Image segmentation based on local motion detection

Richard Beare

Thesis submitted for the degree of

Doctor of Philosophy

Department of Electrical and Electronic Engineering
University of Adelaide
South Australia
5005

November, 1997

Contents

List of Figures

Abstract

Biological systems use visual motion information to help solve an extensive range of complex problems. This thesis explores some of the problems associated with the detection and processing of visual motion information and its application to image segmentation in the context of artificial vision systems. Much of the work, however, is inspired by biological vision systems.

The first half of the thesis addresses the problem of detecting the presence and indicating the direction of local motion. Existing models for motion detection are reviewed and their limitations are identified. A new motion detection architecture is proposed which overcomes most of the limitations of the existing motion detection models. This motion detection architecture correctly indicates the direction of motion of moving edges, independent of their contrast, and has adaptive properties that make it a potentially useful first preprocessing layer in a sensor system. A set of adaptive neurofilters is also proposed. These filters exhibit adaptive properties that are commonly observed in biological visual cells and may be used in the construction of more traditional motion detection architectures. This work is intended to be a basis for smart sensor systems.

The second half of the thesis deals with motion based segmentation. Segmentation is performed using information produced by arrays of local motion detectors. Segmentation is often regarded as an important step in visual processing because it allows a compact and convenient representation of the scene for subsequent processing steps. Typical segmentation schemes depend on velocity flow fields, making velocity estimation the primary task of the early visual system. In this work the problem is reformulated in such a way that segmentation is regarded as the primary task of the early visual system. The segmentation system developed depends on importance measures developed from simple models of human visual perception. Velocity information is not used to make segmentation decisions. This work demonstrates the potential of a new type of visual cue for real applications. Results of processing real scenes using these techniques are also presented.

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 the thesis, when deposited in the University Library, being available for loan and photocopying.

Signed : _____ Date : _____

Acknowledgments

Firstly I must extend many thanks to my supervisor, Dr. Abdesselam Bouzerdoum, who introduced me to the topics of artificial and biological vision systems, and successfully managed the “bugeye” project, and kept me on track with advice and assistance. My predecessors in the vision lab, Thong Nguyen and André Yakovleff, deserve acknowledgment for the work they did to successfully establish the “bugeye” project. I also thank my fellow students, Ali Moini and Andrew Blanksby, for helping maintain the focus of the project on a practical device, being available for discussion and advice and making student life enjoyable and amusing.

Thankyou also to the numerous members of the tea and beer groups who helped to make the postgraduate student experience an enjoyable one (most of the time).

Lastly, thankyou to my parents for not asking, “When are you going to finish?”, too often.

Chapter 1

Introduction and motivation

1.1 Introduction

All mobile creatures require information about the environment in order to function successfully. This information is essential for everyday tasks such as finding food, mates and prey, avoiding predators and navigating, and is available from many different sources. Nature has evolved many different mechanisms, or senses, that are capable of extracting information from the environment in a reliable, real-time fashion. The principles of operation of these senses vary as widely as the environments in which they are used. Some senses, like smell and taste, are chemical in nature, and can be used to indicate the presence of other animals in the vicinity, to mark territories or transmit messages (communal insects). Others, like touch, provide information about the environment in contact with the creature.

Still other senses are available to determine detailed structure of the environment at a considerable range from the creature. In environments that are not conducive to the propagation of ambient electromagnetic energy, or such energy is commonly absent, active mechanisms are commonly used to determine the structure of the environment. Examples include sonar in bats and marine mammals and electrolocation in many fish species.

Perhaps the most complex sense capable of extracting information about the structure of the environment is vision. Vision is the processing of electromagnetic energy that is reflected from or radiated by the environment. As with all senses the nature of the vision system employed by a creature depends upon both the nature of the environment and the types of tasks commonly performed by the creature.

There is a vast number of different types of information that are extracted by biological vision systems. For example, the surface properties of objects can modify the spectrum of light reflected, and this property, i.e. colour, can be sensed by many visual systems. Some

objects, such as the bodies of warm blooded animals, also radiate electromagnetic energy in the non-visible (to humans) part of the spectrum and many animals are capable of detecting this energy (infra-red). Light passing through the atmosphere is polarised in a characteristic way, and a number of species are capable of using this information to orient themselves. Depth is a particularly important property that must often be determined by visual systems, and there is a number of different cues that may be used. For example, humans can use both stereopsis and texture gradients to estimate range.

These examples are far from forming a complete list of cues that are known to be exploited by biological vision systems, yet many species do not exploit the cues just discussed. It is well known that many organisms lack significant binocular vision, and others do not exploit colour information. There is one source of information that is believed to be exploited by all biological vision systems, and that is motion information.

1.2 Motivation

There is a wealth of behavioural and neurophysiological evidence demonstrating the importance of visual motion information to biological systems. Many relatively simple organisms, like flies, are capable of using this information to perform very complex tasks. At present artificial autonomous systems are easily outperformed, in terms of flexibility and reliability in "unfriendly" environments, by even the simplest organisms. Part of the poor performance of autonomous systems is due to the difficulty in designing appropriately robust sensors and using them in sensible ways. Conventional visual sensor and processing systems have power and space requirements that are orders of magnitude larger than typical biological systems, and this severely limits the application domains of these artificial systems. Some of the advantages exhibited by the visual systems of insects are due to the optimal use of limited bandwidth channels and the close coupling between sensing and processing systems. Another important factor is that biological visual systems only attempt to extract the information that is essential for the task being performed — all unnecessary information is eliminated. The capability to create sensors and processing systems that are small and reliable is critical to many fields, including robotics and consumer products.

It seems highly likely that the types of visual motion processing performed by biological vision systems would be useful to many kinds of artificial vision systems and that biological systems could provide useful models for building visual sensors.

1.3 Background

This work has been associated with the "bugeye" project at the University of Adelaide. The "bugeye" project began in 1991 with the aim of implementing a smart sensor based on Professor Adrian Horridge's template model using VLSI technology. All of the work described in this thesis has been at least partially motivated by the experiences of colleagues developing and testing the bugeye sensor. The investigations are largely focussed on areas which could help to improve the performance and flexibility of the sensor.

1.4 Contributions and Roadmap

This thesis addresses two important problems related to visual motion. The first is the design of motion detection systems which include adaptive properties. Several different designs that use biologically inspired architectures and components are proposed and investigated. One of these systems utilises the nonlinear interaction responsible for motion detection as an adaptive mechanism.

The first part of the thesis is organised as follows. Chapter 2 describes the use of motion information by biological systems and reviews the state of the art in motion detection and velocity estimation models. Chapter 3 describes the goals of the motion detector investigation, introduces the criteria for useful local motion detectors, and investigates the performance of existing biologically inspired motion detection models. Chapter 4 introduces some adaptive neurofilters that can be used to construct well known motion detection systems, such as the Reichardt detector. These filters exhibit adaptive properties that are similar to those observed in visual cells. Chapter 5 describes a new local motion detector, called the *directionally sensitive local inhibitory motion detector*, or DSLIMD. The DSLIMD can be used as an adaptive sensory layer and exhibits behaviours that have been observed in biological systems.

The second problem involves processing the information produced by arrays of motion detectors. Motion based segmentation is the specific problem investigated in the second part of this thesis. Segmentation is an important preprocessing step for many visual tasks because it reduces storage requirements and produces a representation that is more useful for higher level tasks. The technique developed explicitly tracks relationships between features rather than relying on accurate velocity estimates. Some important parts of the process are derived from a simple model of human perceptual processes. The results of processing real scenes with the new technique are also shown.

The second part of the thesis is organised as follows. Chapter 6 introduces the traditional

approaches to motion based segmentation and reformulates the problem. Chapter 7 introduces the notion of a perceptual motion structure, that is the basis for motion segmentation used in this thesis. Chapter 8 introduces a new thresholding process. Chapter 9 describes the techniques developed to perform segmentation and includes test results. Conclusions and possible further work are discussed in Chapter 10.

Chapter 2

Biological Motion Processing

2.1 Introduction

Visual motion information is a critical component of the sensory data that organisms use to survive in complex environments. The first part of this chapter reviews some of the ways in which mobile creatures use motion information. This review aims to illustrate the importance of visual motion information to biological systems. These uses of motion information are also likely to be important for autonomous artificial systems.

The second part of the chapter reviews a variety of models and techniques which can be used to extract visual motion information from optical sensors. Some of these are inspired by either behavioural or neurophysiological investigations of biological systems while others have been developed by computer vision practitioners. These systems are of interest because they provide a useful starting point for the design of artificial sensory systems. The distinction between motion detection and velocity estimation will also be discussed.

2.2 Motion Information

The first evidence that motion information is a separate sense in humans was the waterfall illusion, or motion after-effect, where stationary objects appear to be moving in the opposite direction to previously observed moving objects. This effect was discussed by a number of early investigators such as Helmholtz, Aubert and Wohlgemuth, and is a paradox unless motion and position can be regarded as separable senses [Helmholtz 24]. Other early classical studies such as the investigations of "phi" by Gestaltists such as Wertheimer and direct and indirect perception of motion by Exner later led to the discovery of different long and short range motion processing systems [Braddick 74].

It is difficult to completely classify the role of motion information in human activity because of the extensive interactions between different types of visual processes, but its general importance can be illustrated with some examples. Many of these examples are known to be important to a variety of simpler organisms as well.

2.2.1 Depth Perception

Stereopsis is the most commonly known source of depth information in humans, yet people with only one eye or animals with limited spatial separation between eyes are capable of functioning quite effectively. This is possible because a variety of powerful monocular depth cues are available. Motion parallax is probably the most powerful of these (Figure 2.1). The optical velocity field contains information about the relative range of features in the environment, and the slant of surfaces. Absolute range is available if the observer's velocity is known. Humans are capable of extracting depth information from random dot patterns using only motion information.

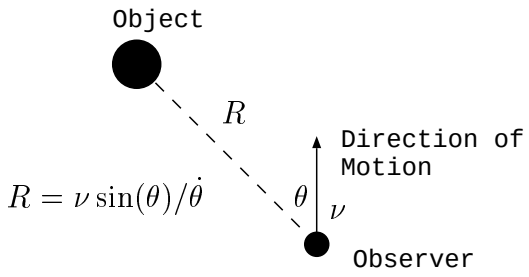


Figure 2.1: Motion Parallax.

2.2.2 Time to Collision

Although only relative, rather than absolute, depth is generally available from the optical velocity field, it is possible to compute time to collision with an approaching object from the apparent rate of expansion of the object (Figure 2.2). It has been shown that this information is used to trigger landing reflexes in flies [Goodman 60] and can be exploited by humans performing interception tasks such as catching [Heuer 93].

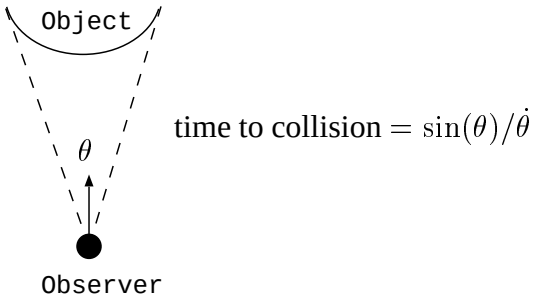


Figure 2.2: Looming.

2.2.3 Image Segmentation

Image segmentation, or grouping, is the process of parsing image data into component objects. Many different cues, including colour, texture and brightness, may assist in this task, but motion is particularly useful. Different parts of an object will tend to be moving at similar velocities, whether the object or the observer is moving. There also tends to be a discontinuity in the velocity field at depth discontinuities. Velocity is therefore useful for associating different parts of an object together while discontinuities assist in locating borders of an object. The assumption that different parts of an object will have similar velocities is related to the "principle of common fate" described by Gestalt psychologists.

The power of motion as a segmentation cue has been demonstrated by moving a randomly textured object in front of a statistically identical, randomly changing textured background. In this case the only information distinguishing the object from the background is motion, as the object is not perceptible when stationary. The changes are random and therefore different to the changes caused by coherent motion of the object. Therefore, detecting only the change in intensity is insufficient to distinguish the object from the background. Both humans and insects are capable of easily perceiving objects in this situation.

2.2.4 Proprioreceptive Sense

Organisms have a number of mechanisms available to determine their own motion. For example, humans possess gravity sensitive organs in the inner ear and stretch receptors in limbs. These sensors provide information about orientation and change in orientation. Gibson proposed that visual motion can also provide this sort of information [Gibson 79]. In many cases it appears that visual motion information is able to override the other proprioreceptive senses. This is particularly evident to anyone viewing wide screen cinema footage or

using sophisticated military simulators.

Visual motion information also appears to play a similar role in simpler organisms. It has been shown that it is possible to use optical motion to modulate the wing beat pattern of flies [Srinivasan 77].

2.2.5 Preattentive processing

These applications of motion information are thought to be examples of *preattentive* or *cognitively impeneterable* processing [Fodor and Pylyshyn 81]. It is widely accepted that cognition plays an important part in human perception because many visual tasks can be influenced by knowledge and experience (i.e. humans operate in a culturally defined framework). However, many tasks can be performed without the need for cognition. Any extremely common action or reflex that requires high speed processing, like walking upright or dodging moving objects, is probably largely dependent on information produced by preattentive processing. In humans, it is very difficult to distinguish between tasks performed by purely preattentive processing and those that involve some cognition because so many activities seem to improve with training. In these situations it is possible that cognitive processes modify the behaviour of the preattentive processes, rather than forming a part of them. Most of the motion processing tasks just discussed are also performed by insects, and it is reasonable to assume that insects operate in a purely preattentive manner.

The lack of understanding of cognitive processing makes modelling of such systems unrealistic at present. Preattentive processes are better understood and can therefore be used as a basis for artificial systems.

2.3 Motion Blindness

Perhaps the most striking demonstration of the importance of motion information in humans are the severe difficulties experienced by a victim of motion blindness as reported by Zihl [Zihl et al. 83]. Cases of motion blindness are far rarer than cases of loss of other visual senses such as colour. Motion blindness made many everyday tasks difficult and dangerous. For example, crossing the road was hazardous because cars appearing far away at one instant would be very close the next, with no sense of approach in between. The patient also experienced severe difficulties in conversation because it was impossible to read facial expressions. In fact the patient was forced to use cognitive processes in many circumstances to compensate for the inability to perceive motion. For example it became necessary to continuously

scan the environment and consciously note changes.

The loss of motion senses while otherwise maintaining normal vision is also one of the strongest pieces of evidence that motion is a separate visual sense in humans.

2.4 Models of Motion Processing Systems

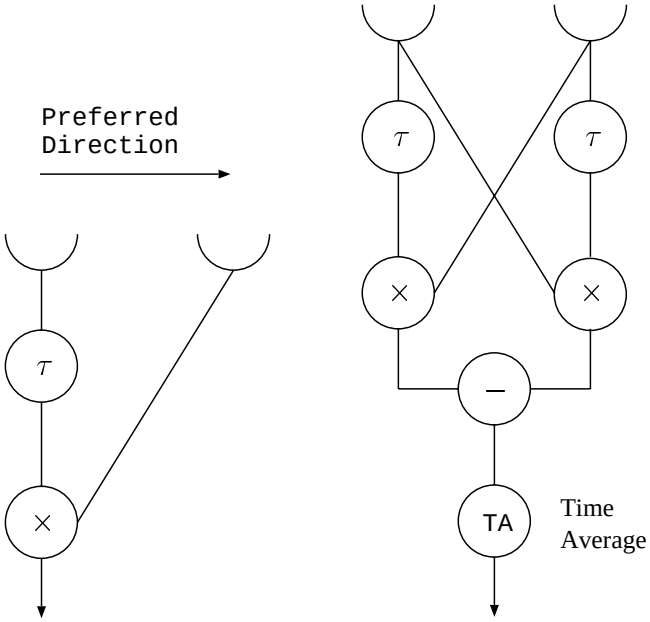
This section is an overview of some of the important results from motion detection studies in biological and artificial systems. Biological studies have produced two broad classes of motion processing systems — delay and compare systems and energy models. Delay and compare systems will be discussed first. However, it will become evident that there are similarities between the two classes. The review of artificial systems will be very brief, only touching on the most widely used results.

2.4.1 Delay and compare systems

Reichardt or correlation detector

The earliest, and probably most famous model of motion detection in biological systems was developed by Reichardt and Hassentein after a series of behavioural experiments examining the optomotor response of insects [Hassentein and Reichardt 56, Reichardt 61]. The Reichardt, or correlation detector possesses a highly parallel architecture. Each elementary motion detector (EMD) detects motion in a preferred direction by comparing a signal from one receptor with a delayed signal from the other receptor (Figure 2.3(a)). The comparison is performed using a nonlinear, multiplicative, interaction between the two channels. Two EMDs tuned to opposite directions are combined to form a bidirectional motion detector (Figure 2.3(b)). Reichardt's system performs infinite time averaging on the output. The time averaging can be eliminated if an array of motion detectors is used and the responses are integrated spatially.

This system was successful in explaining a number of behavioural phenomena that had been observed experimentally, such as the square-law relationship between luminance and response, the reversal of apparent motion due to spatial aliasing, and the reversal due to stepping motion accompanied by a change in contrast (this is known as the reverse phi stimulus and will be discussed further in Section 5.2.2). The reversal in apparent motion due to spatial aliasing is commmon to all schemes that employ spatial sampling.



(a) Reichardt elementary motion detector (EMD).

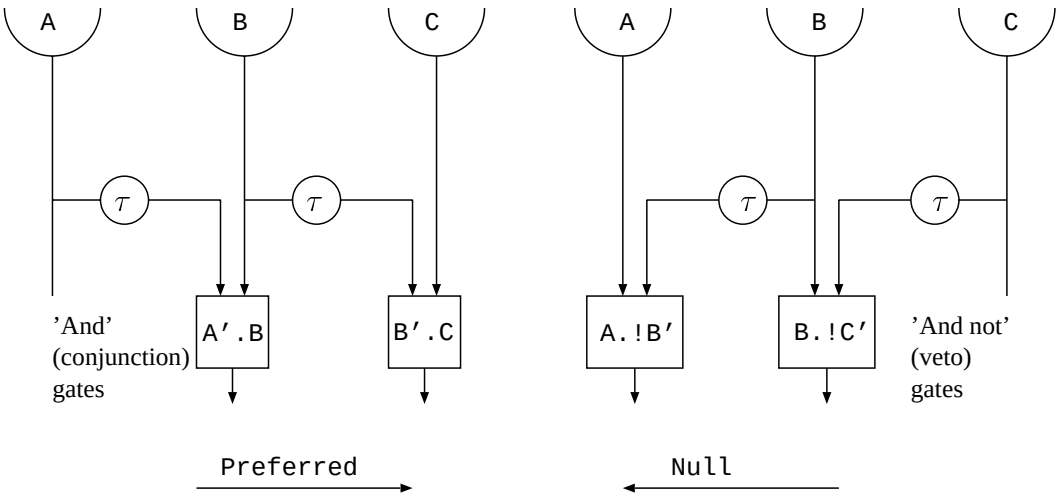
(b) Reichardt motion detector.

Figure 2.3: Reichardt Detector Components.

Inhibitory Detector

The multiplicative interaction employed in the Reichardt detector is an excitatory mechanism. Barlow and Levick [Barlow and Levick 65] pointed out that an inhibitory mechanism was also capable of providing a directionally selective mechanism by vetoing, rather than facilitating, a response. It was demonstrated that inhibition was the dominant component in the motion detection mechanism in rabbits by using a two slit experiment. A pair of closely placed slits were illuminated singly and in sequence and the on and off responses were recorded. The response to the null direction was always less than the sum of the responses to individual slits, indicating that inhibition was occurring. The response to the preferred direction was slightly greater than the sum of the individual responses at very small slit separations, but not at larger separations. Barlow and Levick felt that the facilitory effect detected at small separations was less significant than the inhibitory effect that was detected at all separations. These conclusions received support from experiments using pharmacological agents to disrupt inhibitory interactions [Schmid and Bulthoff 88].

The inhibitory mechanism proposed by Barlow and Levick was implemented using digital logic (Figure 2.4). However, it is equally valid to consider an analog version that replaces "AND" operations with summation and "AND-NOT" operations with subtraction.



(a) Barlow's excitatory system.

(b) Barlow's inhibitory system.

Figure 2.4: Barlow's motion detection architectures.

Alternative inhibitory mechanisms have also been proposed. One of the mechanisms, known as lateral inhibition, was developed as a result of studies of horseshoe crabs and is a

linear interaction [Hartline and Ratliff 74]. A nonlinear version of lateral inhibition, known as *shunting inhibition* was developed by Pinter [Pinter 83] by describing the neurochemistry of visual cells in some detail. Shunting inhibition is described by the following nonlinear differential equation.

$$\dot{m} = L - am - m \sum_i k_i f_i(X_i) \tag{2.1}$$

where m is the response, L is the excitatory input, a is the self decay value, f_i is an activation function, k_i are weights and X_i is an inhibitory input. In the following discussions, excitatory inputs are represented by a dot (•) and inhibitory inputs by a dash (▬).

Shunting inhibition has been used to construct motion detection systems (Figure 2.5) that are successful in describing the same behavioural phenomena as the Reichardt detector [Bouzerdoum and Pinter 93]. The adaptive properties and stability of the shunting inhibitory model make it very important to the first part of this thesis.

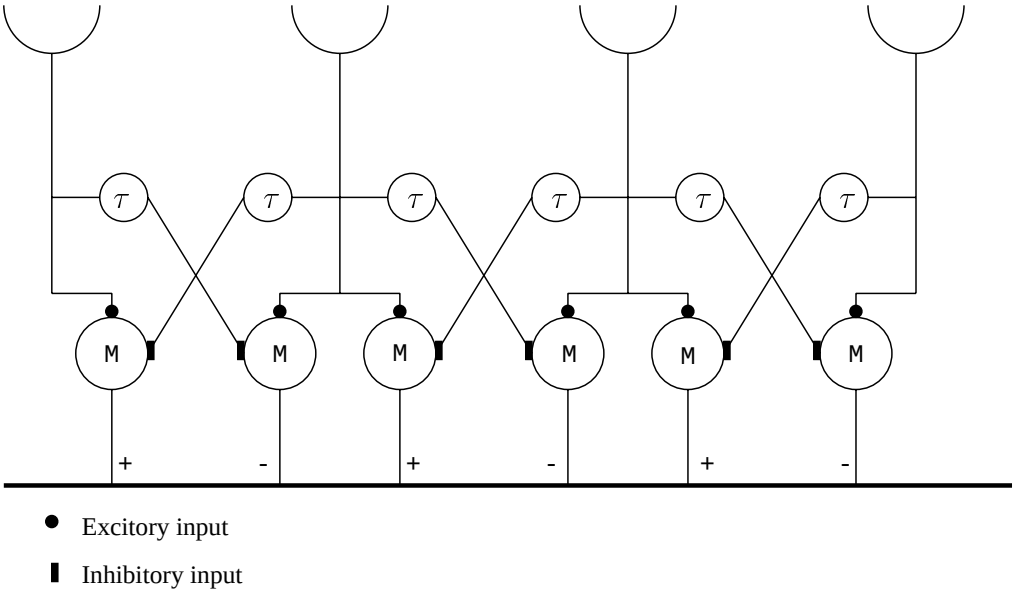


Figure 2.5: Motion Detector using shunting inhibition. M is a shunting inhibitory neuron.

A more comprehensive neurophysiological model of motion detection has been created by Öğmen [Öğmen and Gagné 90b]. Öğmen's system employs nonlinear models of neurotransmitter dynamics and Grossberg inhibitory neurons. The system employs preprocessing layers that model the behaviour of ON-OFF and SUSTAINED neurons in the insect lamina. The behaviour of the SUSTAINED neuron models mimic results gathered by Ar-

nett [Arnett 72]. (Similar results have also been obtained with a much simpler model employing the shunting inhibitory model described above [Beare and Bouzerdoum 96].)

Elaborated Reichardt Detectors

The spatial aliasing predicted by the Reichardt model does not occur in humans. A modified version of the Reichardt detector, known as the Elaborated Reichardt Detector (ERD), has been developed by van Santen and Sperling [van Santen and Sperling 85]. The important modification made in this model was the inclusion of receptive field characteristics (spatial filters) in the input stage (see Figure 2.6). It was demonstrated that careful selection of spatial and temporal filters could produce systems that did not experience spatial aliasing and could also produce EMDs that were equivalent to a motion detector. van Santen and Sperling also pointed out that the energy models capable of estimating velocity (Section 2.4.3) are equivalent to the ERD.

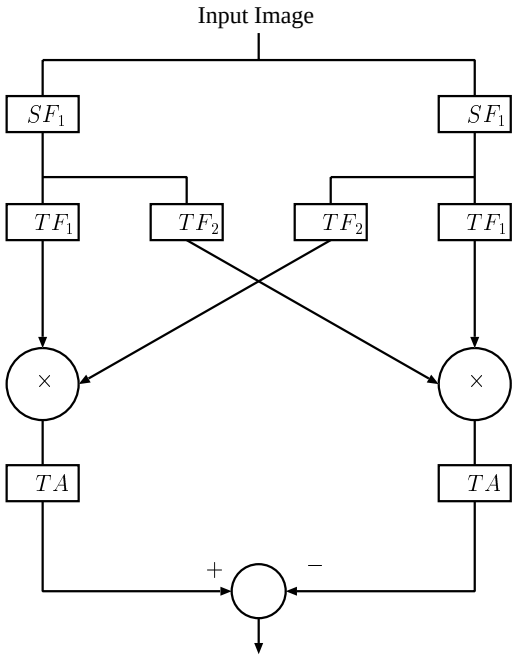


Figure 2.6: Elaborated Reichardt detector. TF is a temporal filter, SF is a spatial filter and TA is a time averaging operator.

2.4.2 Energy Models

The delay and compare models just discussed were derived from behavioural and physiological studies of biological systems. A second class of systems that is also useful in

understanding biological systems were derived to measure spatio-temporal energy characteristics of moving images. Some systems used Fourier domain descriptions of moving images [Watson and Ahumada 85,Fleet and Jepson 89] while others considered time as another spatial dimension [Adelson and Bergen 85].

If one considers the Fourier transform of an unchanging one dimensional image ($c(x)$) being translated at a constant velocity v.

$$c(x - vt, t) \rightarrow \tilde{c}(u, \omega + vu) \tag{2.2}$$

Where u is the spatial frequency and ω is the temporal frequency. This transformation can be considered as a shear in the ω dimension, with temporal frequencies being shifted by $-vu$, while the spatial frequencies are unchanged. In three dimensions the spectrum of the moving image resides in an oblique plane through the origin (Figure 2.7).

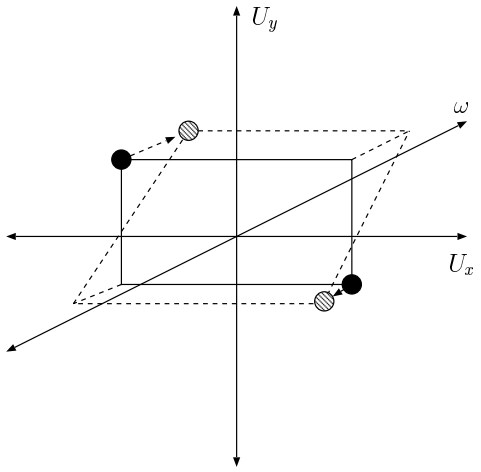


Figure 2.7: Motion shear of a 3 dimensional space-time image. The solid plane is the stationary object, the dotted plane shows the effects of the shear.

Watson and Ahumada designed a scalar motion sensor that exploits this Fourier description of motion (Figure 2.8). The motivation for the design of the sensor was physiological evidence suggesting the existence of spatial frequency tuned channels. The sensor responds to a sine wave grating moving in the preferred direction by producing an output of the same frequency. The output is zero if the grating is moving in the non-preferred direction. This property is provided by Hilbert filters in the quadrature path. The scalar motion sensor only measures the motion energy, not the velocity. Techniques used to determine the velocity will be discussed in the following section.

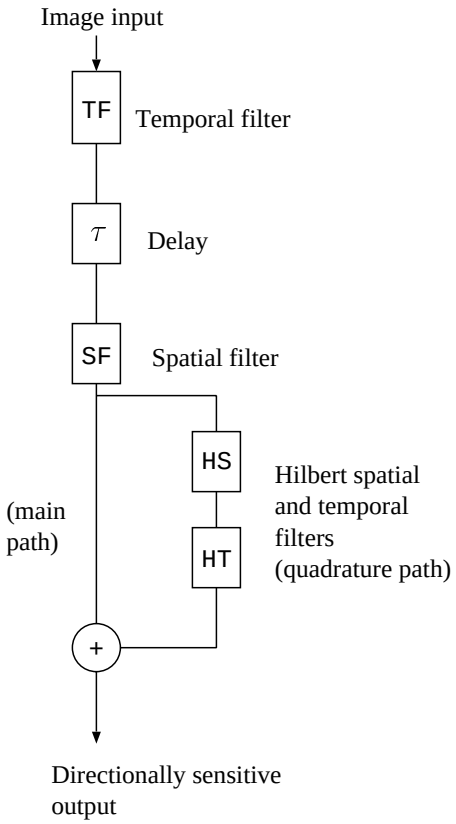


Figure 2.8: Watson's and Ahumuda's scalar motion sensor. The dual paths after the spatial filter are the critical components that provide directional selectivity.

Fleet and Jepson also exploited the Fourier description of moving images using a hierarchical parallel processing scheme. They developed tools to allow the filters used to approach the theoretical limits for velocity tuning and space-time resolution.

A similar approach was used by Adelson and Bergen. They used quadrature pairs of spatio temporal filters, whose outputs were squared and summed to give a measure of spatio-temporal energy. The starting point of Adelson's and Bergen's analysis was not the Fourier description just mentioned, but instead considered time as another spatial dimension, so finding velocity amounted to determining a slope (Figure 2.9).

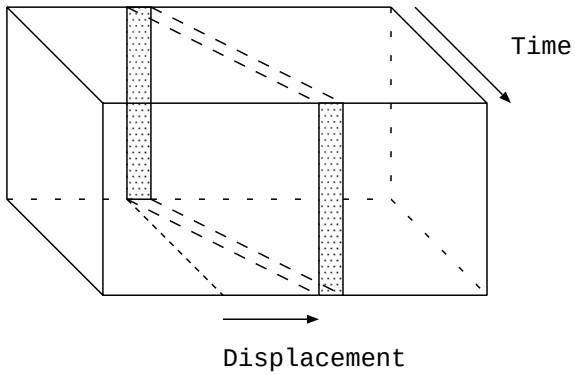


Figure 2.9: Velocity in the spatio-temporal domain as used by Adelson and Bergen.

Heeger [Heeger 87b] has also proposed spatio-temporal energy systems based on three dimensional Gabor filters.

These systems are all measuring motion energy. The mechanisms used to determine velocity information will be discussed next.

2.4.3 Velocity Estimation

The systems just described do not provide a measure of velocity. In fact it is not clear that determining velocity is an essential function of all biological vision systems. The experimental evidence gathered by Reichardt showing a change in response of insects with mean luminance tends to suggest that simple organisms do not measure velocity. However, for many applications, such as determining relative range from motion parallax (Figure 2.1), velocity information is essential.

There is also significant doubt about how a visual system should encode velocity. It is possible to have a number of different detection systems tuned to different velocities. The detector with the largest output would therefore be tuned to a velocity close to the image

velocity. This is termed a “labeling” scheme. This type of scheme is likely to be impractical if a wide range of velocities is expected and a high resolution is required. The second option is to encode the velocity in a particular region using a single signal value. This is economical but cannot be used to represent multiple velocities that may be caused by independently moving objects in the same region.

Watson and Ahumuda proposed a hybrid scheme, which is phsiologically likely. They used several motion sensors, described in Section 2.4.2, at each image location, each tuned to a different spatial frequency and with a different orientation. The outputs were combined by fitting to a cosine function to encode the velocity as an intensity.

Adelson and Bergen needed to eliminate contrast dependency to determine velocity. The scheme proposed also operated by combining several channels, in this case leftward sensitive, static, and rightward sensitive channels. Importantly this scheme could also use the static channel as a form of confidence measure.

van Santen and Sperling claim that the addition of the nonlinear velocity estimation stages make both of these systems equivalent to the ERD [van Santen and Sperling 85].

Snippe and Koenderink [Snippe and Koenderink 94] have proposed a multiple input Reichardt detector capable of extracting velocity. By combining many detectors it was possible to create systems tuned to different velocities. A multiple input detector also eliminates problems posed by evidence suggesting variation in spatial and temporal tuning of detectors in biological systems. This system employs a labeling scheme to encode velocity.

2.4.4 Other Systems

Other methods for measuring optical velocity have been developed by the general engineering and machine vision community. The simplest method was devised as a non-contact method for velocity estimation in applications such as steel rolling mills. The system employs a narrowly tuned spatial filter observing the moving object and providing input to a photosensor. If the input image contains a wide range of spatial frequencies, then the velocity can be determined by dividing the temporal frequency output of the photodetector by the spatial frequency of the spatial filter. This system does not determine the direction of motion.

A second class of methods for velocity estimation known as gradient schemes has been developed by the machine vision community. These systems relate the image velocity to local spatial and temporal derivatives, and are only accurate if the global change in luminance is zero (or known). This relationship is described by the brightness change constancy equation (BCCE) [Horn and Schunck 81]

$$-\frac{\partial f}{\partial t} = \nabla f \cdot \mathbf{v} \tag{2.3}$$

where ∇f is the spatial gradient of the image and $\mathbf{v}$ is the velocity.

The sensitivity of the scheme to global changes in luminance makes the scheme unreliable in many circumstances and therefore unlikely to be used in biological systems [Nakayama 85].

Despite this problem the apparent simplicity of the system has attracted a great deal of interest from the machine vision community. Significant problems such as extracting reliable measurements from noisy conditions have been tackled [Srinivasan 90, Poggio et al. 85].

