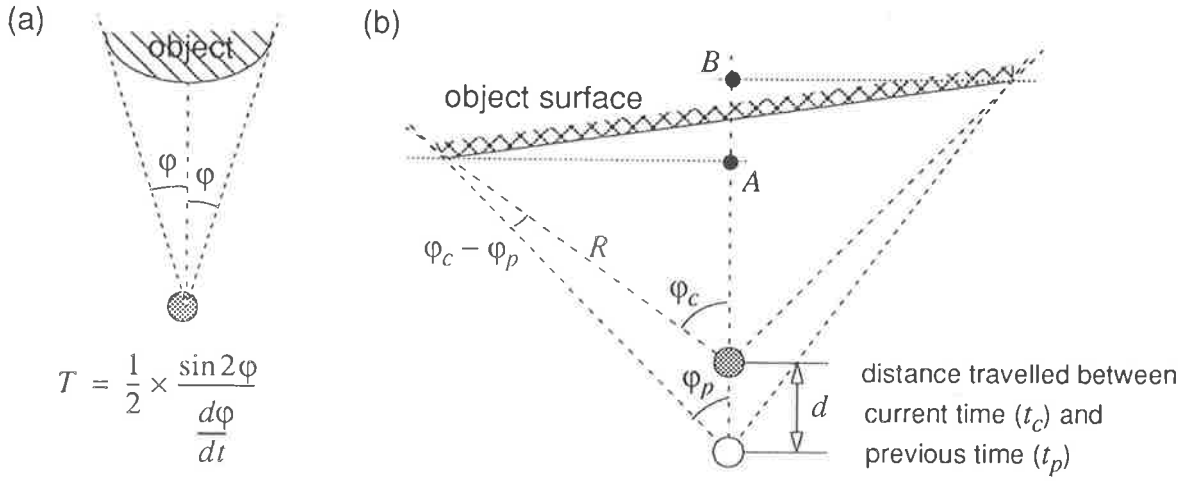
```
1010
1011     format: target: current position (number of time steps, total displacement)
1012
1013  5: 28(1,0)     9: 30(1,0)     5: 31(0,0)
1014  9: 35(2,5)
1015  5: 24(3,-4)    9: 36(3,6)
1016  5: 22(4,-6)    9: 39(4,9)
1017  5: 18(5,-10)   9: 40(5,10)
1018  5: 14(6,-14)
1019  5: 13(7,-15)   9: 43(7,13)
1020  5: 11(7,-17)
1021  9: 1(2,0)      5: 9(6,-15)    9: 45(7,10)
1022  9: 2(3,1)      5: 6(7,-18)    9: 49(7,13)
1023
1024  9: 54(7,14)
1025
```

looming detected

Figure 8.6 Looming picture and tracking data

3.3 Time to collision

In principle, the manner in which the time to collision (T) can be estimated from a looming picture is shown in Figure 8.7a (see Chapter 3). The time to collision formula is derived by assuming that the angular rate of expansion ($d\varphi/dt$) varies smoothly, and that it can be measured with sufficient accuracy for acceleration to be noticeable. In other words, a decrease in the time to collision, which should be noticed in principle, implies that the rate of expansion should increase (i.e., accelerate) faster than the corresponding angular increase, if the angle φ is less than 45 degrees. However, measurements effected with the tracking algorithms of Chapter 7 can vary widely, and hence it may be useful to re-derive the formula for the more general case of Figure 8.7b. The heading direction is not necessarily towards the centre and perpendicular to the surface of the object, and only the angles subtended by the edges at two distinct instants are known.



$$T = \frac{1}{2} \times \frac{\sin 2\varphi}{\frac{d\varphi}{dt}}$$

(from Chapter 3)

$$R \sin(\varphi_c - \varphi_p) = d \sin \varphi_p$$

if v is the velocity of the sensor relative to the object, then:

$$T_A = \frac{R \cos \varphi_c}{v} \quad \text{and} \quad t_c - t_p = \frac{d}{v}$$

and therefore:

$$T_A = (t_c - t_p) \times \frac{\sin \varphi_p \times \cos \varphi_c}{\sin(\varphi_c - \varphi_p)}$$

Figure 8.7 Estimation of time to collision

The time to collision estimates may differ, depending on which edge is chosen for the computation. In Figure 8.7b, T_A is the time required by the bugeye to reach point A, which is in front of the object, and is calculated with data pertaining to the relative motion of the left-hand-side edge. Equivalently, T_B could be computed using the right-hand-side motion, thus T_A and T_B provide lower and upper limits for the "real" time to collision.

Table 8.1 shows the values of T_A and T_B obtained from the data of Figure 8.6, and using the expression derived in Figure 8.7b. Assuming that the velocity of the bugeye with respect to the object remains constant, the values of T_A and T_B should decrease by approximately 15 milliseconds (the sampling period) from one time step to next. Apart from the values of T_B pertaining to the rightward motion measured by the forward tracker between time steps 1021 and 1022, the expected decrease does not appear to occur, and there is even a marked increase between time steps 1023 and 1024. It should be pointed out, however, that the experimental conditions do not guarantee that the relative velocity would remain constant.

time step (ms)	T_A (target: 5, to the left)		T_B (target: 9, to the right)	
	back-track.	forward	back-track.	forward
1021	29	5	53	53
1022	28	5	48	37
1023	-	28	-	48
1024	-	44	-	92

Table 8.1 Time to collision computation
(sampling period is 15.5 milliseconds approximately)

The results are nonetheless consistent when the ratio of T_A to T_B is considered. As pointed out earlier, the data of Figure 8.6 indicates that the leftward motion is roughly twice as fast as the rightward motion, which is fairly well represented in the results of Table 8.1 (i.e., on average, $T_B \approx 2T_A$). However, the variations in the computed values highlight their sensitivity to measurement errors, as reflected by the values of T_A obtained with the forward tracking data at time steps 1021 and 1022, where the angle (φ_p) subtended by the edge at the previous sampling instant is very small.

4. Application to robotics control

The experimental results presented here and in previous chapters indicate how motion detected by the bugeye may be interpreted in order to provide some information about its physical environment. However, it is important to establish the general context in which the sensor operates before determining the appropriate interpretation of the sensing information.

4.1 Environmental context

The current version of the bugeye (see Chapter 6) detects motion in one dimension within an aperture of 60 degrees and with an angular resolution of approximately one degree. The maximum detection range is around 8 to 10 metres, and the sampling rate is in the order of 10 to 15 milliseconds. Not surprisingly, in view of the biological evidence upon which the design

of the chip is based, these characteristics match, in some respects, those of the visual systems of bees and flies (see Chapter 3). However, the application domain of the sensor is constrained, even though the dynamic range would be suitable for a fairly wide range of operating conditions, from dimly lit to bright daylight (notwithstanding aliasing problems under certain artificial lighting conditions which should be corrected in future versions - see Chapter 6).

A concept demonstrator which may be able to exploit successfully the sensing information obtained from the bugeye is an autonomous vehicle whose motion should be restricted to a plane, due to the visual information being analysed in one dimension. Also, considering the detection range and resolution limitations, the sensor would be best utilised in a reasonably confined and human-made environment, and hence a possible application, as a starting point, could be to assist the control of an autonomous mobile domestic appliance (a euphemism for a smart Hoover). It is worth pointing out that recently developed systems designed for similar environments (e.g., Andersen *et al.*, 1992, King & Weiman, 1990, and Kriegman *et al.*, 1989) often require complex sensing and processing capabilities, which would be in marked contrast with a bugeye-based control.

Ultimately, though, it may be possible to develop the sensor to the point where its use on motor vehicles could be envisaged for tasks such as "blind spot" detection, range estimation (see Abbott *et al.*, 1994), and even piloting or steering. In this respect, the system developed by Bruyelle & Postaire (1993) is of particular interest, as visual information is analysed in one dimension, as is the case with the bugeye. A major difference, though, is that range is estimated by triangulation methods from information provided by two scan-line cameras. Other systems are used for steering motor vehicles on the open road, and permit travelling speeds ranging from a few kilometres-per-hour (e.g. Turk *et al.*, 1988), to over fifty (Dickmanns & Graefe, 1988a and 1988b; see also Solder & Graefe, 1990). However, the main difficulty with the current version of the bugeye concerns its optical characteristics, namely, range and resolution.

4.2 Active sensing

The fact that nothing is detected in the absence of relative motion may be viewed either as a restriction, or, *a contrario*, as an advantage. In effect, the preliminary processing step of determining whether objects in the environment are in motion when the sensor (and therefore the vehicle) is static, is a straightforward consequence of the nature of the sensing. Hence, provided that objects are not extremely thin and on a collision course with the sensor, the absence of motion implies "no immediate danger". Moreover, since the range of the sensor is limited to a few metres, no processing is necessary in order to decide that remote objects may be a "threat", simply because the objects would not be detected.

If the total absence of motion on the part of both the sensor and objects in its environment are chosen as initial conditions, an "angular position map" of the immediate surroundings can be constructed by panning, which consists of rotating the sensor about its central axis. The angular positions of edges are then estimated by compensating for the panning motion. In the example of Figure 8.8a, the angular positions of edges *A* and *B* can be evaluated by averaging the maximum and minimum angular positions of the detected tracks.

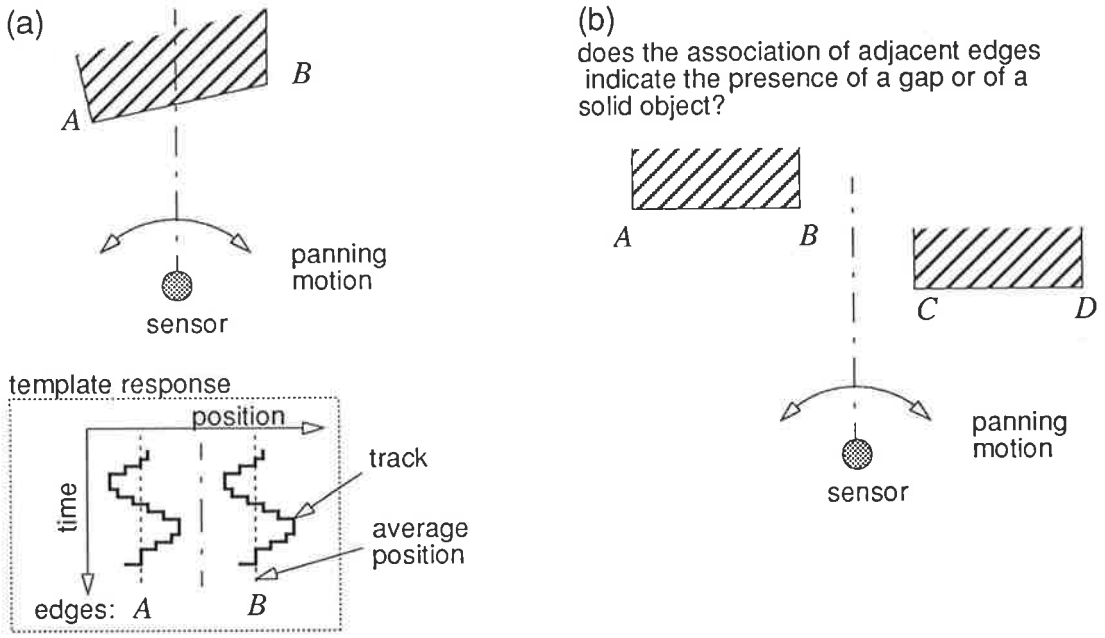


Figure 8.8 Estimation of angular edge positions by panning

The motions of edges *A* and *B* may also be associated in order to infer the possibility of the presence of a solid object. However, the control system should be very circumspect in its decisions concerning track association. For instance, the association of edges *B* and *C* in Figure 8.8b could be interpreted as indicating either a gap, or the presence of a solid object whose contrast is lighter than that of the background (i.e., dark-to-bright change in contrast on the left side and bright-to-dark on the right side).

The examination of the relative positions of "objects" (strictly speaking, associated tracks) may assist the control system in first detecting, and then resolving ambiguities, in order to construct the angular position map. Thus, in the particular circumstances of Figure 8.8b, the correct interpretation may be that the light "object" is in reality a gap, as it is detected in between two possible dark objects, one delineated by edges *A* and *B*, the other by edges *C* and *D*. The degree of confidence in the position map therefore depends on the validity of the assumptions being made concerning the environment. In other words, the control system's perception of the environment would require *a priori* knowledge, such as "objects usually present a uniform contrast", and so forth.

A heading direction can be selected from a list of candidates, based on the angular positions of objects and empty spaces, and according to a preferred direction defined at a higher control level. The motion of the vehicle may then be utilised to estimate the distances separating it from potential obstacles (Figure 8.9), thus yielding a dynamic "relative position map" which may be required by some control schemes. Note that errors in range estimation may be caused by the inaccuracy of the angular velocity measurements as well as by the assessment of self-motion (v), and hence it remains to be seen whether range can be estimated reliably.

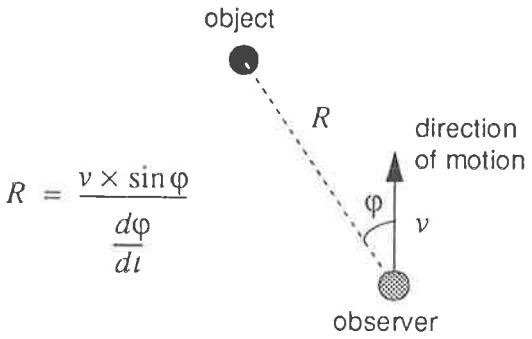


Figure 8.9 Range from self-motion (from Chapter 3)

Ideally, the vehicle should steer around obstacles while avoiding looming conditions such as those of Figure 8.6, where the detected velocities are high. It is clear, though, that due to unforeseen circumstances, such as flawed assumptions, objects or people moving, and so forth, the vehicle may yet find itself on an undesirable collision course with an obstacle. High looming velocities would indicate that collision may be imminent, and hence stopping abruptly would constitute the safest option. However, if the lowest estimated time to collision is deemed to be large enough, taking into account the vehicle's mechanical inertia, the alternative would be to steer the vehicle in the direction of the edge whose relative velocity is lowest. Referring to the example of Figure 8.7b, the vehicle would thus veer (sharply) to the right.

As an aside, it is worth pointing out another possible usage of looming information. In all the examples, it has been assumed that the overall goal is that an autonomous vehicle be able to navigate while avoiding obstacles. However, other goals such as following a person or a moving object could be envisaged. Hence, the idea would be to decelerate or accelerate according to whether looming or receding conditions are detected, respectively. The direction for head-on pursuit would then be maintained by steering the vehicle towards the faster moving edge, until its velocity matches that of the other edge.

4.3 Control structure and operation

The relationships between the perception processes and the control system, summarised in Figure 8.10, present a strong similarity with the "cognitive framework" of Chapter 2. The sen-

sor provides the first measurement, motion, from which more elaborate measurements are progressively derived and interpreted to produce percepts. Therefore, as the "level" of measurement increases, the corresponding percepts become more sophisticated. The percepts are in turn utilised by the control system at the appropriate levels of competence with a view to determining the adequate behaviour, which may imply inducing motion in order to enhance perception. A level of competence may correspond to a control layer, and in general, the principle whereby a control layer "subsumes" the functionality of the next lower level could be applied throughout the control architecture (see Brooks, 1986).

The control system may therefore be built in such a way that the lowest, or reactive, control layer is able to gain control of the vehicle in the case of looming, as the required time-critical reaction is more a reflex than a carefully thought out course of action. As suggested earlier, the reaction consists of steering the vehicle in the direction of the slowest moving edge, implying that the evasive manoeuvre is inferred solely from the difference between the velocities pertaining to the fast moving edges, and is obtained directly from the tracking data. An even simpler scheme, which appears to be used by locusts (Robertson & Johnson, 1993 - see Chapter 3), would be to trigger the collision avoidance mechanism when an object subtends more than a pre-defined portion of the visual field, irrespective of the rate of expansion.

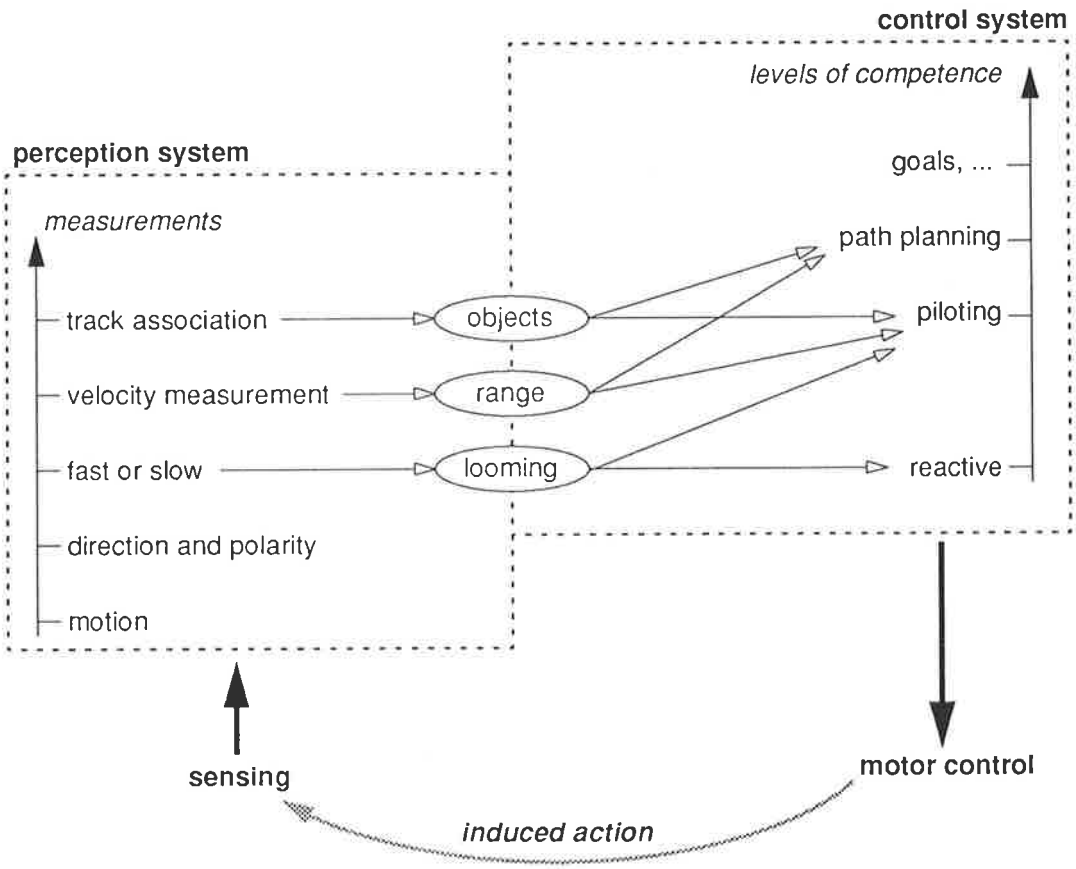


Figure 8.10 Relationships between sensing, interpretation, and control

There are many ways in which "piloting", or short-term path planning, may be accomplished. However, robotics control *per se* is beyond the scope of this thesis, and hence only a few of the available options are suggested, and are based mainly on evidence gathered from natural systems.

Navigation may be controlled incrementally through an appropriate choice of successive fixation points, thus visual feedback is required (Ishiguro *et al.*, 1990). In effect, the vehicle could navigate from one point to another by selecting an edge, and by controlling its motion as a function of the relative angular velocity and the changing position of the edge. From a computational perspective, it could be argued that tracking a moving stimulus may simplify the calculation of self-motion parameters such as translational and rotational velocities (Bandopadhay & Ballard, 1991). Moreover, the computational complexity of depth perception tasks may be reduced as a result of controlling input changes (Fermüller & Aloimonos, 1993). Also in support of such a scheme, insects seem to adjust continuously the heading direction as a function of moving stimuli, as suggested by the optomotor response (see Chapter 3). For instance, the response observed in a number of insect species is generally in a different direction from that of the detected pattern motion (e.g., Lazzari & Varjú, 1990, and Borst *et al.*, 1993). Yet another example is provided by guard bees, who seem to control a stable "hovering" position by responding to the small positional changes of a fixated pattern (Kelber & Zeil, 1990).

Inspiration may also be found in the mechanisms of "oculomotor pursuit" observed in the human visual system (e.g., Lisberger *et al.*, 1987). In particular, it appears that eye orientation is determined as a function of the torque required to maintain the current eye position (Schnabolk & Raphan, 1994). Thus, for a vehicle, it may be advantageous to couple the steering torque with the angular velocity of a tracked edge.

Finally, in order to assist navigational tasks, it may be possible to incorporate mechanisms whereby the relative motion of patterns are learnt. Such patterns could be environment-specific, and, if recognised at a later stage, may be used to provide additional cues. Indeed, it appears that bees can recognise the correct path towards a "reward", based solely on relative pattern motion (Lehrer, 1990), and are able to "store" task-specific cues (Lehrer, 1994).

5. Discussion

While it may be possible with the bugeye to infer the presence of objects, and to detect whether the current heading direction is on a collision course with an obstacle, there remains a fairly high degree of uncertainty, due in part to measurement errors, and also because these interpretations are based on likelihood and past experience. In other words, it is often possible

to find counter-examples where markedly different conditions stimulate the sensor in a similar way. Therefore, it is important that the control system be aware of the general context (e.g., inside a building), as well as of the sensor's limitations, in order to minimise the risk of critically misinterpreting sensory data.