A third method known as the *template model* shares both biological and engineering heritage [Horridge 91, Moini et al. 93, Nguyen 96]. The template model uses bandpass filters to process an image sequence, samples the output and classifies the result as an increase in intensity, a decrease in intensity or a no-change state. Groups of four responses from adjacent spatial channels and successive time instants are formed into templates (Figure 2.10). Only eight of the possible eighty one templates indicate coherent motion. This significantly reduces the computational requirements in processing the data. Velocity may be estimated by tracking these templates [Nguyen et al. 96].

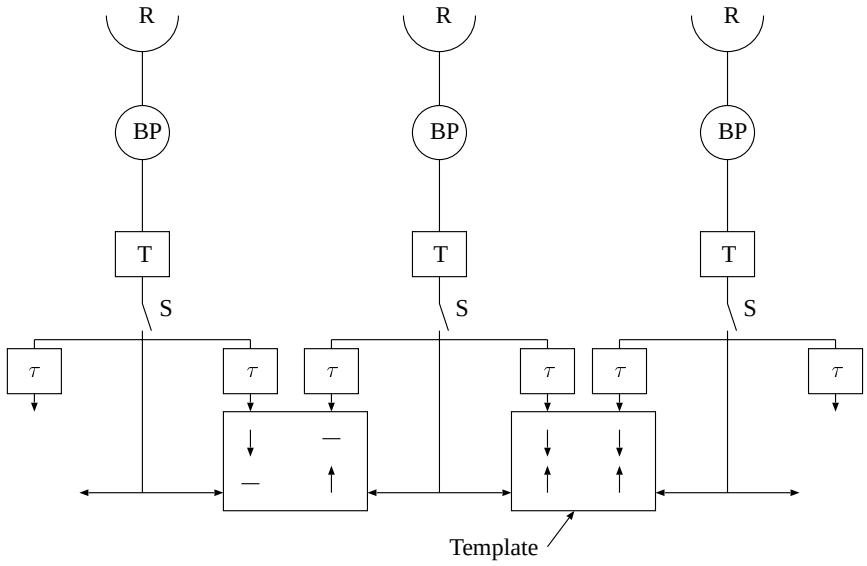


Figure 2.10: The template model. Receptors are indicated by "R", thresholding by "T", sampling by "T" and delays by "τ".

Other schemes have been developed specifically for analog VLSI implementation. A common technique is to explicitly measure the time difference between large changes in intensity at adjacent pixels. The structures used to do this are similar to the delay and compare

schemes discussed earlier. A potential problem with this kind of architecture is the necessity to make thresholding decisions very early in the visual process. An example of this kind of system is presented in [Kramer et al. 95].

It is also possible to apply conventional optimal tracking techniques, like Kalman tracking, to large spatial regions or complex features using a variety of matching techniques. The matching process may be correlation based or may use some other form of heuristic with a lower computational cost. Edges and corners are common choices of features that can be tracked [Deriche and Faugeras 90, Smith and Brady 95], but arbitary regions can also be used as features. In general well defined features like corners and edges are more popular in computer vision while region tracking is used in video coding. Tracking systems generally produce a sparse velocity field.

Fuzzy systems have also been applied to the problem of estimating image velocities by tracking pixel brightness levels [Kouzani et al. 95]. Fuzzy systems are attractive due to their ability to represent uncertainty and the possibility of parallel implementation.

2.5 Conclusions

This chapter has reviewed the biological applications of motion information and techniques for extracting visual motion information from the environment. This review was not a comprehensive one. It concentrated on the most popular models and those models that are of some relevance to this thesis.

There can be no doubt that biological vision systems make extensive use of motion information, and it appears that there is a wide variety of mechanisms employed to extract it. All of the applications of motion information are of potential use to artificial systems. Physiological evidence suggests that perception of these interesting, but relatively complex, quantities is likely to involve some form of hierarchical combination of elementary processes. However, it is also apparent that the early stages of processing which are preattentive and therefore do not invlove cognition are particularly important. Models of biological velocity estimation and motion detection schemes have been developed in the past. Both types of schemes are likely to be useful for different types of application, but velocity estimation schemes are more complex.

Adaptation to environmental conditions and use of minimal hardware are two well known and much envied abilities of biological vision systems. Biological systems have spent millions of years of evolution to optimise the tradeoffs between many conflicting requirements. Many of the solutions, both computational and structural, are likely to be valuable to design-

ers of artificial systems. The development of motion processing systems with these properties has been largely ignored, with many models assuming (not necessarily unreasonably) that the appropriate adaptation happens before motion processing takes place.

The transition from a biological model to an artificial system is rarely a straightforward one. The type of hardware available to biological systems is very different to the VLSI systems available to designers today. Therefore models are often modified and simplified to make an implementation practical.

Chapter 3

Biological Models and Artificial Systems

3.1 Introduction

The traditional approach to constructing artificial vision systems typically involves using a conventional camera followed by a digitising system and either specialised or conventional digital computing hardware. The resulting systems tend to be bulky, expensive and have high power consumption, properties that significantly limit the application domain. More recently the trend has tended towards the design of specialised micro-electronic sensors that perform significant levels of processing in addition to sensing. Such devices are known as *smart sensors* and have the potential to offer a cheap, compact and low power building blocks for artificial vision systems. The discussion of the previous chapter demonstrated the importance of motion information to biological vision systems. It is reasonable to expect that a smart sensor capable of performing similar types of processing would be a useful component for many types of system, especially mobile platforms.

Deciding exactly what type of processing should be performed is a difficult task since both hardware and computational constraints must be considered. This chapter explores the type of motion processing that can be expected to be achievable using VLSI technology, and defines the functionality required of this processing. A local motion detection system will be introduced as an appropriate compromise between complexity and functionality.

The motion processing systems reviewed in the previous chapter are investigated to determine whether they can form a suitable basis for designing local motion detectors. Finally, some modifications that can be made to the basic systems to produce local motion detectors are introduced.

3.2 Goals

A sensor capable of providing an estimate of image velocity at every image pixel is likely to be a useful device. Unfortunately the systems reviewed in Chapter 2 that are capable of estimating velocity are quite complex. They tend to rely on multiple channels and complex spatio-temporal filters. Such filters present a serious challenge for analog VLSI. Simplified versions of many of these systems have been built using VLSI technology (see [Moini 97] for a review), but most of them are one dimensional or have a low resolution and often suffer a lack of robustness due to noise. It is possible to produce a digital implementation of these complex systems, but this does not exploit the potential advantages of a smart sensor system.

The delay and compare schemes reviewed in Chapter 2 appear to be simpler and therefore provide a more attractive basis for a smart sensor design. The simplest delay and compare schemes only detect motion, without estimating velocity. The extensions that enable such systems to estimate velocity involve using multiple inputs motion detectors tuned to different velocities. This increases the system complexity.

The simplest motion detectors were designed to model wide field behaviour, i.e. detect motion in a wide fiels by summing inputs of multiple EMDs. It is not clear that a purely widefield sensor is an especially useful general purpose device, although it could be useful as a proprioreceptive type sensor. This chapter will therefore investigate local motion detectors as a compromise between complexity and functionality. Local motion detectors do not estimate velocity but are capable of detecting the direction of motion between two or three adjacent receptors.

3.3 Local Motion Detection Systems

Most of the work done modelling biological motion detection systems has involved widefield, steady state behaviour. In general the stimuli have also been wide field, typically drifting gratings. Some exception include the transient response of widefield neurons to wide field stimuli [Egelhaaf and Reichardt 87], and the response of wide field neurons to localised, transient stimuli [Franceschini et al. 89]. The response to localised transient stimuli of earlier processing layers, such as the large mono-polar cells (LMCs), has also been investigated [Arnett 72].

The lack of standard tests for local motion detectors makes it necessary to define some operating criteria that are desirable for an artificial sensor.

3.3.1 Operating Criteria

The operating criteria for local motion detectors will be defined in terms of response to moving edges, since edges tend to be a significant perceptual component of most scenes. This is different to the traditional approach involving moving sine wave gratings that are invaluable for determining steady state characteristics of motion processing systems.

The requirements of a local motion detector are:

- The response polarity (or sign) must indicate the direction of motion;
- The response must be independent of the sign of the contrast of the moving object;
- The response should be robust to noise;
- There should be no response to a stationary edge;
- The position of the response should be close to the position of the edge;
- Spatially separated edges should produce spatially separated responses.

The last two points distinguish wide field from local detectors.

No restrictions will be placed on the shape of the response, which will depend on the shape of the edge for most detector types. Sample responses can be seen in Figure 3.1.

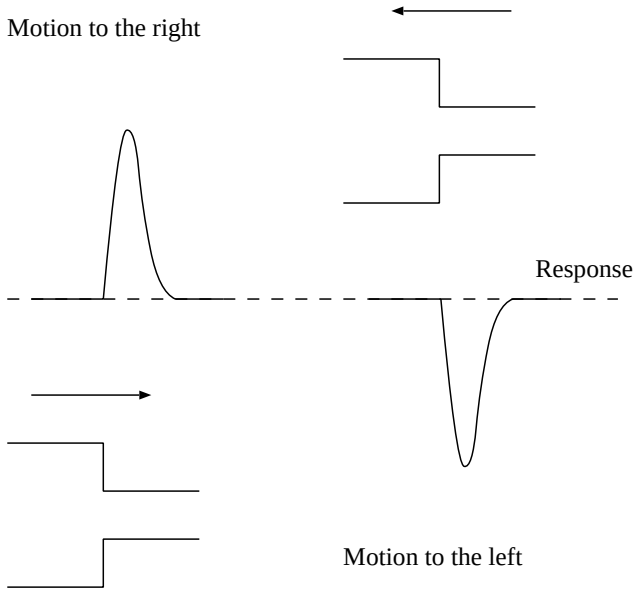


Figure 3.1: Desired forms of response.

Test Input

The standard stimulus used to test local motion detectors is of the form

$$L(t) = L_0(1 + cu(t - t_0))$$

where L_0 is the mean luminance, c is the contrast ($-1 \leq c \leq 1$), and $u(t)$ is a unit step. This stimulus is a simple model of an ideal moving edge. Note that if the mean luminance L_0 is zero then there is no stimulus.

3.4 Practical Considerations

There are two extremely important issues that must be considered when designing visual sensors. These issues have been largely ignored by the models of biological systems discussed so far. They are: performance under noisy conditions and mechanisms to handle a wide dynamic range of inputs.

3.4.1 Dynamic range

The levels of mean luminance experienced by a visual sensor can vary enormously during the course of a day. The difference in luminance between faint starlight and bright sunlight can be as much as 12 orders of magnitude. The human eye is capable of functioning surprisingly well over this range. This is an amazing feat when the often poor characteristics of neural hardware, like low dynamic range and significant internal noise, are considered. The dynamic range of a neuron is usually quite small, so it is essential that sensory layers only transmit the information that is required by the subsequent processing layers in order to make best use of the available communication bandwidth (dynamic range).

For example, an image may be represented by a relative contrast measure or a measure of received light intensity at every receptor. The former representation is independent of mean luminance and requires a much lower dynamic range than the latter while still being suitable for most image processing algorithms. Unfortunately it is not possible to directly measure the contrast at every receptor — some measure of incident light level must be made instead. The dynamic range represented by the signal must be reduced by the early sensory layers to avoid dynamic range problems when designing the post-processing layers. The process of reducing the dynamic range is known as adaptation. Exactly how adaptation is performed is dependent on the requirements of the post-processing layers. In some cases the adaptive processes are closely related to conventional image processing tasks.

In conventional imaging systems the goal is to provide an image representation that is presented to a human. The sensory layer may be film or electronic sensors. The adaptive processes influence the amount of light impinging on the sensing layer by controlling the aperture size and the exposure time. A typical adaptive architecture for a conventional imaging system is shown in Figure 3.2. This is a global approach because the timing signals and aperture size have the same influence on all of the sensor area. Global approaches tend to be quite limited when compared to the more localised methods used by biological systems — it is quite common to see photographs that are partially over or under exposed while humans have no difficulty with the same scene. This is particularly impressive when one considers that the dynamic range of film can be quite large. The iris of the human eye is an adaptive mechanism, but not a dominant one. The dominant adaptive mechanisms are found in the retina.

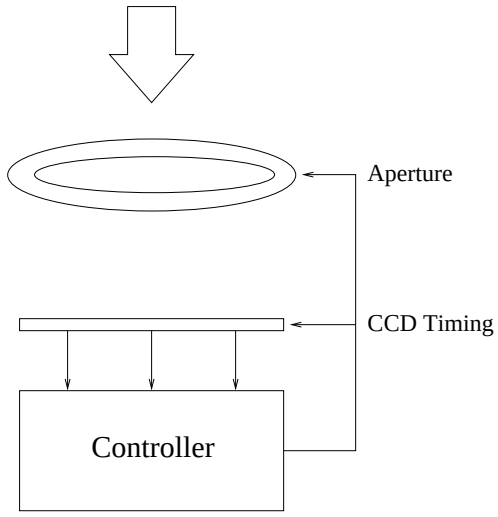


Figure 3.2: Globally adaptive system architecture.

The adaptive processes in the early visual processing layers of biological systems act to eliminate redundancy or maximise information flow [Srinivasan et al. 82, Laughlin 87, Laughlin 89, van Hateren 92]. These processes may be either spatially adaptive, temporally adaptive, or both. Spatially adaptive processes act to eliminate spatial redundancy and may be regarded as a form of edge detection, while temporally adaptive processes eliminate temporal redundancy and are equivalent to bandpass filters. The choice of the adaptive process depends on the type of information that is required by the subsequent processing systems. If an adaptive mechanism that also performs some of the necessary processing can be found (eg edge or motion detection), then a reduction in hardware complexity should be possible.

The ways in which nature optimises the use of limited computational and bandwidth resources is of special interest to designers of artificial sensor systems, offering techniques to improve performance and increase robustness of artificial sensors.

3.4.2 Noise Performance

Noise is always a problem in visual systems. It may be introduced at the sensing or processing stages. An especially important consideration is the way in which the noise characteristics change with luminance. Shot noise is the dominant type of noise that visual systems experience. Shot noise has a square root dependence on the number of particles (or events) involved in a measurement. The signal to noise ratio is therefore proportional to the inverse of the number of events involved in a measurement. The events in question may be photons impinging on photosensors or electrons being formed within the sensors. Therefore the signal to noise ratio in a visual system will be low at low luminance levels and increase as the luminance increases. It is important that the characteristics of the visual system be able to change to take this into account. At low luminance levels derivative type operators, like edge detectors, become error prone because noise tends to be the dominant high frequency component. This means that operator characteristics need to be dependent on luminance, with generally low pass characteristics at low luminance levels and bandpass characteristics at higher luminances. In biological systems this type of effect is predicted by the maximal information flow approach described by van Hateren and has been observed experimentally [van Hateren 92, Srinivasan et al. 90].

3.4.3 Comments

The biologically inspired motion detection architectures described so far were intended to model behaviour under relatively well defined and repeatable experimental conditions and therefore did not take the problems of dynamic range and noise into account. These problems must be considered in any practical system, and inspiration for tackling them can definitely be obtained from biological systems.

In the past, these problems have been tackled by producers of conventional imaging devices such as film and video cameras, however the goals were quite different to those of most smart sensors and the mechanism employed were relatively simple. Typical adaptive mechanisms, employing a mechanical iris and timing control, are globally adaptive rather than locally adaptive and therefore cannot handle a wide dynamic range at one time. Noise at low light levels is less of a problem for imaging devices because no processing of information

is being performed. Long exposure times help to reduce the problem.

The important difference between imaging systems and smart sensors is that imaging systems produce a representation of light intensities in a scene for presentation to a human, whereas smart sensors usually extract a very different type of information, such as velocity, from a scene for presentation to an artificial system. A smart sensor is therefore likely to be able to employ the lessons taught by biological systems and significantly reduce the signal redundancy by using different image representations. If the adaptive mechanisms can produce the appropriate representation, in addition to reducing the dynamic range, then there could be a significant saving in hardware complexity and improved performance. Designing adaptive mechanisms that are well matched to the types of processing being performed, in particular motion detection, is the goal of the first half of this thesis.

3.5 Performance of delay and compare schemes

In this section we investigate the performance of the simplest delay and compare schemes described in Chapter 2. The characteristics of interest are:

- Response to stationary and moving edges.
- Dynamic range requirements.
- Noise performance.

It will be shown that the simplest delay and compare schemes do not meet these requirements.

3.5.1 The Reichardt Detector

A single Reichardt motion detector without time averaging is shown in Figure 3.3. The structure of the detector without time averaging is appropriate for a local motion detector because the transient response to a local stimulus is available. Note that the original Reichardt model described in [Reichardt 61] employed additional low pass filters in each branch of the EMD. These filters were required to produce an accurate behavioural model, but are not necessary to produce a directionally sensitive wide field response.

Edge Response

The response of the local Reichardt detector to a moving edge is shown in Figure 3.4. The response consists of a pair of peaks of opposite sign. The order of the peaks is dependent on

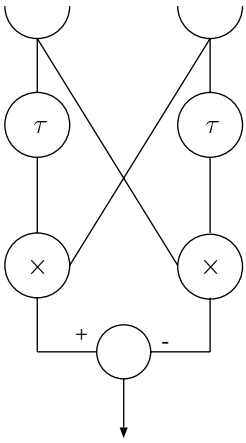


Figure 3.3: Local version of the Reichardt detector. τ is the time constant of a first order low pass filter. This filter acts as a delay element.

both the direction of motion and the change in contrast. This response does not possess the desired form because it is not possible to determine the direction of motion by examining the sign of the response. One desirable feature of the system is that the response of the system is zero if no motion is present. The zero response occurs because the multiplication operation is commutative and the inputs to each EMD are the same.

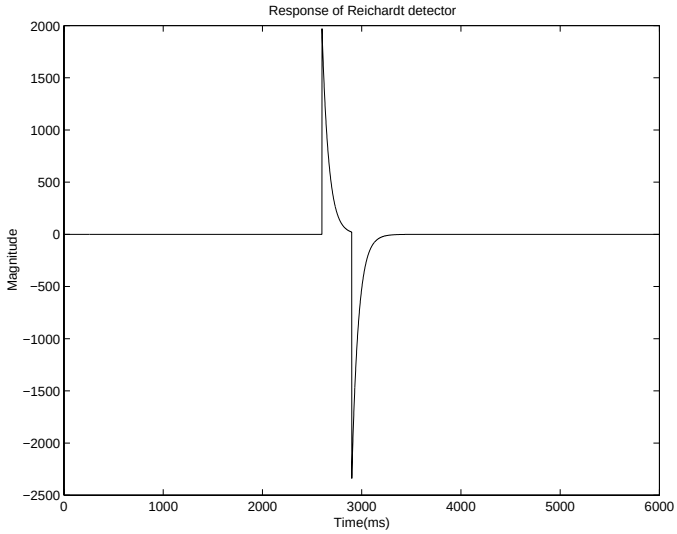


Figure 3.4: Response of local Reichardt detector to a moving edge. $c = 0.2$, mean luminance $(L_0) = 100$, delay filter transfer function $H(s) = 15/(s + 15)$.

3.5.2 Local feedforward Inhibitory Detector

The local version of the feedforward inhibitory system is shown in Figure 3.5.

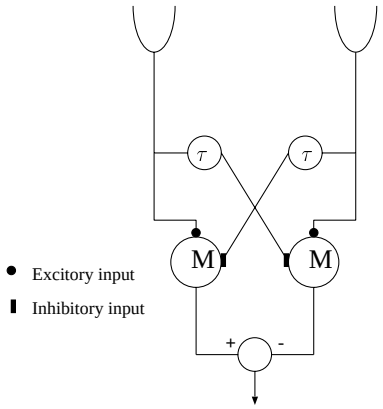


Figure 3.5: Local Inhibitory detector. M is a shunting inhibitory neuron and τ is the time constant of a first order low pass filter.

Edge Response

The response of the feedforward inhibitory system to a moving edge is a pulse as desired, see Figure 3.6. Unfortunately the sign of the pulse is dependent on both the sign of the contrast change and the direction of motion (unless the input is separated into ON and OFF channels as discussed in Section 3.6.2). Therefore, this system is not a useful local motion detector. An additional problem is that the response to a stationary step (i.e. an edge located between the two input receptors that has been stationary for some time without changing magnitude) is nonzero. This problem is caused by the non commutative nature of the division operation that is performed by the shunting inhibitory neuron in the steady state. The input signals to the left and right neurons are reversed — the excitory input to the left neuron and the inhibitory input to the left neuron are the same in the steady state. This means that the steady state response of the local detector is given by

$$Response = \frac{L_1}{a + L_2} - \frac{L_2}{a + L_1} \tag{3.1}$$

where L_1 is the input to the left receptor and L_2 is the input to the right receptor. The response is only zero when $L_1 = L_2$.

A second system based on feedback inhibition has also been investigated. The structure of the feedback system can be found in Appendix A. The feedback architecture eliminates

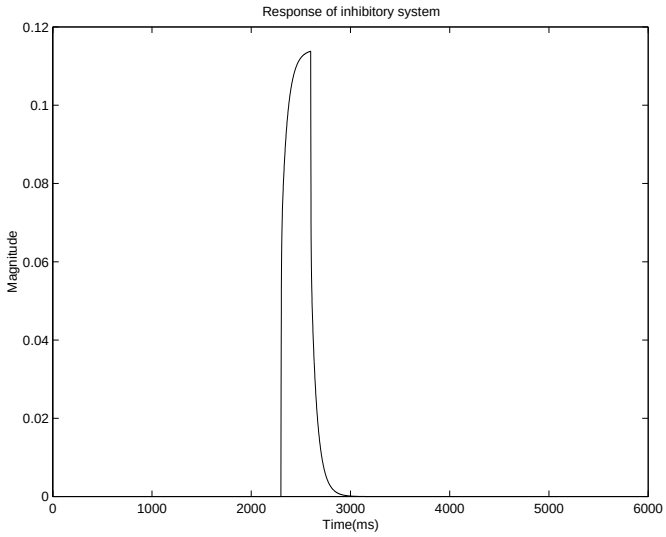


Figure 3.6: Response of local inhibitory detector to moving edge. $c = 0.2$, mean luminance $(L_0) = 100$, self decay parameter $a = 15.0$, delay filter $H(s) = 15/(s + 15)$, gain of inhibitory input $= 1$.

the need for explicit delay filters because the self delay of the neuron is exploited instead. This is attractive because it is not necessary to construct delay elements that operate over a wide dynamic range; however, the form of the edge responses are identical to the feedforward inhibitory system and therefore do not meet the requirements.

Despite the fact that neither of these architectures satisfy even the most basic of requirements, other properties will be examined to guide the design of alternatives.

3.5.3 Noise Performance

Another important issue that must be considered in any artificial vision system is the robustness under noisy conditions. Noise is most likely to be a problem at low luminance levels, when the signal to noise ratio is lowest. Our investigations have demonstrated that inhibitory systems have a significant advantage over the Reichardt detector under these conditions (see Figure 3.7). The reason for the performance disadvantage of the Reichardt detector under these conditions is that the multiplicative interaction acts as a gain and increases the noise power. Details of the analysis of noise performance of the two systems can be found in Appendix A, and the results are outlined in [Beare et al. 95].

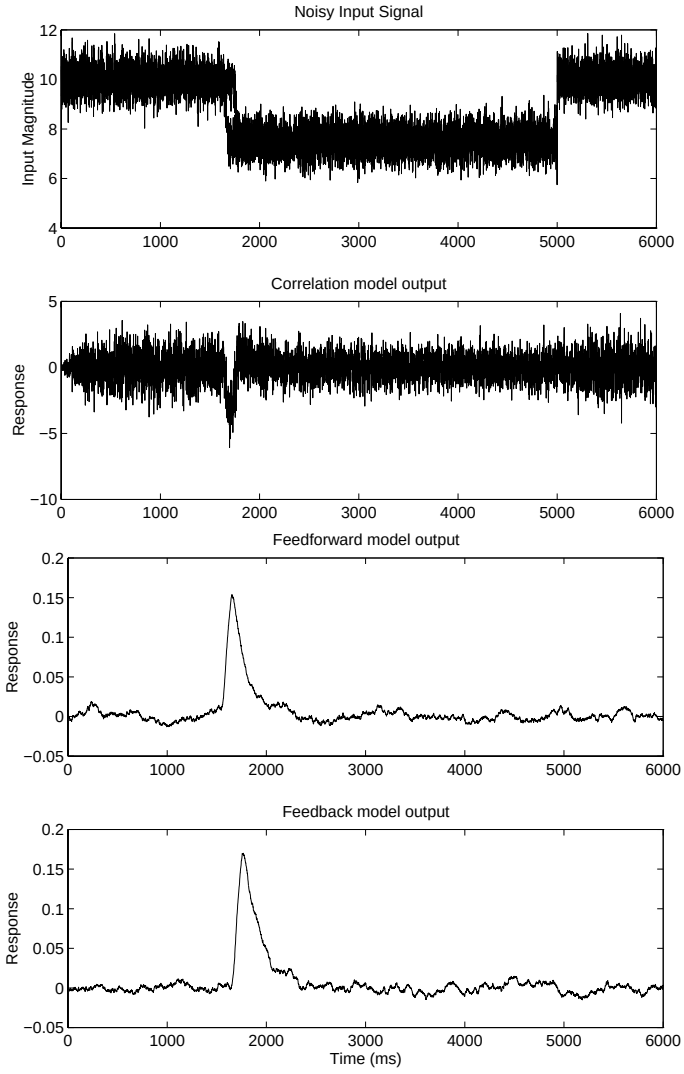


Figure 3.7: Response of various detectors to noisy signals.

3.5.4 Adaptive Properties

The adaptive property of interest at this stage is compression of dynamic range. As mentioned earlier, the luminance levels can change by approximately 12 orders of magnitude over the course of the day, however a typical photosensor may only be expected to operate over 6 or 7 orders of magnitude [Gruss et al. 91]. (The human eye uses two different types of photo receptor to achieve light and dark adaptation). If a moving edge of relatively high contrast ($c = 0.4$) is observed under high luminance conditions then the change in input signal between adjacent receptors will be large enough to cause dynamic range problems for the processing systems if the signal is not compressed in some way.

The Reichardt detector uses a multiplicative interaction between adjacent channels. Such an interaction is very undesirable from the point of view of adaptation as it increases the signal range that the processing systems must be able to handle. The effect is also observable in software simulations of the Reichardt detector, where a significantly larger numerical range is required to represent the signal after the multiplication than before it (which is hardly surprising).

The shunting inhibitory neuron implements a steady state division. Division is really the ideal mechanism with which to eliminate mean luminance and retain the contrast information.

The peak response of a single Reichardt motion detector of Figure 3.3 to a step edge of contrast c, ($c < 1$), is given by

$$peak = L^2 c$$

where L is the mean luminance. (One side of the edge has magnitude L and the other has magnitude $(1 + c)L$).

For the inhibitory system the peak response is given by

$$peak = \frac{cL}{a + (1 + c)L}$$

Note that the selection of the value of a is important to the behaviour of the system at low luminance levels.

These calculations assume that the delay elements have unity gain, and that the gain of the inhibitory input, k in Equation 2.1, is one. The steady state gain of the delay elements do affect the peak response but the dynamics do not. This is because the peak response of the Reichardt detector occurs when the change caused by motion is experienced by the undelayed input to the multiplier but the delayed input has not changed. Therefore the peak

response occurs during the transient, and is velocity independent. The peak response of the inhibitory detector occurs when the output of the delay element has reached steady state, i.e. when the edge is stationary. As the edge velocity increases the peak response is reduced because the low pass filters do not reach steady state. This difference in behaviour is due to the asymmetric nature of the division being performed by the inhibitory neuron.

A comparison of the two responses is shown in Figure 3.8. As expected the dynamic range requirements for the post-processing circuits of the correlation model would be very large, while the inhibitory system reduces the dynamic range requirements to more manageable levels.

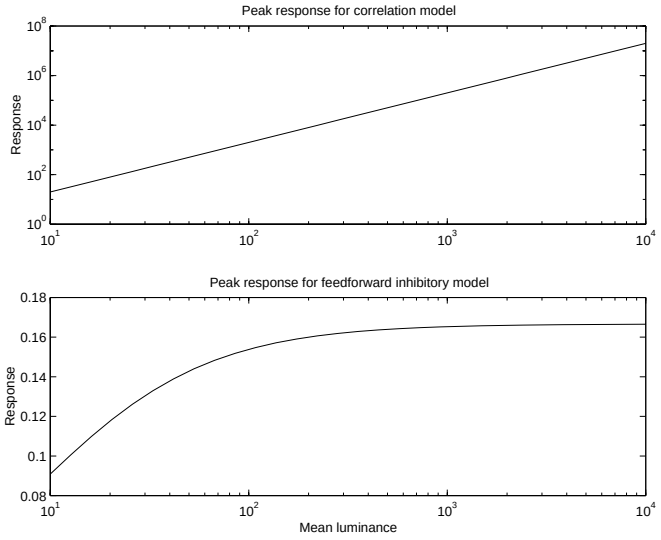


Figure 3.8: Peak responses for the correlation model and feedforward inhibitory model for an edge of contrast 0.2. The inhibitory model has a self decay parameter $a = 10$. Delay elements, τ, have unity gain.

3.5.5 Comments

Neither of these simple delay and compare schemes meet the requirements outlined in Section 3.3.1 when used as local detectors. The responses are not directionally sensitive, the response of the inhibitory detector to a stationary edge is not zero, and the Reichardt detector does not reduce the dynamic range of the signal. Therefore, these systems are not suitable as local motion detection systems.

3.5.6 Extending the basic architectures

There are some simple modifications that can be made to the basic delay and compare architecture to produce useful local motion detectors. Before discussing these modifications, the differences between the widefield operation and local operation of the basic detectors will be investigated in detail.

3.6 Failure Modes for Local Detectors

The basic mode of operation for a delay and compare motion detection system is to store a signal from one receptor using a delay mechanism and compare this signal to the undelayed output of the adjacent receptor. If there is a change which, when delayed, matches the change in the output of the adjacent receptor then motion is detected. Exactly how this "match" is determined is the critical part of a receptor pair scheme.

Let us first examine how a widefield version of the Reichardt detector produces the correct response to a moving edge. The response of a single Reichardt motion detector to a moving edge is shown in Figure 3.4, and a small part of the widefield architecture is shown in Figure 3.9. The response consists of a pair of peaks of opposite sign. The first peak occurs when the moving edge causes a change at one receptor. The decay towards zero commences as the delay element charges towards its steady state value. The second peak occurs as the second photodetector experiences the change in input and the second decay begins as the second delay element charges. The second peak occurs at the same time as the first peak from the adjacent motion detector. The absolute magnitudes of the positive and negative peaks are different due to the nonlinear interaction between channels. If the edge is positive then the first peak will have a magnitude of

$$peak_1 = L^2 c$$

and the second peak will have magnitude

$$peak_2 = -L^2 c(1 + c)$$

where L is the mean luminance and c is the contrast; the delay element has a steady state gain of one. (The time constant of the delay element does not affect the peak value.) An increasing edge has magnitude $L(1 + c)$ while a decreasing edge has magnitude $L(1 - c)$. The absolute magnitude of the second peak is greater than the first. If the simultaneously occurring positive and negative peaks from adjacent motion detectors are added then a net negative response will result.

If the edge is negative then the first peak has a magnitude of

$$peak_1 = -L^2 c$$

and the second has a magnitude of

$$peak_2 = cL^2(1 - c)$$

The sum of these two values is also negative, so the sign of the response is independent of the relative edge contrast. If the edge is moving more rapidly then the delay element will not reach a steady state before the second photodetector experiences the change in input. The absolute magnitude of the second peak will therefore be lower fo a positive edge (the first example) and higher for a negative edge (the second example). The first peak will not change. The sign of the sum of the positive and negative peaks does not change.

In the widefield (or time averaged) configuration the peaks are cancelled by neighbouring detectors to produce a response which correctly indicates the direction of travel of the edge. The summation must be over a sufficient number of motion detectors to produce a response of sufficient magnitude to override the uncancelled peaks at the ends of the array. If the additional low pass filters required to produce the accurate behavioural model are also included (see Section 3.5.1), then the peaks in response of individual motion detectors will not be as sharp; however, the nonlinearity will still result in a directionally sensitive response.

The feedforward inhibitory system has similar behaviour, but some restrictions on the type of input are necessary. Instead of cancelling part of the response caused by each edge, the system cancels the responses caused by symmetrical positive and negative contrast changes.

In a widefield or time averaged configuration both of these architectures are able to exploit their nonlinear characteristics to achieve the desired response by cancelling the linear parts of the response. In the local versions this cancellation does not occur and the linear effects dominate. The nonlinear interaction in the correlation detector means that the positive and negative parts of the response to a moving edge have different magnitudes. After time averaging this difference is all that remains, and the sign of the difference indicates the direction of motion of the edge. In the inhibitory systems the nonlinear interaction is a division and is asymmetric. Therefore the cancellation does not occur for single edges, but will happen for strips with identical contrast at both edges (i.e. some form of symmetrical input is required).

The systems are exploiting their inherent nonlinearities to encode the sign of the change in signal, i.e. to indicate whether the temporal derivative is positive or negative. Unfortunately these widefield and time averaged mechanisms do not function for local versions of

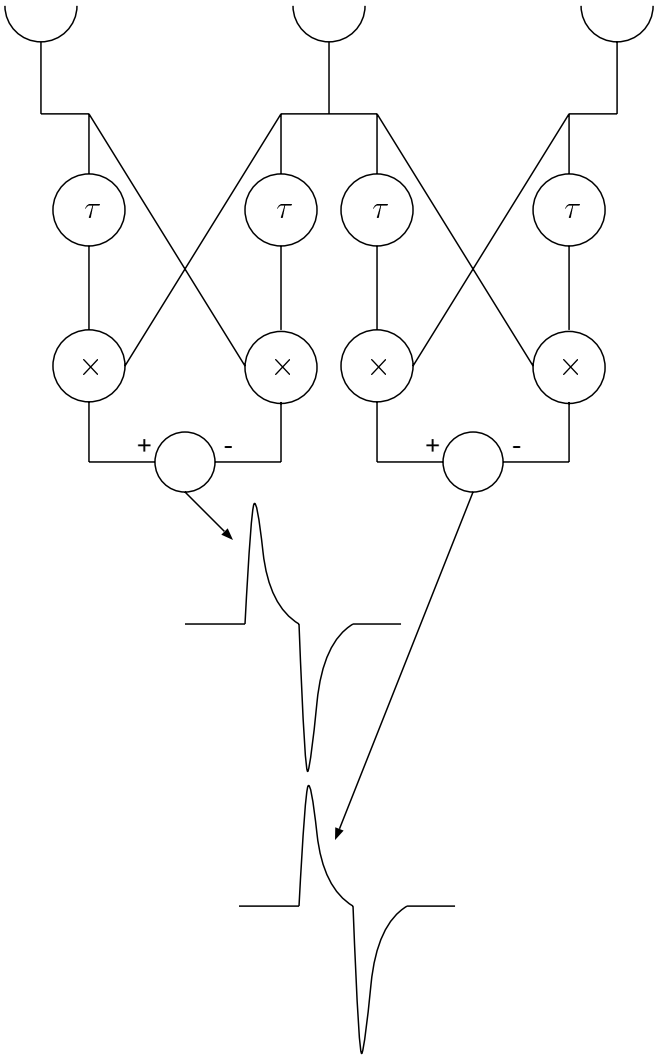


Figure 3.9: Local version of the Reichardt detector.

the detectors using only 2 or 3 receptors, so the output becomes dependent on the sign of the change in input signal.

3.6.1 Modifications

Two important modifications are commonly made to the simplest delay and compare schemes to produce useful local motion detectors. The modifications are:

- Preprocessing with bandpass filters to produce a zero steady state input to the motion detector layer.
- Using ON and OFF channels.

These modifications are inspired by structural studies of early visual processing layers in insects and primates. Temporally adaptive cells, like the large monopolar cells (LMCs) found in the insect lamina, are functionally equivalent to bandpass filters under most conditions and process input signals before they reach the motion detection systems. The properties of LMCs have been extensively studied and they appear to be an important adaptive unit [Arnett 72, van Hateren 92, Pinter et al. 90, Laughlin 89, Srinivasan et al. 90]. There is also evidence supporting the existence of ON and OFF channels in insect visual systems [Franceschini et al. 89, Horridge and Marcelja 90]. Separate ON and OFF channels are used in addition to preprocessing with bandpass filters. Some biologically inspired bandpass filter designs require the use of such channels [Öğmen and Gagné 90a]. Using separate ON and OFF channels has some potential advantages. The design of the multiplier in the Reichardt motion detector is simplified because four-quadrant operation is not required. This permits the use of neural multiplication models such as the one described in [Srinivasan 76]. Separate channels can also help to improve the form of the response in inhibitory detectors (see Section 3.6.2).

Modified Reichardt Detector

If either input to a Reichardt EMD is zero then the response will also be zero, due to the multiplication operation. The multiplication operator is therefore capable of acting as a gate to switch an input on or off depending on the value of the other input. The inputs to the two EMDs forming a motion detector are shown in Figure 3.10. The response of the left EMD will be non-zero between times a and c, when the bandpass filter input is non-zero. The response of the right EMD will be non-zero between times a and b. The difference between the two pulses produces a directionally sensitive response. A consistent motion stimulus

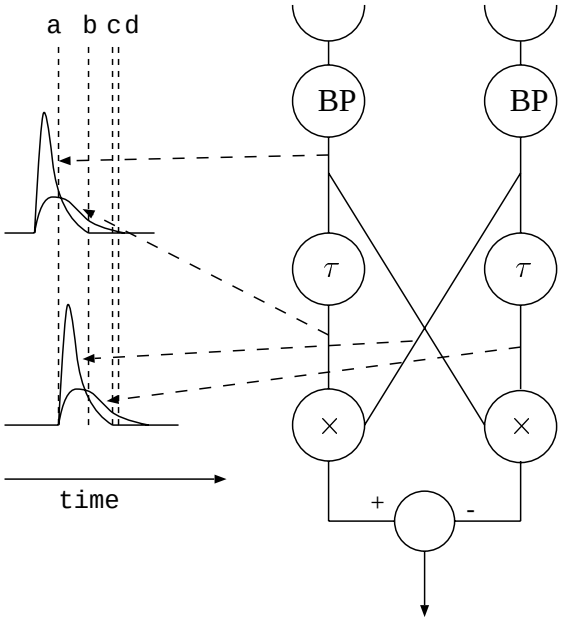


Figure 3.10: Reichardt motion detector with bandpass filter preprocessing.

will produce two bandpass filter responses with the same sign. The motion detector response will therefore be independent of the change in contrast of the edge because the multiplication operation will always produce a positive output. (This is true for a four quadrant multiplier. If the inputs are split into ON and OFF channels then a single-quadrant multiplier is sufficient.) The reverse phi stimulus described in Section 5.2.2 will not produce inputs to the motion detector which have the same sign.

The operation of the Reichardt motion detector with bandpass filter preprocessing is quite different. The nonlinear interaction is acting as a gate and is not needed to code the sign of the change in intensity, since this is being done by the preprocessing layer.

3.6.2 Modified Feedforward Inhibitory Detector Operation

Inhibitory systems do not have the benefit of gating signals in the way the multiplication does. This is a significant disadvantage. The output of an inhibitory neuron will be nonzero whenever the excitory input is nonzero, as can be seen in the equation below describing the steady state response of an inhibitory neuron.

$$m = \frac{L}{a + \sum_i k_i f_i(X_i)} \tag{3.2}$$

The presence of inhibitory signals reduces the magnitude of output from the neuron and this difference is the mechanism that is intended to indicate the direction of motion. Unfortu-

nately, the outputs of the two subunits will not fall within the same time span (due to lack of gating) except under special circumstances, namely sufficiently rapid motion. If the subunit outputs do not fall within the same time frame, as will happen with slow motion, then the net output of the detector will be dual peaks of opposite sign. Under these conditions it is likely that the inhibitory signals will be ineffective. This demonstrates that the performance of inhibitory detectors degrades ungracefully as the velocity changes. This can be observed in Figure 3.11 where the responses of two differently tuned systems to the same input are shown. The bottom response is for a system with a much smaller delay than the upper one (i.e. tuned for higher velocities). The dual peak nature of the response when the input is below the tuned velocity can be clearly seen. There are also some potential problems regarding stability of inhibitory systems when negative inhibitory signals are involved. These problems may be avoided by employing a combination of bandpass filters and ON and OFF channels, as described in [Bouzerdoum 93].

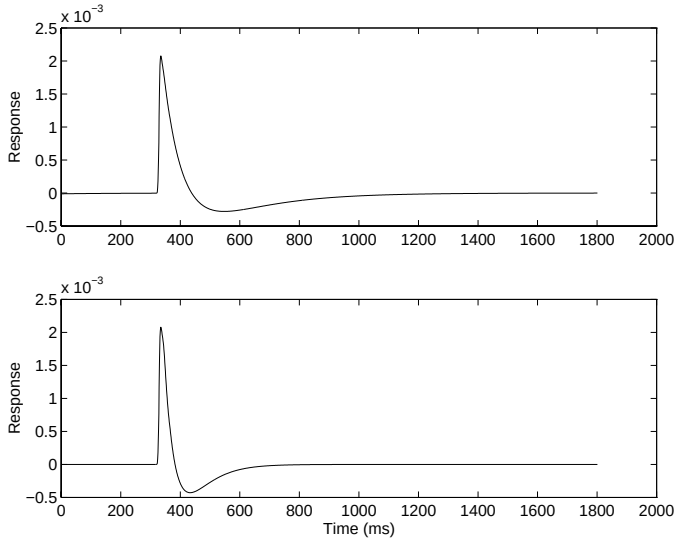


Figure 3.11: Response of differently tuned feedforward inhibitory systems to identical inputs.

3.6.3 Options for Adaptation

The addition of bandpass filters to the input of the motion detection system clearly changes the options available for adapting the system to different luminance levels. It is essential that adaptation occurs immediately after the photodetection stage so that the dynamic range requirements of signals are immediately reduced. The first layer of processing in the sys-

tems just discussed is the bandpass filter layer, so adaptation must occur there since a linear bandpass filter capable of operating over a wide dynamic range is obviously impractical. This means that the opportunity to exploit the nonlinearity inherent in some motion detection architectures as an adaptive mechanism has been lost, unless systems that do not require bandpass filters can be developed (see Section 5.2). Reducing the signal dynamic range in the bandpass filter also means that the multiplication operation may become acceptable even though it does increase the signal dynamic range. The introduction of preprocessing by bandpass filters makes the adaptation and noise performance of the system dependent on the design of the bandpass filter.

3.7 Conclusion

This chapter has investigated local motion detectors because they appear to be a useful compromise between complexity and functionality. The operating criteria for local motion detectors were defined and issues important to real sensors, dynamic range and noise performance, were discussed.

The simplest delay and compare schemes were examined and shown to be incapable of meeting the criteria necessary for useful local motion detection. Some common modifications to the basic systems that were inspired by studies of biological systems were also examined. It was found that the modifications produced useful local motion detectors, but changed the viable options for system adaptation.

Therefore there are now two important avenues of investigation related to local motion detectors — adaptive band pass filters and new local motion detectors that do not require bandpass filters.

Chapter 4

Adaptive Neurofilters

4.1 Introduction

The useful local motion detectors described in the previous chapter used bandpass filters in the preprocessing layer. These preprocessing layers must be adaptive to satisfy the dynamic range and noise requirements outlined in Sections 3.4. The first part of this chapter introduces an adaptive bandpass filter that uses shunting inhibitory neurons and mimics some of the characteristics of large monopolar cells found in insect lamina. Shunting inhibitory neurons are used because they have useful adaptive properties. The design employs both spatial and temporal adaptive mechanisms.

The second part of the chapter describes an adaptive spatial derivative element that can be used in an alternative motion detector design, and a four-quadrant multiplier element that can be used in a Reichardt motion detector. Both of these elements also use shunting inhibitory neurons.

4.2 Adaptive Bandpass Filter

Linear bandpass filters capable of operating over very wide dynamic ranges of inputs are impractical. Therefore, the use of bandpass filters in the preprocessing layers of useful motion detectors obviously means that the adaptation must occur prior to or within the bandpass filters. Adaptation prior to the bandpass filters does not exploit the potential savings that could be made by combining the necessary adaptive characteristics with the computational elements. Thus adaptive bandpass filters are important to the implementation of these architectures.

The approach to designing adaptive bandpass filters (and adaptive elements in general)

taken in this chapter involves the use of biologically inspired building blocks, in particular the shunting inhibitory neuron. The shunting inhibitory neuron is used because it implements a steady state division, is stable and can be constructed relatively cheaply in hardware [Moini et al. 97]. Steady state division is a desirable adaptive property.

The shunting inhibitory neuron possesses excitatory and inhibitory inputs. The adaptive signals are usually provided to the inhibitory inputs. The shunting inhibitory neuron is described by Equation 2.1 and the steady state solution is given by Equation 3.2.

The shunting inhibitory neuron is essentially a low pass filter whose time constant is controlled by the inhibitory input. Simple bandpass filters may be constructed by taking the difference between two low pass filters with different time constants. The structure shown in Figure 4.1 uses this principle. Each shunting inhibitory neuron is arranged in a feedback configuration. The output provides the inhibitory input through a feedback path. The time constants and dynamics of the two subunits are different because a linear delay element is included in one of the feedback paths. The usual dynamic range problems associated with linear elements can be avoided in this situation because the signal has already been compressed. The gain of the two feedback paths should be the same to ensure a steady state response of zero. This very simple structure possesses some interesting and useful adaptive properties. The response of the system to a positive step input followed by a negative one is shown in Figure 4.2.

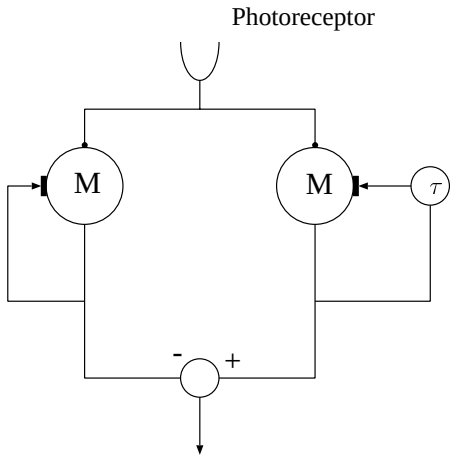


Figure 4.1: Adaptive bandpass filter. M is a shunting inhibitory neuron and τ is the time constant of a first order low pass filter acting as a delay element.

The adaptive properties of interest are the compression of dynamic range and the change in frequency response with mean luminance.

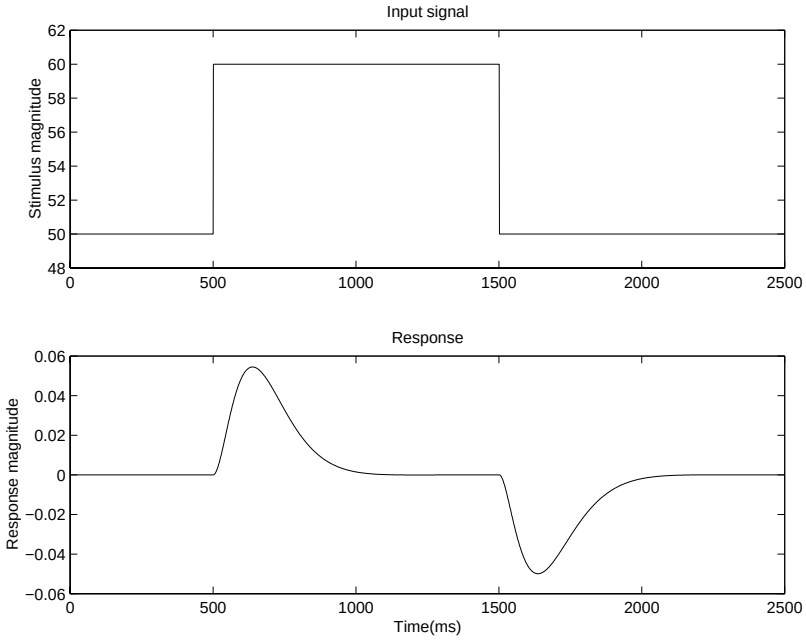


Figure 4.2: Response of adaptive filter to a step input. Neuron self decay parameter $a = 10$, and delay element $H(s) = 10/(s + 10)$.

Dynamic Range Compression

A shunting inhibitory neuron in a feedback configuration provides square root compression of the form

$$m^2 + \frac{a}{k}m - \frac{L}{k} = 0$$

$$\Rightarrow \quad m = -\frac{a}{2k} + \frac{\sqrt{(a/k)^2 + 4L/k}}{2}$$

where a is the self decay and k is the weight, as defined in Equation 2.1. This result is obtained by considering the steady state solution to Equation 2.1 (i.e. $\dot{m} = 0$) with $f(X) = m$.

The peak response of the system to a step input also varies approximately as the square root of the mean luminance, as shown in Figure 4.3(note that this is a log plot). Square-root compression is useful, despite not being as powerful as logarithmic compression. The dynamic range requirements may be reduced further by increasing the gain in the feedback paths, but the square root characteristics are retained.

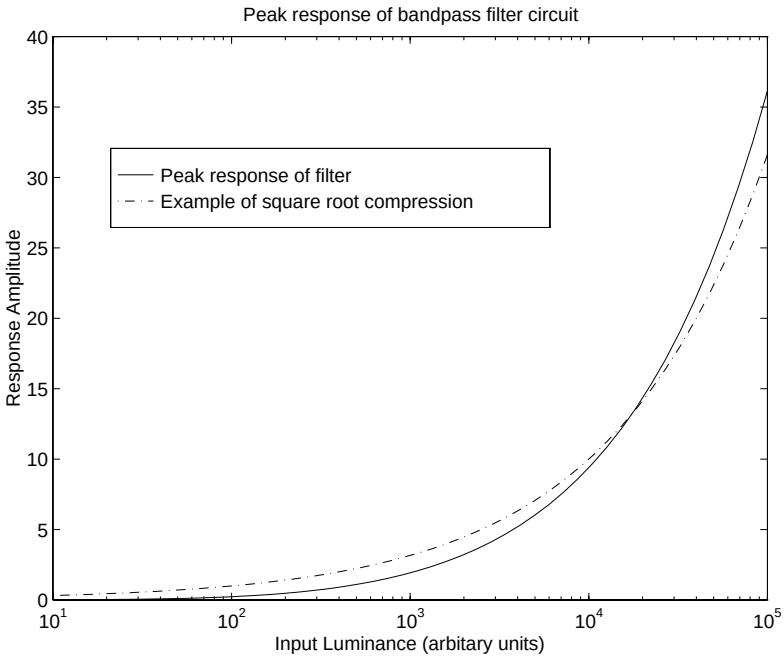


Figure 4.3: Change of peak response of bandpass unit with mean luminance. An example of square root compression is shown for comparison. Note that the x axis is logarithmic.

Frequency Response

The frequency response of the early visual processing layers in biological systems is dependent on the mean luminance because the characteristics of noise present in the systems are highly dependent on the mean luminance. Random photon and electron events are a dominant source of high frequency noise at low levels of mean luminance. Under these conditions the use of derivative operators, like edge detectors or high pass filters, would be highly error prone. Low pass filters and "object" detectors which act as integrators are more appropriate under these conditions. When the mean luminance is higher, the signal to noise ratio increases and derivative operators can function reliably. The adaptive bandpass filter behaves in this fashion. A linearised model has been developed (see Appendix B) and is described by the following equations.

$$\begin{aligned} H(s) &= \frac{sB}{(s^2 + s(X+A) + A(X+B))(s+X+B)} \\ T(s) &= \frac{A}{s+A} \\ X &= a + kz_0 \\ B &= kz_0 \end{aligned} \tag{4.1}$$

$$z_0 \quad = \quad \frac{-a}{2k} + \frac{1}{2}\sqrt{(a/k)^2 + 4L_0/k}$$

where L_0 is the mean luminance, z_0 is the mean output, $T(s)$ is the transfer function of the feedback delay filter and a and k are the neuron parameters described in Equation 2.1.

This linearisation demonstrates that the neural system is acting as a second order band-pass filter in cascade with a first order low pass filter. The characteristics of this system vary with mean luminance. The frequency responses of the system at different mean luminance values are shown in Figure 4.4; the impulse responses are shown in Figure 4.5. Both the centre frequency and bandwidth increase as luminance increases. The system has low gain and low bandwidth with a low frequency cut off at low luminance levels. The operating frequencies increase with luminance. This is appropriate behaviour because the signal to noise ratio also improves and derivative operators become more reliable. These characteristics are typical of those measured in biological visual systems [van Hateren 92, Srinivasan et al. 82]. At very low luminances the model described here remains band pass (as seen in Equation 4.2 and Figures 4.4 and 4.5), while van Hateren's work predicts low pass responses at very low luminance levels.

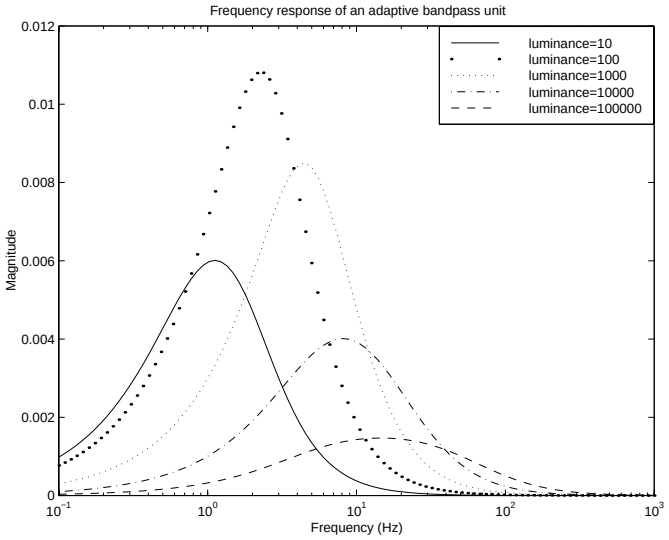


Figure 4.4: Change in frequency response of adaptive bandpass filter with mean luminance ($A = 15$, $a = 5$).

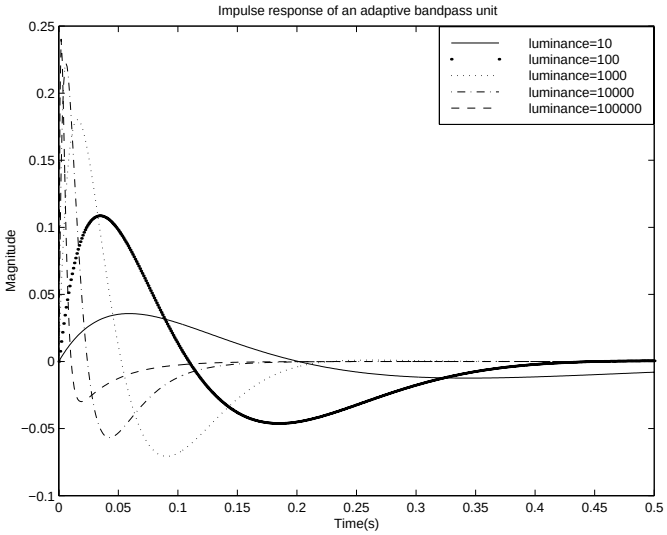


Figure 4.5: Change in impulse response of adaptive bandpass filter with mean luminance ($A = 15$, $a = 5$).

4.2.1 Using spatial adaptation

The system just described is only reducing signal dynamic range by eliminating temporal redundancy (temporal adaptation). Larger reduction of dynamic range can be obtained if spatial redundancy is eliminated as well (spatial adaptation). The adaptive structure just described may be modified to perform spatial adaptation by including inhibitory inputs from adjacent detectors (Figure 4.6). The spatial inhibitory inputs may be either feedforward or feedback. The feedforward system does provide stronger compression because the feedforward inhibitory signals are larger. This produces a structure with properties similar to the SUSTAINED unit described by Arnett [Arnett 72].

Similarities to the SUSTAINED unit in the insect lamina

Arnett explored interactions between signals applied to ON and OFF regions in the insect lamina. The results of his investigations are shown in Figure 4.7. The ON region is equivalent to the excitatory input of the adaptive bandpass filter while the OFF regions are equivalent to the spatial inhibitory regions. The responses of the model described here to equivalent stimuli are illustrated in Figure 4.8 and are similar in character to the experimental results. More complete details can be found in [Beare and Bouzerdoum 96]. Some of the differences in decay rates between the experimental and simulated responses could be due to the presence of additional modes in the real system. These modes could be caused by the photoreceptor

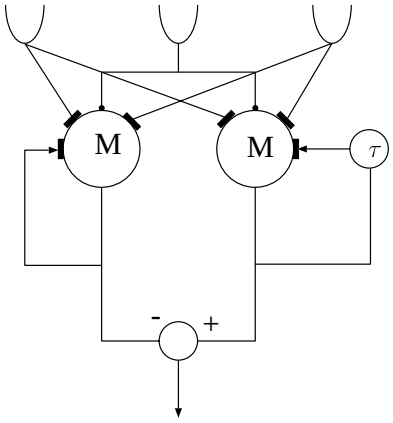


Figure 4.6: Bandpass filter with lateral and feedback shunting inhibition. τ is the time constant of a first order low pass filter.

characteristics, which are not included in the simulated responses.

Adaptive properties

The adaptive properties of the bandpass filter are also interesting. The peak response to a step input varies as shown in Figure 4.9. The signal begins to decrease at higher luminance values as the feedforward inhibitory signals begin to dominate. This is interesting but not especially desirable because important signals will be lost at high luminance levels. This behaviour can be modified by changing the activation functions of the spatial inhibitory signals. Experiments show that an approximately logarithmic relationship between step response and luminance can be achieved by using an activation function of the form $f(x) = x^{0.6}$ for the spatial inhibitory inputs. An example is shown in Figure 4.10.

Another interesting effect is the nonlinear response to different step sizes at constant levels of mean luminance. A similar nonlinearity has been observed in biological systems and has been interpreted as a form of matched amplification, with regions of higher amplification corresponding to the contrast changes that are more likely to occur in real scenes [Laughlin 87]. Responses of this form, with the activation functions described above, are shown in Figure 4.11.

4.2.2 Summary

The elements just described exhibit many of the adaptive properties that have been observed in investigations of biological systems. Such adaptive properties enable near optimum util-

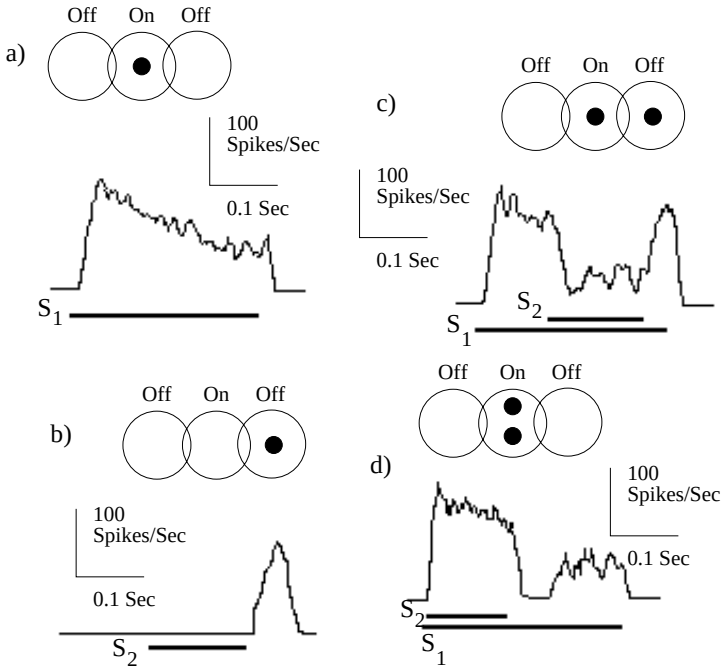


Figure 4.7: Experimental results for the SUSTAINED unit from Arnett 1972. In subfigure c spot S_2 is applied to the "off" region.

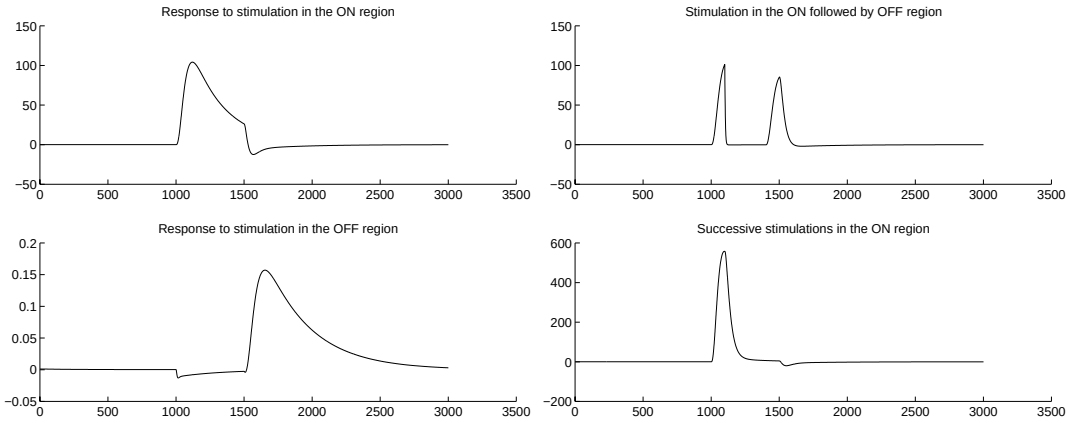


Figure 4.8: Simulated responses of the spatially adaptive bandpass filter.

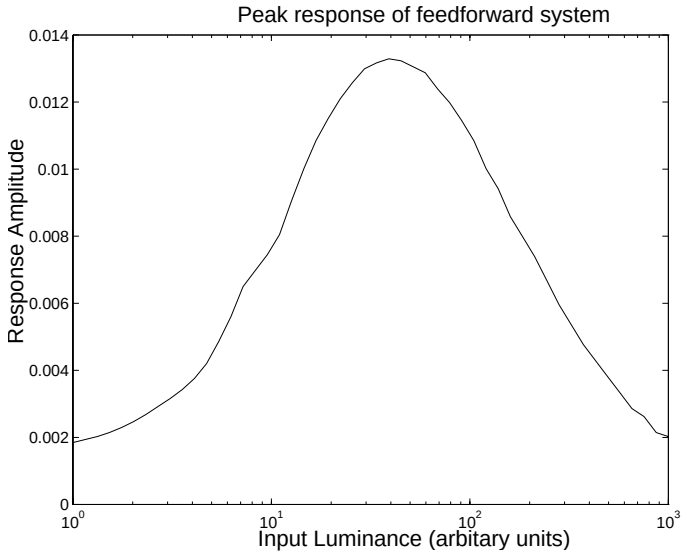


Figure 4.9: Peak response to a step input of the system with feedforward spatial adaptation.

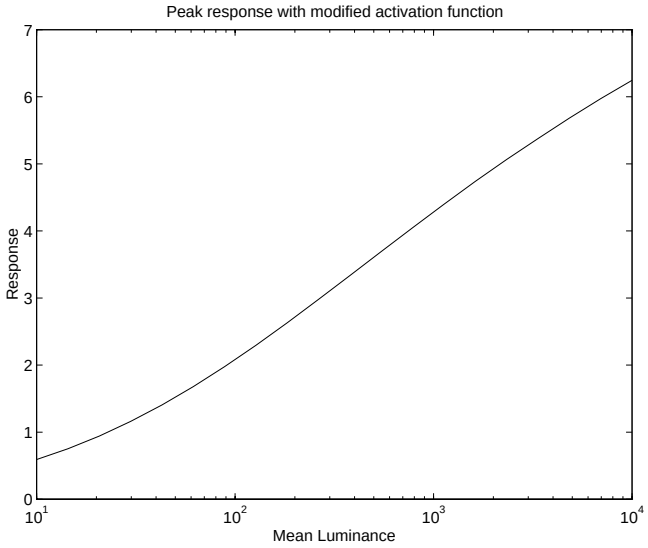


Figure 4.10: Peak response to a step input of the system with feedforward spatial adaptation and nonlinear activation functions.

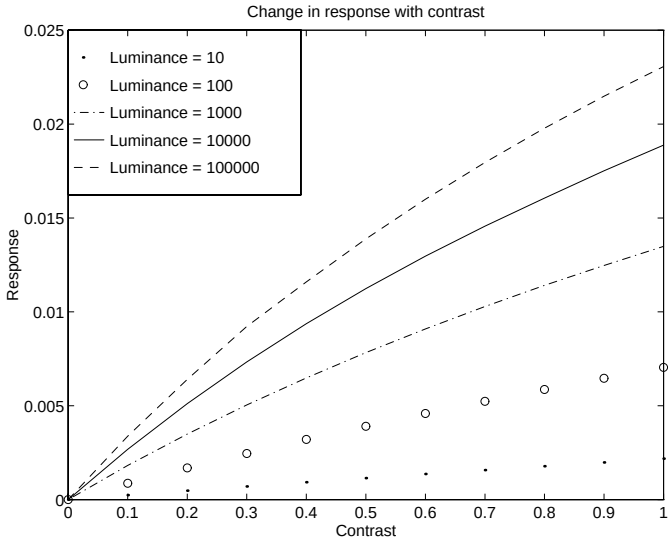


Figure 4.11: Change in response of bandpass filter with contrast.

isation of limited bandwidth channels and limited computational resources, and are vital in producing robust visual processing systems. The bandpass filters just described are ideal for use as the first layer of processing for motion detectors of the form described in the previous chapter. They may also be useful for other applications.

4.3 Spatio-temporal derivative motion detection model

The adaptive bandpass filter may also be used in an alternative type of motion detector based on spatial and temporal derivatives. It is generally believed that biological detectors do not operate using this principle because it is sensitive to changes in light intensity, but its simplicity may outweigh these problems in many circumstances.

The spatio-temporal derivative motion detector is very simple, and hardware implementations have been constructed in the past [Horiuchi and Koch 96]. Spatial and temporal derivatives are calculated at every point in the image. These could be used to determine the velocity as described by the brightness change constancy equation (BCCE) (and this is the usual approach), however the direction of motion can be determined by comparing the signs of the two derivatives. By using this information to indicate motion, rather than estimate velocity, many of the noise problems usually associated with the BCCE are eliminated. (The trade-off is that less information is recovered.) If the changes in intensity are caused by motion (rather than global intensity changes) then spatial and temporal derivatives of the same sign indicate motion in one direction while derivatives with different signs indicate motion in the

opposite direction. If global changes in light intensity occur then edges will also be detected by this process, because both derivatives will be nonzero. A multiplication operation can be used to perform the sign comparison (as seen earlier in the modified correlation detector). If the spatial and temporal derivatives are already compressed then the dynamic range of the multiplication operation should not cause serious problems. An adaptive spatial derivative operator is therefore required.

4.3.1 Adaptive spatial derivative operator

An adaptive spatial derivative circuit that is suitable for this application is very simple (see Figure 4.12) and uses a similar structure to the adaptive bandpass filter. For the purposes of motion detection the only property of the two derivatives that is essential is the sign, although the magnitude is a useful indicator of reliability. An obvious way of capturing this information in an adaptive fashion is to use a division operation. This will compress all negative spatial derivatives into the range 0 to 1 and all positive ones will be greater than 1. This compression is asymmetric, but the contrast in real scenes is usually relatively low, so the asymmetry will be generally insignificant. Shunting inhibition implements a division operation in the steady state, so it should be possible to produce an adaptive spatial derivative circuit using shunting inhibition.