An important question is the extent to which the overall performance of the control system is affected by the limitations of the bugeye. It may be assumed, for instance, that both the reliability and efficiency of a control system should be enhanced by using a more accurate sensor than the bugeye. If the characteristic of an efficient control system is that path planning is optimal, in the sense that travelling distances are minimised, then it could be argued that efficiency is indeed a function of sensing accuracy. As far as reliability is concerned, however, the importance of sensing accuracy is mitigated by mechanical constraints. For instance, the speed with which a vehicle reacts to motor stimuli may not be commensurate with the capabilities provided by fast and accurate sensing. In other words, inertia and other mechanical characteristics could be considered as the limiting factors in terms of reliability. Also, it is debatable whether precise measures of depth and egomotion are prerequisites for "safe" navigation. Instead, piloting may ultimately be more reliant on perception being robust and fast, albeit crude, than accurate, but intrinsically complex and slow.

In support of the argument, it may be useful to consider the underlying nature of perception. In biological systems, perception is based on the detection of observable phenomena, whose occurrences trigger fairly well behaved neuronal responses (see Chapter 3). The "observables", such as colour, the presence (or absence) of motion, its direction, and so forth, are robust indications in that they rely principally on reactions to stimuli being consistent. In this respect, even relative velocity could be considered to be an observable, as an assessment of self-motion does not seem to be critical to velocity perception (Brenner, 1991 - see also Smeets & Brenner, 1994). In contrast, measuring depth and assessing self-motion are accomplished through further interpretation of the observables, and require knowledge of the physical properties of the imaging system and of its "calibration" (Brady & Wang, 1992).

It may be difficult to determine how far successive interpretations of measurements should be carried out while keeping a reasonable degree of confidence in the resulting perceptions. For instance, Wolf & Heisenberg (1990) have found that a model which incorporates second-order computations, such as visual acceleration, best fits experimental data obtained from the optomotor response of the fly. However, Borst *et al.* (1993) argue that the outputs of the local motion detectors of bees undergo no more than spatial integration and temporal filtering, thus suggesting that visually-induced motor control need not rely on acceleration obtained from velocity estimates.

Another aspect concerns the determination of heading direction. It appears that humans make use of depth cues as well as of optic flow in order to estimate the heading direction, particularly under visually "noisy" conditions (Van den Berg & Brenner, 1994). Also, psychophysical evidence suggests that primates may extract heading direction and depth information simultaneously (Perrone & Stone, 1994). Given the nature of biological perception, though, depth information tends to be represented qualitatively (e.g., obstacles are "close" or "far away") instead of being expressed in terms of precise metric measurements.

In view of the sometimes contradictory evidence, it would be unwise to conclude beyond doubt that the proposed interpretations of the motion information provided by the bugeye would be sufficient for piloting a vehicle. In particular, the velocity and range estimation schemes are subject to sometimes significant measurement errors, and hence neither acceleration nor time to collision can be estimated reliably. However, the bugeye constitutes a robust basis upon which a control system capable of simple navigational tasks, such as collision avoidance, could be constructed.

6. Summary

The tracking schemes of Chapter 7 provide velocity as well as polarity information, i.e., the signs of the changes in contrast. Therefore, if an object presents a constant contrast and is delineated by two edges, their motion relative to the bugeye may be associated on the basis of their velocities, directions, relative angular positions, and polarities. For instance, by using panning motion, whereby the sensor is rotated about its central axis, the tracks induced by the presence of a static object indicate the same direction and are of similar velocities, but the polarities are opposite. Conversely, if the sensor is on a collision course with an object, the two tracks indicate high velocities in opposite directions, but are of identical polarity, thus defining looming conditions.

The angular positions of objects could assist in selecting a heading direction, while looming detection may be used to steer around obstacles. The motion of the autonomous vehicle carrying the sensor would then be determined iteratively, from one selected edge, or fixation point, to another. The control scheme is thus based on reacting to stimuli, as distinct from estimating the heading direction quantitatively before deciding whether it should be altered. Arguably, the scheme is feasible, as biological systems seem to rely more on robust, but crude, perception, than on precise measurements.

CHAPTER 9 Summing-up

In this final chapter, the main arguments which have been discussed in the course of the thesis are summarised, and some of the peripheral issues which have arisen are briefly addressed. Possible ways in which the design of the bugeye may be improved are outlined, followed by a brief presentation of ongoing work. Finally, a number of questions are posed concerning future research directions.

1. Thesis summary

1.1 Context

An autonomous vehicle relies on information provided by its sensors, be they visual or otherwise, in order to move freely while avoiding obstacles. Raw sensory data, however, does not provide information which can be utilised by the vehicle's control system, and hence the data needs to be interpreted. The interpretation process may be considered as a "machine" version of perception in that it assists the control system in reaching a decision concerning an adequate course of action. At the lowest control level, a direct link between perception and motor control may be established as a means of decreasing the reaction time in case of an emergency, such as the imminence of a collision with an obstacle. At a higher level, the control system may induce the detection of stimuli in an attempt to enhance its perception of the environment.

The active relationship between sensing and motor control, both direct and via high-level processes, may be viewed in a broad context as constituting the skeleton of a cognitive system. The focus of this thesis, however, is on the interpretation of visual motion, which appears to be the principal source of information used by low-level species, such as insects, for navigational

purposes. Consequently, it may be possible for a control system to interpret motion information to a level which is adequate for simple navigational tasks. In order to assess the validity of the concept, it is proposed to copy some of the characteristics of the insect visual system in a hardware sensor. Experimental results are then utilised to investigate the manner in which sensing information may be exploited by the control system.

1.2 Synopsis of the arguments

The first important discussion, early in the thesis (Chapter 2 - see also Appendix A), concerns the application of biological principles to visual processing methods. Firstly, it is argued that computational methods suffer from "information overload", in the sense that the input images are treated as raw data which require intensive processing in order to extract usable information, be it motion, object contour, and so forth. Secondly, it is generally assumed that every single pixel is potentially relevant, and hence the data is extensively manipulated before any attempt at interpretation is made. The underlying philosophy is therefore that no possibility, however remote, should be ignored for fear of deriving the wrong interpretations, and this approach usually results in high and costly computational requirements. Another drawback is the under-constrained nature of visual processing problems, which can only be solved by making assumptions. It follows that if some of the assumptions are invalid, the problems may become ill-posed.

In comparison, biological systems do not seem to have the luxury of virtually unlimited resources, in terms of processing power and storage capacity, and yet generally out-perform computers in many perception tasks. This point of view, however, is superficial in the sense that even the simplest components of biological systems appear to perform some elementary tasks, and hence the comparison of information processing capabilities is unfair. By performing simple operations, neurons at the lowest level, i.e., close to a sensor, actively participate in perception processes by interpreting the sensor's responses to stimuli (Chapter 3). The interpretation is indirect, in that it is not based on contextual assumptions, and instead reflects the underlying physical characteristics of the biological sensor. However, these characteristics have evolved to the point where they are remarkably suited to environmental conditions.

Biological evidence suggests that, at least in insects, low-level behaviour is determined primarily through the interpretation of relative motion associated with the angular positions of objects in the environment. This raises the possibility that the interpretation of motion information alone may be sufficient to steer a vehicle around obstacles, by detecting their presence, without having to recognise their precise nature. For instance, of critical importance to collision avoidance is the detection of "looming", whereby the relative size of an obstacle can be seen to increase prior to collision. In principle, the "time to collision" may be estimated from

the angle subtended by the obstacle, together with that angle's rate of expansion. Another useful measure, this time for short-term path planning, or "piloting", is the distance between the sensor and an object in its environment. The distance can be evaluated from the motion of the sensor, the angular position of the object, and the angular velocity of the object relative to the sensor. These simple schemes may be exploited in order to construct and maintain a dynamic map of the environment, as a means of controlling the heading direction of the vehicle.

While restricting sensing to the detection of motion would greatly contribute to reducing the computational load in guiding the vehicle, it remains to be seen whether real-time applications are practicable. Consequently, it is proposed to implement a motion detection scheme in hardware for two main reasons. Firstly, experimental evidence may then be utilised in order to assess the feasibility of motion-induced navigation, and secondly, it is difficult to model accurately natural environmental conditions in software.

1.3 Experimental support

The mechanisms by which motion information may be interpreted are reasonably simple, and some models lend themselves to a hardware implementation. One model, in particular, appears to be eminently suited to VLSI, as the transformation of an analog input (i.e., a time-varying image) to a digital interpretation is remarkably efficient (see Chapter 4). In essence, the scheme consists of combining locally detected changes in contrast over time and in one spatial dimension, in order to infer the direction of motion, and the sign of the change in contrast, or "polarity". The model can thus be employed to detect the current positions and the directions of moving edges, which may delineate solid objects.

The VLSI implementation of the motion detection scheme on a single chip, dubbed the "bugeye" (Chapter 5), has been successful, even though the detection of increasing contrasts has been compromised as a result of having underestimated the effects of variations in VLSI fabrication processes. The performance of the chip can nonetheless be characterised, and the directions of moving edges showing a decrease in contrast can be detected in real time, and within a range of 8 metres approximately (Chapter 6). Moreover, the relative velocities of moving edges may be estimated through integrative methods, which generally consist of tracking the displacements of the edges over time (Chapter 7). The presence of a static object whose contrast with respect to the background is reasonably constant can then be inferred by inducing relative angular motion, and by observing that the edges delineating the object should be seen to move at similar velocities, but present opposite polarities. Conversely, the detection of looming consists of establishing that two edges of identical polarities are rapidly moving away from each other (Chapter 8).

1.4 Expectations and results

Experimental results indicate that the accuracy of the velocity measurements may be sufficient for comparative purposes, but not for further computations. In other words, detecting looming conditions or the presence of objects in the environment may be accomplished in real-time, but time to collision and range estimates can be inaccurate.

In a way, inaccurate measurements appear to be a consequence of employing biologically-inspired methods, in that it reflects a particular aspect of natural vision. A characteristic of many natural systems is that they seem to be more adept at perceiving, in the qualitative sense, than at measuring, in the metric sense. However, an autonomous vehicle may still be able to utilise comparatively vague percepts, such as "slow", "fast", "far", and "close", provided that they are robust and consistent, are obtained rapidly, and can be processed efficiently. It may then be desirable to design the control system around biological principles, in order to exploit the strengths of the sensing system, whose weaknesses could thus be minimised. For instance, an evasive manoeuvre may be induced directly from the detection of looming, while navigation could, conceivably, be accomplished by keeping track of the relative velocities of selected fixation points. Further work should be undertaken to demonstrate, preferably with an experimental vehicle, the feasibility of such control schemes, based on the insect "motion-induced" navigation mechanisms.

2. Related questions

In the course of this research, a number of issues have arisen which have not been addressed in the thesis, mainly because doing so would have interrupted the flow of arguments. The following summarises some of these issues, particularly those which other researchers have found to be most relevant.

2.1 Extension of the motion detection model to two spatial dimensions

The main advantage of extending the model to two spatial dimensions would be that it might assist in resolving ambiguities in discerning solid objects from empty spaces, thus enriching the representation of the environment. However, there appears to be a number of disadvantages if real-time operations were attempted.

Firstly, the processing and storage capabilities would need to be significantly enhanced in order to cope with an increased bandwidth, thereby affecting the benefits of low cost and small size inherent to the current scheme (see for instance, Horridge, 1991). Secondly, the biological plausibility of a two-dimensional method is debatable, as suggested by experimental studies (Rubin & Hochstein, 1993). Hence, it might be sufficient to simply use another sensor, ori-

ented perpendicularly to the first, and to derive two-dimensional motion parameters through correlation schemes (e.g., Ancona & Poggio, 1993). Finally, the application domain should also be considered. Assuming that an autonomous vehicle is confined to a flat surface, and is not meant to fly, the most important consideration is for the vehicle to avoid contact with obstacles, irrespective of their shapes. Under such constraints, the purported advantages of a full two-dimensional scheme may be adversely affected by increased computational requirements. However, the "2 × 1-D" compromise is worth exploring, which leads to the next important issue.

2.2 Enhancement of depth perception through stereoscopic vision

It seems reasonable to assume that the combination of angular velocity measurements, obtained from two distinct viewpoints, would result in estimates accurate enough for reliably evaluating the distances between the sensor and detected edges. Navigational tasks may thus be facilitated, and, arguably, at a small cost regarding additional processing requirements. Moreover, the methods used for matching the motion tracks obtained from two bugeyes may well present similarities with those employed by the "2 × 1-D" scheme mentioned previously.

It may be pertinent, though, to consider stereoscopic vision in the context of biological systems. Experimental data suggests that, in humans, motion is more effectively perceived monocularly than stereoscopically (Gerbino, 1984), and that depth perception processes could begin at a monocular stage of visual processing (Fahle & De Luca, 1994). Also, as indicated in Chapter 3, a number of species appear to navigate on the basis of monocular information. Therefore, it could be argued that enhancing depth perception with two bugeyes may not be critical to low-level control tasks. However, the possibility that accurate range measurements could be obtained cheaply, in computational terms, from stereoscopic vision deserves to be investigated.

2.3 Hardware improvements

The characteristics of the bugeye are as much the result of an attempt to copy some of the important features of the insect visual system, as of technological constraints. Therefore, as revealed experimentally (see Chapter 6), it is clear that the implementation of certain analog components could be refined, and indeed, a number of possibilities are currently being investigated. The aim is to improve the characteristics of the responses to changes in contrast, while decreasing the sensitivity to illumination conditions (e.g., Chong *et al.*, 1992, and Ward *et al.*, 1993). Also, one way to improve the accuracy of velocity measurements would be to increase the sampling rate of the photo-detectors, which implies that the processing speed should be increased commensurately.

Other solutions might be more efficient. As an example, architectural modifications may be considered in order to incorporate different features also inspired from natural systems. In particular, there is no reason why the angular resolution should remain constant across the field of view, apart from an understandable desire to keep the implementation simple. In fact, supposing that the aperture is the same as in the present system, a higher resolution in the middle of the visual field than towards its periphery would be closer to the characteristics of natural systems (e.g., McKee & Nakayama, 1984). Velocity estimation towards the centre would then be more accurate than is the case at present. Conversely, the detection of motion towards the periphery could be systematically interpreted as being "fast", due to the coarser resolution, and may thus be utilised to trigger attentional shifts (see Chapter 2). Varying the resolution could be implemented either optically, with a specially designed lens, or by varying the pitch between photo-detecting elements.

3. Ongoing work and investigations

3.1 Bugeye II

The problems with the bugeye, which are described in Chapter 6, have led Ali Moini to design a new chip, dubbed "bugeye II", whose main purpose is to allow the electrical characteristics of the individual components of the analog processing stage to be measured (Moini, 1994). Only the optical characteristics, namely aperture and photo-detector separation, have been carried from the first bugeye, while analog components have been modified in an attempt to correct noted difficulties and to improve the dynamic range. Direct access to components is provided by switches, which route the analog signals to output pads according to a binary instruction, while analog signals may also be provided through input pads. The digital outputs of the contrast change detectors are loaded in parallel into a register, whose contents can then be shifted out serially. Instructions are also shifted serially, but in a distinct "scan-path", since they need not be altered each time a response is read out.

The architecture of the first bugeye is therefore quite different from the second, in that the latter contains very little digital circuitry. However, depending on the instruction, the functionality of the analog stage may be the same as that of the original bugeye. Therefore, the new chip can be used for experimentation in much the same way as the first bugeye, although a different computer interface is required. For instance, the computer reads the detector responses serially and forms the templates in software, which presents the advantage of being able to experiment with their format.

Admittedly, the new chip should have been designed before the original bugeye, at least according to safe engineering practices. However, the tight funding situation at the start of the

project (see Chapter 1) was a strong motivator, and favoured the full integration of detection and processing in the hope of developing a compelling concept demonstrator.

3.2 Tracking algorithms

Current investigations focus mainly on possible modifications to the tracking algorithms presented in Chapter 7, in an attempt to improve velocity estimation. For instance, the back-tracking algorithm may incorporate a predictive scheme similar to that of the forward tracking algorithm (Thong Nguyen, private communication). Ideally, a control system should not have to chose between different sources of information, and hence, it would be desirable to combine the advantages of both methods with a view to implement a robust velocity estimation scheme in a single hardware unit.

4. Questions for the future

4.1 Binary optics

A new technology, called "binary optics", consists of forming tiny lenses, or microlenses, by etching the surfaces of optical materials, using techniques similar to those employed in integrated circuit fabrication. In principle, the physical and optical characteristics of an individual lens can thus be tailored to specifications limited only by current microcircuit technology. Moreover, it might become possible to merge the fabrication processes of microlenses and VLSI circuits (see Veldkamp & McHugh, 1992, for a review).

The surface of a microlens is a discrete approximation of the ideal smooth surface which would produce the desired optical characteristics, and hence the photon-efficiency of a microlens is worse than that of a conventional lens. However, it is not currently feasible to utilise conventional optical technology to fabricate lenses on a scale as small as that permitted by binary optics technology. Moreover, microlenses are particularly suited to discrete receptor arrays in that the number of photons falling in between the receptive surfaces can be minimised, compared with a single lens which focuses light uniformly over the entire surface.

The emergence of binary optics technology as a viable alternative to conventional optics may, arguably, revolutionise the design of visual sensors such as the bugeye, as well as enable novel applications. Indeed, Ogata *et al.* (1994) have recently designed an imaging device based on the insect's compound eye, comprising gradient-index microlenses positioned on top of a small two-dimensional array of photo-detectors. Eventually, the impact on the design of photo-detecting elements could be significant, firstly in terms of physical dimensions, and secondly by alleviating the requirement that these elements be positioned according to sometimes tight constraints imposed by conventional lenses.

As a research tool, binary optics could be exploited to emulate the wide variety of optical characteristics found in nature (see for instance, Land, 1990), as a means of gaining insight into the manner in which different species process visual information. One could go further, and, for instance, combine "traditional" optics with the telescope-like characteristics of the butterfly (Nilsson *et al.*, 1988), thus permitting fresh investigations into building sensors whose performances surpass those of biological systems. In short, the technology may open new avenues of research.

4.2 360 degree vision

The aperture of the sensor could be enlarged, either by using separate chips, or by exploiting binary optics technology. It is clear that extending the aperture to 360 degrees would allow the detection of moving objects which are approaching the sensor from behind, and thus enhance the vehicle's capabilities. It is also worth pointing out that if the angular resolution is coarser at the back than in front, the presumed heading direction, the detection of motion could be employed as a simple alarm mechanism. In other words, motion detected at the back can be systematically considered as being "fast", and hence trigger an attentional shift towards a potential threat. More importantly, however, is the possibility that the vehicle's heading direction could be estimated with reasonable accuracy by taking into account the relative motions of receding as well as approaching objects (see for instance, Nelson & Aloimonos, 1988).

4.3 Control aspects

Combining low-level sensing information with high-level functions was briefly addressed in Chapter 8, and only a few possibilities have been mentioned. Thus, in addition to further investigations into the use of behaviour to extract information from the sensor, many avenues are left to be explored, particularly concerning the limitations of visually-induced navigation. As an example, one of the most important aspects is the control of attention in achieving high-level goals, which may require some form of object representation to be associated with environmental information, i.e., perception inferred from sensing (Tipper *et al.*, 1994).

Experimentation may be the most efficient way to go about implementing "intelligent" behaviour in an autonomous system. Arguably, simulation methods sometimes produce "ideal" results which fall short of the challenges presented by the real world, mainly because of incomplete models. Conversely, experimental evidence often points the way towards simple and effective solutions, which may turn out to be less complex than those obtained by painstakingly adapting the ideal model to real conditions. In turn, it may also be argued that experimental results can be particularly revealing for the understanding of natural processes, such as the biological plausibility of internal representations and reference systems, for instance (e.g., Soechting & Flanders, 1992).

5. Closing comments

Throughout this thesis, I have consistently, even strenuously, referred to our vast but disparate knowledge of biological systems, as I thought it necessary to underline their remarkable simplicity and variety. In doing so, however, I freely admit that the arguments can get somewhat convoluted, but this may be unavoidable.

My main motivation is to highlight some of the advantages which can be gained by using the characteristics of biological systems as a source of inspiration for accomplishing low-level sensing and control tasks. The rationale is that such tasks often rely more on the perception of essential qualities, than on precise quantitative measurements. For instance, the perception that "something is moving fast and is nearby", may be sufficient for a control system to decide on an appropriate course of action. A characteristic of biological systems is that such a simple percept can be derived from the "bottom-up" interpretation of sensory information, and may in some cases directly induce a reaction. In contrast, a "top-down" computational approach may consist of using a set of well-defined rules in order to assess the situation on the basis of velocity and range measurements, and then determining a course of action.

Hence, one advantage of biologically-inspired systems over computational systems, is that some of the low-level decision-making processes may be greatly simplified. Another advantage, as described in this thesis, is that the source of elementary perception processes may be embedded in small hardware devices whose power requirements are modest. Such devices may thus be used for applications where both cost and size are of critical importance.

State-of-the-art technology may only allow us, nowadays, to build fairly crude systems, restricted to menial tasks. However, the emergence of technologies such as binary optics, regular breakthroughs in semiconductor technologies such as Gallium Arsenide, and an increasing understanding of natural systems, augur well for the future.

APPENDIX A Visual representations and computational motion perception

This appendix is meant primarily as a reference for the discussion of Chapter 2, and is therefore descriptive rather than argumentative. There appear to be two main approaches to computational motion perception, which are treated separately before drawing some comparisons. The first approach consists of estimating motion parameters by establishing correspondences between features detected in distinct images, while the second is based on the interpretation of optical flow computed from the image intensity gradient.