The response of a single neuron to identical excitatory and inhibitory inputs (a zero spatial derivative) is dependent on the magnitude of the signals due to the presence of the neuron self decay in the denominator of the steady state solution (Equation 4.2). The presence of the self decay term also prevents division by zero. This is most significant at low luminance levels.

$$Response = \frac{L}{a + L} \tag{4.2}$$

Where L is the luminance.

This means that the response corresponding to a zero spatial derivative (zero point) must be determined before the derivative circuit can be used at low luminances. Fortunately a reference corresponding to the zero point can be easily provided by using an identical neuron with both excitatory and inhibitory inputs coming from the same receptor. The structure is shown in Figure 4.12.

The frequency response of the system changes as a function of luminance in a desirable way. The circuit is a low pass device with a time constant that decreases as luminance increases. Thus integration times are longer when the information is less likely to be reliable.

The compressive power of a division based scheme is very strong, producing a response that is only dependent on contrast when the luminance is high. At lower luminance levels the response decays gracefully (Figure 4.13). The matched amplification that was discussed in relation to the adaptive bandpass filter can also be observed in the operation of the spatial derivative circuit (Figure 4.14), and is of similar form to that observed by Laughlin [Laughlin 87].

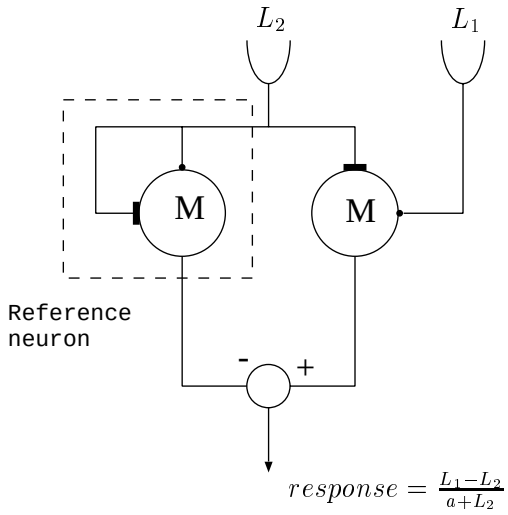


Figure 4.12: Spatial derivative neural circuit.

4.4 Neural Multipliers

The Reichardt motion detector employs a multiplication operation. A neurally plausible multiplier implementation using average neuron firing rates has been proposed by Srinivasan [Srinivasan 76]. The Reichardt motion detector with bandpass filter preprocessing and the spatio-temporal derivative motion detector use the multiplier as a gating operator. Four-quadrant operation is required to perform sign correction if separate ON and OFF channels are not used.

4.4.1 Four-quadrant multiplier using shunting inhibition

The gating and sign correction properties may be implemented using a simple circuit based on shunting inhibition. The circuits shown in Figure 4.15 perform these functions. The steady state responses of the two circuits are given by

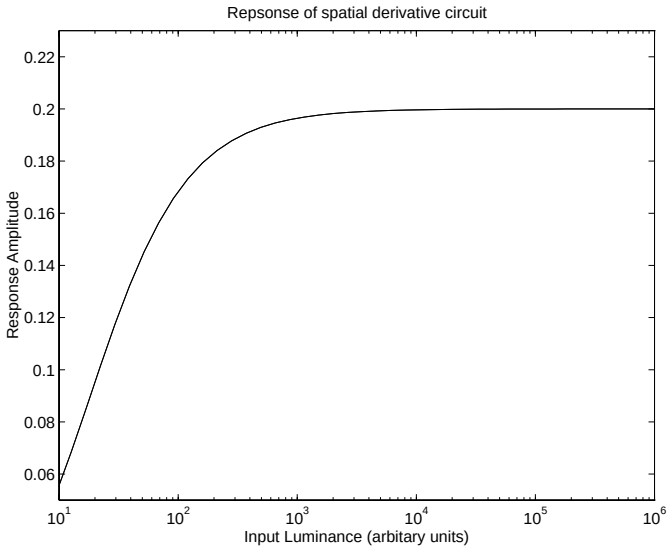


Figure 4.13: Response of spatial derivative circuit to a stationary edge located between the two receptors. Contrast $c = 0.2$.

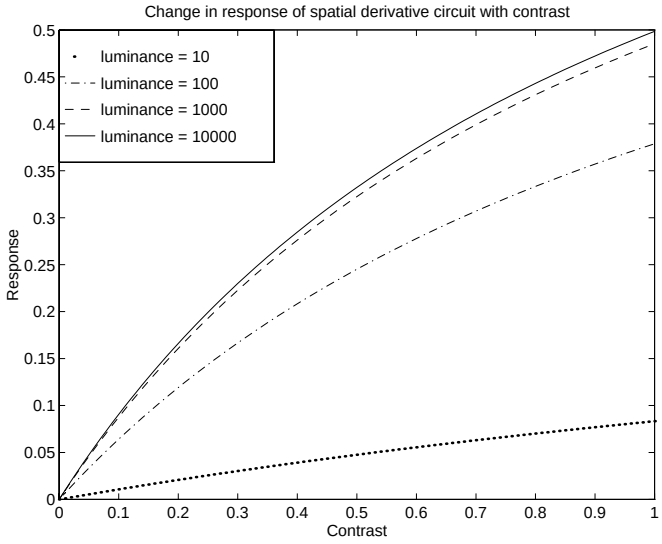


Figure 4.14: Contrast response of spatial derivative circuit.

$$\text{Response} = \frac{x}{a} - \frac{x}{a+y} \tag{4.3}$$

for Figure 4.15(a)

$$\text{Response} = \frac{x}{a+x} - \frac{x}{a+x+y} \tag{4.4}$$

for Figure 4.15(b)

The desired functionality is produced over a limited range of input values. Care must be taken to ensure that the denominators do not approach zero, i.e. $a + y \gg 0$, $a + x \gg 0$ and $a + x + y \gg 0$. This problem may occur when inputs are negative. It can be eliminated by using large values of a, which imply rapid responses, and using appropriate adaptive components to provide the input. The steady state responses of the circuits are shown in Figure 4.16. The gating properties and sign correction are clearly visible.

The multiplicative effect is observable under a limited range of conditions. If $y/a \ll 1$ then Equation 4.3 is approximated by

$$\text{Response} = \frac{xy}{a^2} \tag{4.5}$$

If $y/(a + x) \ll 1$ then Equation 4.4 is approximated by

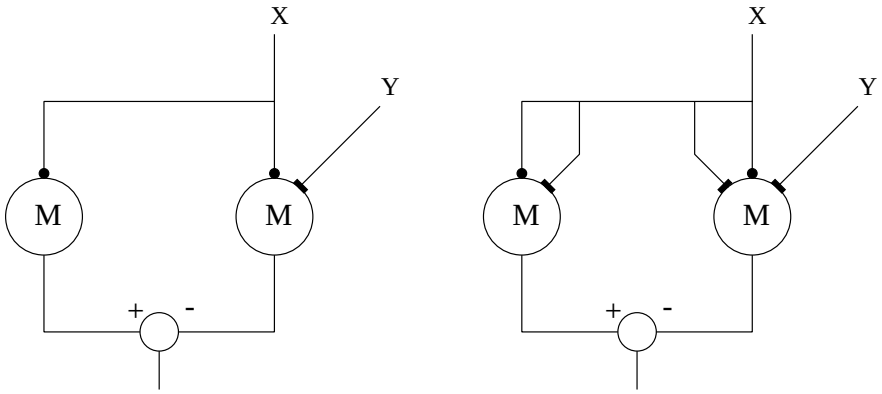
$$\text{Response} = \frac{xy}{(a+x)^2} \tag{4.6}$$

4.5 Conclusion

This chapter has introduced several adaptive elements that are potentially useful in the early layers of visual sensors — adaptive temporal bandpass filters and adaptive spatial derivative elements. Both can be used as components of motion detection systems. These elements have adaptive properties that are similar to those observed in biological visual cells. A four-quadrant neural multiplier circuit was also described. This circuit has gating properties that make it a useful component in some motion detection systems.

All of these elements used shunting inhibitory neurons as the basic building block. It is interesting to note that it is now possible to build several different types of motion detectors using only shunting inhibitory neurons and delay elements.

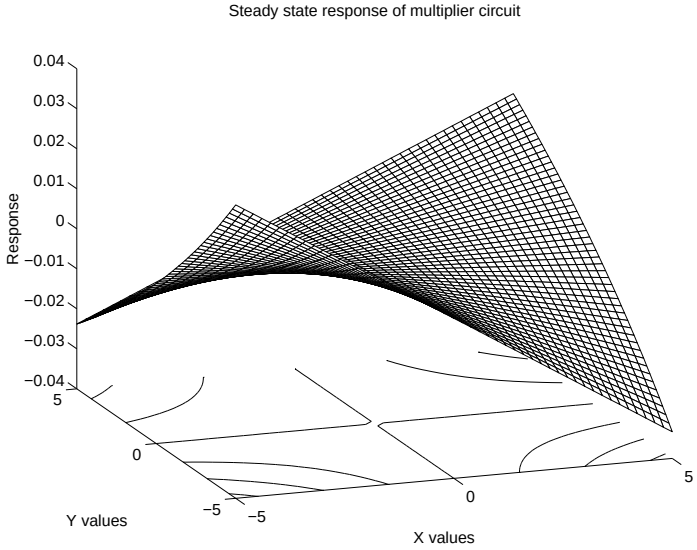
However, using the kinds of adaptive elements described in this chapter in the preprocessing layers means that it is no longer possible to use the nonlinear interaction responsible



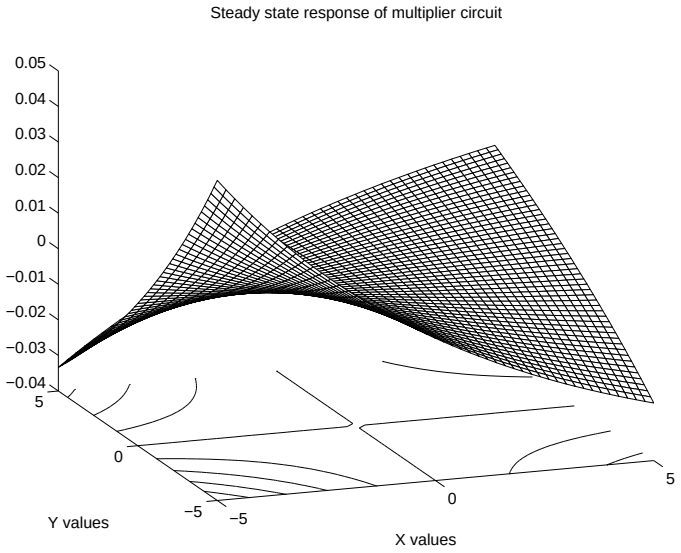
(a) (b)

Figure 4.15: Neural multiplier circuits.

for motion detection as the dominant adaptive mechanism. If an alternative local motion detector that does not require preprocessing can be designed then it may be possible to use the interaction as an adaptive mechanism. This could result in a simpler system. In the next chapter we will introduce a new motion detection architecture that has this property.



(a) Response of circuit 4.15(a).



(b) Response of circuit 4.15(b).

Figure 4.16: Steady state responses of the multiplier circuits.

Chapter 5

The directionally sensitive local inhibitory motion detector

5.1 Introduction

This chapter describes a new local motion detector that does not use a preprocessing layer. The new motion detector is called a *directionally sensitve local inhibitory motion detector* (DSLIMD), and employs a delay and compare architecture. Shunting inhibitory neurons are used as the basic building blocks. The system meets the requirements for a useful motion detector listed in Section 3.3.1.

The DSLIMD is also tested in a wide field configuration. Many of the well known wide field characteristics of biological motion detection neurons are also displayed by an array of DSLIMDs.

5.2 The directionally sensitive local inhibitory motion detector

In the simplest Reichardt and shunting inhibitory motion detectors the nonlinear interactions are vital to the detection of motion. However, these systems operate correctly only in a widefield or time averaged mode. The multiplicative interaction has the advantage of being symmetrical, but has the serious disadvantage of increasing the dynamic range of the signal. The symmetrical interaction means that the basic detector produces no output to a stationary edge. The inhibitory interaction has the opposite problem — desirable adaptive properties but asymmetric operation resulting in a non zero response to a stationary stimulus. The ideal delay and compare architecture would be capable of using an adaptive nonlinear interaction to detect local motion. This section presents DSLIMD scheme that achieves this goal.

The basic mechanism used to achieve this is to provide the excitatory and inhibitory inputs to pairs of neurons from the same sources (Figure 5.1). Asymmetry is necessary, and is provided in the conventional way by a delay element. The delay element has unity gain so that the steady state inputs to each neuron will be the same, meaning that the steady state response of the subunit to a stationary stimulus will be zero. This eliminates one problem with the original inhibitory architecture.

However, as can be seen in Figure 5.2 the sign of the response is dependent on the sign of the contrast change. The operation of the subunit is easy to understand. If a change is experienced by the receptor on the left in Figure 5.1, then it is transmitted to the excitatory inputs of both neurons simultaneously. This causes identical changes to both neurons and therefore the difference between the two outputs will remain zero. If the receptor on the right experiences a change, then the delay will cause the neuron on the left hand side to experience the change after the neuron on the right. This means that the output of the right hand neuron drops, while the left hand neuron output remains constant until the change is transmitted through the delay. This produces a positive response. The response will return to zero when the delay element reaches steady state.

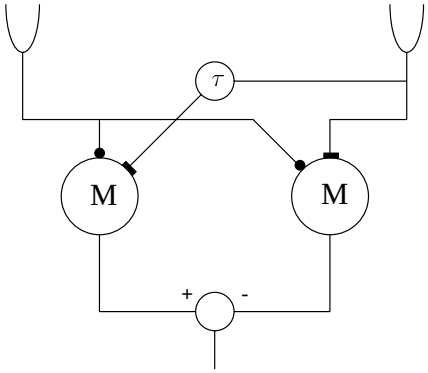
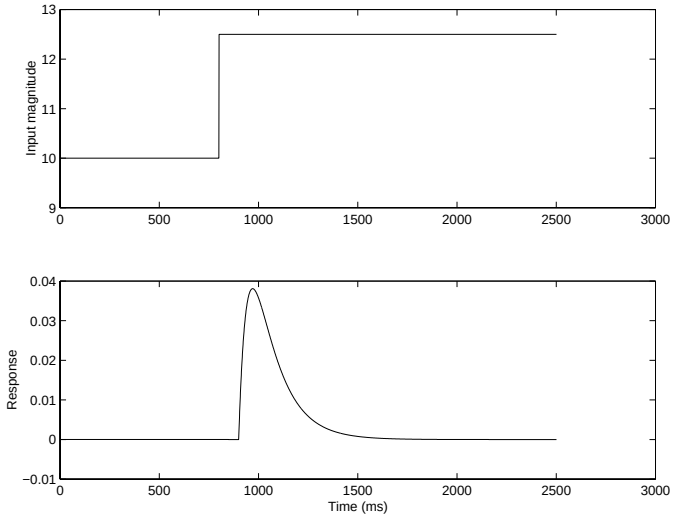
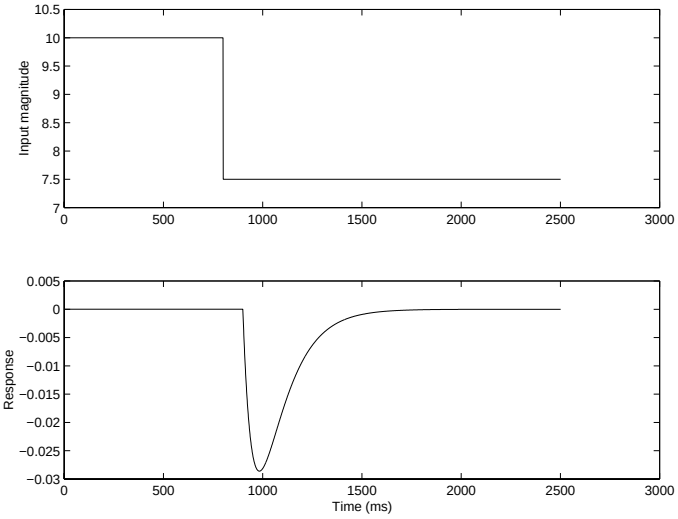


Figure 5.1: Symmetric inhibitory subsystem. M is a shunting inhibitory neuron and τ is the time constant of a first order low pass filter.

Obviously the system is not yet capable of performing local motion detection independent of the sign of the contrast. However, the extension shown in Figure 5.3 corrects this. A local motion detector is created by using a mirror image array of subunits and combining the results as shown in Figure 5.3. This detector is using the nonlinearity of the neuron to correctly determine the direction of motion without widefield summation or infinite time averaging. The system now uses three receptors to provide inputs for a complete local motion detector. The centre receptor provides all of the inhibitory inputs associated with a particular



(a) Response of structure shown in Figure 5.1 to a positive edge moving from left to right.



(b) Response of structure shown in Figure 5.1 to a negative edge moving from left to right.

Figure 5.2: Responses of symmetrical inhibitory subsystem (Figure 5.1) to a moving edge.

local motion detector (an individual motion detector is contained within the dashed line in Figure 5.3). The excitatory inputs are supplied by the outer receptors. (Note that only one delay unit per receptor is required. Multiple delays are illustrated for clarity.)

Consider an edge moving from left to right. If the edge is positive (increasing contrast) then the left subunit will experience an increase in excitatory inputs when the edge reaches the leftmost receptor. The change in excitory inputs occurs at the same time and the inhibitory inputs remain the same so the net output from the left subunit remains zero. As the edge reaches the central receptor, the inhibitory input to the right neuron in the left subunit is increased. The inhibitory input to the left neuron increases less rapidly. The output of the right neuron will therefore be decreased relative to the left neuron resulting in a positive response from the left subunit. At the same time the inhibitory inputs to the right subunit are also changing. The inhibitory input to the left neuron of the right subunit is larger than that to the left, resulting in a net positive output. The activity of the neurons in the left subunit is higher than those in the right subunit because the excitatory inputs are higher. If the motion detector response is given by $left - right$, a net positive response is produced.

If the edge is negative (decreasing contrast) then the excitatory inputs to the left subunit experience a decrease in excitatory inputs followed by a decrease in inhibitory inputs. The inhibitory input for the right neuron in the left subunit will decrease more rapidly making the output of the right neuron larger and producing a net negative output from the subunit. A similar sequence of events occurs to produce a net negative response from the right subunit. In this case the activity of the neurons in the right subunit is highest so the absolute magnitude of the right subunit is greater than the left (although both are negative). Therefore the net response of the motion detector is positive and therefore independent of the change in contrast. If the direction of travel is reversed then the response becomes negative.

This may be summarised as follows:

If the dynamic properties of the neuron are ignored (i.e. the neuron is treated as a division operator, rather than an adaptive filter) then the response of the left subunit then the edge reaches the centre receptor is given by

$$left = \frac{L_2}{a + L_1} - \frac{L_2}{a + L_2} \tag{5.1}$$

while the response of the right subunit is given by

$$right = \frac{L_1}{a + L_1} - \frac{L_1}{a + L_2} \tag{5.2}$$

where L_1 is the background luminance, L_2 is the edge luminance, and a is the internal delay.

If $L_1 > L_2$ then Equation 5.1 is positive. Equation 5.2 is also positive, but smaller. This produces a net positive response from the motion detector.

If $L_1 < L_2$ then Equation 5.1 is negative. Equation 5.2 is also negative, but has a larger absolute magnitude. This also produces a net positive response from the detector.

Therefore the sign of the response is independent of the contrast of the edge. If the direction of motion is reversed then the responses become negative.

The results for edges of opposite signs moving from left to right are shown in Figure 5.4. The response to edges moving in the opposite direction are shown in Figure 5.5.

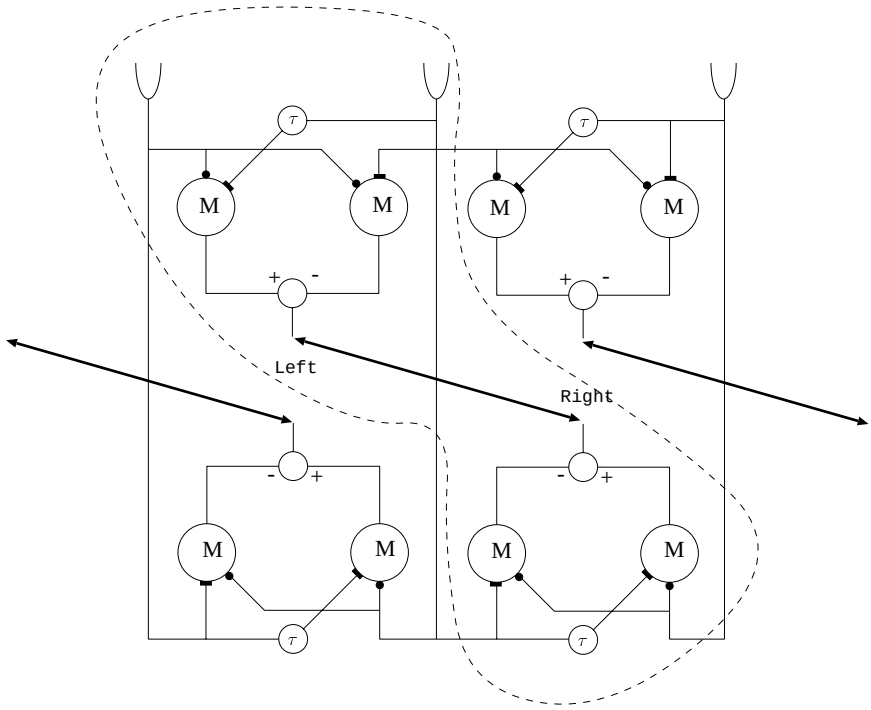
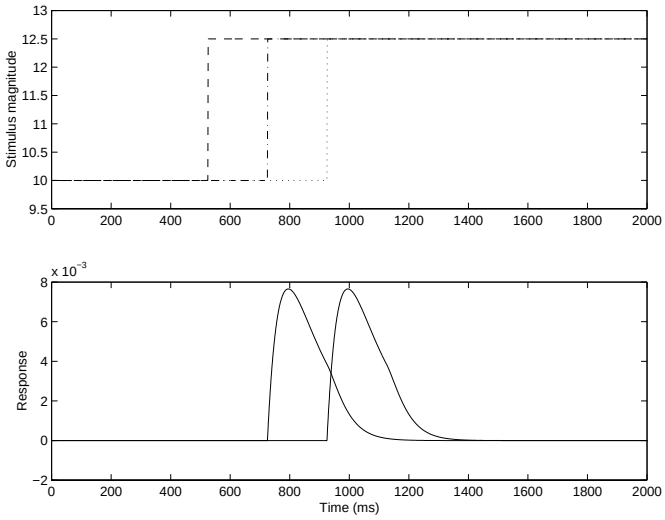


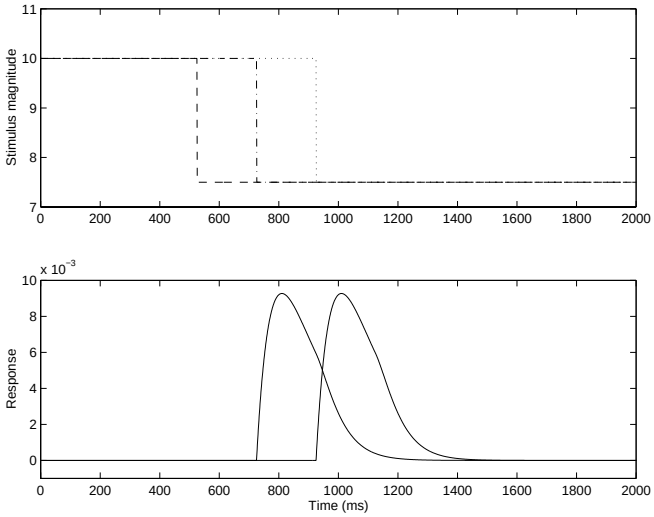
Figure 5.3: DSLIMD architecture. A single detector is indicated by the dashed line. M indicates a shunting inhibitory neuron and τ is the time constant of a first order low pass filter.

Adaptive properties

The adaptive properties of the system are also very interesting. The change in peak response with mean luminance is shown in Figure 5.6. The fact that the response reaches a peak at high luminance levels indicates that the detector is performing very useful compression of dynamic range when it is most important. At this point it should be noted that the system does have a potentially significant problem — a linear delay element capable of operating

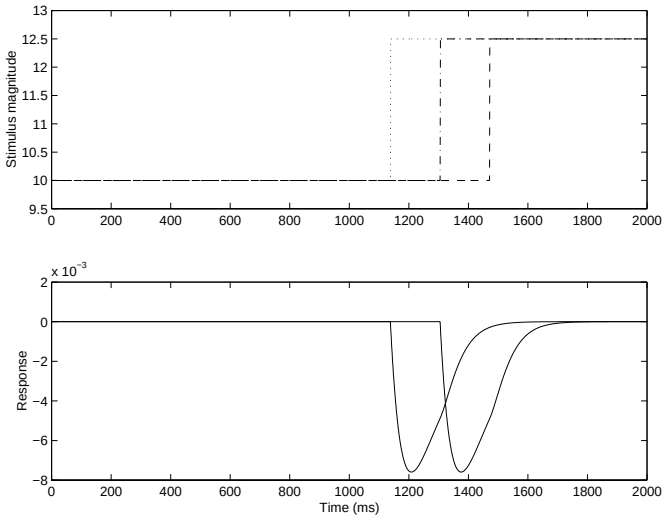


(a) Response of DSLIMD to positive edge.

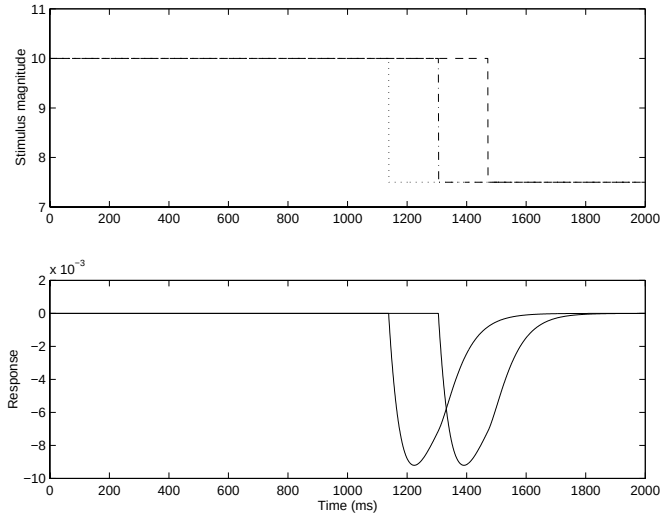


(b) Response of DSLIMD to negative edge.

Figure 5.4: DSLIMD responses to rightward motion. Responses of two adjacent detectors are illustrated. Neuron self decay $a = 10$ and delay filter $H(s) = 8/(s + 8)$.



(a) Response of DSLIMD to positive edge.



(b) Response of DSLIMD to negative edge.

Figure 5.5: DSLIMD responses to right to left motion; neuron self decay $a = 10$ and delay filter $H(s) = 8/(s + 8)$.

over a wide dynamic range is necessary. (Note that this problem also applies to the widefield detectors.)

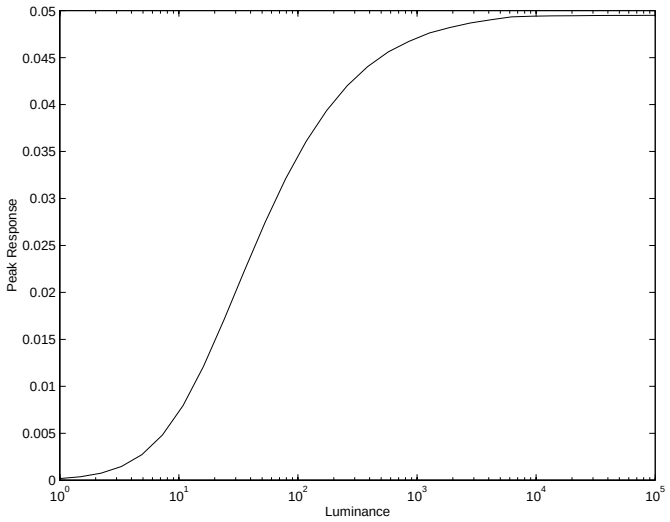


Figure 5.6: Change in peak response to moving edge with mean luminance; velocity $=$ 10 receptors/second.

The peak output to a moving edge when the delay element is more significant than the neural delay is given by

$$\text{Peak Response} = \frac{L^2(1-c)^2}{(a+cL)(a+L)}$$

The change in peak response with edge velocity is also as expected, with a decay in magnitude experienced as velocity increases (Figure 5.7). The point at which the decay begins also increases with mean luminance.

One effect that is not observed for the new architecture is the matched gain effect; the increase in response of a DSLIMD with edge contrast is essentially linear.

Noise performance

The noise performance characteristics of the DSLIMD have not been explored in detail. The results of a preliminary test are shown in Figure 5.8 and appear to be promising. The DSLIMD structure is quite similar to the feedforward inhibitory structure so the noise characteristics should also be quite similar.

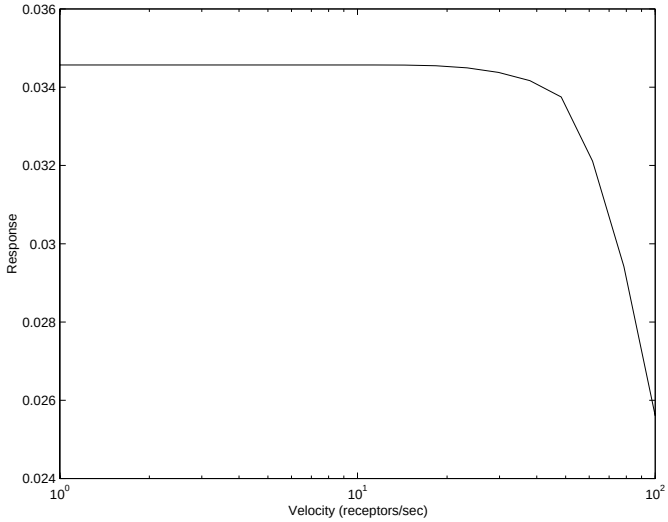


Figure 5.7: Change in peak response with edge velocity; luminance $= 100$.

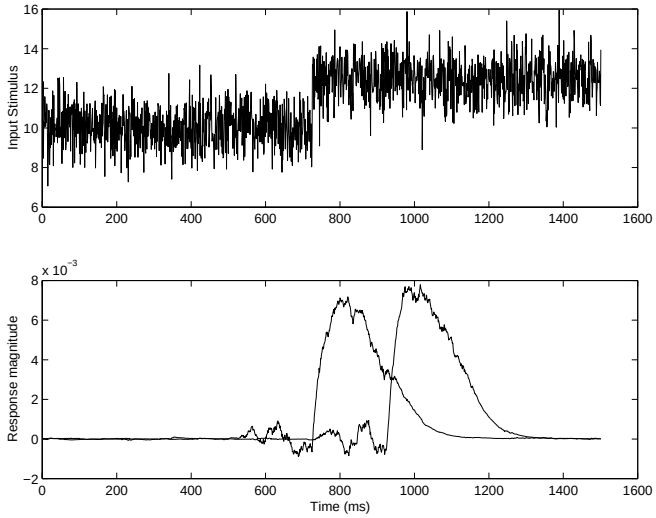


Figure 5.8: Response of a DSLIMD to a noisy moving edge; neuron self decay $a = 10$ and delay filter $H(s) = 8/(s + 8)$.

5.2.1 Steady state implementation

The DSLIMD architecture introduced here can be implemented using a simplified neuron. Using a neuron that only implements the steady state solution of the shunting inhibition equation (i.e. the neuron is a division operator rather than an adaptive low pass filter.) still provides an operational local motion detector. Most of the useful adaptive properties are maintained and the neuron is simpler to implement in silicon. (A steady state version of the neuron is simpler because it does not attempt to mimic any dynamic characteristics. The low pass filtering properties of the neuron are therefore discarded.) Obviously the delay elements, indicated by τ in the figures, are essential.

Properties that are dependent on the internal time constant will not be observed, however these tend to be less important in real scenes. The response is likely to be more susceptible to noise at low luminance levels because the integration properties of the neuron have been removed. The simulations shown in Chapter 9 all use a steady state version of the motion detector layer to save computation time. A sample response of the steady state version is shown in Figure 5.9 and a simple noise test is shown in Figure 5.10. The reduction in performance compared to Figure 5.8 is clearly visible.

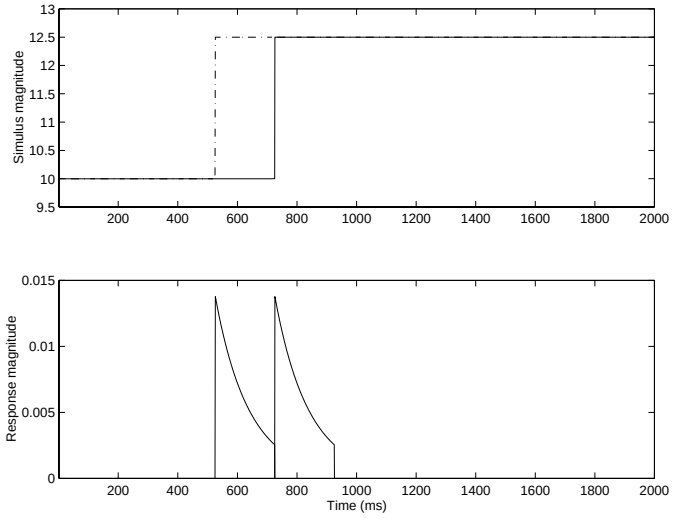


Figure 5.9: Response of the steady state version of the DSLIMD to a moving edge. Neuron self decay $a = 10$ and delay filter $H(s) = 8/(s + 8)$.

The steady state implementation does have some flaws that may be serious in some rare circumstances. When the stimulus consists of a moving line that stimulates only one receptor (a situation that corresponds to a very narrow stimulus and receptive fields that are narrower

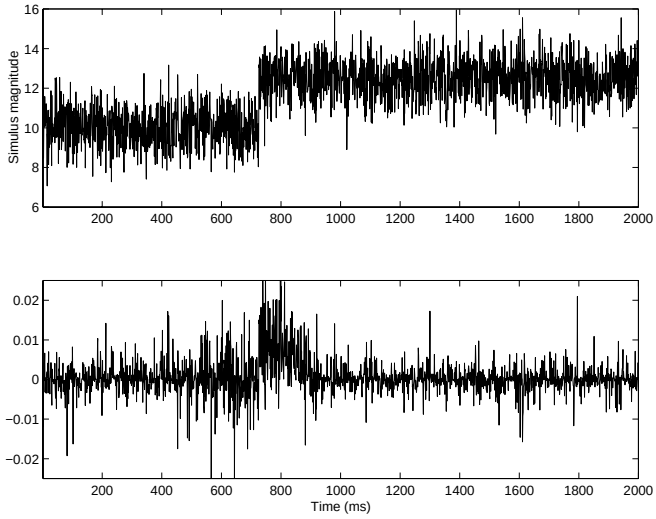


Figure 5.10: Noise response of a steady state DSLIMD. Neuron self decay $a = 10$ and delay filter $H(s) = 8/(s + 8)$.

than the receptor spacing) then the response of the system becomes dependent upon the contrast. This happens because the lack of internal delay in the simplified neuron means that the response value is not stored, so the information is not present for the nonlinearity to code the direction of the change in contrast. The history of the motion of the object is not stored by the simplified neurons.

This effect has not been observed in real scenes because this stimulus is not realistic. Optical blurring makes single receptor stimuli very unlikely and when the stimulus overlaps multiple receptors it is unnecessary for the neurons to maintain any past information because the information is already available.

5.2.2 Reverse Phi Stimulus

The reverse phi stimulus consists of a moving bar stimulating a single pixel that reverses contrast between photo detectors (for example a line initially brighter than the background will become darker than the background). Humans and insects perceive this as motion in the opposite direction [Anstis 80]. The prediction of this effect in insects was one of the Reichardt detectors important successes (see Section 2.4.1). Systems using separate ON and OFF channels will often indicate motion in the correct direction.

The DSLIMD produces a reversed response to a reverse phi stimulus. Consider the reverse phi stimulus moving from left to right across a pair of photo receptors. The first recep-

tor experiences an increase in intensity, which disappears as the second receptor experiences a decreasing intensity. The two neurons in the left subunit are partially charged by the increasing pulse and begin to discharge as the pulse disappears. As the pulse reaches the second receptor, the inhibitory input of the right hand neuron in the left subunit is decreased, causing that neuron to discharge less rapidly and resulting in a net negative response from the left subunit.

The same decrease in inhibitory input of the left neuron in the right subunit produces a net negative response from the right subunit. The peak response of the left subunit will be greater (more negative) because of the higher excitatory input. If the stimulus began with a decrease in intensity for the left receptor then the neurons in the left subunit would begin to discharge. As the stimulus moves the neurons begin to recharge, with the left neuron recharging less quickly because of a larger inhibitory input. The net response would therefore be positive. The right subunit will also produce a net positive output because the left neuron experiences an increase in inhibitory input before the right neuron. In this case the right subunit will have a larger response (more positive) due to the higher excitatory inputs. In both cases the net motion detector response will have the same sign. In this example the response is given by $left - right$, producing a negative response.

The second part of the response shown in Figure 5.11 occurs when the stimulus moves to the third receptor. Both excitatory inputs of the right subunit experience the change. In the first case this is an increase in intensity, resulting in both neurons charging. The left neuron charges less rapidly because its inhibitory input is now at the background level while the other inhibitory input is lower. This results in a net positive response. The left subunit experiences an increase in inhibitory input to the right neuron, discharging it slightly and producing a smaller net positive response. The net motion detector response will also be negative. If the third receptor experiences a decrease in intensity then a similar sequence of events occurs because the inhibitory input to the right hand neuron is now larger.

The reverse phi stimulus mimica an aliasing situation by introducing a temporal phase difference of greater than $180\,\mathrm{deg}$ between adjacent receptors. An "incorrect" response is therefore unsurprising.

The response to a reverse phi stimulus is shown in Figure 5.11. The response to this form of stimulus is reversed because the relative magnitude of inhibitory inputs is reversed when compared to the normal type of stimulus. The reverse phi stimulus is completely artificial so producing the "incorrect" response is not a concern.

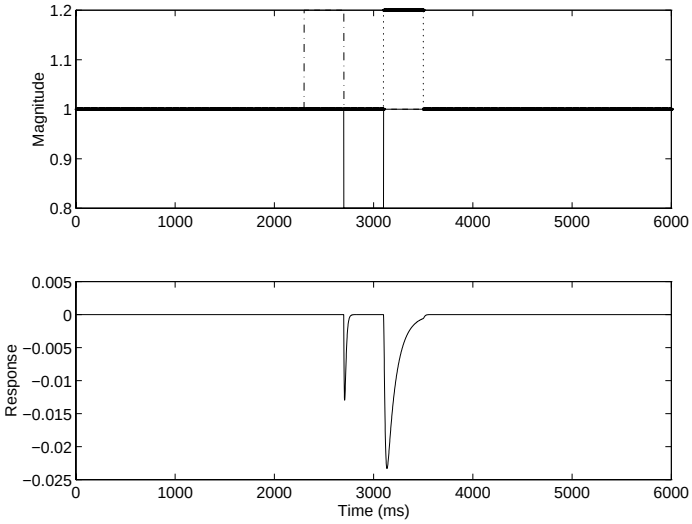


Figure 5.11: Reverse phi response. The upper graph shows the stimulus to 3 adjacent receptors, illustrating the change in contrast. The motion is from left to right.

5.3 Wide field behaviour

The DSLIMD system was developed to explore the use of adaptive mechanisms in local motion detectors. It utilises the nonlinearity inherent in shunting inhibitory neurons to produce the desired forms of responses to moving edges. Having designed a useful local detector it is also interesting to investigate the wide-field properties to determine whether they are similar to those of the traditional models. An array of local motion detectors (ADSLIMD) can be used to implement a wide-field detector simply by summing all of the outputs from individual DSLIMDs.

A number of tests are commonly used to characterise the wide-field behaviour of motion detection systems. These typically include the transient and steady state responses to drifting gratings and the way in which these responses change with luminance. A drifting grating stimulus is described by the following equation.

$$L(s,t) = L_0 + cL_0 \cos\left(\mu\omega_s s + \omega_t t + \phi\right) \tag{5.3}$$

where $\mu = \pm 1$ indicates the direction of motion. The steady state response of the wide-field DSLIMD can be computed in a similar fashion to that described by Bouzerdoum for the basic shunting inhibitory architecture [Bouzerdoum 93]. The result is

$$M = c^2 \frac{G_\alpha}{\alpha} \beta \sin(\phi)(\sin(\gamma) - A_\omega \sin\left(\gamma - \theta_\omega\right)) \tag{5.4}$$

where G_α and γ are the gain and phase changes at frequency ω through a linear filter of the form $H(s) = \frac{1}{s+\alpha}$, $\alpha = a + bf(L_0)$, $\beta = bf'(L_0)$, $\phi = \omega_s \Delta S$ is the phase difference between adjacent channels due to the receptor spacing ΔS, and A_ω and θ_ω are the gain and phase changes at frequency ω through the linear delay filter of the form $H(s) = \frac{A}{s+A}$. The activation function is linear ($f(e) = e$).

The $\sin(\phi)$ term shows a dependence on the array sampling interval. This is typical of wide-field motion detection models, and closely models the experimental results.

5.3.1 Drifting grating tests

Some transient responses to drifting gratings are shown in Figure 5.12. In these tests a grating is held stationary for several seconds and then begins moving. Some transient oscillations are observed before a steady state is reached. These are similar to the observations and predictions made by Egelhaaf and Borst [Egelhaaf and Borst 89].

The dependence of the peak of the transient response and the steady state response on contrast frequency and luminance are shown in Figures 5.13 and 5.14. The contrast frequency at which the maximum transient response occurs increases with mean luminance. There is also a much smaller increase in the contrast frequency of the maximum mean response with mean luminance.

Figure 5.15 illustrates the saturation effects observed as luminance increases and the contrast and contrast frequency remain constant. These effects have all been observed in biological systems [Eckert 80].

5.4 Conclusion

This chapter has described a new local motion detection system, known as the directionally sensitive local inhibitory motion detector, or DSLIMD. The DSLIMD employs a delay and compare structure and does not require the usual temporal bandpass filter preprocessing layer. This means that the DSLIMD can act as the first adaptive layer in a visual system because the shunting inhibitory neurons, which form the basic building block of the detector, have very useful adaptive properties. The DSLIMD therefore meets all of the requirements for a useful local motion detector described in Section 3.3.1.

A simplified version of the detector can also be built using steady state versions of the shunting inhibitory neuron.

Arrays of DSLIMDs exhibit many of the characteristics that have been observed during

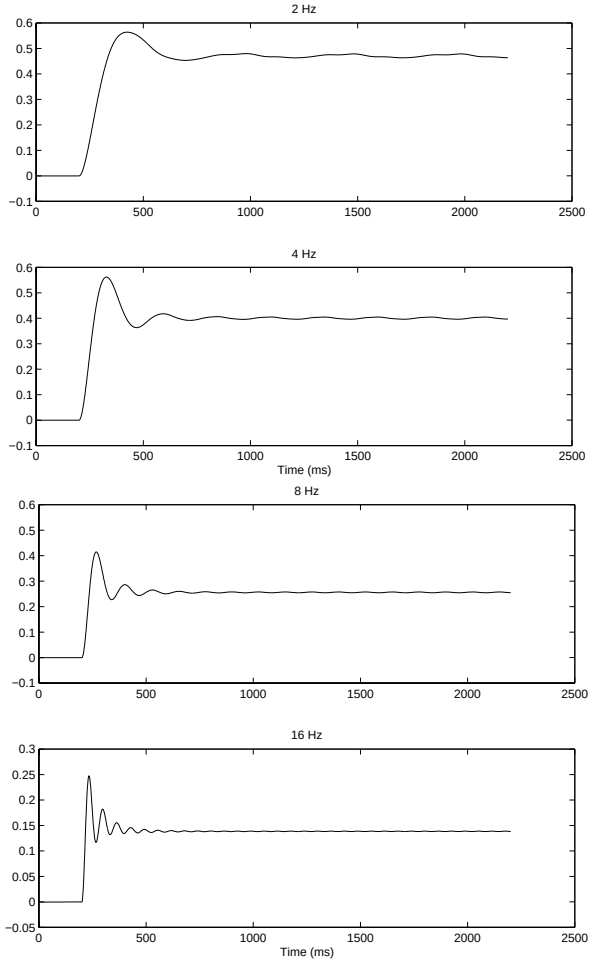


Figure 5.12: Transient responses of ADSLIMD to drifting grating at several different temporal frequencies. Neuron self decay $a = 10$ and delay filter $H(s) = 8/(s+8)$, $arraysize = 20$ and contrast $c = 0.4$.

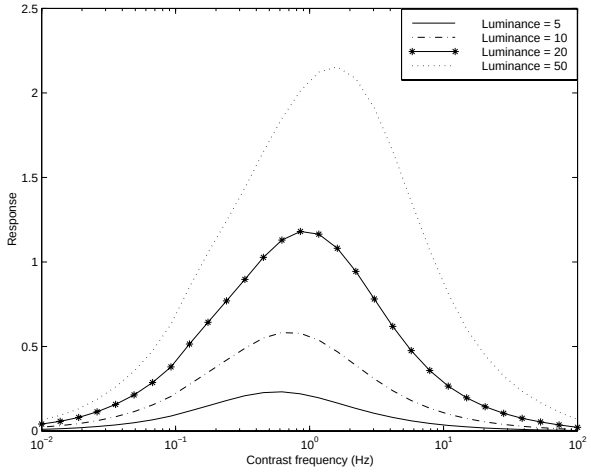


Figure 5.13: Transient peak amplitude. Contrast $= 0.5$, $f_s = 0.25$ cycles/receptor.

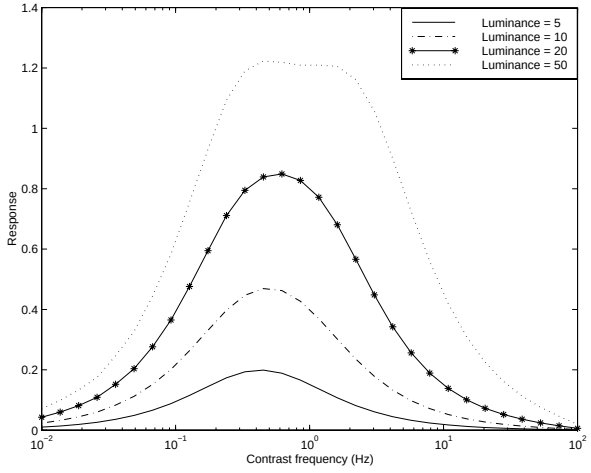


Figure 5.14: Mean steady state response. Contrast $= 0.5$, $f_s = 0.25$ cycles/receptor.

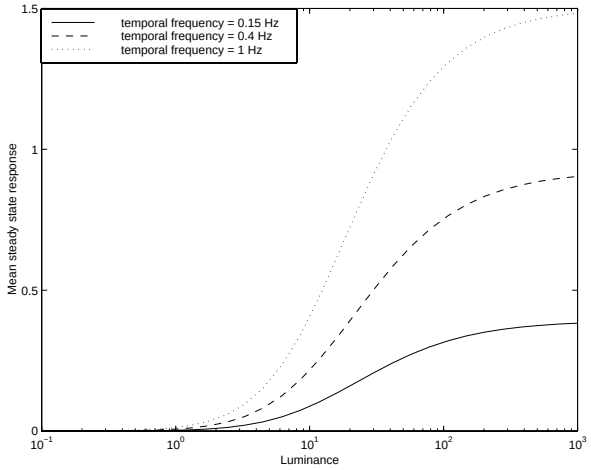


Figure 5.15: Steady state response as a function of mean luminance. $f_s = 0.25$, Contrast $= 0.4$.

widefield tests of insect visual systems. These characteristics include the transient response to drifting gratings and the adaptation of responses to mean luminance changes. It is interesting to see that a detector intended for local operation can explain these wide field results.

Chapter 6

Velocity Estimation and Segmentation

6.1 Introduction

The aim of the second part of this thesis is to explore ways of processing the output of local motion detectors. The two problems of interest are velocity estimation and object segmentation. Traditionally these two tasks have been treated as different problems. Velocity estimation is usually regarded as a low level process which provides information for segmenting the image sequence in some way. Thus, velocity estimation is often viewed as a vital function of early visual processing.

Unfortunately, extracting velocity information in a reliable and scene independent way is not easy. Many velocity estimation techniques have been developed, however all exploit some assumptions that limit the types of environment in which these techniques operate well. This is not necessarily a serious problem. It is unlikely that any single scheme can operate well in all environments and that the most sensible way to produce a reliable velocity estimation system would probably involve combining the results of several different kinds of schemes.

Image segmentation is a critical process in machine vision and is often referred to as figure-background segmentation. Segmentation of a scene into component objects permits relatively simple forms of representation for higher level processing stages, hence reducing the communication bandwidth and storage requirements. It is also regarded as an essential preprocessing step for tasks such as object tracking, recognition and general scene understanding.

This chapter reviews a selection of different velocity estimation schemes and discusses their implicit assumptions. General segmentation principles will then be discussed. The problems of velocity estimation and segmentation will then be reformulated in a fashion

more suitable to the motion detection environment of interest to this thesis. The descriptions of existing techniques presented in this chapter will not be in great detail because the reformulation leads to an approach that is significantly different to previous works.

6.2 Estimating Velocity Flow Fields

The optical flow field is the starting point for almost all previous works on image sequence analysis. Ideally the flow field is the projection of the three-dimensional scene velocities onto the image plane. In some situations it is not theoretically possible to determine the velocity of an object in an image sequence; for example, a featureless spinning sphere under constant illumination will not produce any visible indication of its motion. In practice no scheme can accurately estimate the velocity in all circumstances, and the change in accuracy within a scene can cause problems.

This section will discuss two broad classes of schemes commonly used to estimate the flow field. The solutions to common problems with the schemes will also be discussed. The biologically inspired schemes that were discussed earlier will not be explored further.

6.2.1 Gradient and Texture Schemes

Horn and Schunck's gradient scheme

Optical flow is the term now used to describe the velocity field estimated using any image based technique. However, the term was originally applied to fields generated using the formulation provided by Horn and Schunck [Horn and Schunck 81]. The brightness change constancy equation (BCCE) proposed by Horn and Schunck relates the image velocity, under constant (or slowly changing) lighting conditions, to the spatial and temporal gradients. The BCCE is expressed as

$$-\frac{\partial f}{\partial t} = \nabla f \cdot \mathbf{v}$$

where ∇f is the spatial gradient of the image brightness and $\mathbf{v}$ is the velocity vector.

The BCCE is only capable of determining the component of velocity in the direction of the brightness gradient. This restriction is related to the aperture problem that will be discussed in more detail later.

A possibly more serious problem with this approach is that results are only accurate in regions where the image gradient is high. In other areas the results are likely to be dominated by noise. This problem has been addressed in a variety of ways. For example some form of

smoothness constraint can be applied to the field [Nagel and Enkelmann 85](possibly using regularization theory or relaxation techniques). Unfortunately smoothness constraints tend to result in the blurring of important discontinuities, in addition to reducing noise. Nonlinear smoothing has been proposed to reduce this effect. Alternatively, the response of edge detectors can be used to give an indication of where the gradient scheme is likely to be accurate. All of these different solutions tend to make the gradient schemes computationally intensive.

Energy Models

A number of schemes that are related to the work of Horn and Schunck have been proposed, some of which are considered as biologically inspired and were discussed in Section 2.4. Some other examples include a filtering scheme using six oriented spatial filters [Srinivasan 90]. This scheme operates on patches of an image to produce accurate two-dimensional velocity estimates, avoiding the aperture problem if there is sufficient textural information present. Another scheme [Heeger 87a] uses spatio-temporal filtering to determine the image velocity at every point. This scheme uses filters tuned to different spatio-temporal frequencies in a local region. The strongest filter response is used to select the velocity. The use of information from a local region does result in smoothing.

6.2.2 Tracking Schemes

Tracking schemes of many types are now routinely used to produce velocity fields. Such fields may be sparse and are often called *feature based optical flow fields*. These schemes have become popular due to the high computational costs of optical flow techniques and the availability of specialised matching hardware from related application domains such as video compression. Tracking is also a well understood process.

Feature Tracking

Tracking schemes involve several stages:

- **Feature extraction**. Some type of detector is used to find features in an individual image frame. Feature classes must be selected with some care because it is useful to use features that are important in a wide range of environments. Edges and corners are typical examples of features used in tracking schemes.

- **Determine correspondence**. Locate each feature in the next frame and therefore begin tracking the feature. Some form of error metric is usually used to restrict the number

of possible matches. The error metric will usually include information obtained from the motion model.

- **Tracking**. Once a possible correspondence between features has been established tracking can begin. This involves establishing a motion model for the feature that can be used to restrict the numbers of possible matches. An example of a system developed to track edges in two dimensions can be found in [Deriche and Faugeras 90].

Selecting features for a tracking scheme can be a difficult task. Edge detection has a long history in computer vision and the processes involved are well understood. Consequently edges are a commonly used feature in tracking schemes. The tracking scheme may track individual edge elements produced by edge detectors, in which case the system will suffer from the aperture problem. Alternatively the edge elements may be used to construct longer straight line segments. This will result in fewer objects to track but can only be expected to operate well in structured environments. ([Deriche and Faugeras 90] presents an example of this type of system).

The process of detecting corners is not as well understood; however, corners are easy to locate in two dimensions and are therefore not susceptible to the aperture problem. The only potential problem is that the type of environment in which such a system can be expected to operate well may be restricted. The most promising systems can exploit a combination of both edge and corner information. For example [Smith and Brady 95] uses corners to obtain a sparse flow field and then includes edge information when performing object segmentation.

Patch or region tracking

The availability of specialised region matching hardware for video compression applications has helped to reduce the problem of finding scene independent features. Comparing regions between frames allows the regions to be used as features and tracked using conventional methods. Problems can arise at the borders of objects moving in front of a background because occlusion can cause a large part of the patch to change, resulting in areas of uncertain velocity. Another potential problem is the choice of patch size, which may be fixed by the hardware. Different patch sizes will work well in different environments.

Template model

The template model (see Section 2.4.4) is another feature extraction method that helps to eliminate redundant data. The interesting thing about motion templates is that they are simple, scene independent, *spatio-temporal* features. This means that some motion information

is already encoded in the features. Unfortunately the simplicity of the features means that a relatively powerful tracking scheme is probably necessary for a robust system, although some simple schemes have been reasonably promising [Nguyen 96]. The aperture problem must also be addressed.

Performing Comparisons

A critical part of most of these schemes is the matching process. Correlation techniques may be used, but tend to be computationally expensive so some form of fast heuristic comparison is often used instead. Correlations may not be a useful comparison process in applications where the background is moving. Problems may arise when trying to compare localised features like corners in front of a moving background. The problem is that a large proportion of the pair of regions being correlated may contain "background". In this situation the correlation operation is effectively comparing the two background regions to one another, rather than the features of interest.

6.2.3 Comments

The collection of velocity estimation schemes just discussed share a common "problem". The fundamental nature of this problem is that arbitrary choices of some sort have been included in the design of the systems which may make each useful in only some environments. No automatic way of making those choices is available to allow the system to modify its behaviour while operating. For example, the choice of patch size in patch based systems is a compromise between sufficient textural detail for correct operation and resolution for the task at hand. The compromise is usually made with human guidance. In feature based systems the choice of features cannot be changed at runtime since designing feature detectors is a difficult task.

No single technique is ever going to be ideal for every application, however any improvement in flexibility and robustness will be useful. The different techniques can produce information suitable for different types of application. For example, the patch based techniques are likely to be useful for wide field applications, such as estimation of self motion. More precise feature based techniques are likely to be useful in segmentation.

6.2.4 The Aperture Problem

The aperture problem states that a local velocity measurement on a contour can only measure the component of velocity perpendicular to the contour (see Figure 6.1).

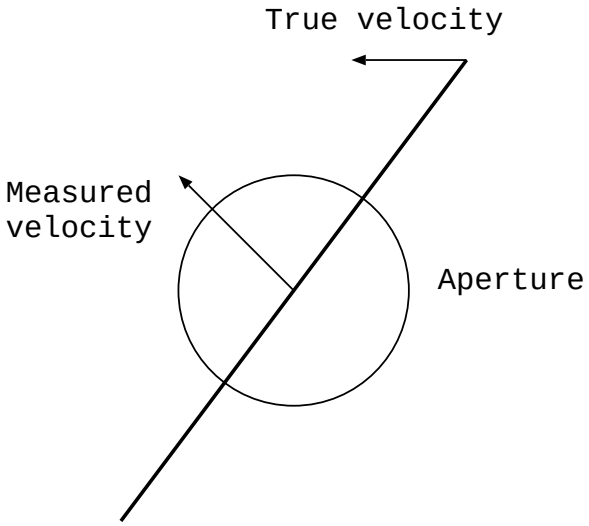


Figure 6.1: The aperture problem.

This problem is evident in both gradient schemes and tracking schemes in which the feature locations are only well defined in one dimension. Two types of approaches have been developed to eliminate the aperture problem.

The first is to combine the many local velocity estimates along the contour to form a more plausible result. Hildreth approached the problem as a minimization of change in velocity along a contour [Hildreth 84]. This approach is inherently sequential, computationally intensive, and assumes that sufficient information is available to locate the contour. Horn and Schunck used a similar approach to reduce the problem in their gradient scheme.

The second avoids the aperture problem by using more information to estimate the initial velocity. The problems of local estimates can be avoided entirely by processing larger areas with sufficient textural detail. This process was examined by Reichardt [Reichardt et al. 88] and is exploited in the generalised gradient scheme [Srinivasan 90]. The use of corners in tracking schemes exploits much the same effect.

Avoiding the aperture problem by using larger image regions introduces other complications. It is necessary to select patch sizes that are large enough to have sufficient textural content to avoid the aperture problem, while being small enough to provide a velocity flow field of useful density. The choice of “best” patch size will be dependent on the visual envi-

ronment and the visual task. Making this choice automatically may be very difficult.

6.3 Segmentation

Segmentation is the process of parsing an input scene into components. Motion is a very powerful cue for performing segmentation. There are several classes of segmentation schemes, some of which assume certain camera or object motion characteristics. Others make assumptions about the structure of the environment. Generally the techniques may be considered as either global or local. Global schemes attempt to find regions of consistency while local schemes find discontinuities in the flow field.

Local schemes can be attractive from the implementation point of view, but they tend to be susceptible to noise. Schemes of this type often require scenes with fine textures, and can only be expected to operate on simple images.

Global schemes commonly attempt to fit regions of the flow field to analytic functions, which may have a variety of parameters for different types of applications. The fitting process may be either top down or bottom up. In the top down approach the original image is broken down into successively smaller regions, while in the bottom up approach regions are merged together to form larger regions that fit the function. The technique proposed by [Adiv 85] is an example of a bottom up approach formulated for an environment of planar objects. Adiv's system used modified Hough transforms to perform clustering on optical flow fields.

Alternative schemes are hybrids of local and global schemes; they usually attempt to formulate the problem as some form of global optimization that can be solved using an iterative, localised computation [Murray and Buxton 87]. These approaches are computationally expensive.

Some simple, biologically inspired, systems have also been proposed. These systems are intended to model insect pursuit behaviour and therefore make restrictive assumptions about the environment. The most significant assumption is that the system must only distinguish a single small moving object. Nonetheless these systems do closely mimic the insect performance under similar circumstances [Reichardt and Poggio 79, Reichardt et al. 83] and may therefore be useful for some applications.

Tracking schemes based on statistical models of shape and motion also perform segmentation [Blake et al. 93, Isard and Blake 96]. These schemes have proved to be quite robust, but only segment objects for which models have been defined.

6.4 Reformulation

Feature tracking schemes estimate velocity by matching features between frames, using information from motion models to reduce the size of the search space. This process may be considered as analogous to the long range motion processing system in humans, and can be expected to function correctly at relatively low frame rates.

The local motion detectors described in the first half of this thesis model the short range motion processing system, and do not provide estimates of velocity. It is likely that the output of motion detector arrays could be treated in an identical fashion to conventional video streams, i.e. feature extraction, matching and tracking, followed by object segmentation. Some computational or performance advantages may result from the motion detection preprocessing, either in the form of simpler feature detection, smaller search spaces due to the elimination of stationary features, or the availability of directional information to initialise the tracking engine. However, this approach is not particularly well suited to processing information from a short range motion detection system.

Short range motion detection systems are characterised by high frame rates and low motion distance per frame relative to the scene texture. The short distances moved per frame make it difficult to obtain relatively accurate velocity estimates quickly, so using conventional velocity based segmentation could result in significant time lags before results can be obtained. Experiments on short range processes in human vision suggest that motion based segmentation can occur very quickly — in some cases only one frame is necessary, and that "conventional" features are not required. The classical test that demonstrates these results involves random dot textures. If an "object" consisting of a random dot pattern with known statistical properties is placed on a background with the same statistical properties, then no object is perceptible. This is an example of the mathematically perfect camouflage used by Julesz in random dot stereopsis experiments [Julesz 71]. If the object is moved then it is immediately distinguishable from the background. This is not a particularly difficult situation to copy, in fact a simple temporal derivative operator will separate the object and the background, since the background is not changing. The interesting part of the test occurs when the background is allowed to change in a random fashion, while maintaining the same statistical properties. In this case the human visual system is still capable of isolating the consistently moving objects very quickly. However, the simple temporal derivative operator would give equivalent responses to foreground and background.

This type of test illustrates a number of important points. Firstly, there are no edges or corners that can be tracked by conventional tracking schemes. Dots are the only features

available for tracking, and while it is possible to track dots, the lack of features for comparison and the large numbers of dots in the scene will tend to result in a complex tracking system. (Patch and flow based schemes should function well in this kind of texture rich environment.) The second important point is the speed with which segmentation takes place. It is usually assumed that high speed, yet global, operations in early vision are an indication of a highly parallel, locally interconnected, computational structure. The high speed of segmentation also suggests that a significant amount of segmentation processing is performed before accurate velocity information is available.

These observations suggest an alternative to the usual segmentation procedure that is more suitable to the short range motion detection environment. Segmentation, rather than velocity estimation, should be the primary goal of the preprocessing scheme. The scheme should use inaccurate velocity estimates, that are readily available, in combination with spatial information to perform some basic segmentation that does not depend on scene characteristics or conventional feature extraction methods. In the best case this will result in well segmented objects. Useful feature extraction will still take place in situations that are not well suited to the idea (i.e. where larger scale motion is occurring).

6.5 Benchmarking

The lack of standardised methods with which to test computer vision techniques has been a known problem for many years [Jain and Binford 91]. Part of the difficulty is due to the complexity of real biological vision systems and the problem of defining exactly what they are doing. It is therefore hard to define what functions a “standard” vision system should perform. The other major difficulty is common to benchmarking in many different fields - defining a set of problems that is sufficiently general to provide an indication of system performance in real applications and not allow over specialisation of the system while being simple enough to provide a useful measure. The wide range of environments and the complexity of tasks being performed make these problems especially difficult for vision systems. For example, the higher level problem addressed in this thesis, motion based segmentation, is very difficult to define. At present we are restricted to somewhat haphazard comparisons to our own perception, but in the future some form of statistical comparison to psychological tests may be practical.

The only practical option available at the present is to measure the overall performance of the system in which the vision system is operating. This approach makes sense for many industrial applications where error rates may be a useful measure.

6.6 Conclusion

This chapter has discussed some of the traditional approaches to velocity estimation and scene segmentation. The difficulties with the different types of velocity estimation scheme have been outlined. It is proposed that segmentation schemes using accurate velocity estimation are more suited to operation in a long range motion estimation environment. This thesis is investigating methods of processing short range motion, and it is suggested that segmentation should be considered as a fundamental goal of this processing and an integral part of the motion estimation process. It is proposed that alternative segmentation techniques that do not require accurate velocity estimates or explicit feature extraction should be developed. The concept of segmentation in short range motion environments will be examined in more detail in the following chapter and the techniques developed to perform this segmentation are described in Chapter 9.

Chapter 7

Perceptual Motion Structures

7.1 Introduction

This chapter proposes that a new structure, called a *perceptual motion structure*, should be used as the building block of short range motion processing systems. This structure is analogous to the spatial structures that convey most of the information present in still images. Perceptual structures are defined in Section 7.2 and perceptual importance is discussed in Section 7.4. The requirements of a system capable of extracting the new structure are examined in Section 7.5.

7.2 Perceptual Structures

Humans use many different sources of information when trying to understand a stationary scene. Colour and texture are often used, however edges are usually considered to be the most important source of information about a scene. The fact that humans can generally extract most of the important scene information by using only edge data is a strong indication of the importance of edges. As a consequence of this importance, edge detection has a long history in computer vision and image processing and the local processes are well understood. The part of the edge detection process in humans that is not well understood is the grouping of local edge detector responses into possibly large spatial structures. There is a large number of possible groupings for any edge detected image, yet the human visual system always makes a consistent choice.

Evolutionary experience has indicated that long, smooth lines are likely to be the most reliable sources of information. The human visual system has therefore developed mechanisms to combine low level information, such as edge segments, into larger structures of this

kind. The structures that meet these requirements are known as *perceptual structures*. In this thesis the perceptual structures that are associated with stationary information will be called *perceptual spatial structures*.

The formation of perceptual spatial structures is probably part of the early visual process, and happens very quickly. Some form of ranking of importance, or *perceptual significance*, of perceptual spatial structures also seems to occur. Longer, smoother lines, which are reliable sources of information, are perceived as more important than shorter or rougher ones, which are likely to be less reliable. The ranking may be vital to the real time operation of our visual system because it allows limited computational resources to be applied where they can be best utilised. The ranking process can quickly isolate the important structures from cluttered environments of locally identical structures. This effect is often called the "pop out" effect, and an example is illustrated in Figure 7.1. Three circular structures in this figure usually become obvious after very brief exposure, even though the colour and thickness characteristics of the lines forming the circles and the "background" are identical.

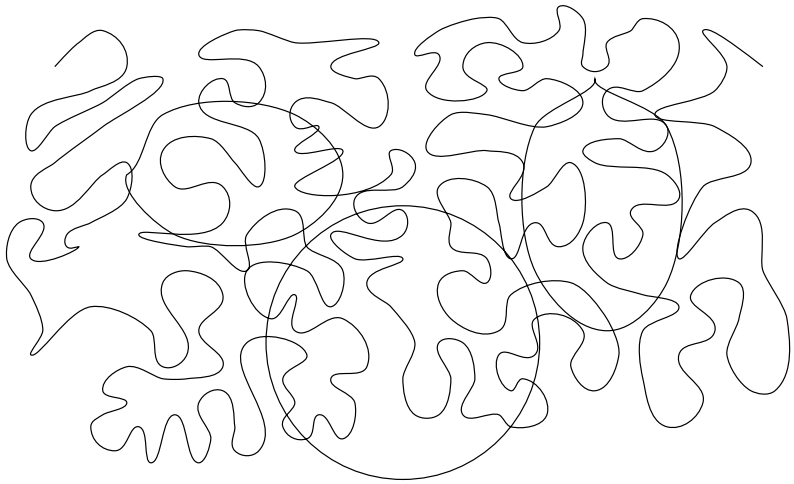


Figure 7.1: An example of the "pop out" effect.

In more complex, real world examples, the ranking process probably helps to eliminate the effects of noise by rejecting structures that are not likely to convey useful information. Random structures formed from noise are likely to be in this category.