1. Computing visual representations

The systematic nature of computational analysis, coupled with a minimum assumption criteria (i.e., as little as possible *a priori* knowledge), implies that the most robust strategies are those based on finding statistically predominant features in an image before incorporating other features in the analysis. Consequently, many of the detectors described here apply primarily to edges, which are often underlying components of other significant features. Due to a common objective and to a certain degree of cross-fertilisation, the proposed methods may sometimes overlap. Hence their listing in separate categories is motivated by clarity of presentation, and is not an attempt at strict classification.

1.1 Feature detection methods

Enhancement or filtering

The contours of objects present in a visual scene are characterised by abrupt changes in the intensity function of the image, which result in steep gradients of the function's derivative. Early methods were based on convolving the discretised image with an operator such as the cross-difference operator proposed by Roberts (1965). In its simplest implementation, i.e., a 2×2 pixel operator, the differences between diagonally opposed pixels are evaluated, thus providing an estimate of partial derivatives of the intensity function. The square of the magnitude of the gradient at a given point is then evaluated as the sum of the squares of the partial derivatives. Sobel suggested using a 3×3 pixel operator (Duda & Hart, 1973), whereby at the point on which the operator is centred, the partial derivatives in the vertical and horizontal directions are calculated through linear combinations of the neighbouring pixels. These operators have since been altered and refined in many different ways to accommodate various kinds of shapes.

A number of schemes developed since then exploit the observation that the second-order directional derivative of the intensity function is equal to zero at the locations at which changes occur, and are therefore commonly referred to as "zero-crossing" methods (see Figure A.1).

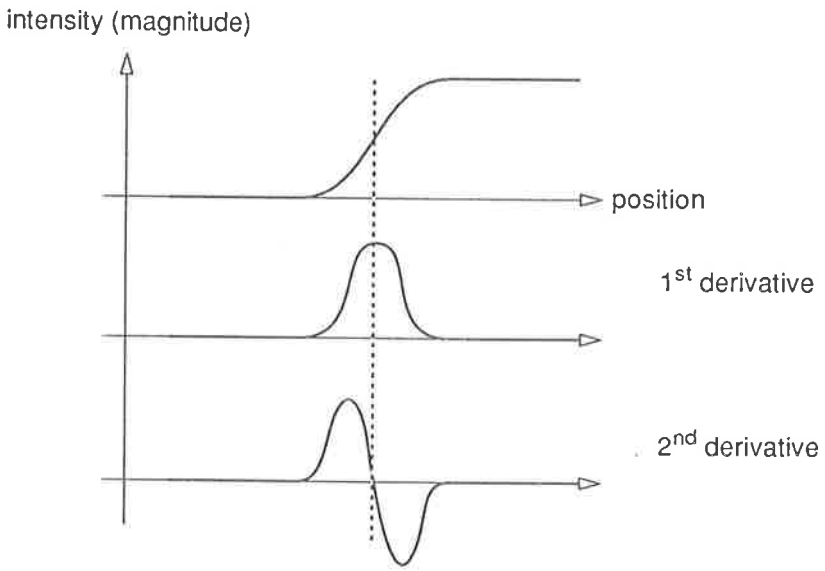


Figure A.1 Intensity function, first- and second-order derivatives

One of the most popular method, often used as a benchmark, has been proposed by Marr & Hildreth (1980). Their scheme consists of first convolving the image with a two-dimensional Gaussian filter, in order to limit the scale at which intensity changes occur (i.e., smoothing), followed by the application of the Laplacian operator, which produces the second-order differential (the use of the Laplacian is motivated by the fact that it is orientation-independent). Edge points in the image can be detected by finding the zero-crossings of the resulting function at different scales, each one set by the Gaussian's standard deviation, assuming that since the image has been smoothed, the intensity variations are locally linear. The edge points are then joined into small segments, by exploiting the fact that neighbouring points on either side of the zero-crossing have opposite signs. Lastly, an *amplitude* measure is associated with each segment. The amplitude characterises the intensity change, and is defined as the slope of the directional derivative taken perpendicularly to the segment. Together, the segments and their amplitudes form the basic elements of the primal sketch, as defined previously by Marr (1976).

Watt (1987) noted that while a form of primal sketch is biologically plausible, and indeed very likely, the methods used to obtain it do not account for the "flexible" nature of spatial representation in human vision. In particular, it is argued that the spatial scale over which representations are made is not rigid. Also, zero-crossings are susceptible to noise, and hence a more reliable way of determining edge points would consist of using the average of the responses of spatial filters. Consequently, Watt & Morgan (1984, 1985) have proposed a

model, called MIRAGE, whereby the responses of a range of spatial filters are first split into their negative and positive components, to preserve reliably the sign of the response, and then averaged separately. The resulting signals contain non-zero regions which are analysed to provide the following primitives: the centroids of each zero-bounded region are indicative of edge locations, blur is a function of the separation between the centroids, and contrast is represented by the area of the regions. The sequences of primitives are then examined to fully characterise edges.

The problem of noise in images is ubiquitous as it may be present in the image itself as well as being introduced by the sensing apparatus. Moreover, if the intensity function presents noise-induced local extrema, it is clear that several zero-crossings of the second-order derivative would occur. Therefore, characterising edges obtained through such methods requires a filtering step before computing the image derivatives, as the numerical problem is generally ill-posed (Torre & Poggio, 1986 - see also the comments at the end of this appendix). Even then, zero-crossings may not always correspond to the exact locations of edges, and may sometimes be caused by inflection points of the intensity function, thus additional tests on zero-crossing points may be necessary (Clark, 1989, Ulupinar & Medioni, 1990). Canny (1986) developed performance criteria for edge detection error rate and localisation, as well as accounting for multiple responses induced by noisy data. In effect, Canny's approach consists of providing the optimal operator depending on the edge profile and the signal-to-noise ratio. An earlier attempt at optimisation had been proposed by Shanmugan *et al.* (1979), and consisted of improving visibility by reducing resolution, or vice-versa, depending on the amount of blurring in the original image.

A variety of non-linear filters are also employed, such as median filters, which constitute the basis of order statistics detectors (e.g., Pitas & Venetsanopoulos, 1986). An advantage of non-linear filtering is that some of them present desirable characteristics under noisy conditions. Finally, a subspace method put forward by Tewfik & Deriche (1993) eliminates the preliminary smoothing step altogether, and is based on computing the eigenvalues of a matrix obtained from the discrete Fourier transform of the unfiltered image, and deriving the eigenvectors corresponding to the smallest eigenvalues along different directions. The local minima of the FFT of the eigenvectors yield the edge points. While this last technique does produce good results, it is computationally intensive, especially for a large number of edges, and its performance is adversely affected in the presence of highly correlated noise.

Morphological operators

Morphological filtering consists of applying an operator, called a "structuring element", to an image in the same way as a convolution mask. However, each pixel is evaluated as a logical combination of the structuring element and the image neighbourhood defined by the structur-

ing element, and hence morphological filtering is sometimes considered as a special case of non-linear filtering. The basic and most commonly used operations of *dilation* and *erosion* can be described as follows: strictly speaking, the dilation of an image by a structuring element is the set of points such that the intersection of the structuring element and the corresponding image neighbourhood is not empty, while erosion consists of finding the set of points for which the structuring element is included in the image neighbourhood (Serra, 1986).

When applied to greyscale images, the dilated image is computed by evaluating the maximum of the sum of the structuring element's gray levels with each corresponding pixel of the original image, while the eroded image is determined by the minimum of the differences (Sternberg, 1986). Lee *et al.* (1987) have proposed an edge detector based on evaluating the difference between the original image and its dilation or erosion, thus providing an edge strength map. Many filters are based on combinations of dilation and erosion, namely opening (i.e., dilation followed by erosion) and its dual, closing (i.e., erosion followed by dilation). A formal treatment of those and other combinations has been formulated by Serra & Vincent (1992) and Haralick *et al.* (1987).

Surface fitting

A somewhat different approach to the above consists of fitting a surface element to a neighbourhood of pixels, and is based, for instance, on least-square error methods. The surface's gradient is used as an estimate of the rate of change of the image's gray levels. The concept of surface fitting in two dimensional arrays was originally proposed by Prewitt (1970), and generalised by Morgenthaler & Rosenfeld (1981) to any dimension, thus permitting the application of the concept to edge detection in volumetric representations. The types of surfaces used are generally cubics, hence the idealised surface's derivatives can easily be computed to yield edge points.

A number of variations of the surface-fitting method have been proposed. For instance, Nalwa & Binford (1986) have suggested a successive approximation method using combinations of one-dimensional surfaces (i.e., surfaces constant in one direction). The direction of variation is first approximated by fitting a planar surface to an image neighbourhood and refined with a cubic surface fit. A one-dimensional hyperbolic tangent ("tanh") surface fit in the specified direction then localises an edge element, or *edgel*, which is a short linear segment characterised by its direction and position. The use of the tanh as the basis for surface-fitting is motivated by its reduced approximation error for step edges, and also by the fact that its combinations adequately describe roof and line edges. In a similar vein, Morgenthaler (1981) proposed superposing a low-order polynomial surface function to a local picture model of a step edge in an attempt to accommodate edges as well as ramps. Haralick (1980) advocated fitting the image with small planar surfaces (or "facets") on the assumption that edges will cause the

parameters of adjacent facets to be inconsistent. Subsequently, Haralick (1984) proposed fitting the neighbouring pixels of a supposed edge with a cubic polynomial, computing its second-order derivative in the direction of the gradient, and thresholding the slope at the zero-crossing to validate the original hypothesis.

Texture

While edges are marked by abrupt changes in the intensity function, they may also be viewed as the boundaries of similarly textured areas. Texture can be defined as the spatial organisation of local tonal primitives (Haralick, 1979), and has been used in combination with spectral features in remote sensing imagery (e.g., Haralick & Shanmugan, 1974, who have proposed a set of computable texture features for image classification). For instance, to detect edges in images described by texture models, Davis & Mitiche (1980) have proposed computing for each pixel the differences of averages of the image function between adjacent neighbourhoods, followed by thresholding steps to eliminate non-edge points. Among others, Geman *et al.* (1990) investigated the use of relaxation methods.

Texture information can also be utilised to recover orientation. Witkin (1981) found that the distortion of texture elements under projection is systematic, assuming that the elements have random direction, and hence by inverting the projection it should be possible to recover the orientation of the image by choosing the most uniform reconstruction. Davis *et al.* (1983) then proposed more efficient algorithms for estimating the maximum likelihood of the orientation, while Ikeuchi (1984) advocated the use of a spherical projection model, instead of the orthographic projection originally used by Witkin. Brady & Yuille (1984) put forward a method whereby orientation is recovered from a contour by maximising the ratio of its area to the square of its perimeter. This last method works best for regular shapes, while the previous are based on assumptions concerning the characteristics of texture elements. Alternative methods make use of spectral information to recover orientation by analysing the change in texture density caused by the projection (e.g., Jau & Chin, 1990).

Contours

A contour may be described as a connected set of short linear edge segments, each comprising a small number of edge points. Therefore, additional processing is required to obtain contours from the edge points unless, of course, the boundary or edge detection operator already provides such a representation. Apart from the method advocated by Marr & Hildreth (1980), which yields short linear segments, there are a number of different approaches to linking edge points into segments, which are then connected to form contours. Among others, Nevatia & Babu (1980) have suggested detecting edges by convolving the image with edge masks at several orientations, followed by linking short edge elements on the basis of proximity and orientation criteria. Edge points may also be linked by iterative methods, such as that proposed by

Williams & Shah (1990), who begin by placing the sets of possible edge points in priority queues, each set corresponding to a scale of the edge detection operator. Starting from a known edge point, the next point on the contour which best fits the direction, magnitude, or curvature, is selected, until the contour is either closed, or the queue is empty, in which case the search proceeds at the next finer scale. Other techniques are based on heuristic search algorithms (e.g., Martelli, 1976) whereby the "cheapest" path from a beginning to an end point is determined.

1.2 Managing the computational load

Computational motion perception schemes are based on either optical flow or correspondence matching. In the latter case, strategies generally consist of evaluating the displacement of objects defined by shapes represented as contours (or a subset thereof), or by similarly textured areas. However, it is not always necessary to fully characterise the shape or the contour of an object. The computational load may be significantly reduced if motion can be inferred from the displacements of so-called "dominant" or "salient" points which are particularly interesting as they may be relatively insensitive to scaling and orientation (Ansari & Delp, 1989). In order to obtain such points, some methods may employ conventional techniques to detect edges before determining which edge points are also corner points (Xie *et al.*, 1993), while other methods consist of finding discontinuities or curvature extrema along a contour (e.g., Fischler & Bolles, 1986, extended by Fischler & Wolf, 1994). Alternatively, on the basis that many edge detectors do not perform well in the presence of corners, other approaches make use of specially designed operators to detect curvature extrema within small neighbourhoods (Seeger & Seeger, 1994). Similarly, Mehrotra *et al.* (1990) suggested detecting half-edges oriented in different directions.

The process of finding salient features may be viewed as a somewhat random form of discrimination, in the sense that it is based on the analysis of local discontinuities, which is carried out on the entire image. By contrast, segmentation techniques proceed from systematic analyses, from coarse to fine resolutions, and may thus be considered as the computing equivalent of the pre-attentive mechanisms found in natural vision. However, the intent of segmentation is often to partition an entire image into areas of interest which can then be individually analysed, whereas attracting the focus of attention implies providing a preliminary clue which may, or may not, result in further analysis.

Morphological filtering is well suited to a top-down approach to segmentation, as shown by Salembier & Serra (1992) who employ operators at progressively finer scales. Each step consists of extracting the features detectable at that particular scale, using those features for segmentation, and then determining where processing at the next finer scale should be carried out

to more precisely define area contours. In a similar vein, Wang *et al.* (1993) use a recursive algorithm, where at each step the size of the structuring element is chosen as a function of an error estimation, and its position for each pixel as a function of the neighbouring pixels.

In contrast to segmentation methods using morphological filtering, which is based on feature size, Lifshitz & Pizer (1990) propose a decomposition into nested light and dark regions, thus creating a tree structure obtained by progressively blurring the image and following the paths of intensity extrema. Each region is delineated by an "iso-intensity" contour, in the manner of a geographical map which incorporates elevation. Other forms of decomposition have been proposed, notably in the case of noisy images which may be better partitioned into regions where the intensity variations are small (Veijanen, 1994). Finally, distinct sources of information, obtained through various feature detection methods and from different types of sensors, may be combined in an attempt to improve the accuracy (Chu & Aggarwal, 1993).

1.3 Towards shape characterisation

Features extracted from an image may be combined to provide a formal description which is then used for further tasks, though it should be emphasised that, in general, the proposed descriptions are not meant to be task-specific. In the case of motion estimation, however, exhaustive shape characterisation is not always necessary, and it may be advantageous to trade-off accuracy of representation for improved efficiency. Finding the optimal trade-off is a delicate balancing act, as it consists of carefully exploiting circumstantial assumptions without overly confining the application domain.

Whether the ultimate goal is to recognise objects or to compute motion, depth information is often essential to resolve ambiguities. In limited cases, however, a single view of an object may be successfully interpreted to recover its three-dimensional shape, by making a number of assumptions. For instance, some artificial objects can be described almost entirely in terms of planar surfaces and parallel or more or less symmetrical lines (e.g., Kanade, 1981). Alternatively, the curvature of some surfaces may present such characteristics that contours provide sufficient information for a 2-D description (e.g., Rom & Medioni, 1993) or even for 3-D shape recovery (Ulupinar & Nevatia, 1994). In a wider context, though, preliminary assumptions tend to fail, and hence more robust methods utilise more than one view. In particular, structure may be inferred from stereo images, thus drawing on the natural visual systems which extract depth information partly from binocular disparities. In terms of computations, therefore, the two major steps consist of establishing a point-to-point correspondence between the two images (the "correspondence problem"), and then using the associations to determine distances.

Basing their work on psychophysical evidence, Marr & Nishihara (1978) have suggested that early visual processes result in the formation of an "orientation and depth map", dubbed the "$2^1/_2$ -D" sketch, and thus constitutes an intermediate representation between detected features and a full object (or scene) characterisation. Marr & Poggio (1979) proposed that binocular images be first filtered with masks of various orientations and whose sizes increase with eccentricity. Filtering is then followed by a feature extraction step, using zero-crossing methods, which is accomplished independently on each image. Finally, detected features are matched between both images, within a range of disparities corresponding to the size of the filter, and on the basis of the zero-crossings at the feature locations being in the same direction in each image (see also Grimson, 1981). Volumetric primitives can then be obtained with a view to fully represent 3-D shapes in a modular and hierarchical fashion (Nishihara, 1981).

The notion that distinct "primal sketches" are to be computed independently in each visual channel was subsequently challenged by Mayhew & Frisby (1981), who conjectured that binocular matches are effected as early as the feature extraction stage, and that correspondences are established between different spatial-frequency-tuned channels. In fact, physiological evidence suggests the presence of correspondence-matching mechanisms at the earliest stages of binocular interaction (Poggio & Poggio, 1984). It thus appears that solving the correspondence problem is inextricably linked to motion estimation and structure recovery, which will be addressed in the next section.

2. Computational motion estimation

Establishing the correspondence between two images is a fundamental problem in computer vision, as it permits the recovery of depth information from stereopsis as well as constituting the basis of a large number of motion estimation algorithms. Moreover, the latter are often used either in conjunction with stereopsis for 3-D shape characterisation, or as an alternative. Therefore, it is hardly surprising that similar matching algorithms are sometimes used in different contexts. Commonly used techniques are briefly reviewed, before focusing on motion estimation strategies, which appear to be based on either optical flow principles or feature matching techniques.

2.1 Image correspondence

Generally speaking, matching strategies operate either on features such as edges, corners, and other salient points, or are area-based (Dhond & Aggarwal, 1989). Feature-based strategies are often the more economical, as relaxation techniques operating on a limited set of points may be employed (e.g. Ranade & Rosenfeld, 1980, who proposed one of the earliest method), but their performance is a function of the conspicuousness of those features. Thus in

the presence of noise, it is sometimes necessary to rely on best-fit approaches (e.g., Boyer & Kak, 1988). By contrast, area-based techniques employ correlation methods to establish correspondences between intensity patterns in local neighbourhoods, but tend to be confused in the presence of occluding boundaries and distortion. Since neither method is optimal, Cochran & Medioni (1992) recently suggested that area- and feature-based methodologies be combined in order to minimise matching errors, while another approach consists of merging the feature extraction and matching stages (Jones & Malik, 1992). The outputs of spatial filters at different scales and orientations are used to compute a disparity map, which is then iteratively refined through a process which exploits surface smoothness and geometrical constraints. Hoff & Ahuja (1989) have also advanced the idea of combining different processes, by interpolating surfaces at progressively finer resolutions. The surfaces are obtained at each step through a feature matching process, followed by planar and quadratic surface fitting to extract contours.

Approaches to dynamic correspondence (i.e., matching in a time-varying sequence of images) have been broadly classified into two categories (Aggarwal *et al.*, 1981). The first consists of identifying areas likely to contain moving objects and constructing an iconic representation, or template, while the second involves matching structural models in successive frames. As in stereopsis, some constraints may be exploited to reduce the computational load, such as limiting the number of frames under consideration by assuming that portions of the image sequence contain sufficient motion information (Lee & Joshi, 1993). Alternatively, Sethi & Jain (1987) make the assumption that motion is generally smooth, and have cast correspondence as an optimisation problem through the use of a coherence function. Such methods also appear to be particularly suited to neural network approaches (e.g., Chang *et al.*, 1993).

Several methods have been proposed to reduce the errors due to flawed assumptions and constraints. For instance, the combination of stereoscopic and dynamic correspondence processes may result in relaxing the assumed constraints of either process (Liu & Skerjanc, 1993). Alternatively, the constraints used in one process can serve to facilitate the other matching process (e.g., Jenkin & Tsotsos, 1986). Also of note is the use of neural networks, which may be beneficial in that constraints are derived through an optimisation process, instead of being explicitly formulated (Khotanzad *et al.*, 1993).

Lastly, tracking objects in a sequence of images also involves solving a form of correspondence problem (sometimes referred to as "data association"), although, strictly speaking, it may be considered as an integral part of motion estimation and structure recovery methods. Briefly, to illustrate the similarity, tracking normally consists of matching features detected through conventional techniques in successive images, based on predictions of the features' motion. Deriche & Faugeras (1990) have suggested employing a similarity function which uses the distance between the features' attributes to determine the best matches, within a search area

obtained through Kalman filtering methods. The measure used here is the *Mahalanobis* distance, which is derived from the covariance matrices of candidate feature vectors (see also Cox, 1993, for a review of various techniques).

2.2 Motion from feature correspondences in image sequences

Motion can be computed by evaluating the displacements of features obtained from a time-varying sequence of images. This approach to motion analysis thus involves solving the aforementioned dynamic correspondence problem, which may be a non-trivial task when the effects of occlusion caused by bodies in motion are considered, not to mention luminance changes, structural discontinuities, and so on. Assuming that all these difficulties have been overcome, the displacement of a point p between times t_0 and t_1 can then be expressed mathematically as follows (Huang & Netravali, 1994):

$$p_1 = R \times p_0 + T$$

In the above equation, R is a matrix representing rotation, T is a translation vector, and p_0 and p_1 are the point's vector coordinates at times t_0 and t_1 respectively. The dimensions of the vectors and of the matrix depend on the type of representation, thus, a 3-D to 3-D correspondence will entail finding twelve unknowns (9 rotation and 3 translation). However, the rotational matrix can be specified in different ways, for instance as three successive rotations around the coordinate axes, or as rotation around an axis passing through a pre-defined point. Hence, the components of the matrix are not independent, and in the 3-D to 3-D case yield six free parameters. Another quantity often used is the so-called E-matrix, the product of R by T, which, under certain circumstances, presents some desirable mathematical properties permitting the recovery of missing correspondence points (Huang & Faugeras, 1989).

The methods for recovering motion parameters (i.e., determining the components of R and T) differ mainly in terms of the number of frames and correspondence points under consideration, the type of projection (orthographic or perspective), and whether binocular or monocular sequences are employed. Of particular interest is the number of possible solutions derived from the motion equations, and is a function of the number of correspondence points. To constrain the solutions, it is often assumed that objects undergoing motion remain rigid, which seems reasonable as long as a small number of frames, encompassing a short duration, are considered. In fact, the problem of non-rigid motion can be addressed incrementally, by modifying an internal object model by the minimal change that accounts for the transformation (Ullman, 1984). However, most of the methods, which will now be briefly described, employ only two or three frames (see also Aggarwal & Nandhakumar, 1988, for a review).

Recovering structure

Ullman (1979) was among the first to propose a method which uses motion to determine structure. The structures of groups of four non-coplanar points are first recovered from three orthographic projections, followed by independent rigidity tests for each group, which are finally combined to provide a full description. Roach & Aggarwal (1979, 1980) employed the same number of points as Ullman, but considered the more realistic, and more complex, perspective projection model instead of the orthographic projection.

A general solution to the problem of reconstructing a scene from two perspective projections arbitrarily separated in space and time, was formulated by Longuet-Higgins (1981), and employed eight point correspondences in order to reduce the problem to a set of linear equations. An identical number of correspondence points was also used by Tsai & Huang (1981 - see also Tsai *et al.*, 1982), who determined the translation parameters, assumed to be non-zero, of small planar surfaces from the solutions of linear equations. Tsai & Huang (1984) subsequently addressed the problem of finding a unique solution to the set of equations, while Zhuang *et al.* (1986) eliminated the translation assumption. Also of note is a method recently proposed by Kontsevich (1993) whereby images in a sequence are compared pairwise to provide linear criteria from which depth can be inferred.

As it is believed that natural binocular visual systems make use of cues provided by stereo and motion to recover the structure of objects in a scene, the principle of integrating both structure-from-stereo and structure-from-motion approaches may prove to be beneficial. Moreover, the problem can be reduced to solving a set of linear equations for which a unique solution exists (Richards, 1985). Strategies generally consist of obtaining three-dimensional features by using the depth information provided by stereo correspondences, followed by matching those features in a sequence of images, and lastly computing the motion parameters. To match the three-dimensional features, Kim & Aggarwal (1987) employed a two-pass relaxation method in order to reduce the possibility of false matches, while Weng *et al.* (1992b) suggested a more complex method (but less noise-sensitive) whereby the motion parameters are first approximated and then iteratively optimised.

Noisy images

The problem of noisy images has been addressed mainly by considering more correspondence points than are theoretically necessary in the ideal case, and by using more than two or three image frames, for instance a few dozen. The advantage of over-determination combined with a long time sequence, is that features may disappear for a short while, due to occlusion, but their motion may still be recovered when they eventually re-appear, by using motion smoothness constraints (Weng *et al.*, 1987). Fang & Huang (1984) studied the effects of varying the number of feature correspondences, using simulated and real data, while Weng *et al.*

(1989) employ the redundancy to estimate the noise-induced errors of the motion parameters. Arun *et al.* (1987) have proposed a mathematical solution, based on least-square error methods, to the problem of finding the motion parameters when a noise vector is introduced. Finally, a different approach has recently been suggested by Vaidya & Haralick (1993), who determine the locations of objects by exploiting the characteristics of the Wigner distribution, which permits simultaneous analyses in both the time and frequency domains.

Problem confinement

A suitable interpretation of *a priori* knowledge concerning the environment and the dynamics of objects in motion may significantly improve the efficiency of motion estimation and structure recovery methods, to the point where real-time and low-cost applications can be realistically envisaged. The rigidity assumption is widely employed, but it is not the only possibility. For instance, if motor vehicles are specifically targeted, their motion would likely be confined to planar surfaces and hence the recovery of motion parameters reduces to tracking a limited set of features, since shape and size are more or less pre-established. Dreschler & Nagel (1982) were among the first to use real image sequences, from which corner features belonging to cars were extracted and tracked. However, their method was fairly complex, requiring non-linear optimisation techniques, and more efficient algorithms have since been proposed (e.g., Tan *et al.*, 1993). The assumption that motion is constant in both translation and rotation over an extended period has also been made (Broida & Chellappa, 1986), but is probably too restrictive to be useful outside of crafted situations.

Lastly, the correspondence problem itself has been addressed somewhat radically. Huang (1986) offers a mathematical formulation whereby, if an object's shape parameters are known and if the brightness of the considered object points remains unaffected by motion, then the initial step of establishing correspondences can be avoided altogether. A possibly more realistic approach to avoiding the correspondence problem has recently been proposed by Lin *et al.* (1994), and consists of recovering the 3-D centre of gravity and autocorrelation matrix from statistics provided by the 2-D image at distinct instants. The eigenvector decomposition of the matrix then yields candidate solutions for the motion parameters.

2.3 The aperture problem

The aperture problem arises from the limited field of view of a receptive array, which, in response to a moving edge, can only constrain the direction of motion to within 180 degrees. In Figure A.2, an edge represented by the solid lines at times t_0 and t_1, moves towards the bottom-right of the picture, although the field of view, delineated by the circle, is too small for the phenomenon to be correctly interpreted.

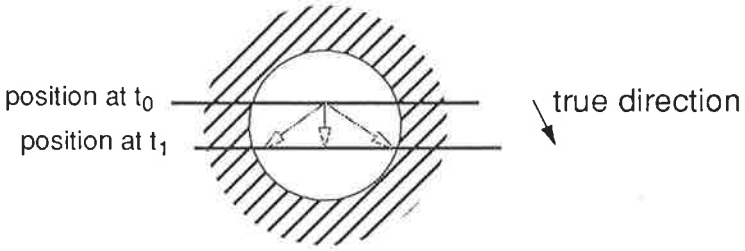


Figure A.2 The aperture problem

Following on earlier work (see the beginning of this appendix), Marr & Ullman (1981) proposed that the sign of the time derivative of the Laplacian of Gaussian convolution at an edge's zero-crossing be evaluated to constrain motion to one side of the edge segment. Hence, the motion direction can be further restricted by associating the constraints of several edge segments (Figure A.3a). Moreover, if the magnitude of the temporal derivative of the Laplacian of Gaussian were to be preserved, the velocity could be approximated by evaluating the ratio of the spatial and temporal derivatives (Harris, 1986). Another approach, based on motion coherence, was proposed by Adelson & Movshon (1982), who suggested that the constraints provided by two separate views be combined to indicate the true direction of motion (Figure A.3b).

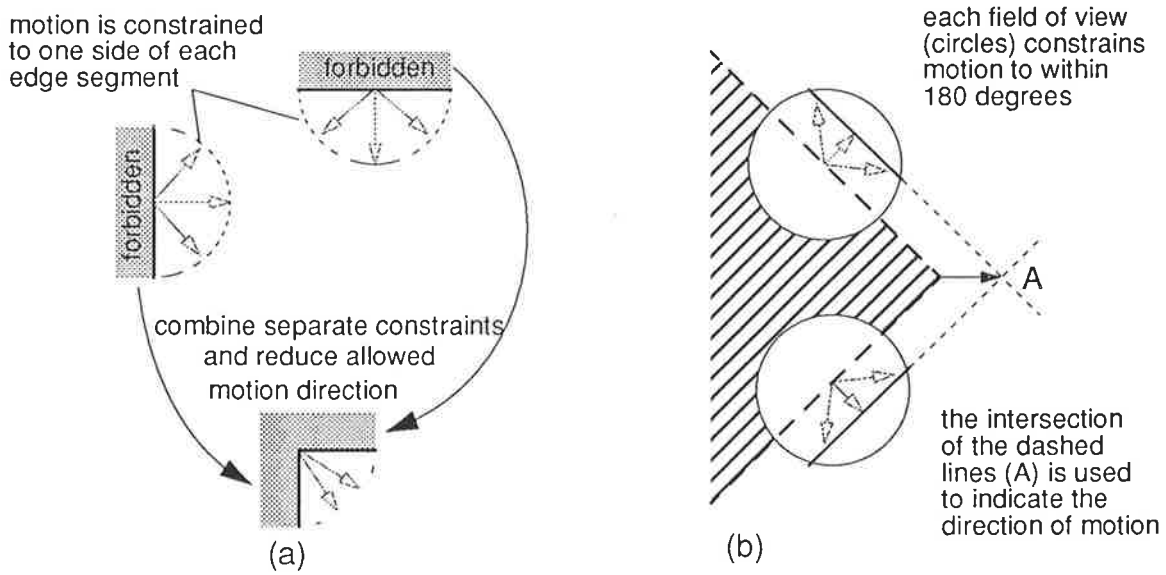


Figure A.3 Possible solutions to the aperture problem
(a) Marr & Ullman, (b) Adelson & Movshon

Yuille & Grzywacz (1988) have argued that in order to fully account for the assumption of coherent movement, local measurements of motion should be combined over large areas. Their technique consists of computing an energy function, which is obtained by combining the available local velocity measurements with a smoothing function. The minimum of the energy function yields an estimate of the overall velocity field, which is the 2-D projection of 3-D

velocities. Smoothing is meant to provide solutions for sparse data (i.e., the regions where no measurements are available), as well as to constrain the measured velocities.

It should be pointed out that the proposed methods to solve the aperture problem appear to rely mostly on measurements which provide local directional constraints and velocity estimates. Biological visual systems, on the other hand, appear to exploit the interactions between motion sensitive units, by taking into account differences in integration time constants as well as contrast sensitivities (Lorenceau *et al.*, 1993). However, the precise nature of these interactions is not clearly understood, thus their implementation remains problematic.

2.4 Optical flow

Optical flow can be broadly defined as a representation of temporal changes in the intensity values which form an image. Thus, objects in motion, as well as the relative motion of the observer with respect to the scene, give rise to three-dimensional velocities which are projected on the image plane, forming the motion field. The latter is often represented by short oriented segments (vectors) representing the displacements of image points in time. Optical flow is an estimate of the motion field, itself derived from first-order variations of image brightness patterns, and may therefore be distinct from the motion field, except under particular circumstances (Verri & Poggio, 1989). The aperture problem is central to the determination of optical flow, which is also affected by stationary changes in brightness.

The methods employed to recover motion and structure from optical flow generally consist of using local changes in intensity in the image to compute velocities, assuming that the overall intensity is conserved. The optical flow equation can be derived as follows (Singh, 1991).

Suppose that at time t, the intensity value of a point in the image is $I(x, y, t)$, where (x, y) are the point's coordinates, and the implicit assumption is that $dI/dt = 0$. If the point is in motion, relative to the viewing apparatus, the same intensity will be found in another location $(x + \Delta x, y + \Delta y)$, and at time $t + \Delta t$, thus:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$

A Taylor expansion can be used to approximate the right-hand-side of the equation, which yields:

$$I(x, y, t) = I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + \text{higher order terms}$$

By ignoring the higher order terms, and if the time interval Δt is small enough, the instantaneous optical flow equation can be written as:

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0$$

In the optical flow equation, the velocity components $u = dx/dt$ and $v = dy/dt$ are the unknowns, since the partial derivatives can be estimated from the image. The optical flow equation constrains the solutions, but does not specify a unique solution. Also, quantisation errors imply that the image intensity will not necessarily be conserved, therefore additional assumptions are required, such as smoothness of motion.

Computing motion parameters

Horn & Schunk (1981) formulated the motion smoothness constraint as the minimisation of the "deviation from smoothness", which is derived from the square of the magnitude of the spatial gradient of the optical flow velocity. This approach may be generalised by taking into account the previously ignored second-order terms of the Taylor expansion, in which case two non-linear equations can be derived for the two unknowns (Nagel, 1983). The equations may then be simplified when considering the intensity levels corresponding to corner features, thus permitting the application of an "oriented smoothness" constraint, which was formulated by Nagel & Enkelmann (1986). Knowledge of a contour may also be employed to restrict the possible solutions, as suggested by Hildreth (1984), who assumed that surfaces are generally smooth in order to derive a unique solution by integrating the velocity components along a closed contour. However, the latter method may be considered as being a special case of the more general "oriented smoothness" approach (Nagel, 1987). Least-squares error techniques are often used to solve the smoothness constraint, although it may be more efficient to begin by partitioning the image into regions at different resolutions, depending on local intensity variations, thus reducing the computational requirements for uniformly textured areas (e.g., Kirchner & Niemann, 1992, who use triangular elements of various sizes). Similarly, if sub-sections of the image present relatively noise-free contrasts and distinct orientations, the overall motion parameters may be adequately recovered from combinations of locally reliable optical flow estimations (Negahdaripour & Lee, 1992).

The optical flow equation based on the intensity gradient is not the only constraint which can be applied to the velocity components u and v, and in fact, any function specifying motion-invariant properties may be substituted for the intensity, thus yielding a system of equations in two unknowns. The directional derivative of the image function has been employed (Wohn *et al.*, 1983), but may be somewhat unreliable as the direction of motion can vary. Other functions, perhaps more appropriate, may be obtained by applying spatial operators to the image intensity to compute local properties, such as contrast and local averages (Mitiche *et al.*, 1987). Also, Koenderink & van Doorn (1976) have shown how the motion of a sufficiently small neighbourhood may be represented by elementary components of translation, rotation and expansion (or deformation). Consequently, changes in brightness may be decomposed in a similar fashion (Verri *et al.*, 1990), and angular and three-dimensional velocity can then be extracted from these components (De Micheli *et al.*, 1993).

Lastly, the optical flow equation may be considered, for a given point in the image, as the representation of a line in the velocity space defined by u and v. The velocity estimates from a neighbourhood of points would tend to intersect within a small area, provided that the motions are similar, and hence a number of methods have been proposed to recover the best estimate, based, for instance, on the Hough transform (Ben-Tzvi *et al.*, 1993). Furthermore, instead of using a cartesian coordinate system, the analysis of discontinuities may be facilitated by re-formulating the equation in polar coordinates (Schunck, 1986). Using the polar form of the equation, Schunck (1989) derived a one-dimensional form of cluster analysis to extract the motion estimate from contradictory data (i.e., in the presence of discontinuities).

Segmentation and structure recovery

The optical flow induced by the independent movements of several objects, coupled with the observer's own motion, gives rise to ambiguities caused by apparently contradictory directions, not to mention occlusions. However, the recovery of the motion parameters of distinct objects, and hence their structure, may be facilitated by segmenting the flow field into groups of vectors which are consistent in some way or another. For instance, assuming that the objects are composed of roughly planar surfaces and that motion is rigid, Adiv (1985) suggested that the three-dimensional motion parameters could be recovered by finding which of these groups are compatible. Stochastic methods have also been investigated, notably by Murray & Buxton (1987), who have modelled the optic flow field as spatio-temporal Markov random fields and used simulated annealing to optimally segment the image. More recently, Konrad & Dubois (1992) have employed the Bayesian reconstruction of images proposed by Geman & Geman (1984) to estimate motion, using simulated annealing and an iterative averaging method.

Longuet-Higgins & Prazdny (1980) have formulated a system of non-linear equations which relates optical flow, its first and second-order derivatives, and structure, thus showing that the structure of a static scene can be recovered from the instantaneous optical flow. However, it appears that the motion of the observer relative to the scene cannot be uniquely specified, as the projection of the instantaneous velocity field is purely translational (Prazdny, 1983). Also, while the system of equations is over-determined, its solutions are not always unique as the equations may become inter-dependent under certain circumstances. Most techniques, therefore, employ restrictions on the shape of objects in the scene and the nature of the motion.

For instance, Lawton (1983) assumes that the observer is undergoing translation in a direction corresponding to a unique focus of expansion. The direction which minimises the mismatch between a number of features in a sequence of images is then selected, thus permitting the recovery of the three-dimensional structure. Conversely, it has recently been shown that depth information may be extracted from pure rotations (Hadani *et al.*, 1994). However, more

degrees of freedom are necessary for the general case of an observer navigating in an unknown environment. Rieger & Lawton (1985) utilised the observation that, in neighbourhoods presenting large depth variations, the orientations of the differences in velocities point towards the focus of expansion, from which the translational component of motion at every image point may be obtained. The velocity components of a moving observer can then be recovered, since motion perpendicular to the translational direction would be due to rotation. Hildreth (1992) then extended the method to account for a dynamically varying scene.

As an intermediate step towards recovering the relative motion of more complex shapes, the case of planar surfaces has received special attention. The problem is that the interpretation of the optic flow arising from the motion of a planar surface relative to an observer is not unique, except in special cases, such as the observer moving directly towards the surface. Longuet-Higgins (1984) showed that the observer's velocity and the distance between the observer and the planar surface can be expressed as a ratio, but not as independent quantities. Also, the orientation of the surface with respect to the direction of motion of the observer is ambiguous, although the correct interpretation may be that for which all the image elements are in the observer's field of view.

Waxman & Ullman (1985) proposed a formulation, applicable to any curved surfaces, based on the use of deformation parameters which describe the rate of change of an image neighbourhood (see also Waxman & Wohn, 1985). A closed form solution may be derived for planar surfaces, and the ambiguities resolved by considering either two distinct surfaces, or the same surface at two different instants (Subbarao & Waxman, 1986). A similar approach was taken by Kanatani (1987), who specified the velocity components u and v in terms of parameters which reflect the invariant characteristics of the projected flow with respect to coordinate changes. The parameters are derived by expressing the one-to-one correspondence between a point in the scene and its projection on the image plane. Surfaces belonging to the same object are recovered on the basis of their having induced optical flows which have a fixed intersection (the "adjacency condition"). Recently, Shu & Shi (1991) have extended the instantaneous optical flow equation by incorporating observer position with the intensity distribution, and then showed how surfaces characterised by an n^{th} degree polynomial may be recovered using stereo images, provided the degree of the polynomial is known beforehand (Shu & Shi, 1993).

Another aspect concerns extending the use of temporal information in order to account for non-uniform motion. Provided the transformations are smooth, a closed form solution may be found for the system of equations relating the instantaneous interpretation of the optical flow to its first-order temporal derivatives (Subbarao, 1989). A fairly similar approach has been suggested by Bandopadhay & Aloimonos (1991), although candidate features, i.e., the locations where contrast disparities are significant, are matched in a sequence of images. The smooth-

ness constraint is formulated by assuming that, for a given feature, the estimated optical flow and the depth vary proportionally, to a first order approximation, thus restricting the possible matches. Lastly, first order temporal variations can be estimated by subtracting images from one another, and convolving the difference images with standard feature detection operators to provide the locations of moving features (e.g., Haynes & Jain, 1983). However, stationary illumination changes will be indistinguishable from motion. To account for this problem, Duncan & Chou (1992) proposed that the second time derivative of the Gaussian smoothing function be convolved with the intensity history at each pixel. Only moving features, not illumination variations, produce zero-crossing contours in the resulting images, which combine information over a number of successive frames. The resulting images may also be used to compute the normal components of the optical flow at the zero-crossings.

3. Comments

Some differences between optical flow and feature-based techniques

Feature-based motion estimation methods generally require four distinct steps: filtering, feature extraction, establishing correspondences, and lastly estimating motion parameters. As a consequence, since a number of sometimes very different techniques may be employed at each step, the results tend to vary widely, in general as a function of the applicability of the methods to various types of images. In comparison, optical flow methods only require two steps, i.e., computing optical flow from the intensity gradient, followed by an interpretation step to estimate the motion parameters. In a sense, the establishment of correspondences is implicit in the optical flow computation. By contrast with feature-based techniques, it has been observed that the different methods to compute optical flow tend to produce similar results, which may be attributed to the simplicity of the spatial structure of the motion field produced by moving rigid bodies (Verri *et al.*, 1992).

When used individually, neither optical flow nor feature-based techniques can provide a complete solution to the recovery of motion information from images. For instance, optical flow methods tend to fail when the displacements from one frame to another are too large or not well-behaved (i.e., the smoothness assumption does not hold), and, conversely, a high signal-to-noise ratio is required for the estimation of small displacements to be accurate. Feature-based methods suffer most from occlusions and the possible similarities between features, thus causing false matches, and from the difficulty of reconstructing surfaces from sparse features. Consequently, it may be advantageous to develop strategies which exploit the strengths of both methods (e.g., Weng *et al.*, 1992a). Higher levels of perception may also be included, if, due to adverse circumstances, motion parameters cannot be retrieved by the application of either method. The grouping of several features under the assumption of coherence, which is a

preliminary step towards recognition, may then be employed to interpolate missing trajectories. It therefore appears that describing shapes in terms of sets of coherent salient points may not be as robust as using contour descriptors, which, on the other hand, are computationally more expensive.