General perceptual spatial structures have found limited uses in computer vision to date. Creating simple global structures, like straight lines, from local edge detected data is reasonably common, but is only useful to limited applications involving man made environments. A more general method of representing and extracting perceptual spatial structures is likely to be important to many applications. The obvious importance of global shape characteris-

tics in the perceptual process has inspired a number of researchers. Sha'ashua and Ullman developed a locally interconnected network that measured line importance as a function of length and curvature [Sha'ashua 88, Ullman 92]. The network was capable of filling gaps and could eliminate cluttered backgrounds. Ahuja and Tuceryan investigated the classification of dots in dot patterns using a hierarchical approach that utilised both local and global structural information [Ahuja and Tuceryan 89].

The use of perceptual spatial structures in the domain of motion processing has been limited to the tracking of simple structures like straight lines and corners. It is likely that a considerable improvement in flexibility of tracking systems could be achieved if a more general representation of spatial features was available. It is probable that perceptually significant spatial structures would be a useful model of the type of features that should be tracked. The problem of extracting and representing perceptual spatial features will not be investigated in this thesis, although the preprocessing methods that will be developed may be useful for this application.

7.3 Perceptual Motion Structures

We have seen that the human visual system appears to have developed ways of grouping edge detected data based upon the type of structures that are likely to provide useful and reliable information. Perhaps it may be useful to apply similar ideas to the processing of short range motion information. This section will introduce the concept of a *perceptual motion structure*, which may be considered as an equivalent to the perceptual spatial structure in the spatio-temporal domain. A perceptual motion structure should be a reliable source of information for segmentation and other applications.

Most real moving objects possess significant levels of perceptual spatial structure, which helps to distinguish them from their surroundings. When in motion these objects possess both perceptual spatial structure and perceptual motion structure. However, it is possible to construct objects with little or no perceptual spatial structure, yet these objects are only perceptible when moving (see Section 6.4). (Perhaps more precisely, these objects have the same level of perceptual spatial structure as their background.) The perception of this kind of object when moving is extremely powerful and indicates the potential importance of the perceptual motion structure. Examination of objects like this will help to isolate the properties of a perceptual motion structure.

A randomly textured object does not possess any significant perceptual spatial structure, so purely spatial properties are unlikely to be an important component of the perceptual

motion structure. In conventional motion tracking schemes, purely spatial properties are used because simple comparisons may be possible and real objects, especially man made ones, often possess some easily detectable and definable spatial property, like straight lines. The crucial property of a perceptual motion structure is that the spatial structure remains constant over time, despite being possibly difficult to define. A consistent, moving structure will be a more reliable and useful source of information than the collection of moving fundamental components (like edge segments), since binding the fundamental elements together should allow local imperfections to be eliminated. As mentioned in Chapter 6, the binding is also beneficial to subsequent processing stages. Thus, mechanisms to explicitly bind motion information into structures of this kind are potentially very useful.

A perceptual motion structure may therefore be defined as *a collection of fundamental elements that appear to be related due to their common motion.*

This notion is closely related to the Gestalt "principle of common fate", which says that objects moving together appear to belong together.

The problems with using conventional tracking schemes in environments with poorly definable spatial structures were outlined in Section 6.4. The remainder of this thesis is dedicated to forming perceptual motion structures from the responses of local motion detectors with the hope that the structures will prove to be robust in many different environments and useful in a variety of different applications such as object tracking. These perceptual motion structures are then used to track and group moving objects in real and synthetic scenes.

7.4 Perceptual Importance

The goal of this work is to provide a framework for segmenting images using motion information. Unfortunately, it is extremely difficult to obtain benchmarks to enable results to be quantified. The obvious "perfect" system to use as a baseline is the human visual system; however, it is extremely difficult to isolate different computational components of the human visual system. It is therefore difficult to determine whether human image segmentation in a given situation is based upon motion information, spatial information or something more complex like object recognition. In most cases, a combination of many different cues is probably used.

As mentioned in the discussion of perceptual spatial structures, the human visual system does seem to have the ability to rapidly rank the importance of large structures, and the ranking order is likely to be related to the reliability of information provided by the structure. It seems likely that such ranking is also an important part of processing perceptual motion

structures and may in fact be critical to the detection process. It is therefore important to consider what makes a moving object perceptually important.

Let us first consider the consistent motion of a point object in an environment consisting of other randomly moving point objects. A conventional tracking system will take a series of measurements of the point positions and produce a track for the object, which may or may not agree with a human's perception of the object's track. However, the ability to track an object does not make it perceptually significant; in fact, any reasonable tracking scheme should also be capable of providing plausible tracks for the randomly changing points in the background as well.

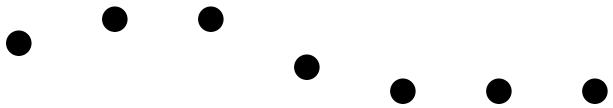
Conventional tracking systems are not an appropriate mechanism to determine whether one track is likely to be more significant than another. Now consider Figures 7.2. These figures show the same set of measurements in 3 very different sets of circumstances that help to indicate some of the factors that influence perceptual importance. Figure 7.2(a) illustrates an isolated track that appears to be significant. If the track is displayed in the presence of some similar surrounding clutter then it becomes less significant until its presence is completely disguised by the clutter surrounding it (Figures 7.2(b) and 7.2(c)). These figures indicate that both track consistency and local geometry have a strong influence on the perceptual importance of individual moving points. In general terms the relationship may be described as follows "a track may be perceptually significant if the uncertainty in the track is small compared to the surrounding spacing", where the track uncertainty is related to the prediction errors in the tracking process.

This same concept also applies to lines in space. It is possible to fit smooth curves through the interior of arbitrary groups of dots, but most of the candidates would have no perceptual significance. Humans would not consider such curves as important.

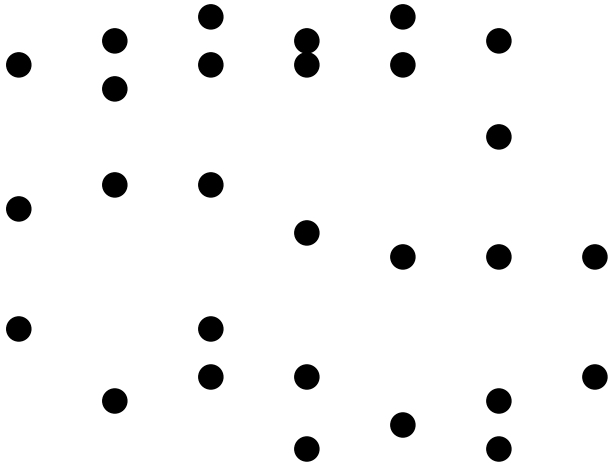
In summary — *it seems to be as useful to measure the importance of visual structures as it is to estimate the parameters of the structures.*

These ideas are also important in defining the concept of rigidity in the context of perceptual motion structures. Noise in imaging systems combined with inconsistent motion (rotations and deformations) means that objects do not tend to be perfectly rigid. Thus, the perception of rigidity is also important and will also be dependent upon the local geometric structure.

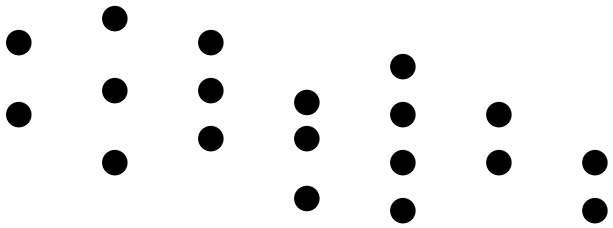
It would be useful to design tests to attempt to determine what the relationships between the geometric structure and perceptual importance of moving objects in humans actually are. However, designing experiments to isolate the relevant parts of the visual system is extremely difficult. For now the focus will be on designing a system in a fashion that makes



(a) Isolated measurements.



(b) Partially isolated measurements.



(c) Surrounded measurements.

Figure 7.2: 3 identical measurements in different surroundings.

this information readily available.

A moving structure tends to be isolated more quickly by the human visual system and is generally regarded as more important than a moving point under the same circumstances. The relationship between the perception of moving points and moving structures in humans is not clear, but this work will assume that similar processing mechanisms are involved. It will also be assumed that the difference in response time for the different stimuli is related to the amount of information that is available to the segmentation system — a consistent spatial structure provides additional information. If it is assumed that a fixed amount of information is necessary to segment an object from its surroundings, then it is easy to understand why a moving structure could be isolated more rapidly than a point object. Information about the point object can only be obtained from the time dimension and is therefore not available immediately. If an object has a significant spatial structure then some additional information is available and may be used immediately.

7.5 Computational and Structural Requirements

There are implementation requirements that are relevant to the formation of perceptual motion structures as well as some that should be considered in any real vision system to ensure robustness and flexibility. These include:

- **Parallel Computations**. Humans form perceptual spatial structures and rank their relative importance extremely quickly. This high speed is usually taken to indicate that the grouping is performed using a highly parallel computational structure. Due to the types of neural interactions that are believed to be responsible for high speed computations, it is also assumed that the parallel computations must rely on localised interactions. It will be assumed that mechanisms forming perceptual motion structures have similar properties.

  It is desirable that techniques developed should have a conceptually parallel structure to permit a high speed parallel hardware implementation, but given the generally serial nature of most digital signal processing devices, especially memories, it is also important that a serial implementation be realistic. For example, techniques that might use an interaction between one pixel and every other pixel in an image should be avoided because although they are conceptually parallel the memory access will still need to be serial, making the entire scheme unrealistic.

- **Eliminating arbitrary thresholds**. Thresholding is an early stage in many forms of image analysis. In a self contained system it is desirable that any thresholding that is performed be sufficiently conservative so that vital information is not lost, while also being aggressive enough to stop the vital information being excessively cluttered by noise. Some form of optimal statistical filtering is often used, but such techniques require knowledge about foreground and background image statistics, both of which will be scene dependent. This is a reasonable technique to use in applications where the environment does not change.

  Many other types of processing also use thresholding. A threshold of some form is essential wherever a decision is made. This thesis will aim to avoid basing this kind of decision on absolute values, instead formulating the threshold decision in terms of relative values or probabilities. In the next chapter we will introduce Voronoi thresholding, which attempts to meet these requirements in a scene independent fashion.

- **Avoiding expensive searches**. Techniques that attempt to perform global optimizations or perform searches of large image spaces are unrealistic and should be avoided. Ideally the search space should be kept small without making arbitrary decisions that may be scene dependent.

- **Scene dependence**. The goal of this thesis is to produce a very flexible approach to short range motion processing. In order to achieve this, care must be taken to ensure that scene dependent assumptions are not included. If any heuristics are to be used, then they should be based on experimental human evidence. Any assumptions about camera motion or object motion should also be avoided.

- **Doing too much.** Processing of short range motion information is a preprocessing step, so it should not be expected to solve all problems all the time. The output of the system should be considered as appropriate for some form of higher level controller. For example, a tracker could be used to maintain distinct objects and detect occlusion. Ideally the preprocessing can take advantage of knowledge gathered by the controller, but this is beyond the scope of this thesis.

7.6 Conclusion

This chapter has proposed the concept of a perceptual motion structure that could form the basis for short range motion processing systems. This concept is an extension of the idea of

perceptual spatial structures, like smooth curves in the spatial domain, to the spatio-temporal domain and is very different to the conventional approach to velocity estimation and clustering. The approach is specifically designed to operate on the output of local motion detectors which do not provide velocity estimates. Detecting perceptual motion structures will provide a robust and flexible preprocessing step for visual systems. It is also proposed that psychologically inspired interpretations of importance and rigidity are likely to be useful in designing the system. It is hoped that this approach will help produce a more robust and flexible system.

Chapter 8

Voronoi Thresholding

8.1 Introduction

Thresholding of processed images is often a vital step in an artificial visual processing system. It is the point at which a decision is made about the presence or absence of the type of feature or property being detected. For example, thresholding is usually performed at some stage during the edge detection process.

Deciding on a threshold level is extremely difficult, especially if the aim is to have close correspondence to human decisions while eliminating noise. Many different thresholding techniques exist. Some use statistical knowledge of image properties and knowledge of detector operations while others are generated using a more heuristic basis. Generally an implementation of a thresholding scheme will only be suitable for a particular application.

This chapter describes a new technique that is inherently parallel, does not make any assumptions about image structure, and creates data structures that are useful in subsequent processing stages. The emphasis of the technique is on producing a representation that is suitable for subsequent processing stages, rather than creating an image that has a high perceptual quality.

8.2 Overview

This chapter is structured as follows. Sections 8.3.1 and 8.3.2 discuss the need for and requirements of an automated thresholding process. Here the Voronoi neighbourhood is proposed as a useful way in which to represent the important geometric properties. In Section 8.3.3 a method of computing modified Voronoi neighbourhoods from motion detected images is introduced. Two possible implementations of the scheme are described in Section

8.3.4. Sections 8.4 and 8.5 include some comments and conclusions.

8.3 Fundamentals

It is difficult to threshold data produced by local image processing operations in a way that agrees with human perception. This is partly because humans can use many different sources of information when making the same decision. For example, human edge detection decisions can utilise a more global concept of what a line is than most thresholding schemes (see the discussion about perceptual spatial structures in Section 7.2), as well as an understanding of the scene being viewed.

Thresholding is usually a vital step that reduces the amount of data that must be processed. Therefore it is essential to develop a thresholding scheme that does not require intervention by humans, or any scene dependent statistical knowledge.

It is also desirable that no global statistical information be used because this would be difficult to achieve in a parallel fashion (parallel operations are desirable for the reasons outlined in Section 7.5). In reality, a vision system using global statistics that are not scene dependent would probably be a realistic compromise because the statistics should not change quickly and therefore would not need to be updated every frame.

The first step in designing the thresholding scheme will be to consider the requirements of later processing stages.

8.3.1 Thresholding System Requirements

The data provided by the thresholding process is going to be used to extract the perceptual motion structures. In particular the data is going to be related to the spatial component of the perceptual motion structure. Thus, the thresholding process should provide a framework from which relevant spatial information can be extracted quickly. The temporal component of the perceptual motion structure will be developed by examining successive frames, so only spatial operations should be used in the thresholding scheme.

It is also essential that the thresholding scheme be flexible. There should be no dependence on scene brightness or on the spatial scale of the data. Ideally, most, but not all, of the brightness dependence should be eliminated by sensor adaptation, however differing scene contrasts and velocities can still produce a wide range of motion detector outputs.

Since the thresholding is generating data for more sophisticated processing, it is acceptable to have some noise remaining. In fact, given the restrictions being imposed, elimination

of all noise is impossible. However, it is essential that important features be distinguished in some way. This requirement makes the simple approach of a very low threshold unacceptable.

The output of the motion detectors is a signed intensity. The sign indicates the direction, so the thresholding scheme should operate on the absolute values of motion detector outputs. The absolute magnitude of the motion detector output should be considered as a measure of reliability. A large motion detector response may indicate an object moving at a velocity close to the detector's tuned velocity, or an object that has a very high contrast (or both). This means that the scheme should give preference to "large" detector outputs, where the term "large" means relative to some local region.

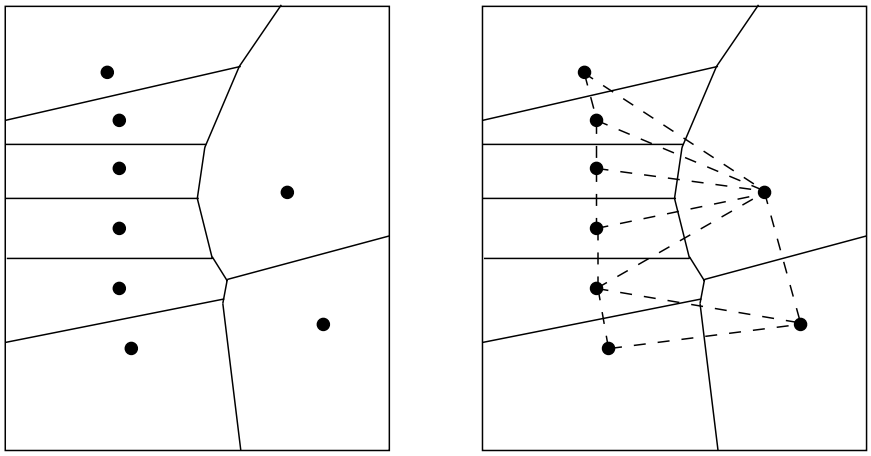
8.3.2 Local Geometry

The requirement of spatial information will be examined first. Consider a simple dot scene (possibly something that has already been "well" thresholded). A method of capturing the local spatial structure of these dots, in a scale independent manner, is required.

There are many different possibilities. One of the most common is a graph connecting a dot to its n nearest neighbours. Ahuja and Tuceryan pointed out the problems with this simple idea, and instead proposed that the *Voronoi neighbourhood* of the dot could be used as a powerful representation of the local structure [Ahuja and Tuceryan 89].

The Voronoi neighbourhood of a dot is the region that is closer to that dot than any other. Voronoi neighbourhoods, and the dual Delaunay graph, which connects Voronoi neighbours, have been used previously in computer vision for representation and approximation of three dimensional data. Ahuja and Tuceryan used the Voronoi neighbourhood and the Delaunay graph of dot patterns to generate classifications of the perceptual roles of individual dots in the patterns. The initial estimates of a class membership were obtained by analysing the characteristics of the Voronoi neighbourhoods. Figures 8.1(a) and 8.1(b) show an example of a Voronoi tessellation and the dual Delaunay graph.

A lot of information about the local structure of a point can be determined from the shape of the Voronoi neighbourhood. The example shown in Figure 8.1(a) illustrates some simple properties. The neighbourhoods of points on the line in the left part of the picture tend to be long and thin. Ahuja and Tuceryan used this type of property, as well as more complex ones, to estimate the roles of dots in a pattern. This sort of information can be expected to be helpful in isolating perceptual motion structures. The structures are also likely to be a useful mechanism in isolating perceptual spatial structures, like curves, but this is beyond the scope



(a) A Voronoi Tessellation.

(b) The Delaunay graph (dashed lines).

Figure 8.1: A simple example of a Voronoi Tessellation and the corresponding Delaunay graph. The graph edges are indicated by dashed lines.

of the thesis.

8.3.3 Voronoi Thresholding Scheme

The Voronoi tessellation of a set of points is only dependent on the positions of the points (The tessellation for a planar set of points can be computed using an $O(n \log n)$ algorithm). Therefore, the true Voronoi neighbourhood is only useful in situations where the points may be considered as dimensionless quantities, e.g. after thresholding has been carried out. Voronoi neighbourhoods and Delaunay graphs appear to be a reasonable solution to the problem of representing the spatial information; however, an appropriate scheme is still required to produce the thresholded output.

This section investigates an alternative, modified neighbourhood scheme. In this scheme the formation of neighbourhoods is closely coupled to the thresholding process. The scheme meets the requirements for scene independence and parallel computations.

Description

The scheme begins by assuming that every point in the motion detected image is potentially a salient point. (A salient point is one that is retained after thresholding). A spatial decay

function is started at every image point. This decay function is of the form

$$f(x - x_0) = \alpha c^{|x - x_0|} \tag{8.1}$$

where α is the motion detector response at x_0 and c is a constant between 0 and 1. $\mid x - x_0 \mid$ is the distance between points x and x_0 — this will be a distance in two dimensional image space in most circumstances.

The simplest way to visualise the operation of the scheme is to consider the decay function associated with each point to be evaluated independently over the entire image. The characteristics of the decay function are dependent on the location of the starting point and the value of motion detector response at that location.

All of the decay functions are then overlaid (Figure 8.2 shows a one dimensional example), the value of the largest decay function at each image location is retained to form an envelope as shown in Figure 8.3. Salient points are those at which the envelope value does not exceed the motion detector response. The neighbourhood of each salient point is defined by the extent of the decay function associated with it that survives the process of combining all of the decay functions.

If the input image had already been thresholded, so that the image contained only points of magnitude zero and one, then this procedure would generate correct Voronoi neighbourhoods. However, in typical situations a neighbourhood will be affected by the value of the motion detector response with which it is associated. This dependence will be most obvious at strong discontinuities, where the neighbourhoods of points with higher magnitude will be significantly larger than those adjacent neighbourhoods associated with lower magnitude points. This means that the scheme is not producing true Voronoi neighbourhoods. However, this is not a disadvantage. In fact it means that strong motion detector responses tend to be well distinguished because the noise in a wide surrounding area is removed. This is a desirable property.

A one dimensional example of the growth of decay functions is shown in Figures 8.2 and 8.3. An example of the operation on a single frame of motion detected video is shown in Figures 8.4, 8.5 and 8.6. Note that the Voronoi neighbourhoods are two dimensional regions. The responses of horizontally oriented motion detectors of the type described in Section 5.2 are shown in Figure 8.4(b). In this example, no attempt has been made to tune the detectors to the velocities present in the scene, so the response to the car moving from left to right is very low (the contrast between the car and the background is also low). The results of thresholding are shown in Figure 8.5(a), with the direction indicated by the colour (white indicates motion to the right, black indicates motion to the left). Even though the motion detector response to

the car moving to the right was very low, it was not discarded by the voronoi scheme. Some noise is also detected, as would be expected. The neighbourhoods are illustrated in Figure 8.5(b), with colour coding used to distinguish the different neighbourhoods. Note that this form of representation is not perceptually useful.

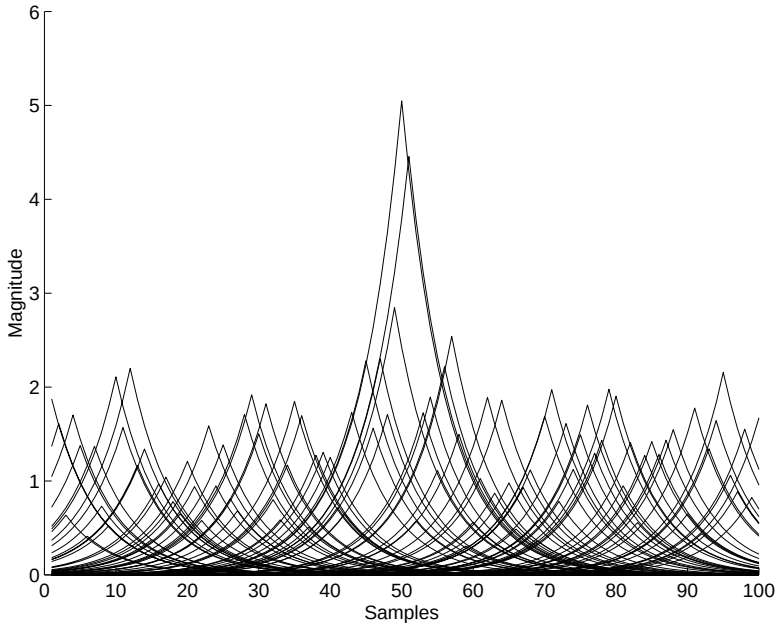


Figure 8.2: All decay functions for an edge detected signal.

It is essential that the information required by later processing stages can be readily extracted from these data structures. The important requirements are that the nearest salient point to any image location can be readily found (this will eliminate time consuming searches), and that the Delaunay graph be available. Other properties are likely to be important to different applications, but these will be discussed in a later chapter.

The process of constructing the decay functions makes the first requirement easy to satisfy — each point in a given neighbourhood knows the location from which the decay function begins. (At a salient point this "knowledge" refers to itself.)

Building the Delaunay graph may be the most time consuming part of the process. It involves finding the borders between neighbourhoods and constructing an edge between the two salient points if there isn't one there already.

The Delaunay graph is extensively used in the motion processing techniques described in this thesis.

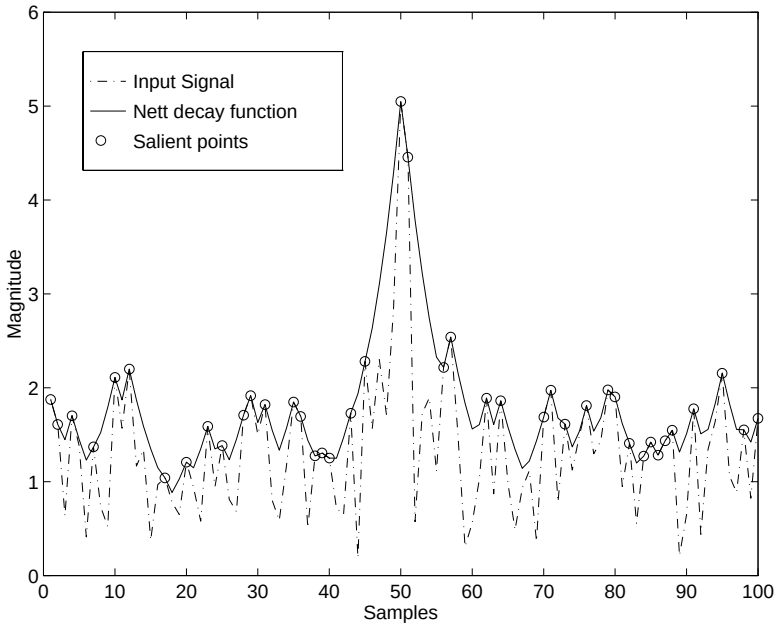


Figure 8.3: Result of applying Voronoi thresholding to an edge detected signal.

8.3.4 Implementation

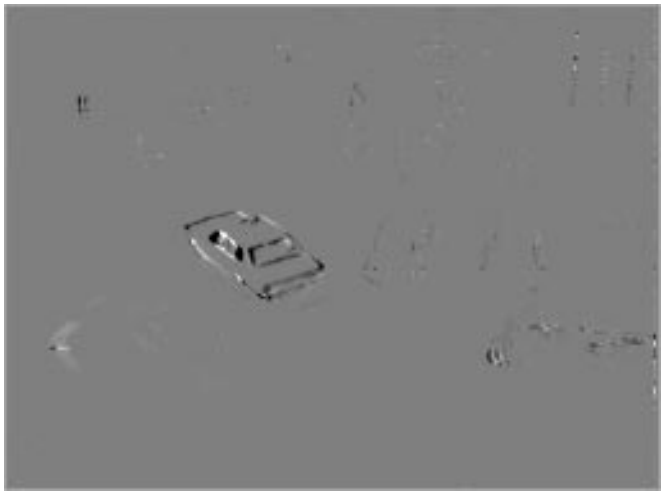
It is obviously inefficient to evaluate the decay function for each image location over the entire image; a more efficient approach is actually used.

The Voronoi thresholding scheme just described can be implemented in a serial or parallel fashion. The parallel approach employs cellular automata. The cellular automata approach is easier to understand, and may be considered as a very simple biological model, however it would be an inefficient architecture for a digital VLSI implementation. The serial approach uses identical local computations to the cellular automata, but is more efficient for a serial computer.

Parallel Approach

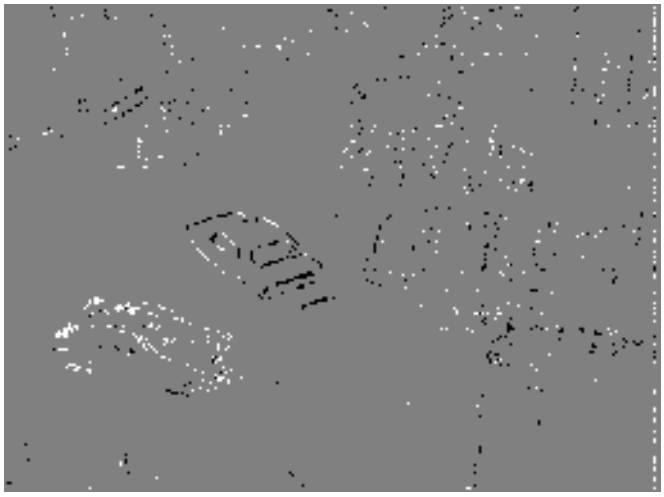
The parallel structure requires a computational element (automata) at each pixel which interacts with each adjacent automata. The decay functions will "grow" at one pixel per iteration, so a number of iterations equal to the larger image dimension are necessary to ensure a correct tessellation. The parallel scheme can be stopped at any time to produce an incomplete result that may contain most of the useful information. If the image contains some structure, which will typically partition the image, then fewer interactions will be required to produce the correct tessellation.

(a) Unprocessed image frame.

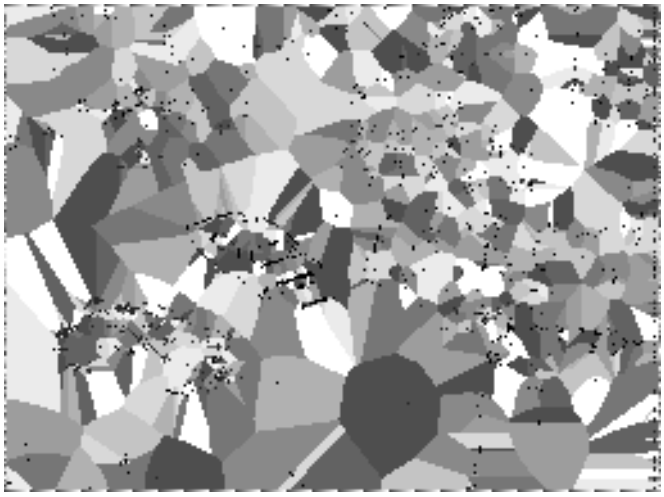


(b) The motion detector response.

Figure 8.4: The raw “Hamburg Taxi” input and the motion detector response. No attempt was made to tune the motion detectors to the car velocity.



(a) The salient points resulting from Voronoi thresholding.



(b) The modified Voronoi neighbourhoods (neighbourhoods indicated by colour).

Figure 8.5: The results of Voronoi processing.

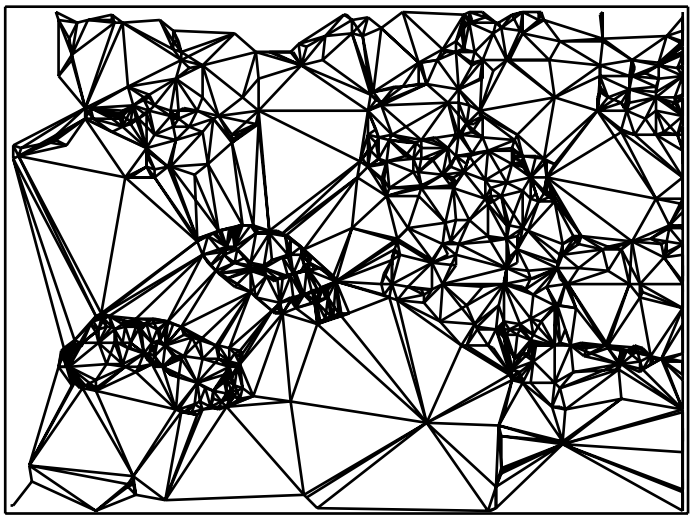


Figure 8.6: The corresponding Delaunay graph.

The value of the decay function described in Equation 8.1 can be computed at any location if that location and the location of the origin of the decay function are known. Each automota therefore requires a reference (a pointer in a software implementation) that indicates the location of the nearest salient point. The salient point indicated by the reference is the origin of the decay function to which the location currently belongs.

Algorithm

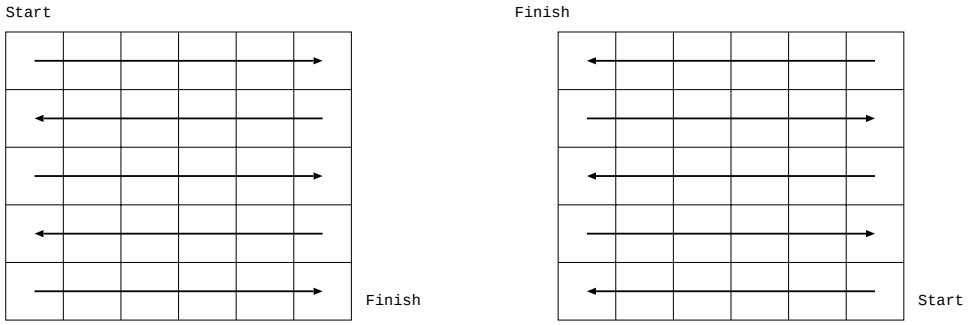
```
Each pixel has eight neighbours, each with a reference (pointer)
indicating the origin of the decay function to which they
currently belong.

Compute the value of each of these decay functions at the
centre pixel and select the maximum (MaxDecay). Keep a copy of
the reference associated with this maximum (MaxRef).

Determine the value of the decay function (LocalDecay)
associated with the current pixel's reference (LocalReference)

if MaxDecay > LocalDecay then
  LocalReference := MaxNeighReference
end if
```

The process of copying the reference causes the decay functions to grow. Regions which have a low value of α at their origin will be overwritten by their neighbours (the region to which a pixel belongs may change if the $LocalReference$ is changed. If the computation is repeated many times at every location then regions will grow until an equilibrium is reached.

Serial Approach

The pixel based computations for the serial scheme are identical to those described for the parallel scheme. However, only a few operations are required at each pixel instead of many iterations of simultaneous operations. Regions in the parallel implementation grow due the continual copying of references between neighbouring automata. A region may expand by one pixel each time the computation is carried out at a pixel. When implementing this scheme on serial hardware it is necessary to perform the computation once on each image pixel before repeating, resulting in a computationally expensive scheme.

The cost of the scheme for a serial implementation may be significantly reduced by carefully selecting the order in which pixels are visited (i.e. computations performed). If pixels are visited as shon in Figures 8.7(a) and 8.7(b) then only two passes (two iterations per pixel) are necessary. The improvement is possible because the visit order corresponds to the possible "wavefront" of region growth, so the visit order effectively causes the regions to grow more efficiently.



(a) First pass (b) Second Pass

Figure 8.7: Visit order for serial evaluation of neighbourhoods.

The potential disadvantage of the serial approach is that the computation cannot be halted with the expectation of sensible results because only a small number of regions are affected by the computations occuring at a given instant. However it is far more suitable for digital implementation than the parallel version.

Biological Implications

There is no evidence to suggest that the thresholding scheme or the spatial representation described above are exploited by biological systems. If the ideas were used by biological systems then the hardware implementation would be very different. The problems of neural implementation are beyond the scope of this thesis. However it is a possibility that coupled neural oscillations (temporal binding) could be used to create and represent the neighbourhood structures.