Regularisation and perception

In a general context, many approaches to computational vision consist of developing techniques based upon limited aspects of early vision processes. The relationships between distinct processes are often overlooked, at least in the original problem formulation. As a result, the problems are in many cases ill-posed, and hence their resolution often relies on making a number of assumptions. Then, once a particular problem has been solved within a restricted domain, some initial constraints may be progressively relaxed. Such an "inside-out" method of generalisation is not conducive to the elaboration of truly global solutions, but it is a clear indication that the nature of the interactions between early vision processes are only superficially understood. To further complicate the matter, it is now assumed that vision processes are not purely "bottom-up" (see the discussion in Chapter 2).

Consequently, the need arises for an approach which unifies the common characteristics of early vision processes. In recent years, Poggio *et al.* (1985) have addressed the issue of ill-posed problems globally, by formulating the so-called "regularisation theory". Essentially, the theory consists of utilising suitable *a priori* knowledge, which is meant to relate to the general statistical properties of the problem formulation, instead of only considering context-specific aspects of the problem. The application of regularisation theory has led, for instance, to methods for continuously controlling the smoothness constraint, particularly in the presence of discontinuities (Terzopoulos, 1986). Three-dimensional shape modelling is another area which has benefited from this approach, as surface models can be made to adapt to local or time-varying deformations (e.g., Terzopoulos, 1992).

It should be noted that if, as argued by its proponents, regularisation theory is indeed physically plausible, the "criterion for perception" proposed in Chapter 2 may be validated. Moreover, consider the subtraction motion estimation methods mentioned at the end of the previous section on optical flow techniques. These methods may be viewed as possible implementations of the hierarchical motion perception scheme used as an example in Chapter 2, where each step produced measurements starting from changes in luminance, followed by motion directions, and so forth.

Image compression

A trait common to many methods is that they appear to be based on systematic analyses of images which require a large memory capacity combined with substantial processing power.

One way to address this problem is by compressing the signals obtained from the sensing apparatus before they are processed, keeping in mind that useful information be preserved in a form which is suitable for further processing. Compression does not necessarily preclude feature extraction from either the restored image (e.g., Reichenbach *et al.*, 1993), or even the compressed image (McLean, 1993), however it remains to be seen whether traditional methods would be optimal. Of particular interest, therefore, are schemes which attempt to reproduce some of the characteristics of animal perception. For instance, the phenomena of noise and distortion masking may be modelled by a non-uniform time and frequency analysis which, having identified the critical bands, also takes into account the non-stationary nature of the inputs (Jayant *et al.*, 1993).

APPENDIX B Bugeye chip operation and testing procedures

This appendix provides details concerning the bugeye's testability, interfacing requirements, and internal operating modes, with a view to clarifying the testing procedures used in Chapter 6. The implementation of one particular testable component, the processor, is also described in order to complement Chapter 5.

1. Internal structure and testability

The chip can be decomposed into a number of functional blocks, as depicted in Figure B.1. The template and save RAMs ("Random-Access Memories") are individually accessible for testing purposes, and can be written into and read from without interfering with any other block. The photo-detectors and the output multiplexer can only be tested indirectly by reading the save RAM after sampling. The processor can only be tested with its RAM, as the processor output is not fed onto the main bus. The blocks which are testable are depicted as filled boxes.

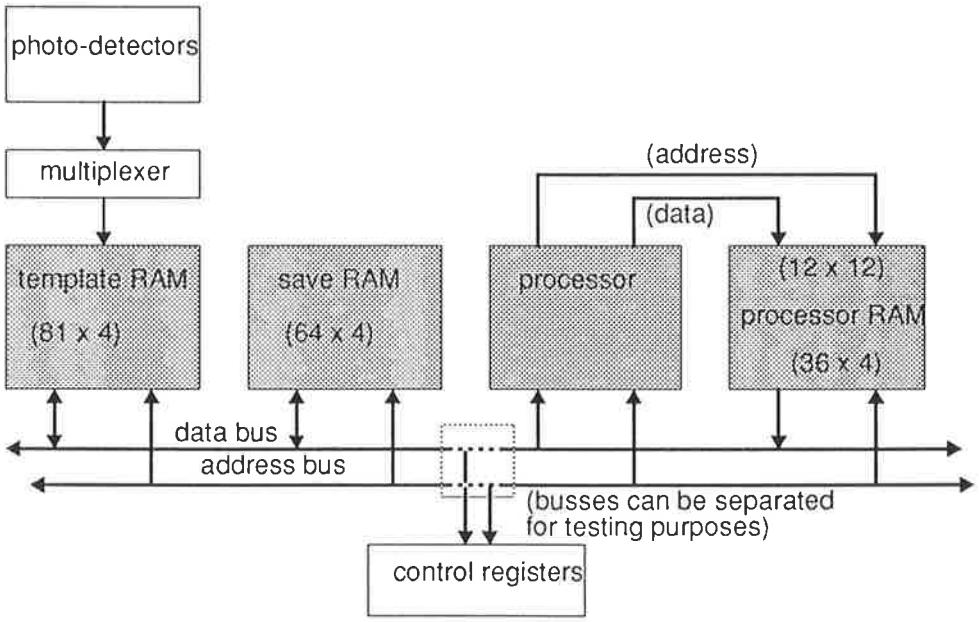


Figure B.1 Bugeye internal structure

2. Testable blocks

2.1 Processor (excerpted from Yakovleff *et al.*, 1993)

Architecture

The processor detects the occurrences of pre-determined templates and calculates their relative displacement within pre-set areas of the visual field, which can be partitioned into up to six areas that may overlap. Hence, while the template RAM output is stored in an intermediate result memory at an address corresponding to its position in the visual field, the analysis stage compares the current template with a desired value. If a match is found, the current position is stored and subtracted from the previously stored value, if any, thus yielding the displacement. The latter is in turn stored along with the current position into another result memory at an address corresponding this time to the area.

The processing stage (Figure B.2) comprises a controller and 6 tracking units, each with two processing elements, and hence two templates may be tracked in each visual partition. The internal data bus is 12 bits wide, as a processing element output comprises the current address, or position within the 60 degree visual angle (6 bits), and the displacement (5 bits + sign bit). The RAM is organised accordingly, which means that while processing it operates as a 12x12 memory (two 12-bit locations per area), and results are read on the 4-bit external data bus.

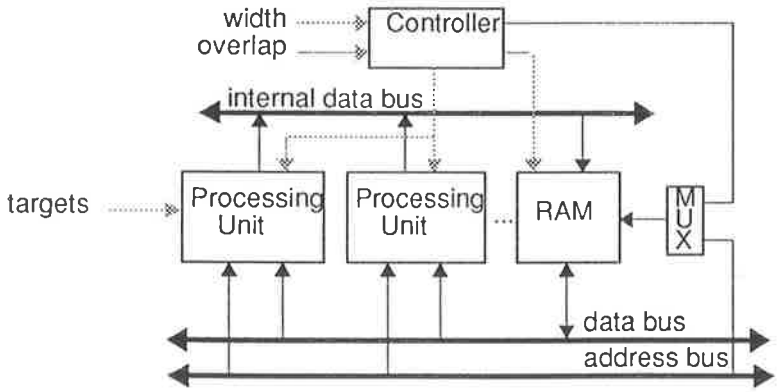


Figure B.2 Processor architecture

Control structure

The controller requires a signal from the interpretation unit that indicates the presence of valid data and address (that signal is essentially the "write enable" for the intermediate result memory). It produces the control signals to the appropriate processing units and generates the RAM address and write enable at each area boundary. Processing completes a few clock cycles after the last template has been analysed. There are only 4 system parameters: width, overlap and two target template values, which implies that areas are identical, and two templates are tracked independently in each area.

The controller comprises two 5-bit counters, Count1 and Count2, which are multiplexed into two comparators, one each for width and overlap as shown in Figure B.3. Initially, only Count1 and the first area are enabled. When the overlap value is reached, Count2 and the second area are enabled. The overlap value is therefore the position at which the next area begins, rather than the amount of overlap, and its value must be more than half the width to ensure correct operation. When Count1 reaches the width value, the multiplexer switches to Count2, Count1 is reset and the first area is disabled. Later on, when Count2 reaches the overlap value, Count1 and the third area are enabled. The second area is disabled when Count2 reaches the width value and the multiplexer switches back to Count1. Operations proceed in this manner until either the sixth area is the last enabled, or the address has reached 60. The processor output values are stored in the result RAM each time the width value is reached.

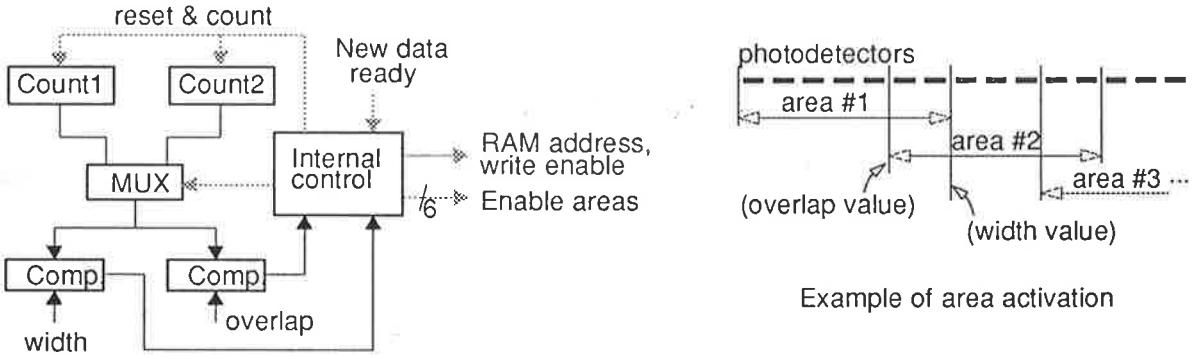


Figure B.3 Controller

Processing unit

A processing unit comprises three registers, a comparator and a subtractor (see Figure B.4). Within an enabled area, each time a new template is on the data bus, its value is compared with a target value. If there is a match, the current address is stored in Reg1, subtracted from Reg2, and the result is stored in Reg3. Reg1 is then copied into Reg2, which will therefore hold the previous position of the template the next time that particular area is scanned.

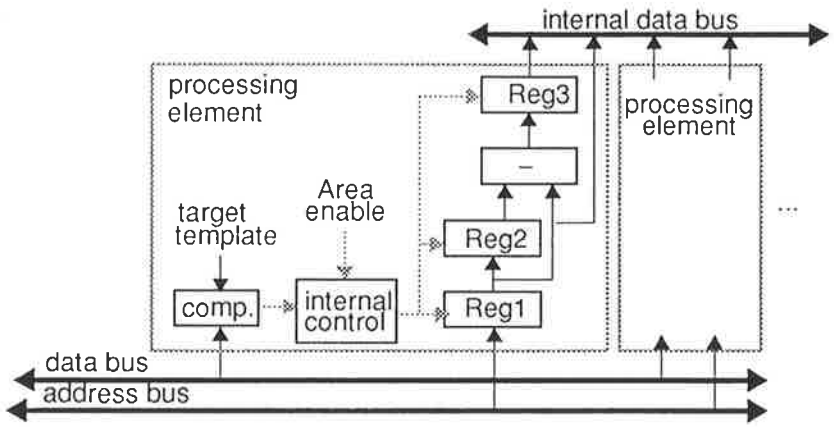


Figure B.4 Processor internal structure

2.2 Memory structures

The various memory structures are shown in Figure B.5. All memory locations are 4 bits wide. To facilitate the hardware implementation, some locations are not used (hashed boxes).

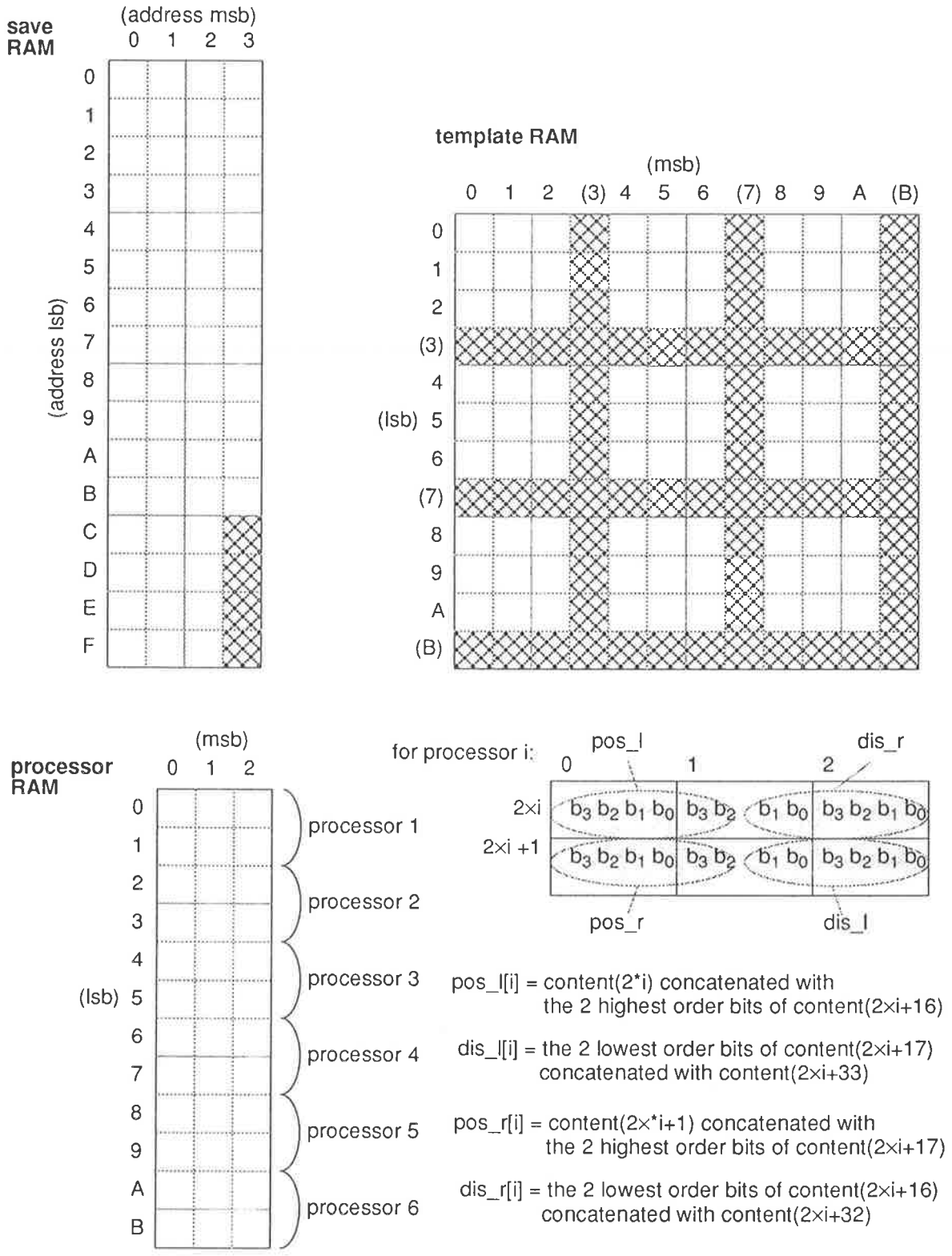


Figure B.5 Save, template, and processor RAMs

3. Bugeye interfacing requirements and basic operations

The testing apparatus described in Chapter 6 comprises a computer connected via its parallel input/output port to a "test rig", which itself contains the bugeye and interfacing circuitry. The interface consists of an address register, a bi-directional data register, and a state machine implemented with programmable logic devices. The state machine handles all the control signals to and from the bugeye, i.e., signals other than data, address, reset, and the digital clock (which it uses itself and is generated independently). The signals that are of interest to functional testing are described in Table B.1. The direction indicates whether a signal is a bugeye input (I), an output (O), or both (I/O).

signal name	direction	function
address	I	6-bit memory and register address
data	I/O	4-bit data bus for memories and control register
clock	I	digital clock (4 MHz)
clock_out	O	synchronisation signal (clock ÷ 4)
bus_available	O	status signal indicating whether sampling has completed
sample	I	sample photodetector outputs
chip_select	I	activate I/O functions
read/write	I	data transfer direction
processor_test	I	data and address valid for processor external testing
reset	I	asynchronous reset

Table B.1 Signal description

The state machine receives data and control signals from the computer, carries out the requested operation, indicates its completion to the computer, and, in case of a read operation, also drives data to the computer. Communication is asynchronous, and employs a "request-acknowledge" hand-shaking scheme, which is fairly standard. The computer may thus request that the sample signal be set or reset, that the various memories and registers be read from or writing into, or that some data and address be supplied to the processor for testing purposes.

Communication with the bugeye is generally similar to a memory access, with the proviso that the bus_available signal be asserted, thus indicating that the storage and processing of new templates is complete, following the falling edge of the sampling signal. If such is the case, read and write operations can be carried out in customary fashion, using the address, data, chip_select, and read/write signals.

The only exception concerns the processor test, which can only be effected while the portion of data and address buses pertaining to the processor are internally disconnected from the save and template RAMs (see Figure B.1). The state machine must first reset the internal control by applying the sampling signal, and then synchronise the bugeye's internal digital clock with its own, using the clock_out signal. The next step is a normal write operation, and is carried out at the same time as processor_test is pulsed, which emulates an internal signal indicating that the current address and data are ready to be processed.

4. Testing procedures

4.1 Bugeye operating modes

The control register comprises eight 4-bit locations (Table B.2). The highest location, 0xFF, is assigned to the chip's mode of operation, and the other locations contain the processor parameters. The "mask" parameter is employed in conjunction with a target, and specifies which individual bits are to be taken into account for the comparison between a detected template and the target, which therefore allows the latter to be matched with a group of templates.

location	contents	location	contents
0xF8	target 1	0xFC	overlap (bits 4, 3, 2, 1)
0xF9	mask 1	0xFD	overlap (bit 0), [unused], width (bits 4, 3)
0xFA	target 2	0xFE	width (bits 2, 1, 0), [unused]
0xFB	mask 2	0xFF	mode

Table B.2 Control register

The bugeye mode of operation is set by writing to the control register at address 0xFF, and hence, if the highest address bit of a subsequent operation is low, the test block specified in Table B.3 is selected.

mode register content	mode	type
0x00	photodetector	(internal)
0x01	template RAM	read/write
0x02	save RAM	read/write
0x03	processor	read RAM, write to processor
0x04	processor test	(internal)

Table B.3 Operating modes

Photodetector mode (0x00)

Only the sample signal is to be applied from the outside. bus_available goes high (inactive) with the rising edge of the sampling signal and remains so until the photodetector outputs have been decoded in the template RAM and stored in the save RAM. This occurs approximately 60 internal clock cycles after the falling edge of sample. Storing and processing templates take place concurrently.

Save and template RAM modes (0x01 and 0x02)

In both the save and template RAM modes, the memories can be read from and written into. Normally, the template RAM need only be loaded following power-up, while each location of the save RAM would be read once between sampling.

Processor mode (0x03)

The buses are internally disconnected. A write operation is directed to the processor, while a read fetches data from the processor RAM.

Processor test mode (0x04)

Read or write operations are ignored by the bugeye, while applying the sample signal simply resets the processor control.

4.2 Software procedures

A number of tests can be effected from the computer. For instance, testing the various memories generally consist of setting the appropriate operating mode, i.e., writing 0x01 or 0x02 to location 0xFF, and then reading from the bugeye and displaying the data, or downloading data to the bugeye. The processor test is slightly more complex, and involves carrying out the following steps:

(i) set mode to 0x04, and apply the sample signal once to reset the processor

(ii) set mode to 0x03, and write 60 times, supplying addresses 0 to 59 (0x3B) along with test data

(iii) read processor RAM (36 operations) and display data.

Finally, the standard test used in Chapter 6 consists of:

(i) set mode to 0x01, and load template RAM (81 write operations)

(ii) set mode to 0x00, and apply sample signal

(iii) set mode to 0x02, read save RAM (60 read operations), and display results

[repeat from step (ii)]

REFERENCES

Abbott D., Cui S., Eshraghian K. & McCabe E. (1991) "Photovoltaic Gate Biasing Edge Effect in GaAs MESFETs", *Electronics Letters*, Vol. 27, No. 21, pp 1900-1902.

Abbott D., Moini A., Yakovleff A., Nguyen X.T., Blanksby A., Kim G., Bouzerdoum A., Bogner R.E. & Eshraghian K. (1994) "A new VLSI smart sensor for collision avoidance inspired by insect vision", *SPIE Proceedings, Intelligent Vehicle Highway Systems*, Vol. 2344, pp 105-115.

Abbott L.F. & Kepler T.B. (1990) "Model Neurons: From Hodgkin-Huxley to Hopfield", in *Statistical Mechanics of Neural Networks (Lecture Notes in Physics, 368)*, Luis Garrido ed., Springer-Verlag, pp 5-18.

Adelson E.H. & Bergen J.R. (1985) "Spatiotemporal energy models for the perception of motion", *Journal of the Optical Society of America A*, Vol. 2, No. 2, pp 284-299.

Adelson E.H. & Bergen J.R. (1991) "The Plenoptic Function and the Elements of Early Vision", in *Computational Models of Visual Processing*, Michael S. Landy and J. Anthony Movshon eds, MIT Press, pp 3-20.

Adelson E.H. & Movshon J.A. (1982) "Phenomenal coherence of moving visual patterns", *Nature*, Vol. 300, pp 523-525.

Adiv G. (1985) "Determining Three-Dimensional Motion and Structure from Optical Flow Generated by Several Moving Objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, No. 4, pp 384-401.

Aggarwal J.K., Davis L.S. & Martin W.N. (1981) "Correspondence Processes in Dynamic Scene Analysis", *Proceedings of the IEEE*, Vol. 69, No. 5, pp 562-572.