Simplified Version

A simpler form of the scheme may be used if only the thresholding properties are desired (rather than the data structures as well). Instead of maintaining the reference at each pixel, the value at the point where the decay began and the distance to that point are stored. The distance can be updated as the neighbourhood expands. The salient points are those where the decay value is equal to the input value.

Including noise information

It is possible that some knowledge of noise mechanisms in the imaging system may be available. Such information should be used. If a reliable noise threshold is known then it may be utilised by ignoring all values below the threshold (ie performing a simple initial thresholding step) and then proceeding as before.

8.3.5 Building the Delaunay Graph

There are two steps involved in the current method of creating the Delaunay graph — finding edges in the Voronoi diagram and checking whether an edge connecting the two neighbourhoods already exists.

The second part could be improved by using traditional fast search techniques, rather than the simple list search used in the prototype.

Edges in the Voronoi diagram are found using a simple 2×2 kernel. The edge detection is operating on essentially ideal (noiseless) data and is actually comparing image locations (pointers). This is very simple and could be easily implemented in hardware.

This pixel based technique was used because it permits a conceptually parallel system and is in character with the method used to form the Voronoi diagrams. The Delaunay graph can be computed analytically with worst case complexity $O(n^2)$ for the planar case or an

average complexity of $O(n \log n)$ using randomisation methods. These techniques may be considerably faster but have not been investigated.

8.4 Comments

Voronoi thresholding is certainly not the perfect thresholding scheme. It is still necessary to select one parameter (c). If c is too small then significant features will not be isolated from their surroundings. If c is too large then significant points may eliminate too much of the surrounding data. Experience has shown that a value between 0.6 and 0.75 usually produces perceptually satisfactory results. (Note however that perceptual quality was not the original criterion for this scheme.) Possibilities for modifying c as part of the segmentation process will be discussed later (Section 9.6.6).

A second potential problem relates to the pixel based representation of the Voronoi diagram. The processing of the Voronoi diagram to produce the Delaunay graph involves an edge detection operation. The discrete nature of the boundaries of the Voronoi neighbourhoods may mean that some graph edges are created that would not be produced by an analytical solution, while others may be missed. This is most likely to happen in areas where there is a high density of salient points because boundaries are short and are therefore represented by few pixels. In most cases these errors should not cause serious problems.

8.5 Conclusion

This chapter has developed a thresholding scheme that can be implemented in a parallel fashion without making any assumptions about the nature of the scenes being observed. The process of performing the thresholding is also responsible for creating data structures that capture the types of information that are expected to be important when extracting perceptual motion structures.

The perceptual performance of the Voronoi thresholding scheme is certainly not as high as an application specific thresholding scheme would be since the Voronoi scheme will consider low magnitude noise as salient if it is sufficiently isolated. However, it is important to note that all important structures above the noise level are located, and that this is achieved without any *a priori* knowledge about the scene structure.

Chapter 9

Segmentation using Perceptual Motion Structures

9.1 Introduction

While the concept of a perceptual motion structure that was discussed in Section 7.3 is relatively straightforward, the precise definition, and therefore isolation of such structures, is not as easy. Perceptual motion structures do not need to conform to any particular spatial structure, provided that whatever structure they do possess remains consistent over time. This rigidity in motion is what distinguishes a perceptual motion structure from its surroundings.

This chapter describes a proof of concept system that operates by maintaining a measure of certainty about the relationships between simple features. The system uses some principles derived from human visual perception to define the basic quantities that are used. The system has been used to process real, noisy scenes and the results are shown in this chapter.

9.2 Overview

This chapter is structured as follows. Section 9.3 introduces the properties required of a general purpose motion segmentation system, including perceptually meaningful measures of importance, parallel implementation and deriving decision criteria from an understanding of human perception of equivalent phenomena. Section 9.4 discusses the human perception of rigidity that relates to some of the segmentation criteria that are developed later. The segmentation system is introduced in Section 9.5. The segmentation system involves several stages — estimation of point correspondence, estimation of graph edge correspondence, making uncertainty estimates, performing spatial interactions and evaluating some local geometric

properties.

These processes produce the information required to make segmentation decisions, as discussed in Section 9.5.6. The error metrics employed by the correspondence estimation processes are used to develop the uncertainty estimates. The uncertainty estimates and the local geometric properties are the critical factors on which segmentation decisions are made. The decision criteria are based on a simple understanding of how humans may perceive rigidity.

Section 9.6 describes a technique that stabilises the results of the segmentation decisions over time and provides a useful representation of the segmented regions. Test results for real and artificial scenes are provided in Section 9.7. Section 9.9 contains some general discussion.

9.3 System criteria

The system described in this chapter has been designed with a number of goals in mind.

- Mechanisms should be present to produce a perceptually meaningful measure of track importance for point objects.
- The system should segment independently moving objects. Objects with spatial structure should be isolated more quickly than those without.
- These tasks should be performed using simple, local interactions.
- Where possible any decision criteria should be flexible and include ideas derived from human perception.

9.4 Rigidity

The distinguishing feature of perceptual motion structures is their rigidity. In conventional segmentation techniques this rigidity is inferred by measuring velocities — sufficiently similar velocities in the same region imply rigidity. The process described in this chapter is more direct — rigidity is the property we are interested in, and it is "estimated" by explicitly monitoring the relationships between simple features using the Delaunay graph representation of the image.

By definition a rigid object does not change shape or size. This is obviously not a very helpful definition in the context of a vision system where only a projection of the object is

available and all measurements are affected by noise in some way. If the effects of noise can be accurately characterised then a simple modification to the scheme would probably be effective in many situations. This approach is most likely to be useful in well controlled environments. The alternative investigated in this work is based on the human perception of rigidity and the perceptutal importance of moving objects. It is hoped that this would help to produce a more robust and versatile system.

Section 7.4 discussed the concept of perceptual importance and the reasons why conventional tracking schemes do not provide an estimate of perceptual importance. It was proposed that the human perception of importance is related to some form of uncertainty or error and the statistics of the local geometric structure. It will be assumed that the human perception of importance of a relationship between two features, i.e. the rigidity, involves similar quantities. Careful psychological experiments may assist in estimating the nature of the relationship more precisely, however such experiments are beyond the scope of this thesis. Instead, mechanisms will be developed to measure the parameters of interest, and very simple and arbitrary comparisons to human visual perception will be carried out.

9.5 The Segmentation System

The segmentation system consists of several stages:

- Generating the Delaunay graph (see Chapter 8).
- Estimating correspondence between salient points (graph vertices) in successive frames, and graph edges in successive frames.
- Evaluate an uncertainty measure describing each graph vertex and edge.
- Perform spatial interactions that utilise spatial information to reduce uncertainty.
- Estimate the important local geometric quantities.
- Make segmentation decisions.
- Stabilise the object representation over time.

9.5.1 Estimating point correspondence

The starting point for the segmentation process is an estimate of point correspondence between frames. The thresholding process eliminates a significant amount of data and provides

a mechanism to rapidly locate points, but the problem is not trivial.

There are many different ways in which correspondence may be estimated and maintained. If some form of velocity measure is available (for example from a region based scheme) then it could be used to predict the location of points in subsequent frames. The nearest points to the predicted locations could be quickly located using the Voronoi diagram. This could be regarded as a primitive form of data fusion.

The method used in this work involves two stages. The first stage predicts positions of points in a new frame using information from previous frames. This prediction is based on a constant displacement motion model. A displacement vector, indicating the per frame displacement of the salient point, is maintained for each salient point. The value for a new salient point is initialised to a default value based on directional information.

The displacement vector is used to predict a new location for the point in the new frame. It is assumed that the matching point is either the nearest point to the predicted location (which can be quickly located using the Voronoi diagram), or one of the Voronoi neighbours of that point (see Figure 9.1). The matching point is decided using an error metric. The error is computed for all of the possible matches located using the Voronoi diagram. The error for a pair of points in successive time frames is given by

$$E = |P_L - M_L| + C \tag{9.1}$$

where P_L is the predicted location, M_L is the measured location, and C is a penalty term. If the direction of motion of the two points is the same then $C = 0$, otherwise $C = 3$. The choice of $C = 3$ is completely arbitary, and was selected as a moderate fraction of the starting error, which was arbitarily set to 10. The salient point in the new frame that minimizes E is selected as a match. The value of the error, E, is maintained for each salient point.

The second stage checks all salient points in the new frame that were not matched during the first stage. If the unmatched salient point has a Voronoi neighbour that was successfully matched by the first stage, then the displacement vector is copied from that neighbour. If multiple Voronoi neighbours were matched, then the displacement vector is copied from the neighbour with the lowest error, E. The displacement vector is then used to find a matching point in the first frame, using exactly the same techniques as in the first stage.

There are many other cues that may be used when estimating point correspondence. The lengths and angles of Delaunay graph edges leaving a point would provide a more powerful feature vector that would help make matching more reliable. Improvements of this nature will not be investigated because the aim here is to use only the simplest features. A large improvement in system performance is potentially possible if results of the segmentation

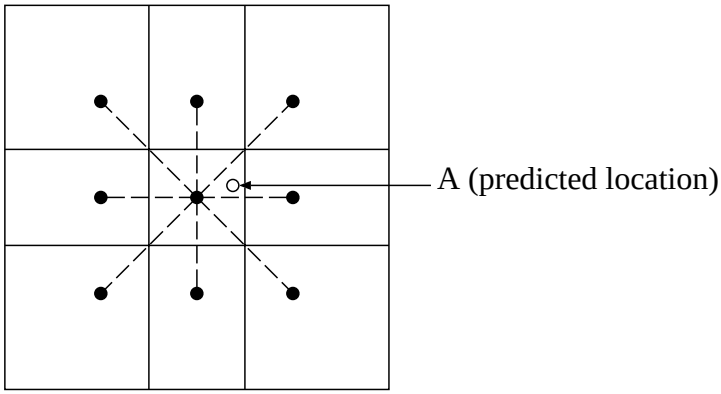


Figure 9.1: Initial correspondence assumption.

processing can be used to modify the predictions (feature space feedback). This will be discussed in more detail in Section 9.8.

9.5.2 Matching Delaunay graph edges

After an estimate of point correspondence has been made the process must be repeated for the Delaunay graph edges. The problem is now much simpler because constraints have been introduced by matching the endpoints of the edges.

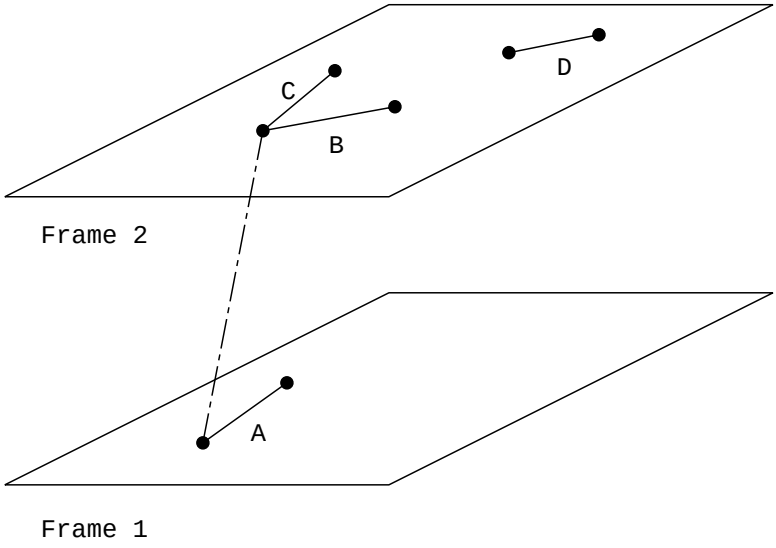


Figure 9.2: Constraints imposed on edge matching by the point matching process.

Consider Figure 9.2. The point matching process has connected the points in successive frames as indicated by the dashed line. When deciding on a match for graph edge A in Frame

1, it is only necessary to consider edges attached to the points matching the end points of A. Edges B and C are the only candidates. Edge D does not need to be considered.

An error metric is computed for pairs of Delaunay graph edges so that edge matching may be carried out in similar fashion to the point matching just described. The error metric is

$$E = \mid \vec{P} - \vec{N} \mid + C$$

where $\vec{P}$ is the edge vector in the previous frame and $\vec{N}$ is the edge vector in the new frame. C is the penalty term applied if the end points of $\vec{N}$ appear to be moving in different directions (i.e. they have motion detector responses with different signs)

It is possible that graph edges could be tracked instead of points. The correspondence problem for edges would therefore become similar to that for points. However, relationships between features would be tracked explicitly. Unfortunately no information about perceptual importance would be gathered for points. The second problem is that the Voronoi diagram data structures do not help to quickly locate graph edges directly.

9.5.3 Representing uncertainty about rigidity and tracks

It is important to form an estimate of uncertainty for use in this scheme. This work will use a temporal average of errors as the basic indicator. Many alternatives are possible, however this has the advantage of simplicity.

The errors calculated as part of the salient point and graph edge matching procedure are filtered over time to produce uncertainty estimates for all salient points and graph edges. The filter used to produce the uncertainty estimate is described by the following equation

$$\begin{aligned} \sigma' &= \sigma(1-\gamma) + \gamma E \text{ if } \sigma' > Q \\ \sigma' &= Q \text{ otherwise} \end{aligned} \tag{9.2}$$

where σ is the uncertainty for the feature in the previous frame, E is the error, $0 < \gamma < 1$ is the filter coefficient and Q is the measurement error. ($\gamma = 0.25$ was used in this work. An arbitary value of $Q = 0.05$ was used. This was really intended to eliminate the possibility of zero uncertainty in artificial scenes.) Thresholds were used to restrict the maximum uncertainty and large default values of σ were used for new objects. For salient points an arbitrary default of 10 was typical, while for graph edges the value used was $Max(10, length/2)$.

Multidimensional error metrics are an obvious extension to this scheme.

9.5.4 Spatial interactions

The system described so far produces uncertainty measures for each graph vertex (salient point) and graph edge. These uncertainty estimates develop over time. If the relationship between two salient points remains consistent then the uncertainty associated with that relationship will decrease with time. This means that the uncertainty of relationships in a large object will decrease at the same rate as those in a small object.

A large object represents more information than a small one, and it should therefore be possible to isolate a larger object more rapidly than a smaller one. This can be achieved by allowing spatial interactions (i.e. interactions between graph edges) to modify uncertainties. Therefore, the uncertainties in a large area of consistent spatial structure can fall more rapidly than those in a small area.

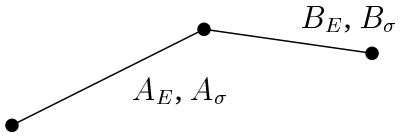


Figure 9.3: Spatial interactions.

The spatial interaction is very simple and takes place between neighbouring salient points and graph edges. Each graph edge and vertex possesses an error, E, and an uncertainty estimate, σ. (A_E indicates the error for graph edge A.) During a spatial interaction at iteration n, edge A in Figure 9.3 is able to influence the uncertainty of edge B. The reverse is not allowed to occur in the same iteration because it would allow the uncertainty of a single pair of consistent edges to fall very quickly. This is an interaction between an instaneous error and an uncertainty (a recursivey filtered error) of a neighbour. This interaction causes a more rapid decrease in uncertainty when neighbours are in agreement.

The interaction proceeds as follows. If $A_E < B_\sigma$ and $B_E < B_\sigma$ then $B_\sigma = kB_\sigma$, where $0 < k < 1$ is a decay factor that reduces the uncertainty of edge B. Equivalent interactions may also occur between edges and salient points (which also have error and uncertainty estimates). After a point or edge uncertainty has been modified by its neighbours, it is not allowed to influence any other edges or points during the same iteration. This prevents random pairs of consistent edges from having an excessive influence.

The average local length of graph edges discussed in Section 9.5.5 can be computed while the spatial interactions are being performed.

9.5.5 Local geometric quantities

Many statistical properties of the geometric structure of an image may be important to the human perception of importance of moving objects. Only the simplest and most obvious properties will be considered in this work (effectively a first order approximation). The two properties that will be assumed to influence the perception of importance of the relationship between salient points are:

- the distance between them, and
- the average inter-point distance in the local region.

The distance between neighbouring salient points can be easily calculated from the Delaunay graph.

The second property is equivalent to the local average spatial density of salient points. (It is possible that the rate of change of spatial feature density and other higher order properties could also be important.) This property can be calculated by applying a recursive spatial filter to the Delaunay graph. The computation proceeds as follows.

- Initialise the local average for each graph edge to the length of that edge.

Repeat the following for a number of iterations (usually 5 to 10 in this work).

- For each graph edge compute the average of the local average lengths (Na) of edges directly connected to that edge (see Figure 9.4).

$$Na = \frac{\sum_{i=1}^{j} La_i}{j}, \qquad i \in [\text{connected edges}]$$

  La is the local average graph edge length computed during the previous iteration.

- Update the local average for each edge using

$$La' = (La + Na)/2$$

  where La' is the local average edge length computed for the next iteration.

The graph edge length and local average graph edge length are the basic geometric quantities used in the segmentation process.

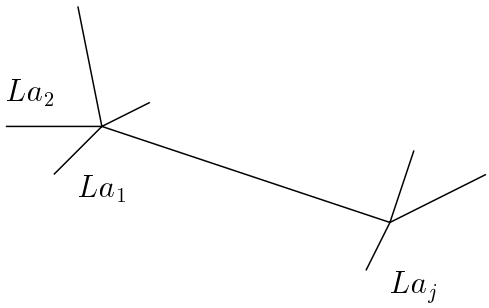


Figure 9.4: Computing the local average length.

9.5.6 Segmentation Decisions

The procedures just described extract the information required for segmentation decisions from the image sequence. The decision that must be made using this data is whether or not a Delaunay graph edge connects two points belonging to the same rigid object (or connects two objects moving at very similar velocities). One of the simplest ways in which this decision can be made is to compare the uncertainty associated with an edge to some function of the length of the edge. If the ratio $uncertainty/f(length)$ is less than some threshold then the edge could be considered to connect two parts of the same object. In this work it was found that a nonlinear function of length was desirable, and a logarithmic function was used. This is because the acceptable relative uncertainty (for humans) drops as the length of an edge increases. After a decision has been made for each edge, a graph traversal can be performed to isolate independent objects (discussed in Section 9.6). Such a simple decision criteria frequently produces incorrect results for a number of reasons, including:

- **Noise.** Image noise can reduce the certainty of edges connecting parts of the same object by changing the structure of the graph. Quantization noise will also be present, and can be a significant proportion of the uncertainty for short edges.

- **Connections between objects.** Graph edges connecting independently moving objects will have an uncertainty that is related to the difference in velocity of the objects. If the distance between the objects is large then the uncertainty could easily be low enough to satisfy the simple criteria mentioned above. In many cases this uncertainty could be comparable to the uncertainty caused by noise, making it difficult to select a threshold for the simple decision criteria that mimics human perception.

Some of the problems can be eliminated by using a higher level grouping process that is described in Section 9.6; however, it is also very useful to consider a slightly more complex

decision criterion. The simple criterion uses a definition of rigidity that does not include any information about the local region. This does not agree with the observations made about perceptual importance in Chapter 7. It is therefore necessary to consider how the local structure affects the perception of rigidity. The local structure tends to have a significant impact on the perception of rigidity at a discontinuity in average edge length. At a discontinuity in the length of edges (as tends to occur between objects) the connection criteria should be modified. A long connection must be significantly more certain than a short one in order to be perceived as rigid. Shorter edges can be less certain than they might be in a uniform environment, and still be perceived as rigid.

The criterion used therefore becomes

$$\sigma < k(\log(l) + 1)\frac{La}{l}$$

where l is the edge length, La is the local average edge length, σ is the uncertainty, and k is a weight ($k < 1$).

This may be considered as including a simple spatial cue. It seems that the perception of moving objects can be made far more stable by including some spatial information. This may indicate that motion and spatial information are not entirely separable in early visual processing. At the very least it does appear to be an appropriate way to eliminate problems with quantization noise.

The choice of k is dependent upon the noise in the image. A high value of k will produce a connection criteria that is too generous and will tend to connect independently moving objects. A low value of k will tend to break an object into its component parts. At present there is no way of selecting this weight automatically, however it is suggested that the stability of regions formed by the higher level process described in the next section may be a useful cue to help make this decision. Tests have indicated that values of k between 0.4 and 0.6 seem to be appropriate for most scenes.

9.6 Exploiting short range motion information

The scheme just described is intended to perform short range motion based segmentation. The scheme produces a set of Delaunay graph edges that satisfies the criteria for connecting two parts of the same object (or two objects that are moving together). In ideal circumstances independent objects could be isolated by finding all connected Delaunay graph edges that satisfy the criteria. A set of graph edges indicating an independent object can be located by starting at an edge that satisfies the criteria and following all connected edges that also meet

the criteria in a recursive fashion. The process of collecting all connected edges meeting the criteria will be called a *graph traversal*. In ideal circumstances one traversal will be required to isolate each object. The redundancy present in the graph structure helps to provide some robustness against noise, but the results of such traversals do not tend to remain consistent over time. Independent objects tend to be connected occasionally while single objects may be broken into several parts. A simple higher level scheme has been developed that maintains consistency over time and provides feedback to some of the earlier processing layers.

The scheme described in this sections forms regions over time, and these regions are used to limit and therefore stabilise the results of the graph traversal. The basic procedure is as follows.

- Update the position of the regions from the previous frame by using the displacement model.
- Create a new region by beginning graph traversals at several points within each old region. Average the newly created region with the previous one and update the age of the region.
- Any unvisited points remaining after this may belong to newly visible objects. Begin building a new region from these points by assuming they belonged to a region from the previous frame with default properties. A default region is simply a new region with an arbitary radius. In this work a radius equal to $1/6$ of the image size was chosen.
- Now check to see whether any regions should be merged. Various merge criteria are possible. The ones that were used in this chapter involved region overlap and relative age. Merging is necessary to quickly form larger objects.

9.6.1 Region Representation

A set of graph edges is not a partcularly convenient way in which to represent a spatial region. The main requirement for shape or region representation in this scheme is that the outline be readily available and the shape can be easily averaged over time. A radial map representation was selected. A radial map represents shape using an array of distances from a centre, and can represent concavities but not holes (see Figure 9.5). Two radial maps can be easily averaged by averaging the matching radii. The radial map representation of a set of points is computed by calculating the radius and angular position of each point relative to a

centre. The positions are used to index the array and the largest radius at each array position was retained. The radii of array positions with no corresponding point are computed by interpolating the nearest non zero neighbours. The centre of gravity of the endpoints of all graph edges contained in a region was used for the original centre of the radial map. After averaging two maps the centre of gravity was recalculated and the array values modified.

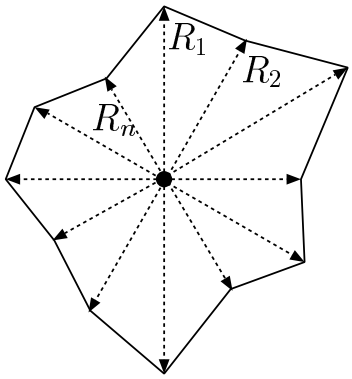


Figure 9.5: Radial map representation.

Smith also used radial maps to represent shapes for similar reasons, however in his scheme the convex hull of the cluster of points was used to define the outline [Smith 92]. This approach could have been used in this work. The other major difference is that there is a close coupling between the selection of the points forming the region and the representation of the region from the previous frame. In Smith's work the clustering process produces a set of points and the radial map is used to maintain the shape in a sensible way over time — no region merging is necessary. Smith's work also uses a more sophisticated region tracking procedure which is capable of correctly operating in the presence of occlusion.

9.6.2 Creating regions

The short range segmentation process described in this chapter is not guaranteed to produce isolated objects. This is not surprising since only motion information available from consecutive image frames is being used. Therefore, a traversal of all edges that meet the segmentation criteria will sometimes break a rigid object or connect two independently moving objects. In some simple situations a frame-based graph traversal may produce acceptable results, however experience has shown that real scenes processed in this way tend to produce unstable segmentation results.

The process of creating regions still involves the graph traversal, but the traversal is

limited by the extent of the corresponding region in the previous frame. The positions of all regions are updated using the same simple motion model as was used in the point matching process (constant displacement). A traversal of the graph commences from several points within each region. The aim of the traversal is to produce a set of points that are thought to belong to the same object. Points are collected by being visited during the graph traversal, and may only be visited if one of the edges leaving from or arriving at the point meets the connection criteria. If an edge leaves the region then the point outside the region is included in the set of points which will form the new region, but the traversal does not continue along that path and the point is not marked as visited. The outside point can therefore be visited during traversals beginning at other points. A traversal is completed when no more points are available to be visited. The set of points collected is used to create a new radial map which is averaged with the previous region representation.

Any points that are left unvisited after all of the regions from the previous frame have been processed may be the result of a new object becoming visible. The region formation procedure remains the same except the region that is used to limit the extent of the traversal is a default circular one. The radius is arbitrarily set at some fraction of the image size. This choice is a tradeoff between rapid formation of regions and getting stuck with large regions containing several objects.

The reason that points falling outside the region are included in the new region is to allow the region to grow. The averaging process prevents large changes in region shape and size between frames so that the edge leaving the region needs to satisfy the segmentation criteria for several frames for the region to change size significantly. This helps maintain stability of segmentation results.

The region is assigned motion parameters by averaging the motion of all of the points within the region. This result may also be averaged with the previous region. Any region comprising only two points is eliminated.

9.6.3 Averaging Regions

The radial map averaging process performs low pass filtering on pairs of shapes by low pass filtering corresponding elements of the radial map. The filter coefficients are time dependent so that shapes may be modified more quickly while “young” but be more difficult to modify when aged. An age threshold T_{age} is used as the point at which filter coefficients stop changing. This was usually set at 5 frames.

The filtering process is described by the following equation.

$$R'_{new}(i) = R_{old}(i) - c[R_{old}(i) - R_{new}(i)] \tag{9.3}$$

where $R_{old}(i)$ is the ith array element of the older region, $R_{new}(i)$ is the ith array element of the new region and $R'_{new}(i)$ is the ith filtered value of the new region.

The filter coefficient c is calculated as follows.

$$c = \begin{cases} k - lA_1 & \text{if } A_1 < T_{age} \\ m & \text{otherwise} \end{cases}$$

where A_1 is the age of the region, k, l and m are constants. In the examples presented in Section 9.7 these values are 0.6, 0.1 and 0.1, respectively.

Smith used a similar process, with additional terms to modify relative rates of expansion and contraction Smith [Smith 92].

9.6.4 Merging Regions

It is necessary to merge regions if the objects in the scene are larger than the default size provided. Merging regions allows large regions to form more quickly. Two criteria were used to test for a merge in this work. The first was that there must be some overlap along the radii connecting the centres of the two regions, see Figure 9.6. The second was that the regions must be "established" and the ages of the regions must be "similar". This prevents sudden changes to well established regions that could be caused by merging with randomly occurring regions.

- Don't merge if $age = 1$.
- Don't merge if $min(A_0, A_1)/max(A_0, A_1) < 0.4$, where A_0 and A_1 are the ages of the two regions being considered. If A_0 or A_1 is greater than T_{age} then replace it with T_{age}.

This is somewhat arbitrary, but seemed to function reasonably well.

At present these merge criteria do not include any motion information. This means that occluding objects will tend to be merged together due to the overlap of regions representing the objects. Motion information could be incorporated in the merge criteria, but has not been in this work.

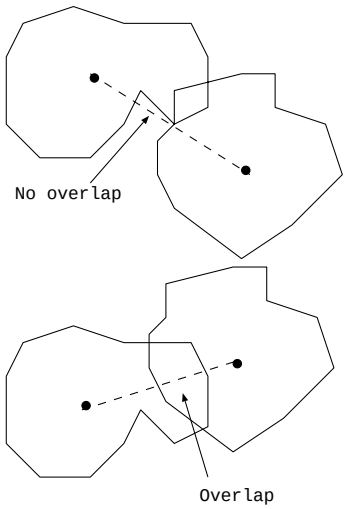


Figure 9.6: Overlap conditions for merging.

9.6.5 Other Options

The technique just described is region based. It is possible that other models could be used to maintain consistency over time. For example, a model of the path taken during the graph search may be more effective although more difficult to implement. This idea would probably help to eliminate the merging step in the procedure described above because connections between independent objects are unlikely to remain constant over time. Alternatively, a top down approach may also be appropriate. A top down process would break large regions into smaller ones rather than combine smaller ones into larger ones.

9.6.6 Uses of the segmentation results

Ideally the technique described should produce regions with shapes closely matching the outline of the moving objects. Graph edges connecting objects that occasionally staisfy the segmentation criteria, the discrete nature of the radial map and the averaging process will all decrease the accuracy of the outline. Despite this, the regions can be used for a number of things. For example, the region information can be used to provide feedback to the earlier processing stages. In this work the feedback operated to reduce the uncertainty of all graph edges that had both ends within the same region. This helps to maintain the consistency of moving objects. The velocity of objects can also be computed by tracking regions or by evaluating an average velocity of all salient points within a region. Feedback could also be used to modify the behaviour of the thresholding system by modifying the decay parameter

c. It is not clear what the appropriate criteria for modifcation of the thresholding proces should be, but some measure of region stability may be appropriate. If the "correct" balance between threshold level and region stability can be found then some savings in computational resources can be made. (If the threshold is too low then more salient points are produced so more computational resources are required to process the image.)

A motion based binding of regions could also be useful. As the long Delaunay graph edges connecting different objects need to have a high certainty to satisfy the connection criteria, it is possible that more perceptually realistic binding of objects may occur if similar processing is performed using the results of the region formation stage. When operating in this mode the motion processing system could be regarded as acting as a motion feature detector, rather than object segmentation process. (This can be easily achieved by reducing the value of the weight k.)

9.6.7 Ignoring the aperture problem

It is important to note that the techniques described in this chapter do not make any attempt to solve the aperture problem, and in most cases it seems to be unnecessary. The motion detectors that provide information to the segmentation system do not detect velocity, and only one orientation of detectors is being used. Thus, the detectors are only producing some measure of the horizontal component of motion. This is an even greater restriction than the usual aperture problem where the normal component of velocity is available. This means that the point tracking process will tend to be wrong when the motion is not horizontal, however this is not a major problem.

Consider the line shown in Figure 9.7 that is moving with a significant vertical component. The point matching process will begin with the assumption that motion is horizontal, since the detectors are oriented to detect horizontal motion. This means that the process predicting locations in the new frame will produce several incorrect matches (shown at the top of the image, near A). The process that looks back in time will produce similar errors (shown at the bottom of the image with dashed lines, near B). Since each point in the new frame can only maintain one match, and this will be the one with the lowest error, the end point of the line will tend to be matched to the point on the line in the old frame with the smallest vertical difference. The points on the other end of the line will be matched to the same point, which is allowable. The next step is matching graph edges. If the line is represented by a roughly uniform density of points then any graph edge oriented along the line could match any other with a low error. In this example, the edges at the bottom of the line in region B will all

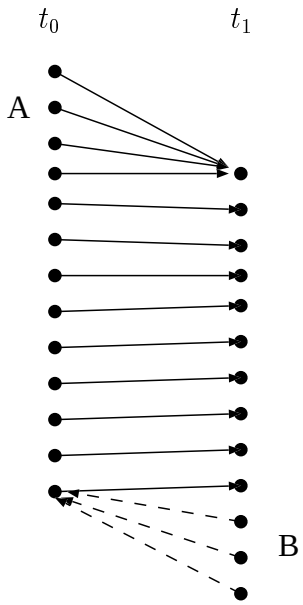


Figure 9.7: Avoiding the aperture problem. Lines show the results of the matching process.

be matched to the same graph edge. Since all of the graph edges on the new line will be matched to a similar edge on the old line, segmentation can still proceed corectly.

Errors can obviously still occur if nearby objects interfere with the matching process. If the results of the segmentation process are tracked, rather than the low level features, then accurate velocity estimates are possible. These results could be used to modify the prediction process so that the original matching is performed correctly. This is a potentially important advantage of performing segementation before velocity estimation.

9.7 Test Results

This section shows some test results for 4 different scenes after processing with the Delaunay graph matching scheme and the simple region averaging scheme. The first two tests demonstrate the response to random noise tests of the form typically used in demonstrations of human motion perception. The other two are real world scenes involving motion of cars. Some artifacts of the digitization process are visible in both of the real scenes. In the Hamburg taxi sequence (Figures 9.16 to 9.19) this is visible as a vertical line on the right side of the image while in the ambulance sequence (Figures 9.20 to 9.24) it is visible on the left. Animated (mpeg) versions of these results (plus some others) may be found on the CD provided with this thesis or on the world wide web at http://www.eleceng.adelaide.edu.au/

Personal/rbeare/Animations/index.html

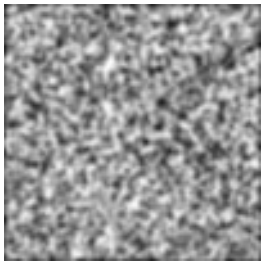
Each test result shows five images. The first is the unprocessed scene, the second is the motion detected sequence, the third is the thresholded version of the motion detected sequence. In the motion detection and thresholded images the colour indicates the direction of travel, with black indicating motion to the left and white motion to the right. The fourth image is the processed Delaunay graph representation derived from the neighbourhood thresholding — dark lines satisfy the connection criteria. The fifth image shows the results of simple region formation, with the brightness of lines representing region age (white regions are the oldest). The regions are overlaid on the original image.

9.7.1 Consistently moving object with random background

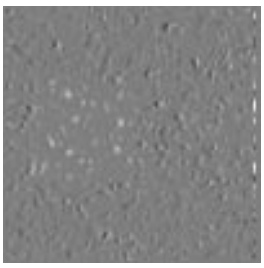
Figures 9.8 to 9.11 show the results of processing a randomly textured object moving in front of a statistically identical but randomly changing background texture. The image size is 100×100. In the stationary images that show the original scene (9.8(a), 9.9(a), 9.10(a) and 9.11(a)) the object is imperceptible. The position of the object can be determined by closely examining the motion detector response to find a region of consistent direction. This is the classical test used to demonstrate the importance of motion information. The object is moving from left to right and is quickly isolated from the background by the region formation process.