Aggarwal J.K. & Nandhakumar N. (1988) "On the Computation of Motion from Sequences of Images - A review", *Proceedings of the IEEE*, Vol. 76, No. 8, pp 917-935.

Allman J., Miezin F. & McGuinness E. (1985) "Direction- and velocity-specific responses from beyond the classical receptive field in the middle temporal visual area (MT)", *Perception*, Vol. 14, pp 105-126.

Aloimonos J. (1988) "Visual Shape Computation", *Proceedings of the IEEE*, Vol. 76, No. 8, pp 899-916.

Ancona N. & Poggio T. (1993) "Optical Flow from 1D Correlation: Application to a simple Time-To-Crash Detector", *Proceedings, Fourth IEEE International Conference on Computer Vision*, Berlin, Germany, pp 209-214.

Andersen C.S., Madsen C.B., Sorensen J.J., Kirkeby N.O.S., Jones J.P. & Christensen H.I. (1992) "Navigation using range images on a mobile robot", *Robotics and Autonomous Systems*, Vol. 10, pp 147-160.

Ansari N. & Delp E.J. (1989) "A Note on Detecting Dominant Points", *SPIE Proceedings, Visual Communications and Image Processing IV*, Vol. 1199, pp 821-832.

Anstis S.M. (1980) "The perception of apparent movement", *Philosophical Transactions of the Royal Society of London B*, Vol. 290, pp 153-168.

Anstis S.M. & Ramachandran V.S. (1985) "Kinetic occlusion by apparent motion", *Perception*, Vol. 14, pp 145-149.

Anstis S. & Ramachandran V.S. (1987) "Visual Inertia in Apparent Motion", *Vision Research*, Vol. 27, No. 5, pp 755-764.

Arun K.S., Huang T.S. & Blostein S.D. (1987) "Least-Squares Fitting of Two 3-D Point Sets", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 5, pp 698-700.

Bair W. & Koch C. (1991) "Real-time motion detection using an analog VLSI zero-crossing chip", *SPIE Proceedings, Visual Information Processing: From Neurons to Chips*, Vol. 1473, pp 59-65.

Baker C.L. (1988) "Spatial and Temporal Determinants of Directionally Selective Velocity Preference in Cat Striate Cortex", *Journal of Neurophysiology*, Vol. 59, No. 5, pp 1557-1574.

Baker C.L. & Braddick O.J. (1985) "Eccentricity-Dependent Scaling of the Limits for Short-Range Apparent Motion Perception", *Vision Research*, Vol. 25, No. 6, pp 803-812.

Baker C.L., Friend S.M. & Boulton J.C. (1991) "Optimal Spatial Displacement for Direction Selectivity in Cat Visual Cortex Neurons", *Vision Research*, Vol. 31, No. 10, pp 1659-1668.

Ballard D.H. (1986) "Cortical connections and parallel processing: Structure and function", *Behavioral and Brain Sciences*, Vol. 9, pp 67-120.

Bandopadhay A. & Aloimonos J. (1991) "Image Motion Estimation by Clustering", *International Journal of Imaging Systems and Technology*, Vol. 2, pp 345-355.

Bandopadhay A. & Ballard D.H. (1991) "Egomotion perception using visual tracking", *Computational Intelligence*, Vol. 7, No. 1, pp 39-45.

Barlow H.B. & Levick W.R. (1965) "The Mechanism of Directionally Selective Units in Rabbit's Retina", *Journal of Physiology*, Vol. 178, pp 477-504.

Ben-Tzvi D., Del Bimbo A. & Nesi P. (1993) "Optical Flow from Constraint Lines Parametrization", *Pattern Recognition*, Vol. 26, No.10, pp 1549-1561.

Bergen J.R. & Adelson E.H. (1988) "Early vision and texture perception", *Nature*, Vol. 333, pp 363-364.

Bishop L.G., Keehn D.G. & McCann G.D. (1968) "Motion Detection by Interneurons of Optic Lobes and Brain of the Flies *Calliphora phaenicia* and *Musca domestica*", *Journal of Neurophysiology*, Vol. 31, No. 4, pp 509-525.

Boman D.K. & Hotson J.R. (1992) "Predictive Smooth Pursuit Eye Movements Near Abrupt Changes in Motion Direction", *Vision Research*, Vol. 32, No. 4, pp 675-689.

Borst A. & Egelhaaf M. (1989) "Principles of motion detection", *Trends in Neurosciences*, Vol. 12, No. 8, pp 297-306.

Borst A., Egelhaaf M. & Seung H.S. (1993) "Two-Dimensional Motion Perception in Flies", *Neural Computation*, Vol. 5, pp 856-868.

Boulton J.C. & Hess R.F. (1990) "The Optimal Displacement for the Detection of Motion", *Vision Research*, Vol. 30, No. 7, pp 1101-1106.

Bouzerdoum A. (1993) "The elementary movement detection mechanism in insect vision", *Philosophical Transactions of the Royal Society of London B*, Vol. 339, pp 375-384.

Boyer K.L. & Kak A.C. (1988) "Structural Stereopsis for 3-D Vision", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-10, No. 2, pp 144-166.

Braddick O. (1974) "A Short-Range Process in Apparent Motion", *Vision Research*, Vol. 14, pp 519-527.

Braddick O.J. (1980) "Low-level and high-level processes in apparent motion", *Philosophical Transactions of the Royal Society of London B*, Vol. 290, pp 137-151.

Brady M. & Wang H. (1992) "Vision for mobile robots", *Philosophical Transactions of the Royal Society of London B*, Vol. 337, pp 341-350.

Brady M. & Yuille A. (1984) "An Extremum Principle for Shape from Contour", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 3, pp 288-301.

Brenner E. (1991) "Judging Object Motion During Smooth Pursuit Eye Movements: The Role of Optic Flow", *Vision Research*, Vol. 31, No. 11, pp 1893-1902.

Broida T.J. & Chellappa R. (1986) "Estimation of Object Motion Parameters from Noisy Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 1, pp 90-99.

Brooks R.A. (1986) "A Robust Layered Control System for a Mobile Robot", *IEEE Journal of Robotics and Automation*, Vol. RA-2, No. 1, pp 14-23.

Brooks R.A. (1991a) "Intelligence without representation", *Artificial Intelligence*, Vol. 47, pp 139-159.

Brooks R.A. (1991b) "New Approaches to Robotics", *Science*, Vol. 253, pp1227-1232.

Bruyelle J-L. & Postaire J.G. (1993) "Direct range measurement by linear stereovision for real-time obstacle detection in road traffic", *Robotics and Autonomous Systems*, Vol. 11, pp 261-268.

Burr D.C., Ross J. & Morrone M.C. (1986) "Seeing objects in motion", *Proceedings of the Royal Society of London B*, Vol. 227, pp 249-265.

Caelli T. & Julesz B. (1979) "Psychophysical evidence for global feature processing in visual texture discrimination", *Journal of the Optical Society of America*, Vol. 69, No. 5, pp 675-678.

Canny J. (1986) "A Computational Approach to Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 6, pp 679-698.

Carpenter G.A. & Grossberg S. (1987) "A Massively Parallel Architecture for a Self-Organising Neural Pattern Recognition Machine", *Computer Vision, Graphics, and Image Processing*, Vol.37, pp 54-115.

Cavanagh P., Arguin M. & von Grünau M. (1989) "Interattribute Apparent Motion", *Vision Research*, Vol. 29, No. 9, pp 1197-1204.

Chang J-Y., Lee S-W. & Horng M-F. (1993) "Image sequence correspondence via a Hopfield neural network", *Optical Engineering*, Vol. 32, No. 7, pp 1531-1538.

Chong C.P., Salama C.A.T. & Smith K.C. (1992) "Image-Motion Detection Using Analog VLSI", *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 1, pp 93-96.

Chu C-C. & Aggarwal J.K. (1993) "The Integration of Image Segmentation Maps Using Region and Edge Information", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-15, No. 12, pp 1241- 1252.

Clark J.J. (1989) "Authenticating Edges Produced by Zero-Crossing Algorithms", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-11, No. 1, pp 43-57.

Cochran S.D. & Medioni G. (1992) "3-D Surface Description from Binocular Stereo", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-14, No. 10, pp 981-994.

Coletta N.J., Williams D.R. & Tiana C.L.M. (1990) "Consequences of Spatial Sampling for Human Motion Perception", *Vision Research*, Vol. 30, No. 11, pp 1631-1648.

Coogan T.A. & Burkhalter A. (1993) "Hierarchical Organization of Areas in Rat Visual Cortex", *Journal of Neuroscience*, Vol. 13, No. 9, pp 3749-3772.

Cox I.J. (1993) "A Review of Statistical Data Association Techniques for Motion Correspondence", *International Journal of Computer Vision*, Vol. 10, No. 1, pp 53-66.

Cremieux J., Orban G.A., Duysens J. & Amblard B. (1987) "Response Properties of Area 17 Neurons in Cats Reared in Stroboscopic Illumination", *Journal of Neurophysiology*, Vol. 57, No. 5, pp 1511-1535.

Davis L.S., Janos L. & Dunn S.M. (1983) "Efficient Recovery of Shape from Texture", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-5, No. 5, pp 485-492.

Davis L.S. & Mitiche A. (1980) "Edge Detection in Textures", *Computer Graphics and Image Processing*, Vol. 12, pp 25-39.

Delbrück T. (1993) "Silicon Retina with Correlation-Based Velocity-Tuned Pixels", *IEEE Transactions on Neural Networks*, Vol. 4, No. 3, pp 529-541.

Delbrück T. & Mead C.A. (1991) "Time-derivative adaptive silicon photoreceptor array", *SPIE Proceedings, Infrared Sensors: Detectors, Electronics, and Signal Processing*, Vol. 1541, pp 92-99.

De Micheli E., Torre V. & Uras S. (1993) "The Accuracy of the Computation of Optical Flow and the Recovery of Motion Parameters", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-15, No. 5, pp 434-447.

Deriche R. & Faugeras O. (1990) "Tracking line segments", *Image and Vision Computing*, Vol. 8, No. 4, pp 261- 270.

Derrington A.M., Badcock D.R. & Holroyd S.A. (1992) "Analysis of the Motion of 2-Dimensional Patterns: Evidence for a Second-Order Process", *Vision Research*, Vol. 32, No. 4, pp 699-707.

Dhond U.R. & Aggarwal J.K. (1989) "Structure from Stereo - A Review", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 6, pp 1489-1510.

Dickmanns E.D. & Graefe V. (1988a) "Dynamic Monocular Machine Vision", *Machine Vision and Applications*, Vol. 1, pp 223-240.

Dickmanns E.D. & Graefe V. (1988b) "Applications of Dynamic Monocular Machine Vision", *Machine Vision and Applications*, Vol. 1, pp 241-261.

Dobkins K.R. & Albright T.D. (1993) "What Happens if it Changes Color when it Moves?: Psychophysical Experiments on the Nature of Chromatic Input to Motion Detectors", *Vision Research*, Vol. 33, No. 8, pp 1019-1036.

Dreschler L. & Nagel H-H (1982) "Volumetric Model and 3D Trajectory of a Moving Car Derived from Monocular TV Frame Sequences of a Street Scene", *Computer Graphics and Image Processing*, Vol. 20, pp 199-228.

Duda R.O. & Hart P.E. (1973) *Pattern Classification and Scene Analysis*, Wiley-Interscience, pp 271-272.

Duncan J.H. & Chou T-C. (1992) "On the Detection of Motion and the Computation of Optical Flow", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-14, No. 3, pp 346-352.

Dzhafarov E.N., Sekuler R. & Allik J. (1993) "Detection of changes in speed and direction of motion: Reaction time analysis", *Perception and Psychophysics*, Vol. 54, No. 6, pp 733-750.

Egelhaaf M., Borst A. & Reichardt W. (1989) "Computational structure of a biological motion-detection system as revealed by local detector analysis in the fly's nervous system", *Journal of the Optical Society of America A*, Vol. 6, No. 7, pp 1070-1087.

Egelhaaf M., Hausen K., Reichardt W. & Wehrhahn C (1988) "Visual course control in flies relies on neuronal computation of object and background motion", *Trends in Neurosciences*, Vol. 11, No. 8, pp 351-358.

Emerson R.C., Bergen J.R. & Adelson E.H. (1992) "Directionally Selective Complex Cells and the Computation of Motion Energy in Cat Visual Cortex", *Vision Research*, Vol. 32, No. 2, pp 203-218.

Eriksen C.W. & Murphy T.D. (1987) "Movement of attentional focus across the visual field: A critical look at the evidence", *Perception and Psychophysics*, Vol. 42, No. 3, pp 299-305.

Etienne-Cummings R.R., Fernando S.A. & Van der Spiegel J. (1992) "Real-Time 2-D Analog Motion Detector VLSI Circuit", *Proceedings, International Joint Conference on Neural Networks*, Baltimore, U.S.A., Vol. IV, pp 426-431.

Fahle M. & De Luca E. (1994) "Spatio-Temporal Interpretation in Depth", *Vision Research*, Vol. 34, No. 3, pp 343-348.

Fang J-Q. & Huang T.S. (1984) "Some Experiments on Estimating the 3-D Motion Parameters of a Rigid Body from Two Consecutive Image Frames", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 5, pp 545-554.

Fermüller C. & Aloimonos Y. (1993) "The Role of Fixation in Visual Motion Analysis", *International Journal of Computer Vision*, Vol. 11, No. 2, pp 165-186.

Fischler M.A. & Bolles R.C. (1986) "Perceptual Organisation and Curve Partitioning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 1, pp 100-105.

Fischler M.A. & Wolf H. C. (1994) "Locating Perceptually Salient Points on Planar Curves", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-16, No. 2, pp 113-129.

Fodor J.A. & Pylyshyn Z.W. (1981) "How Direct is Visual Perception? Some Reflections on Gibson's 'Ecological Approach'", *Cognition*, Vol. 9, pp 139-196.

Forchheimer R., Chen K., Svensson C. & Ödmark A. (1993) "Single-Chip Image Sensors with a Digital Processor Array", *Journal of VLSI Signal Processing*, Vol. 5, pp 121-131.

Franceschini N., Pichon J.M. & Blanes C. (1992) "From insect vision to robot vision", *Philosophical Transactions of the Royal Society of London B*, Vol. 337, pp 283-294.

Frau J., Llario V. & Oliver G. (1990) "Polynomial regression analysis for estimating motion from image sequences", *SPIE Proceedings, Mobile Robots V*, Vol. 1388, pp 329-340.

Ganz L. (1984) "Visual Cortical Mechanisms Responsible for Direction Selectivity", *Vision Research*, Vol. 24, No. 1, pp 3-11.

Geman D., Geman S., Graffigne C. & Dong P. (1990) "Boundary Detection by Constrained Optimisation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-12, No. 7, pp 609-628.

Geman S. & Geman D. (1984) "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 7, pp 721-741.

Gerbino W. (1984) "low-level and high-level processes in the perceptual organization of three-dimensional apparent motion", *Perception*, Vol. 13, pp 417-428.

Giaschi D., Douglas R., Marlin S. & Cynader M. (1993) "The Time Course of Direction-Selective Adaptation in Simple and Complex Cells in Cat Striate Cortex", *Journal of Neurophysiology*, Vol. 70, No. 5, pp 2024-2034.

Gibson J.J. (1979) *An Ecological Approach to Visual Perception,* Boston: Houghton Mifflin.

Gorea A., Papathomas T.V. & Kovacs I. (1993) "Motion Perception with Spatiotemporally Matched Chromatic and Achromatic Information Reveals a 'Slow' and a 'Fast' Motion System", *Vision Research*, Vol. 33, No. 17, pp 2515-2534.

Granlund G.H., Knutsson H., Westelius C-J. & Wiklund J. (1994) "Issues in robot vision", *Image and Vision Computing*, Vol. 12, No.3, pp 131-148.

Gregory R.L. (1980) "Perceptions as hypotheses", *Philosophical Transactions of the Royal Society of London B*, Vol. 290, pp 181-197.

Grimson W.E.L. (1981) "A computer implementation of a theory of human stereo vision", *Philosophical Transactions of the Royal Society of London B*, Vol. 292, pp 217-253.

Grzywacz N.M. & Yuille A.L. (1990) "A model for the estimate of local image velocity by cells in the visual cortex", *Proceedings of the Royal Society of London B*, Vol. 239, pp 129-161.

Gulyas B., Orban G.A., Duysens J. & Maes H. (1987) "The Suppressive Influence of Moving Textured Backgrounds on Responses of Cat Striate Neurons to Moving Bars", *Journal of Neurophysiology*, Vol. 57, No. 6, pp 1767-1791.

Gupta M.M. (1991) "Neuronal-Morphology of Biological Vision: A Basis for Machine Vision", *SPIE Proceedings, Intelligent Robots and Computer Vision X*, Vol. 1608, pp 199-211.

Hadani I., Kononov A., Ishai G. & Frisch H.L. (1994) "Two metric solutions to three-dimensional reconstruction for an eye in pure rotations", *Journal of the Optical Society of America A*, Vol. 11, No. 5, pp 1564-1574.

Haralick R.M. (1979) "Statistical and Structural Approaches to Texture", *Proceedings of the IEEE*, Vol. 67, No. 5, pp 786-804.

Haralick R.M. (1980) "Edge and Region Analysis for Digital Image Data", *Computer Graphics and Image Processing*, Vol. 12, pp 60-73.

Haralick R.M. (1984) "Digital Step Edges from Zero Crossing of Second Directional Derivatives", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 1, pp 58-68.

Haralick R.M. & Shanmugan K.S. (1974) "Combined Spectral and Spatial Processing of ERTS Imagery Data", *Remote Sensing of Environment*, Vol. 3, pp 3-13.

Haralick R.M., Sternberg S.R. & Zhuang X. (1987) "Image Analysis Using Mathematical Morphology", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 4, pp 532-550.

Harris M.G. (1986) "The Perception of Moving Stimuli: A Model of Spatiotemporal Coding in Human Vision", *Vision Research*, Vol. 26, No. 8, pp 1281-1287.

Hausen K. (1982a) "Motion Sensitive Interneurons in the Optomotor System of the Fly - I. The Horizontal Cells: Structure and Signals", *Biological Cybernetics*, Vol. 45, pp 143-156.

Hausen K. (1982b) "Motion Sensitive Interneurons in the Optomotor System of the Fly - II. The Horizontal Cells: Receptive Field Organization and Response Characteristics", *Biological Cybernetics*, Vol. 46, pp 67-79.

Haynes S.M. & Jain R. (1983) "Detection of Moving Edges", *Computer Vision, Graphics, and Image Processing*, Vol. 21, pp 345-367.

He J.Z. & Nakayama K. (1994) "Perceived Surface Shape not Features Determines Correspondence Strength in Apparent Motion", *Vision Research*, Vol. 34, No. 16, pp 2125-2135.

Heeger D.J. (1987) "Model for the extraction of image flow", *Journal of the Optical Society of America A*, Vol. 4, No. 8, pp 1455-1471.

Heeger D.J. (1993) "Modeling Simple-Cell Direction Selectivity With Normalized, Half-Squared, Linear Operators", *Journal of Neurophysiology*, Vol. 70, No. 5, pp 1885-1898.

Hildreth E.C. (1984) "Computations Underlying the Measurement of Visual Motion", *Artificial Intelligence*, Vol. 23, pp 309-354.

Hildreth E.C (1992) "Recovering heading for visually guided navigation in the presence of self-moving objects", *Philosophical Transactions of the Royal Society of London B*, Vol. 337, pp 305-313.

Hoff W. & Ahuja N. (1989) "Surfaces from Stereo: Integrating Feature Matching, Disparity Estimation, and Contour Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-11, No. 2, pp 121-136.

Horiuchi T., Bair W., Bishofberger B., Moore A., Koch C. & Lazzaro J. (1992) "Computing Motion Using Analog VLSI Vision Chips: An Experimental Comparison Among Different Approaches", *International Journal of Computer Vision*, Vol. 8, No. 3, pp 203-216.

Horn B.K.P. & Schunk B.G. (1981) "Determining Optical Flow", *Artificial Intelligence*, Vol. 17, pp 185-203.

Horridge G.A. (1986) "A theory of insect vision: velocity parallax", *Proceedings of the Royal Society of London B*, Vol. 229, pp 13-27.

Horridge G.A. (1987) "The evolution of visual processing and the construction of seeing systems", *Proceedings of the Royal Society of London B*, Vol. 230, pp 279-292.

Horridge G.A. (1990) "A template theory to relate visual processing to digital circuitry", *Proceedings of the Royal Society of London B*, Vol. 239, pp 17-33.

Horridge G.A. (1991) "The compromise between seeing spatial layout and making visual discriminations", *Current Science*, Vol. 60, No. 12, pp 686-693.

Horridge G.A. & Marcelja L. (1992) "On the existence of 'fast' and 'slow' directionally sensitive motion detector neurons in insects", *Proceedings of the Royal Society of London B*, Vol. 248, pp 47-54.

Horridge G.A. & Sobey P. (1991) "An artificial seeing system copying insect vision", *International Journal of Optoelectronics*, Vol. 6, Nos. 1/2, pp 177-193.

Horridge G.A., Wang X. & Zhang S.W. (1990) "Colour inputs to motion and object vision in an insect", *Philosophical Transactions of the Royal Society of London B*, Vol. 329, pp 257-263.

Huang T.S. (1986) "Three-dimensional motion analysis by direct matching", *Journal of the Optical Society of America A*, Vol. 3, No. 9, pp 1501-1503.

Huang T.S. & Faugeras O.D. (1989) "Some Properties of the *E* Matrix in Two-View Motion Estimation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-11, No. 12, pp 1310-1312.

Huang T.S. & Netravali A.N. (1994) "Motion and Structure from Feature Correspondences: A Review", *Proceedings of the IEEE*, Vol. 82, No. 2, pp 252-268.

Hubel D.H. & Wiesel T.N. (1962) "Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex", *Journal of Physiology*, Vol. 160, pp 106-154.

Ikeuchi K. (1984) "Shape from Regular Patterns", *Artificial Intelligence*, Vol. 22, pp 49-75.

Ishiguro H., Stelmaszyck P. & Tsuji S. (1990) "Acquiring 3-D Structure by Controlling Visual Attention of a Mobile Robot", *Proceedings, IEEE International Conference on Robotics and Automation*, Cincinnati, U.S.A., Vol. 2, pp 755-760.

Jacobson L. & Wechsler H. (1987) "Derivation of Optical Flow Using a Spatiotemporal-Frequency Approach", *Computer Vision, Graphics, and Image Processing*, Vol. 38, pp 29-65.

Jau J.Y. & Chin R.T. (1990) "Shape from Texture Using the Wigner Distribution", *Computer Vision, Graphics, and Image Processing*, Vol. 52, pp 248-263.

Jayant N., Johnston J. & Safranek R. (1993) "Signal Compression Based on Models of Human Perception", *Proceedings of the IEEE*, Vol. 81, No. 10, pp 1385-1422.

Jenkin M. & Tsotsos J.K. (1986) "Applying Temporal Constraints to the Dynamic Stereo Problem", *Computer Vision, Graphics, and Image Processing*, Vol. 33, pp 16-32.

Johnston W.A. & Dark V.J. (1986) "Selective Attention", *Annual Review of Psychology*, Vol. 37, pp 43-75.

Jones D.G. & Malik J. (1992) "Computational framework for determining stereo correspondence from a set of linear spatial filters", *Image and Vision Computing*, Vol. 10, No. 10, pp 699-708.

Julesz B. (1980) "Spatial nonlinearities in the instanteneous perception of textures with identical power spectra", *Philosophical Transactions of the Royal Society of London B*, Vol. 290, pp 83-94.

Julesz B. (1981) "Textons, the elements of texture perception, and their interactions", *Nature*, Vol. 290, pp 91-97.

Julesz B. (1991) "Early vision and focal attention", *Reviews of Modern Physics*, Vol. 63, No. 3, pp 735-772.

Kanade T. (1981) "Recovery of Three-Dimensional Shape of an Object from a Single View", *Artificial Intelligence*, Vol. 17, pp 409-460.

Kanatani K-I. (1987) "Structure and Motion from Optical Flow under Perspective Projection", *Computer Vision, Graphics, and Image Processing*, Vol. 38, pp 122-146.

Kelber A. & Zeil J. (1990) "A robust procedure for visual stabilisation of hovering flight position in guard bees of *Trigona (Tetragonisca) angustala* (Apidae, Meliponinae)", *Journal of Comparative Physiology A*, Vol. 167, pp 569-577.

Khotanzad A., Bokil A. & Lee Y-W. (1993) "Stereopsis by Constraint Learning Feed-Forward Neural Networks", *IEEE Transactions on Neural Networks*, Vol. 4, No. 2, pp 332-342.

Kim Y.C. & Aggarwal J.K. (1987) "Determining Object Motion in a Sequence of Images", *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 6, pp 599-614.

King S.J. & Weiman C.F.R. (1990) "HelpMate™ Autonumous Mobile Robot Navigation System", *SPIE Proceedings, Mobile Robots V*, Vol. 1388, pp 190-198.

Kirchner H. & Niemann H. (1992) "finite element method for determination of optical flow", *Pattern Recognition Letters*, Vol. 13, pp 131-141.

Koch C. (1989) "Seeing Chips: Analog VLSI Circuits for Computer Vision", *Neural Computation*, Vol. 1, pp 184-200.

Koenderink J.J. & van Doorn A.J. (1976) "Local structure of movement parallax of the plane", *Journal of the Optical Society of America*, Vol. 66, No. 7, pp 717-723.

Kohonen T. (1990) "The Self-Organizing Map", *Proceedings of the IEEE*, Vol. 78, No. 9, pp 1464-1480.

Konrad J. & Dubois E. (1992) "Bayesian Estimation of Motion Vector Fields", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-14, No. 9, pp 910-927.

Kontsevich L.L. (1993) "Pairwise comparison technique: a simple solution for depth reconstruction", *Journal of the Optical Society of America A*, Vol. 10, No. 6, pp 1129-1135.

Kowler E. (1989) "Cognitive Expectations, Not Habits, Control Anticipatory Smooth Oculomotor Pursuit", *Vision Research*, Vol. 29, No. 9, pp 1049-1057.

Kriegman D.J., Triendl E. & Binford T.O. (1989) "Stereo Vision and Navigation in Buildings for Mobile Robots", *IEEE Transactions on Robotics and Automation*, Vol. 5, No. 6, pp 792-803.

Kröse B.J.A. & Julesz B. (1989) "The Control and Speed of Shifts of Attention", *Vision Research*, Vol. 29, No. 11, pp 1607-1619.

Laird J.E., Newell A. & Rosenbloom P.S. (1987) "Soar*: An architecture for general intelligence", *Artificial Intelligence*, Vol. 33, pp 1-64.

Land M.F. (1990) "Direct Observation of Receptors and Images in Simple and Compound Eyes", *Vision Research*, Vol. 30, No. 11, pp 1721-1734.

Lawton D.T. (1983) "Processing Translational Motion Sequences", *Computer Vision, Graphics, and Image Processing*, Vol. 22, pp 116-144.

Lazzari C. & Varjú D. (1990) "Visual lateral fixation and tracking in the haematophagous bug *Triatoma infestans*", *Journal of Comparative Physiology A*, Vol. 167, pp 527-531.

Ledgeway T. & Smith A.T. (1994) "Evidence for Separate Motion-Detecting Mechanisms for First- and Second-order Motion in Human Vision", *Vision Research*, Vol. 34, No. 20, pp 2727-2740.

Lee C-H. & Joshi A. (1993) "Correspondence Problem in Image Sequence Analysis", *Pattern Recognition*, Vol. 26, No.1, pp 47-61.

Lee D.N. (1980) "The optic flow field: the foundation of vision", *Philosophical Transactions of the Royal Society of London B*, Vol. 290, pp 169-179.

Lee J.S.J., Haralick R.M. & Shapiro L.G. (1987) "Morphologic Edge Detection", *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 2, pp 142-156.

Lehrer M. (1990) "How bees use peripheral eye regions to localize a frontally positioned target", *Journal of Comparative Physiology A*, Vol. 167, pp 173-185.

Lehrer M. (1994) "Spatial Vision in the Honeybee: the Use of Different Cues in Different Tasks", *Vision Research*, Vol. 34, No. 18, pp 2363-2385.

Lehrer M., Srinivasan M.V., Zhang S.W. & Horridge G.A. (1988) "Motion cues provide the bee's visual world with a third dimension", *Nature*, Vol. 332, pp 356-357.

Lifshitz L.M. & Pizer S.M. (1990) "A Multiesolution Hierarchical Approach to Image Segmentation Based on Intensity Extrema", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-12, No. 6, pp 529-540.

Lin C-T. D., Goldgof D.B. & Huang W-C. (1994) "Motion estimation from scaled orthographic projections without correspondences", *Image and Vision Computing*, Vol. 12, No. 2, pp 95-108.

Liu J. & Skerjanc R. (1993) "Stereo and motion correspondence in a sequence of stereo images", *Signal Processing: Image Communications*, Vol. 5, pp 305-318.

Lisberger S.G., Morris E.J. & Tychsen L. (1987) "Visual Motion Processing and Sensory-Motor Integration for Smooth Pursuit Eye Movements", *Annual Review of Neuroscience*, Vol. 10, pp 97-129.

Lorenceau J., Shiffrar M., Wells N. & Castet E. (1993) "Different Motion Sensitive Units are Involved in Recovering the Direction of Moving Lines", *Vision Research*, Vol. 33, No. 9, pp 1207-1217.

Longuet-Higgins H.C. (1981) "A computer algorithm for reconstructing a scene from two projections", *Nature*, Vol. 293, pp 133-135.

Longuet-Higgins H.C. (1984) "The visual ambiguity of a moving plane", *Proceedings of the Royal Society of London B*, Vol. 223, pp 165-175.

Longuet-Higgins H.C. & Prazdny K. (1980) "The interpretation of a moving retinal image", *Proceedings of the Royal Society of London B*, Vol. 208, pp 385-397.

Marlin S.G., Hasan S.J. & Cynader M.S. (1988) "Direction-Selective Adaptation in Simple and Complex Cells in Cat Striate Cortex", *Journal of Neurophysiology*, Vol. 59, No. 4, pp 1314-1330.

Marr D. (1976) "Early processing of visual information", *Philosophical Transactions of the Royal Society of London B*, Vol. 275, pp 483-524.

Marr D. (1980) "Visual information processing: the structure and creation of visual representations", *Philosophical Transactions of the Royal Society of London B*, Vol. 290, pp 199-218.

Marr D. & Hildreth (1980) "Theory of edge detection", *Proceedings of the Royal Society of London B*, Vol. 207, pp 187-217.

Marr D. & Nishihara H.K. (1978) "Representation and recognition of the spatial organization of three-dimensional shapes", *Proceedings of the Royal Society of London B*, Vol. 200, pp 269-294.

Marr D. & Poggio T. (1979) "A computational theory of human stereo vision", *Proceedings of the Royal Society of London B*, Vol. 204, pp 301-328.

Marr D. & Ullman S. (1981) "Directional selectivity and its use in early visual processing", *Proceedings of the Royal Society of London B*, Vol. 211, pp 151-180.

Martelli A. (1976) "An Application of Heuristic Search Methods to Edge and Contour Detection", *Communications of the ACM*, Vol. 19, No. 2, pp 73-83.

Mather G. (1984) "Luminance Changes Generates Apparent Movement: Implications for Models of Directional Specificity in the Human Visual System", *Vision Research*, Vol. 24, No. 10, pp 1399-1405.

Mather G., Radford K. & West S. (1992) "Low-level visual processing of biological motion", *Proceedings of the Royal Society of London B*, Vol. 249, pp 149-155.

Maunsell J.H.R. & Van Essen D.C. (1983) "Functional Properties of Neurons in Middle Temporal Visual Area of the Macaque Monkey. I. Selectivity for Stimulus Direction, Speed, and Orientation", *Journal of Neurophysiology*, Vol. 49, No. 5, pp 1127-1147.

Mayhew J.E.W & Frisby J.P. (1981) "Psychophysical and Computational Studies towards a Theory of Human Stereopsis", *Artificial Intelligence*, Vol. 17, pp 249-385.

McDonald J.D. & Bahill A.T. (1983) "Zero-latency tracking of predictable targets by time-delay systems", *International Journal of Control*, Vol. 38, No. 4, pp 881-893.

McKee S.P. (1981) "A Local Mechanism for Differential Velocity Detection", *Vision Research*, Vol. 21, pp 491-500.

McKee S.P. & Nakayama K. (1984) "The Detection of Motion in the Peripheral Visual Field", *Vision Research*, Vol. 24, No. 1, pp 25-32.

McLean G.F. (1993) "Codebook Edge Detection", *CVGIP: Graphical Models and Image Processing*, Vol. 55, No.1, pp 48-57.

Mead C. (1989) *Analog VLSI and Neural Systems*, Addison-Wesley.

Mead C. (1990) "Neuromorphic Electronic Systems", *Proceedings of the IEEE*, Vol. 78, No. 10, pp 1629-1636.

Mead C.A. & Delbrück T. (1991) "Scanners for visualizing activity of analog VLSI circuitry", *Analog Integrated Circuits and Signal Processing*, Vol. 1, No. 2, pp 93-106.

Mehrotra R., Nichani S. & Ranganathan N. (1990) "Corner Detection", *Pattern Recognition*, Vol. 23, No.11, pp 1223-1233.

Mitiche A., Wang Y.F. & Aggarwal J.K. (1987) "Experiments in Computing Optical Flow with the Gradient-Based, Multiconstraint Method", *Pattern Recognition*, Vol. 20, No.2, pp 173-179.

Moini A. (1993) "Design of a VLSI Motion Detector Based Upon the Insect Visual System", *Master of Engineering Science Thesis, The University of Adelaide.*

Moini A. (1994) "Bug-Eye II, the Second Generation of an Insect Vision Based Motion Detection Chip: Technical & Programming Information", *Internal Report, Centre for GaAs VLSI Technology, The University of Adelaide.*

Moini A., Bouzerdoum A., Yakovleff A., Abbott D., Kim O., Eshraghian K. & Bogner R.E. (1993) "An Analog Implementation of a Smart Visual Micro-Sensor Based on Insect Vision", *Proceedings, International Symposium on VLSI Technology, Systems, and Applications*, Taipei, Taiwan, pp 283-287.

Moore A. & Koch C. (1991) "A multiplication based analog motion detection chip", *SPIE Proceedings, Visual Information Processing: From Neurons to Chips*, Vol. 1473, pp 66-75.

Moravec H.P. (1983) "The Stanford Cart and the CMU Rover", *Proceedings of the IEEE*, Vol. 71, No. 7, pp 872-884.

Morgenthaler D.G. (1981) "A New Hybrid Edge Detector", *Computer Graphics and Image Processing*, Vol. 16, pp 166-176.

Morgenthaler D.G. & Rosenfeld A. (1981) "Multidimensional Edge Detection by Hypersurface Fitting", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-3, No. 4, pp 482-486.

Murray D.W. & Buxton B.F. (1987) "Scene Segmentation from Visual Motion Using Global Optimization", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 2, pp 220-228.

Mulligan J.B. (1993) "Nonlinear Combination Rules and the Perception of Visual Motion Transparency", *Vision Research*, Vol. 33, No. 14, pp 2021-2030.

Nagel H-H. (1983) "Displacement Vectors Derived from Secon-Order Intensity Variations in Image Sequences", *Computer Vision, Graphics, and Image Processing*, Vol. 21, pp 85-117.

Nagel H-H. (1987) "On the Estimation of Optical Flow: Relations between Different Approaches and Some New Results", *Artificial Intelligence*, Vol. 33, pp 299-324.

Nagel H-H. & Enkelmann W. (1986) "An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Image Sequences", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 5, pp 565- 593.

Nakayama K. (1985) "Biological Image Motion Processing: A Review", *Vision Research*, Vol. 25, No. 5, pp 625-660.

Nakayama K. & Silverman G.H. (1984) "Temporal and Spatial Characteristics of the Upper Displacement Limit for Motion in Random Dots", *Vision Research*, Vol. 24, No. 4, pp 293-299.

Nalwa V.S. & Binford T.O. (1986) "On Detecting Edges", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 6, pp 699-714.

Negahdaripour S. & Lee S. (1992) "Motion Recovery from Image Sequences Using Only First Order Optical Flow Information", *International Journal of Computer Vision*, Vol. 9, No. 3, pp 163-184.

Nelson R.C. & Aloimonos J. (1988) "Finding Motion Parameters from Spherical Motion Fields (Or the Advantages of Having Eyes in the Back of Your Head)", *Biological Cybernetics*, Vol. 58, pp 261-273.

Nevatia R. & Babu K.R. (1980) "Linear Feature Extraction and Description", *Computer Graphics and Image Processing*, Vol. 13, pp 257-269.

Newell A., Rosenbloom P.S. & Laird J.E. (1989) "Symbolic Architectures for Cognition", in *Foundations of Cognitive Science*, Michael I. Posner ed., MIT Press, pp 93-131.

Nguyen X.T., Eshraghian K., Moini A., Bouzerdoum A., Yakovleff A., Abbott D. & Bogner R.E. (1993a) "An Implementation of a Smart Visual Micro-Sensor Based on Insect Vision", *Proceedings, 12th Australian Microelectronics Conference*, Gold Coast, Australia, pp 129-134.

Nguyen X.T., Bouzerdoum A., Bogner R.E., Eshraghian K., Abbott D. & Moini A. (1993b) "The Stair-Step Tracking Algorithm for Velocity Estimation", *Proceedings, First Australian and New Zealand Conference on Intelligent Information Systems*, Perth, Australia, pp 412-416.

Nguyen X.T., Yakovleff A., Moini A., Eshraghian K., Bouzerdoum A. & Bogner R.E. (1994) "VLSI Architecture of a Low Computational Load Processor for a Visual Sensory System based upon Insect Vision", *Proceedings, European Design and Automation Conference, User Forum*, Paris, France, pp 85-89.

Nilsson D.-E., Land M.F. &Howard J. (1988) "Optics of the butterfly eye", *Journal of Comparative Physiology A*, Vol. 162, pp 341-366.

Nishihara H.K. (1981) "Intensity, Visible-Surface, and Volumetric Representations", *Artificial Intelligence*, Vol. 17, pp 265-284.

Ogata S., Ishida J. & Sasano T. (1994) "Optical sensor array in an artificial compound eye", *Optical Engineering*, Vol. 33, No. 11, pp 3649-3655.

Orban G.A., de Wolf J. & Maes H. (1984) "Factors Influencing Velocity Coding in the Human Visual System", *Vision Research*, Vol. 24, No. 1, pp 33-39.

Orban G.A. & Gulyas B. (1988) "Image Segregation by Motion: Cortical Mechanisms and Implementation in Neural Networks", in *Neural Computers*, Rolf Eckmiller and Christoph v.d. Malsburg eds., pp 149-158.

Orban G.A., Gulyas B. & Vogels R. (1987) "Influence of a Moving Textured Background on Direction Selectivity of Cat Striate Neurons", *Journal of Neurophysiology*, Vol. 57, No. 6, pp 1792-1812.

Orban G.A., Kennedy H. & Bullier J. (1986) "Velocity Sensitivity and Direction Selectivity of Neurons in Areas V1 and V2 of the Monkey: Influence of Eccentricity", *Journal of Neurophysiology*, Vol. 56, No. 2, pp 462-480.

Orban G.A., Kennedy H. & Maes H. (1981a) "Response to Movement of Neurons in Areas 17 and 18 of the Cat: Velocity Sensitivity", *Journal of Neurophysiology*, Vol. 45, No. 6, pp 1043-1058.

Orban G.A., Kennedy H. & Maes H. (1981b) "Response to Movement of Neurons in Areas 17 and 18 of the Cat: Direction Selectivity", *Journal of Neurophysiology*, Vol. 45, No. 6, pp 1059-1072.

Osorio D. & Sobey P.J. (1992) "Insect vision as model for machine vision", *SPIE Proceedings, Intelligent Robots and Computer Vision XI*, Vol. 1826, pp 253-260.

Pal N.R. & Pal S.K. (1993) "A Review on Image Segmentation Techniques", *Pattern Recognition*, Vol. 26, No.9, pp 1277-1294.

Papathomas T.V., Gorea A. & Julesz B. (1991) "Two Carriers for Motion Perception: Color and Luminance", *Vision Research*, Vol. 31, No. 11, pp 1883-1891.

Pasternak T. (1987) "Discrimination of Differences in Speed and Flicker Rate Depends on Directionally Selective Mechanisms", *Vision Research*, Vol. 27, No. 11, pp 1881-1890.

Perrone J.A. & Stone L.S. (1994) "A Model of Self-Motion Estimation Within Primate Extrastriate Visual Cortex", *Vision Research*, Vol. 34, No. 21, pp 2917-2938.

Pitas I. & Venetsanopoulos A.N. (1986) "Edge Detectors Based on Nonlinear Filters", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 4, pp 538-550.

Posner M.I. & Petersen S.E. (1990) "The Attention System of the Human Brain", *Annual Review of Neuroscience*, Vol. 13, pp 25-42.

Poggio G.F. & Poggio T. (1984) "The Analysis of Stereopsis", *Annual Review of Neuroscience*, Vol. 7, pp 379-412.

Poggio T. & Reichardt W. (1973) "Considerations on Models of Movement Detection", *Kybernetik*, Vol. 13, pp 223-227.

Poggio T., Torre V. & Koch C. (1985) "Computational vision and regularisation theory", *Nature*, Vol. 317, pp 314-319.

Prazdny K. (1983) "On the Information in Optical Flows", *Computer Vision, Graphics, and Image Processing*, Vol. 22, pp 239-259.

Prewitt J.M.S. (1970) "Object Enhancement and Extraction", in *Picture Processing and Psychopictorics*, Bernice Sacks Lipkin and Azriel Rosenfeld eds, Academic Press, pp 75-149.

Pylyshyn Z.W. (1984) "The bridge from physical to symbolic", in *Computation and Cognition*, MIT Press, pp 147-191.

Ramachandran V.S. (1987) "Interaction between colour and motion in human vision", *Nature*, Vol. 328, pp 645-647.

Ramachandran V.S., Ginsburg A.P. & Anstis S.M. (1983) "Low spatial frequencies dominate apparent motion", *Perception*, Vol. 12, pp 457-461.

Ranade S. & Rosenfeld A. (1980) "Point Pattern Matching by Relaxation", *Pattern Recognition*, Vol. 12, pp 269-275.

Regan (1986) "Visual Processing of Four Kinds of Relative Motion", *Vision Research*, Vol. 26, No. 1, pp 127-145.

Regan D. & Beverley K.I. (1978) "Looming Detectors in the Human Visual Pathway", *Vision Research*, Vol. 18, No. 4, pp 415-421.

Regan D. & Hamstra S.J. (1993) "Dissociation of Discrimination Thresholds for Time to Contact and for Rate of Angular Expansion", *Vision Research*, Vol. 33, No. 4, pp 447-462.

Reichenbach S.E., Rahman Z. & Narayanswamy R. (1993) "Transform-Coding Image Compression for Information Efficiency and Restoration", *Journal of Visual Communication and Image Representation*, Vol. 4, No. 3, pp 215-224.

Reichardt W. (1961) "Autocorrelation, a Principle for the Evaluation of Sensory Information by the Central Nervous System", in *Sensory Communication*, Walter A. Rosenblith Ed., MIT Press and John Wiley & Sons (New York, London), pp 303-317.

Reichardt W. & Poggio T. (1979) "Figure-Ground Discrimination by Relative Movement in the Fly Visual System", *Biological Cybernetics*, Vol. 35, pp 81-100.

Richards W. (1985) "Structure from stereo and motion", *Journal of the Optical Society of America A*, Vol. 2, No. 2, pp 343-349.

Rieger J.H. & Lawton D.T. (1985) "Processing differential image motion", *Journal of the Optical Society of America A*, Vol. 2, No. 2, pp 354-360.

Riehle A. & Franceschini N. (1984) "Motion Detection in Flies: Parametric Control over ON-OFF Pathways", *Experimental Brain Research*, Vol. 54, pp 390-394.

Roach J.W. & Aggarwal J.K. (1979) "Computer Tracking of Objects Moving in Space", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-1, No. 2, pp 127-135.

Roach J.W. & Aggarwal J.K. (1980) "Determining the Movement of Objects from a Sequence of Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-2, No. 6, pp 554-562.

Roberts L.G. (1965) "Machine Perception of Three-Dimensional Solids", in *Optical and Electro-Optical Information Processing*, James T. Tippett *et al.* eds, MIT Press, pp159-197.

Robertson R.M. & Johnson A.G. (1993) "Collision Avoidance of Flying Locusts: Steering Torques and Behaviour", *Journal of Experimental Biology*, Vol. 183, pp 35-60.

Rom H. & Medioni G. (1993) "Hierarchical Decomposition and Axial Shape Description", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-15, No. 10, pp 973-981.

Rubin N. & Hochstein S. (1993) "Isolating the Effect of One-Dimensional Motion Signals on the Perceived Direction of Moving Two-Dimensional Objects", *Vision Research*, Vol. 33, No. 10, pp 1385-1396.

Salembier P. & Serra J. (1992) "Morphological Multiscale Image Segmentation", *SPIE Proceedings, Visual Communications and Image Processing*, Vol. 1818, pp 620-631.

Sayre K.M. (1986) "Intentionality and information processing: An alternative model for cognitive science", *Behavioral and Brain Sciences*, Vol. 9, pp 121-166.

Schmid A. & Bülthoff H. (1988) "Using Neuropharmacology to Distinguish Between Excitatory and Inhibitory Movement Detection Mechanisms in the Fly *Calliphora erythrocephala*", *Biological Cybernetics*, Vol. 59, pp 71-80.

Schnabolk C. & Raphan T. (1994) "Modeling Three-Dimensional Velocity-to-Position Transformation in Oculomotor Control", *Journal of Neurophysiology*, Vol. 71, No. 2, pp 623-638.

Schunck B.G. (1986) "The Image Flow Constraint Equation", *Computer Vision, Graphics, and Image Processing*, Vol. 35, pp 20-46.

Schunck B.G. (1989) "Image Flow Segmentation and Estimation by Constraint Line Clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-11, No. 10, pp 1010-1027.

Seeger U. & Seeger R. (1994) "Fast corner detection in grey-level images", *Pattern Recognition Letters*, Vol. 15, pp 669-675.

Sekuler A.B. & Sekuler R. (1993) "Representational development of direction in motion perception: A fragile process", *Perception*, Vol. 22, pp 889-915.

Sekuler A.B., Sekuler R. & Sekuler E.B. (1990) "How the visual system detects changes in the direction of moving targets", *Perception*, Vol. 19, pp 181-195.

Sekuler R., Anstis S., Braddick O.J., Brandt T., Movshon J.A. & Orban G. (1990) "The perception of motion", in *Visual Perception: The Neurophysiological Foundations*, Lothar Spillmann and John S. Werner eds., Academic Press, pp 205-230.

Serra J. (1986) "Introduction to Mathematical Morphology", *Computer Vision, Graphics, and Image Processing*, Vol. 35, pp 283-305.

Serra J. & Vincent L. (1992) "An Overview of Morphological Filtering", *Circuits, Systems, and Signal Processing*, Vol. 11, No. 1, pp 47-108.

Sethi I.K. & Jain R. (1987) "Finding Trajectories of Feature Points in a Monocular Image Sequence", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 1, pp 56-73.

Shanmugan K.S., Dickey F.M. & Green J.A. (1979) "An Optimal Frequency Domain Filter for Edge Detection in Digital Pictures", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-1, No. 1, pp 37-49.

Shaw S.R. (1984) "Early Visual Processing in Insects", *Journal of Experimental Biology*, Vol. 112, pp 225-251.

Shechter S. & Hochstein S. (1990) "On and Off Pathway Contributions to Apparent Motion Perception", *Vision Research*, Vol. 30, No. 8, pp 1189-1204.

Shu C.Q. & Shi Y.Q. (1991) "On Unified Optical Flow Field", *Pattern Recognition*, Vol. 24, No.6, pp 579-586.

Shu C.Q. & Shi Y.Q. (1993) "Direct Recovering of *N*th Order Surface Structure Using Optical Flow Field", *Pattern Recognition*, Vol. 26, No.8, pp 1137-1148.

Singh A. (1991) "Optic Flow Computation: A Unified Perspective", *IEEE Computer Society Press*.

Smeets J.B.J. & Brenner E. (1994) "The Difference Between the Perception of Absolute and Relative Motion: a Reaction Time Study", *Vision Research*, Vol. 34, No. 2, pp 191-195.

Snippe H.P. & Koenderink J.J. (1994) "Extraction of optical velocity by use of multi-input Reichardt detectors", *Journal of the Optical Society of America A*, Vol. 11, No. 4, pp 1222-1236.

Snowden R.J. & Braddick O.J. (1990) "Differences in the Processing of Short-Range Apparent Motion at Small and Large Displacements", *Vision Research*, Vol. 30, No. 8, pp 1211-1222.

Sobel E.C. (1990) "The locust's use of motion parallax to measure distance", *Journal of Comparative Physiology A*, Vol. 167, pp 579-588.

Sobey P.J. (1990) "Determining range information from self-motion - the template model", *SPIE Proceedings, Intelligent Robots and Computer Vision IX: Neural, Biological, and 3-D Methods*, Vol. 1382, pp 123-131.

Sobey P.J. & Horridge G.A. (1990) "An implementation of the template model of vision", *Proceedings of the Royal Society of London B*, Vol. 240, pp 221-229.

Soechting J.F. & Flanders M. (1992) "Moving in Three-Dimensional Space: Frames of Reference, Vectors, and Coordinate Systems", *Annual Review of Neuroscience*, Vol. 15, pp 167-191.

Solder U. & Graefe V. (1990) "Object Detection in Real Time", *SPIE Proceedings, Mobile Robots V*, Vol. 1388, pp 104-111.

Spelke E.S. (1990) "Principles of Object Perception", *Cognitive Science*, Vol. 14, pp 29-56.

Srinivasan M.V. (1985) "Shouldn't Directional Movement Detection Necessarily be 'Colour-Blind'?", *Vision Research*, Vol. 25, No. 7, pp 997-1000.

Srinivasan M.V. (1992) "How bees exploit optic flow: behavioural experiments and neural models", *Philosophical Transactions of the Royal Society of London B*, Vol. 337, pp 253-259.

Srinivasan M.V., Pinter R.B. & Osorio D. (1990) "Matched filtering in the visual system of the fly: large monopolar cells of the lamina are optimized to detect moving edges and blobs", *Proceedings of the Royal Society of London B*, Vol. 240, pp 279-293.

Sternberg S.R. (1986) "Grayscale Morphology", *Computer Vision, Graphics, and Image Processing*, Vol. 35, pp 333-355.

Subbarao M. (1989) "Interpretation of Image Flow: A Spatio-Temporal Approach", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-11, No. 3, pp 266-278.

Subbarao M. & Waxman A.M. (1986) "Closed Form Solutions to Image Flow Equations for Planar Surfaces in Motion", *Computer Vision, Graphics, and Image Processing*, Vol. 36, pp 208-228.

Tan T.N., Baker K.D. & Sullivan G.D. (1993) "3D structure and motion estimation from 2D image sequences", *Image and Vision Computing*, Vol. 11, No. 4, pp 203-210.

Tanner J. & Luo J. (1991) "A single chip imager and feature extractor", *SPIE Proceedings, Visual Information Processing: From Neurons to Chips*, Vol. 1473, pp 76-87.

Tanner J.E. & Mead C. (1984) "A Correlating Optical Motion Detector", *Proceedings, Conference on Advanced Research in VLSI*, Paul Penfield, Jr., Ed., pp 57-64.

Tanner J. & Mead C. (1989) "Optical motion sensor", in *Analog VLSI and Neural Systems*, Addison-Wesley, pp 229-255.

Terzopoulos D. (1986) "Regularization of Inverse Visual Problems Involving Discontinuities", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 4, pp 413-424.

Terzopoulos D. (1992) "Dynamic Surfaces", *SPIE Proceedings, Curves and Surfaces in Computer Vision and Graphics III*, Vol. 1830, pp 128-139.

Tewfik A.H. & Deriche M. (1993) "An Eigenstructure Approach to Edge Detection", *IEEE Transactions on Image Processing*, Vol.2, No. 3, pp 353-368.

Thompson P. (1984) "The Coding of Velocity of Movement in the Human Visual System", *Vision Research*, Vol. 24, No. 1, pp 41-45.

Tipper S.P., Weaver B., Jerreat L.M. & Burak A.L. (1994) "Object-Based and Environment-Based Inhibition of Return of Visual Attention", *Journal of Experimental Psychology: Human Perception and Performance*, Vol. 20, No. 3, pp 478-499.

Tolhurst D.J., Walker N.S., Thompson I.D. & Dean A.F. (1980) "Non-linearities of Temporal Summation in Neurones in Area 17 of the Cat", *Experimental Brain Research*, Vol. 38, pp 431-435.

Torre V. & Poggio T. (1978) "A synaptic mechanism possibly underlying directional selectivity to motion", *Proceedings of the Royal Society of London B*, Vol. 202, pp 409-416.

Torre V. & Poggio T.A. (1986) "On Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 2, pp 147-163.

Treisman A.M. & Gelade G. (1980) "A Feature-Integration Theory of Attention", *Cognitive Psychology*, Vol. 12, pp 97-136.

Tremblay M., Laurendeau D. & Poussart D. (1993) "High resolution smart image sensor with integrated parallel analog processing for multiresolution edge extraction", *Robotics and Autonomous Systems*, Vol. 11, pp 231-242.

Tsai R.Y. & Huang T.S. (1981) "Estimating Three-Dimensional Motion Parameters of a Rigid Planar Patch", *IEEE Transactions on Accoustics, Speech, and Signal Processing*, Vol. ASSP-29, No. 6, pp 1147-1152.

Tsai R.Y. & Huang T.S. (1984) "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 1, pp 13-27.

Tsai R.Y., Huang T.S. & Zhu W-L. (1982) "Estimating Three-Dimensional Motion Parameters of a Rigid Planar Patch, II: Singular Value Decomposition", *IEEE Transactions on Accoustics, Speech, and Signal Processing*, Vol. ASSP-30, No. 4, pp 525-534.

Tsotsos J.K. (1990) "Analyzing vision at the complexity level", *Behavioral and Brain Sciences*, Vol. 13, pp 423-469.

Turk M.A., Morgenthaler D.G., Gremban K.D. & Marra M. (1988) "VITS - A Vision System for Autonomous Land Vehicle Navigation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-10, No. 3, pp 342-361.

Turvey M.T., Shaw R.E., Reed E.S. & Mace W.M. (1981) "Ecological laws of perceiving and acting: In reply to Fodor and Pylyshyn (1981)", *Cognition*, Vol. 9, pp 237-304.

Ullman S. (1979) "The interpretation of structure from motion", *Proceedings of the Royal Society of London B*, Vol. 203, pp 405-426.

Ullman S. (1984) "Maximizing rigidity: the incremental recovery of 3-D structure from rigid and nonrigid motion", *Perception*, Vol. 13, pp 255-274.

Ullman S. (1992) "Low-level aspects of segmentation and recognition", *Philosophical Transactions of the Royal Society of London B*, Vol. 337, pp 371-379.

Ulupinar F. & Medioni G. (1990) "Refining Edges Detected by a LoG Operator", *Computer Vision, Graphics, and Image Processing*, Vol. 51, No.3, pp 275-298.

Ulupinar F. & Nevatia R. (1994) "Recovery of 3-D objects with multiple curved surfaces from 2-D contours", *Artificial Intelligence*, Vol. 67, pp 1-28.

Vaidya V.G. & Haralick R.M. (1993) "Wigner Distribution for 2D Motion Estimation from Noisy Images", *Journal of Visual Communication and Image Representation*, Vol. 4, No. 4, pp 281-297.

Van den Berg A.V. & Brenner E. (1994) "Humans Combine the Optic Flow with Static Depth Cues for Robust Perception of Heading", *Vision Research*, Vol. 34, No. 16, pp 2153-2167.

Van Doorn A.J. & Keonderink J.J. (1984) "Spatiotemporal Integration in the Detection of Apparent Movement", *Vision Research*, Vol. 24, No. 1, pp 47-53.

Van Essen D.C., Olshausen B., Anderson C.H. & Gallant J.L. (1991) "Pattern recognition, attention, and information bottlenecks in the primate visual system", *SPIE Proceedings, Visual Information Processing: From Neurons to Chips*, Vol. 1473, pp 17-28.

Van Essen D.C. & Maunsell J.H.R. (1983) "Hierarchical organisation and functional streams in the visual cortex", *Trends in Neurosciences*, Vol. 6, pp 370-375.

Van Santen J.P. & Sperling G. (1984) "Temporal covariance model of human motion perception", *Journal of the Optical Society of America A*, Vol. 1, No. 5, pp 451-473.

Van Santen J.P. & Sperling G. (1985) "Elaborated Reichardt detectors", *Journal of the Optical Society of America A*, Vol. 2, No. 2, pp 300-321.

Veijanen A. (1994) "Unsupervised Image Segmentation Using an Unlabeled Region Process", *Pattern Recognition*, Vol. 27, No. 6, pp 841-852.

Veldkamp W.B. & McHugh T.J. (1992) "Binary Optics", *Scientific American*, May 1992, pp 50-55.

Verghese P. & Pelli D.G. (1992) "The Information Capacity of Visual Attention", *Vision Research*, Vol. 32, No. 5, pp 983-995.

Verri A., Girosi F. & Torre V. (1990) "Differential techniques for optical flow", *Journal of the Optical Society of America A*, Vol. 7, No. 5, pp 912-922.

Verri A. & Poggio T. (1989) "Motion Field and Optical Flow: Qualitative Properties", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-11, No. 5, pp 490-498.

Verri A., Straforini M. & Torre V. (1992) "Computational aspects of motion perception in natural and artificial vision systems", *Philosophical Transactions of the Royal Society of London B*, Vol. 337, pp 429-443.

Wang D., Haese-Coat V., Bruno A. & Ronsin J. (1993) "Texture Classification and Segmentation Based on Iterative Morphological Decomposition", *Journal of Visual Communication and Image Representation*, Vol. 4, No. 3, pp197-214.

Ward V., Syrzycki M. & Chapman G. (1993) "CMOS photodetector with built-in light adaptation mechanism", *Microelectronics Journal*, Vol. 24, pp 547-553.

Warren S., Hamalainen H.A. & Gardner E.P. (1986) "Objective Classification of Motion- and Direction-Sensitive Neurons in Primary Somatosensory Cortex of Awake Monkeys", *Journal of Neurophysiology*, Vol. 56, No. 3, pp 598-622.

Watamaniuk S.N.J. & Sekuler R. (1992) "Temporal and Spatial Integration in Dynamic Random-Dot Stimuli", *Vision Research*, Vol. 32, No. 12, pp 2341-2347.

Watson A.B. & Ahumada Jr. A.J. (1985) "Model of human visual-motion sensing", *Journal of the Optical Society of America A*, Vol. 2, No. 2, pp 322-342.

Watt R.J. (1987) "An outline of the primal sketch in human vision", *Pattern Recognition Letters*, Vol. 5, No. 2, pp 139-150.

Watt R.J. & Morgan M.J. (1984) "Spatial Filters and the Localization of Luminance Changes in Human Vision", *Vision Research*, Vol. 24, No. 10, pp 1387-1397.

Watt R.J. & Morgan M.J. (1985) "A Theory of the Primitive Spatial Code in Human Vision", *Vision Research*, Vol. 25, No. 11, pp 1661-1674.

Wavering A.J. & Lumia R. (1993) "Predictive visual tracking", *SPIE Proceedings, Intelligent Robots and Computer Vision XII*, Vol. 2056, pp 86-97.

Waxman A.M & Ullman S. (1985) "Surface Structure and Three-Dimensional Motion from Image Flow Kinematics", *International Journal of Robotics Research*, Vol. 4, No. 3, pp 72-94.

Waxman A.M & Wohn K. (1985) "Contour Evolution, Neighborhood Deformation, and Global Image Flow: Planar Surfaces in Motion", *International Journal of Robotics Research*, Vol. 4, No. 3, pp 95-108.

Weng J., Ahuja N. & Huang T.S. (1992a) "Matching Two Perspective Views", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-14, No. 8, pp 806-825.

Weng J., Cohen P. & Rebibo N. (1992b) "Motion and Structure Estimation from Stereo Image Sequences", *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 3, pp 362-382.

Weng J., Huang T.S. & Ahuja N. (1987) "3-D Motion Estimation, Understanding, and Prediction from Noisy Image Sequences", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 3, pp 370-389.

Weng J., Huang T.S. & Ahuja N. (1989) "Motion and Structure from Two Perspective Views: Algorithms, Error Analysis, and Error Estimation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-11, No. 5, pp 451-476.

Westheimer G. & Wehrhahn C. (1994) "Discrimination of Direction of Motion in Human Vision", *Journal of Neurophysiology*, Vol. 71, No. 1, pp 33-37.

Williams D.J. & Shah M. (1990) "Edge Contours Using Multiple Scales", *Computer Vision, Graphics, and Image Processing*, Vol. 51, No.3, pp 256-274.

Williams D., Phillips G. & Sekuler R. (1986) "Hysteresis in the perception of motion as evidence for neural cooperativity", *Nature*, Vol. 324, pp 253-255.

Williams D.W. & Sekuler R. (1984) "Coherent Global Motion Percepts from Stochastic Local Motions", *Vision Research*, Vol. 24, No. 1, pp 55-62.

Witkin A.P. (1981) "Recovering Surface Shape and Orientation from Texture", *Artificial Intelligence*, Vol. 17, pp 17-45.

Wohn K., Davis L.S. & Thrift P. (1983) "Motion Estimation Based on Multiple Local Constraints and Non-Linear Smoothing", *Pattern Recognition*, Vol. 16, No. 6, pp 563-570.

Wolf R. & Heisenberg M. (1990) "Visual control of straight flight in *Drosophila melanogaster*", *Journal of Comparative Physiology A*, Vol. 167, pp 269-283.

Wyatt H.J. & Pola J. (1987) "Smooth Eye Movements With Step-Ramp Stimuli: The Influence of Attention and Stimulus Extent", *Vision Research*, Vol. 27, No. 9, pp 1565-1580.

Wyatt Jr J.L., Keast C., Seidel M., Standley D., Horn B., Knight T., Sodini C., Lee H-S. & Poggio T. (1992) "Analog VLSI Systems for Image Acquisition and Fast Early Vision Processing", *International Journal of Computer Vision*, Vol. 8, No. 3, pp217-230.

Xie X., Sudhakar R. & Zhuang H. (1993) "Corner Detection by a Cost Minimization Method", *Pattern Recognition*, Vol. 26, No. 8, pp 1235-1243.

Yakovleff A., Moini A., Bouzerdoum A., Nguyen X.T. Bogner R.E., Eshraghian K. & Abbott D. (1993) "A micro-sensor based on insect vision", *Proceedings, Computer Architectures for Machine Perception Workshop*, New Orleans, U.S.A., pp 137-146.

Yakovleff A., Nguyen X.T., Bouzerdoum A., Moini a., Bogner R.E. & Eshraghian K. (1994) "Dual-purpose interpretation of sensory information", *Proceedings, IEEE International Conference on Robotics and Automation*, San Diego, U.S.A., Vol. 2, pp 1635-1640.

Yuille A.L. Grzywacz N.M. (1988) "A computational theory for the perception of coherent visual motion", *Nature*, Vol. 333, pp 71-74.

Zavidovique B.Y. & Bernard T.M. (1992) "Generic Functions for On-Chip Vision", *Proceedings, 11th IAPR International Conference on Pattern Recognition*, The Hague, Netherlands, Vol. 4, pp 1-10.

Zhuang X., Huang T.S. & Haralick R.M. (1986) "Two-view motion analysis: a unified algorithm", *Journal of the Optical Society of America A*, Vol. 3, No. 9, pp 1492-1500.