9.7.2 Two moving random textures

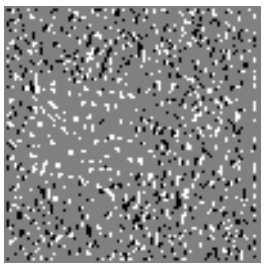
The second test (Figures 9.12 to 9.15) illustrates the separation of the same object from a background that is moving consistently at a different velocity. The image size is also 100×100. In this case the object is also segmented quickly, but the background is not grouped to form a single object (and the background regions are not formed as quickly). (Note that the radial representation would fail in such circumstances because it cannot support the notion of a hole which would be required to represent the foreground object inside the background). The reason that the background only forms small regions is that the texture is relatively dense while the velocity is quite high. This means that the assumptions used to initialise the tracking procedure are only valid in restricted areas where there is a significant discontinuity in texture density. The implications of this will be discussed in Section 9.8.



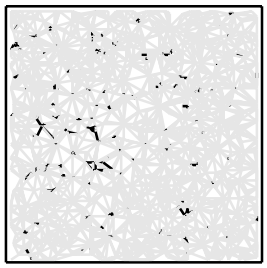
(a) Original image.



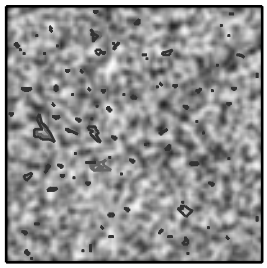
(b) Motion detected image.



(c) Thresholded image.

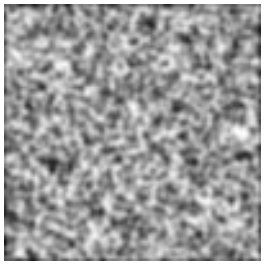


(d) Processed Delaunay graph.

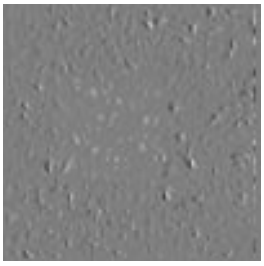


(e) Regions.

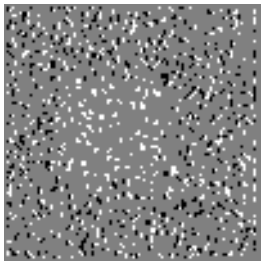
Figure 9.8: Random texture motion frame 5.



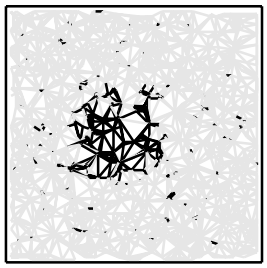
(a) Original image.



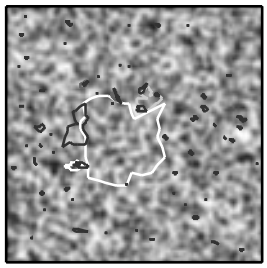
(b) Motion detected image.



(c) Thresholded image.



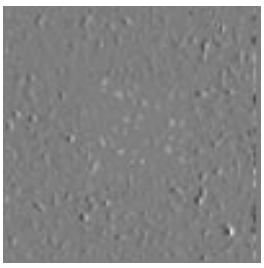
(d) Processed Delaunay graph.



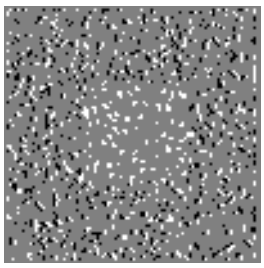
(e) Regions.

Figure 9.9: Random texture motion frame 15.

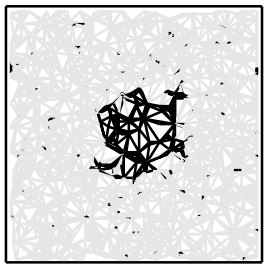
(a) Original image.



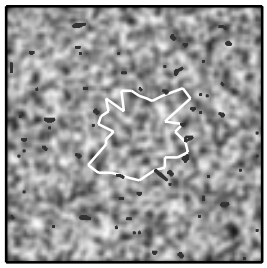
(b) Motion detected image.



(c) Thresholded image.

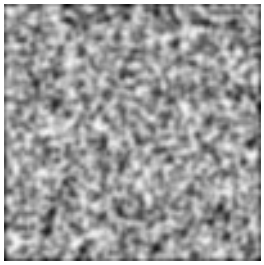


(d) Processed Delaunay graph.

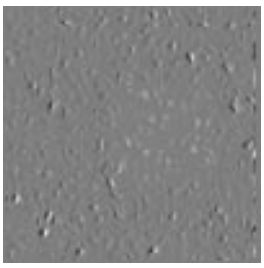


(e) Regions.

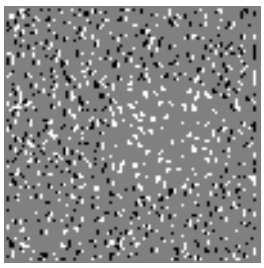
Figure 9.10: Random texture motion frame 25.



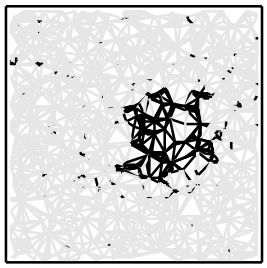
(a) Original image.



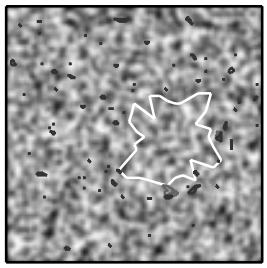
(b) Motion detected image.



(c) Thresholded image.

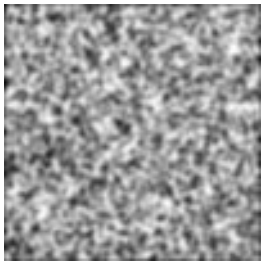


(d) Processed Delaunay graph.

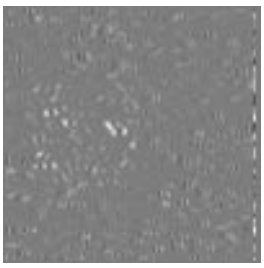


(e) Regions.

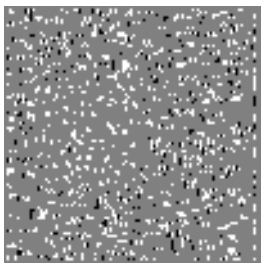
Figure 9.11: Random texture motion frame 35.



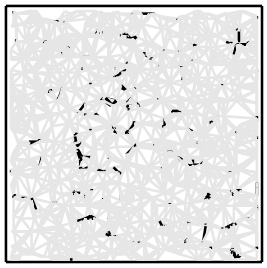
(a) Original image.



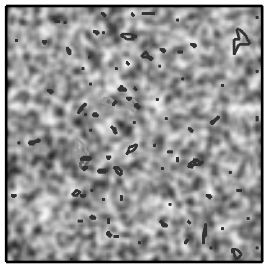
(b) Motion detected image.



(c) Thresholded image.

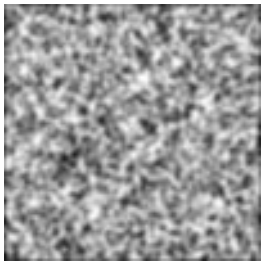


(d) Processed Delaunay graph.

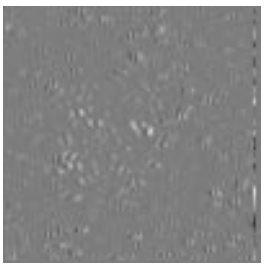


(e) Regions.

Figure 9.12: Consistent object and background texture motion frame 5.

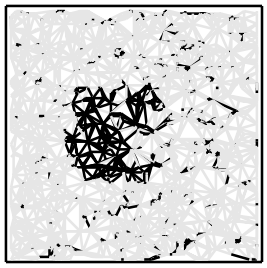


(a) Original image.

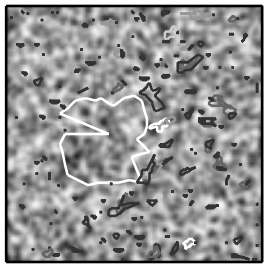


(b) Motion detected image.

(c) Thresholded image.

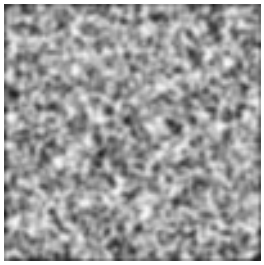


(d) Processed Delaunay graph.

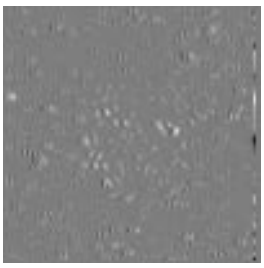


(e) Regions.

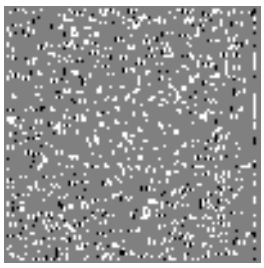
Figure 9.13: Consistent object and background texture motion frame 15.



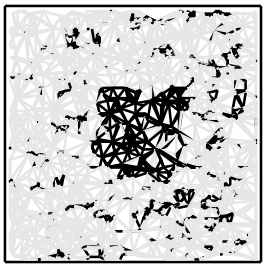
(a) Original image.



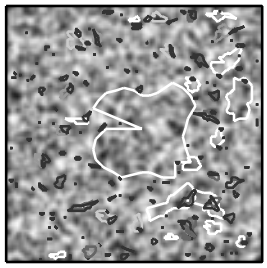
(b) Motion detected image.



(c) Thresholded image.

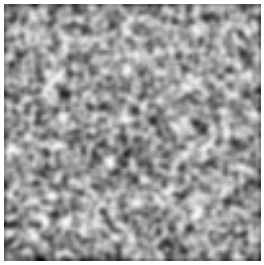


(d) Processed Delaunay graph.

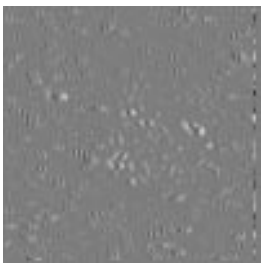


(e) Regions.

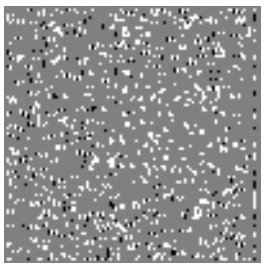
Figure 9.14: Consistent object and background texture motion frame 25.



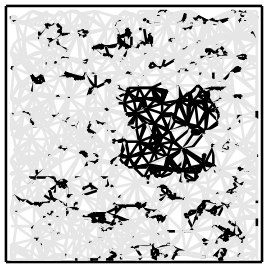
(a) Original image.



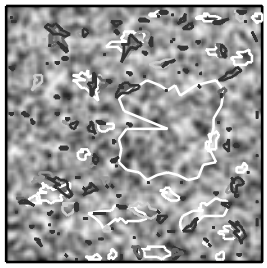
(b) Motion detected image.



(c) Thresholded image.



(d) Processed Delaunay graph.



(e) Regions.

Figure 9.15: Consistent object and background texture motion frame 35.

9.7.3 Hamburg Taxi sequence

The Hamburg taxi sequence (Figures 9.16 to 9.19, image size 256×190) contains four moving objects — 3 vehicles and a person (top left). The scene is quite noisy, with intensity oscillations being clearly visible under different colour maps. This noise has the effect of causing the motion detectors to respond very faintly to stationary edges, and this faint response is detected by the Voronoi thresholding process. Thus, some stationary objects are also isolated by the region formation procedure.

The results stabilise by the end of the sequence and show a range of different levels of performance. The central car, which has a high contrast and strong motion detector response, is well segmented. The person in the top left also has a strong motion detector response and is isolated from its surroundings by the region formation procedure.

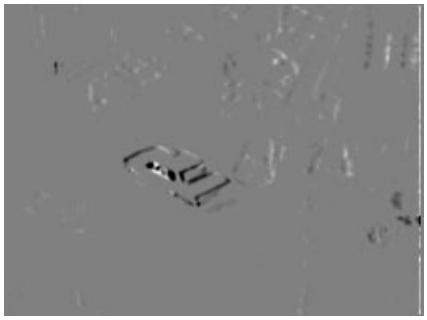
The other two vehicles are extremely low contrast (one is also obscured by a tree) and have very low motion detector response, yet some features are isolated by the segmentation process. The system represents the car on the left as a collection of five "blobs" which correspond to the major corners. The blobs do not get connected together because of the high levels of noise in the scene. In this situation a hierarchical region formation process may be more appropriate. The vehicle obscured by the tree is also represented in a similar fashion.

9.7.4 Ambulance sequence

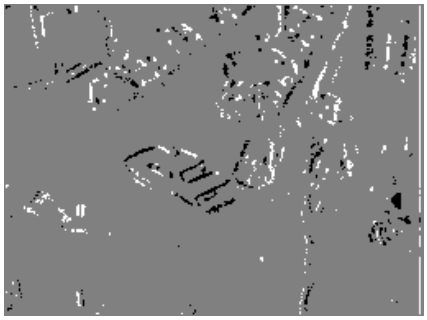
The ambulance sequence (Figures 9.20 to 9.24, image size 256×256) was provided by Stephen Smith, formerly of Keble College, University of Oxford. The sequence was taken from a *moving vehicle* and shows one vehicle to the left of screen (land-rover) that is moving at almost the same velocity as the camera vehicle, and a second vehicle (ambulance) which overtakes both the camera vehicle and the land-rover. This sequence obviously does possess a consistently moving background.

A number of interesting effects can be observed in this sequence. The most important is probably the consequences of not using motion information in the region merging process. In Figure 9.21(e) the region representing the land-rover is merged with regions in the background while in Figure 9.24(e) the two vehicles are merged together. Another interesting effect is the evolution of the region representing the ambulance. In Figure 9.20(e) the ambulance is represented by several regions of different ages, however by Figure 9.20(e) one region has emerged. The effects of noise in the background are also visible. The background is represented by several different regions that are not bound together because of the noisy

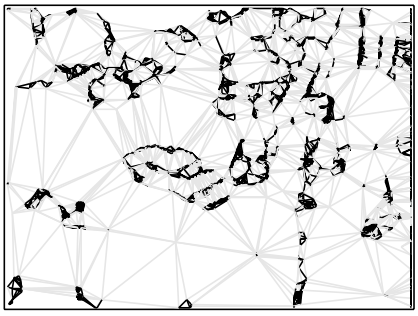
(a) Original image.



(b) Motion detected image.



(c) Thresholded image.

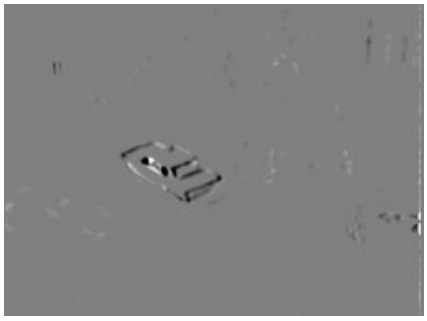


(d) Processed Delaunay graph.

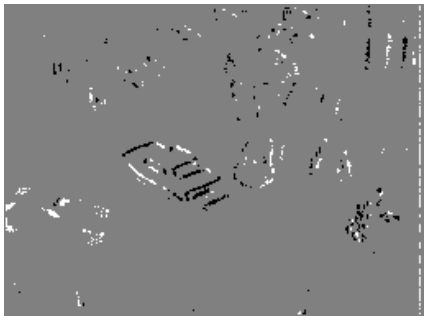
(e) Regions.

Figure 9.16: Hamburg Taxi frame 5.

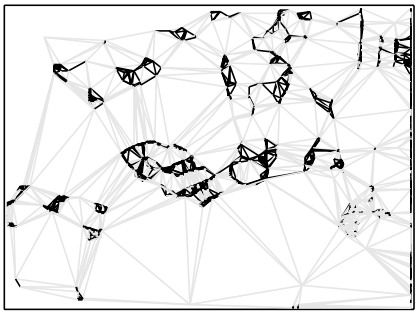
(a) Original image.



(b) Motion detected image.



(c) Thresholded image.

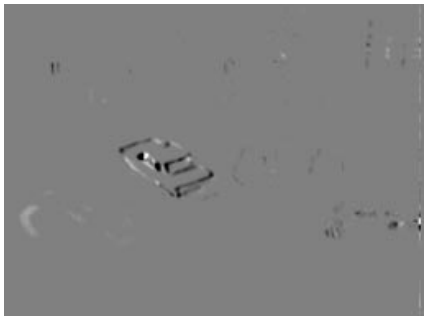


(d) Processed Delaunay graph.

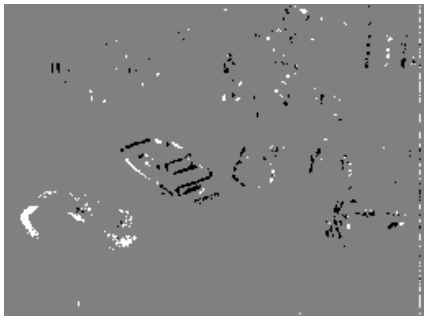
(e) Regions.

Figure 9.17: Hamburg Taxi frame 10.

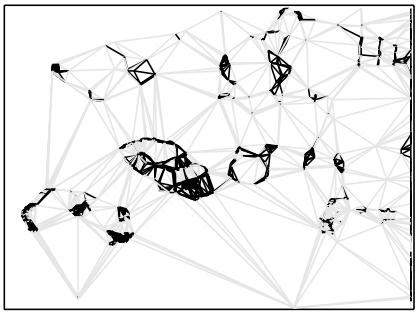
(a) Original image.



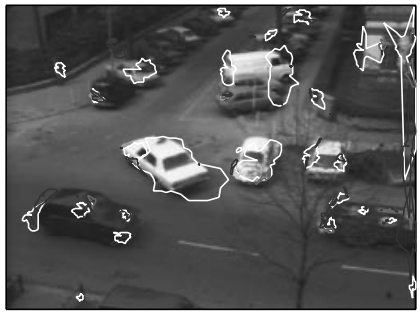
(b) Motion detected image.



(c) Thresholded image.



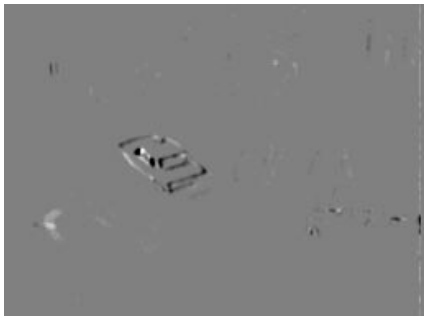
(d) Processed Delaunay graph.



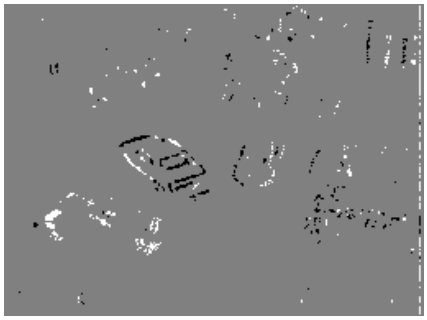
(e) Regions.

Figure 9.18: Hamburg Taxi frame 15.

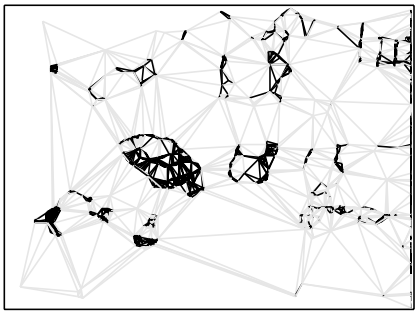
(a) Original image.



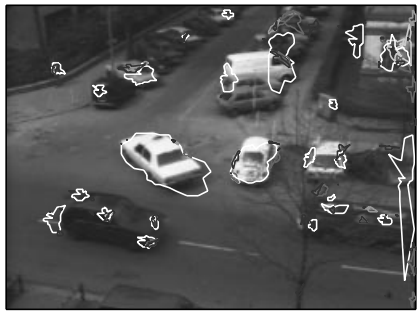
(b) Motion detected image.



(c) Thresholded image.



(d) Processed Delaunay graph.



(e) Regions.

Figure 9.19: Hamburg Taxi frame 20.

conditions. The road is not segmented in this sequence because the detectors are oriented to detect horizontal motion. The dominant motion of the road is vertical, so the response of motion detectors is very weak.

9.8 Failure Modes

The Delaunay graph structure provides a significant level of redundancy that helps improve robustness to noise. However, there are some situations where the effect of noise with a magnitude considerably less than the motion detector responses can be very significant. The effects of noise can manifest themselves in unexpected ways.

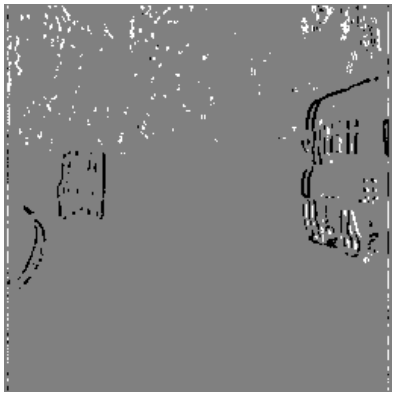
9.8.1 Interrupted graph structure

The most serious cause of failure results from extensive changes to the structure of the Delaunay graph. The graph structure may be disrupted if the location of salient points changes significantly. It is unlikely that noise will cause a substantial shift in location for bright image points, however it can cause disruption to the graph structure representing "sparse" objects by introducing additional salient points. If an object's surface is textured in a fashion that produces widely separated salient points after thresholding then the representation can be considered as "sparse". If even low noise levels are present then extra salient points can be introduced in the gaps. If the number of additional points is sufficiently high and their positions are unstable (as would be expected from noise), then the interconnection structure of the graph will be changed completely. This will produce connecting edges with a high uncertainty. The Voronoi thresholding process does not eliminate noise based on absolute value (unless some preprocessing is done, see Section 8.3.4). The thresholding is a function of both magnitude and distance from other significant points, so that low noise levels can produce salient points when isolated. This effect will obviously occur in otherwise empty backgrounds, but will be eliminated by the matching process. It can also occur in the interior of sparse objects and cause the problems just mentioned.

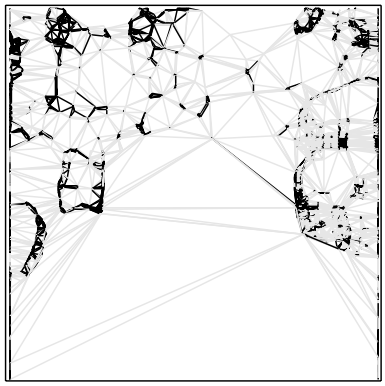
Fortunately most real objects do not produce a sparse representation. This is a result of the perceptual spatial structure possessed by most real objects. For example, moving lines will tend to produce dense distributions of responses and are therefore more difficult to disrupt in this way. The complexity of the interaction between the noise and the type of object being segmented makes the effects difficult to measure.

(a) Original image.

(b) Motion detected image.



(c) Thresholded image.



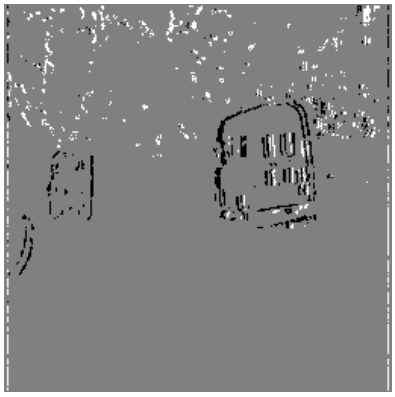
(d) Processed Delaunay graph.

(e) Regions.

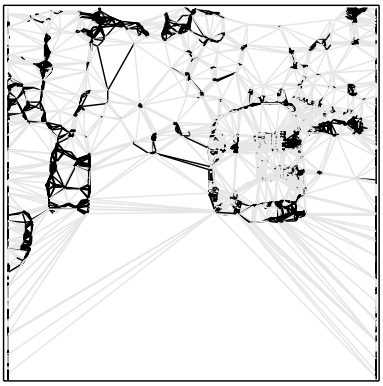
Figure 9.20: Ambulance frame 20.

(a) Original image.

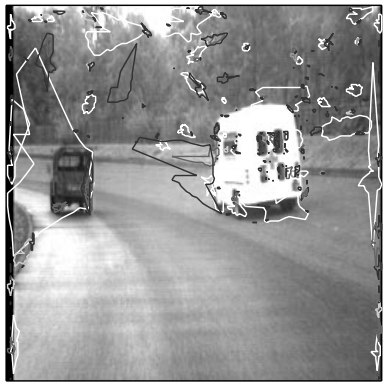
(b) Motion detected image.



(c) Thresholded image.



(d) Processed Delaunay graph.

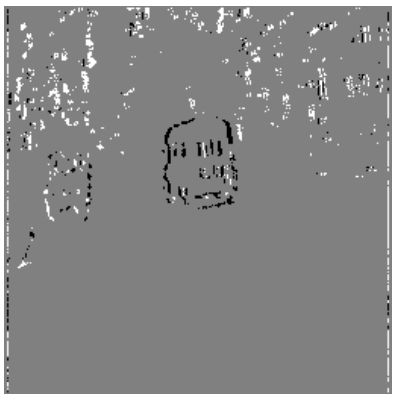


(e) Regions.

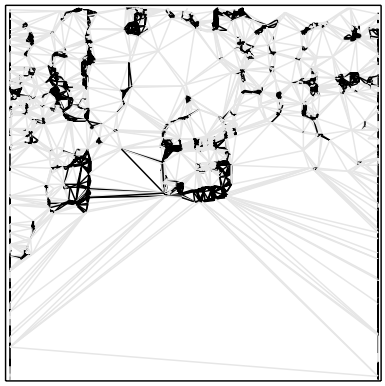
Figure 9.21: Ambulance frame 40.

(a) Original image.

(b) Motion detected image.



(c) Thresholded image.



(d) Processed Delaunay graph.

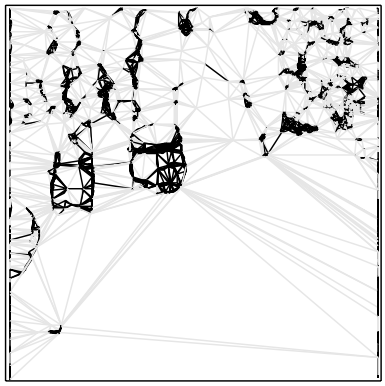
(e) Regions.

Figure 9.22: Ambulance frame 60.

(a) Original image.

(b) Motion detected image.

(c) Thresholded image.



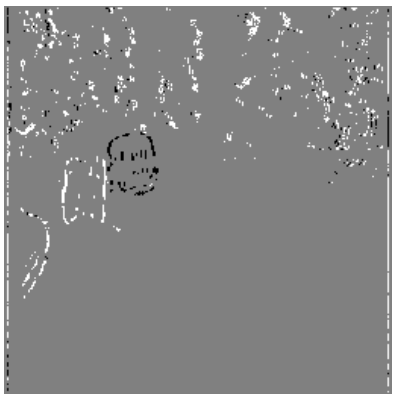
(d) Processed Delaunay graph.

(e) Regions.

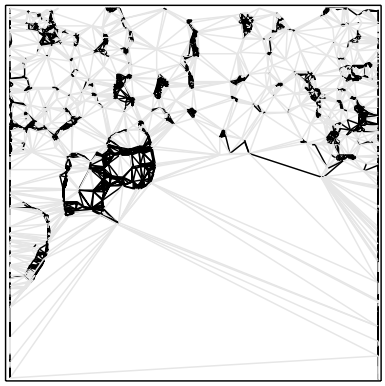
Figure 9.23: Ambulance frame 80.

(a) Original image.

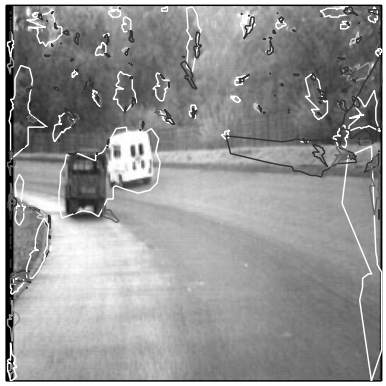
(b) Motion detected image.



(c) Thresholded image.



(d) Processed Delaunay graph.



(e) Regions.

Figure 9.24: Ambulance frame 100.

9.8.2 Tracking Failure

Another failure mode is caused by the simple tracking procedure. If the feature density and image velocity are sufficiently high then a form of aliasing can occur. In this situation the point matching stage will be wrong, resulting in graph edges with low certainty. This illustrates a useful property of the segmentation system. When errors are made in the early processing stage the subsequent stages fail gracefully by producing no response rather than producing ridiculous results.

Some heuristic techniques are possible to help eliminate the aliasing problem in many situations. The region formation system is a useful indicator of success for the early matching process, so the assumptions made that produced successful matches could be used by adjacent, unsuccessfully matched points. This procedure should produce a gradual propagation of correct information from successful areas (usually discontinuities) to the unsuccessful ones. If the assumptions are wrong then segmentations are unlikely to result. An alternative is to simply test different assumptions if no segmentation occurs in a particular region.

9.8.3 Merge Failure

The region merging process is also susceptible to a significant form of failure. As mentioned in Section 9.6.4 the region binding process is only using spatial information, rather than motion and spatial information. This is a serious problem that has not been addressed. The effect is that any regions that overlap due to occlusion will be merged. This is incorrect behaviour that could be corrected by including some higher level information, such as region motion information, into the merge criteria.

9.9 Discussion

This thesis has proposed the idea that segmentation should be treated as the primary goal in short range motion processing. Measures of consistency of relationships between features were used as the basic segmentation cues. This chapter has investigated a particular approach to implementing this idea. The results of this relatively simple technique demonstrate the promise of the idea. This section will discuss limitations and potential improvements of the existing system.

9.9.1 Performance optimisation

It is difficult to compare the performance of the approach described here to that of other motion processing techniques in a meaningful way. Not only is the system designed with different goals in mind and using different starting information, but the current implementation is also an unoptimised prototype. This software implementation was intended to be used as a tool to test the new concepts that have been introduced in this thesis.

The implementation used to test these ideas is certainly not an appropriate starting point for a real time software solution. In fact many of the tasks implemented in software were designed to be implemented by specialised hardware. It is unlikely that even the motion detection stage could be performed cheaply in software in real time. The motion detectors were designed with adaptive properties in mind and therefore must be considered as part of the imaging system in any real implementation. In the prototype system the motion detectors were simulated in software. The Voronoi thresholding scheme could also be performed by specialised hardware, but it may be reasonable to expect a careful software implementation to achieve real time operation on moderate image sizes. The Voronoi thresholding process is simple enough to make a specialised digital hardware implementation attractive.

Using the Voronoi neighbourhood structure to construct the Delaunay graph is the most time consuming preprocessing step in the prototype system, and is also the least optimised.

The stages of processing up to and including the formation of the Delaunay graph have been implemented in near real time on the MIT "Cog head". The Cog system uses a network of Texas Instruments 60MHz C40 DSP's and was easily capable of processing 15, 64×64 frames per second. The Cog "digital brain" is far more complex and expensive than would be desirable in any mobile application, but it does at least demonstrate that some parts of the system can be expected to run quickly in software without significant amounts of effort spent on optimization.

Another important criterion when building a real system is the representation of segmented objects. The prototype system is using simple representations to demonstrate that the principles work. If the ideas are used to form a component of a larger, more complex vision system then the representations required could be very different.

9.9.2 Alternatives

As stated earlier, the aim of the system described in this chapter is to test the idea of treating segmentation as the major goal of early vision. There are possibly many other approaches that could achieve similar results and may be considerably more computationally

efficient. One of the more interesting alternatives involves techniques originally used to track sophisticated models involving both shape and motion in a computationally efficient way [Isard and Blake 96]. The stochastic image sampling techniques used in this process could possibly be adapted to the more general task of segmenting rigid objects by utilising simple perceptual models of the type described here. It may be possible to perform the task without the costly preprocessing steps.

It is also possible that the matching could be performed in a more conventional fashion, but be made more flexible by using the Delaunay graph structure. For example, correlations could be performed in graph space rather than the usual pixel space. This could allow decisions about the suitable size of correlation regions to be made based on image structure and reduce the computational requirements of the correlation process. Alternatively the Delaunay graph of a more conventional feature set, such as corners, could be computed and processed in a similar way. More complex features like corners are less common than the features used in this work and therefore easier to match between frames. The reduction in complexity may make a cheap real time implementation more realistic.

In this chapter the Delaunay graph has been used to explicitly represent the relationship between neighbouring features and to provide a mechanism to quickly locate the neighbours. The work here has demonstrated the benefits of explicitly representing relationships between features, but the construction of the Delaunay graph to represent these relationships is computationally expensive. It is likely to be interesting to investigate other ways of representing the relationships between features. The connection criteria used for graph edges does seem to capture the perceptually relevant information so any new form of representation should also make similar information available. Other ways of representing the uncertainty values may also be worth investigating.

It is also possible that other types of visual cues could be included in the processing. One of the more important cues that is not being used at this stage is spatial information. Certain types of structure of salient points (lines and curves) and even differing densities of points in a stationary image are strong indicators for segmentation. This can be demonstrated by looking at the thresholded frame in Figure 9.18(c). No motion information is available to our visual system when this frame is observed in isolation, yet we can still segment the image based on spatial cues. The cars produce a much denser cluster of dots than the surroundings (as well as containing straight lines) and are therefore isolated from their surroundings by our visual system.

This thesis has addressed the processing of motion information and has therefore largely ignored other cues, however it is clear that much of the power and flexibility of the human

visual system is derived from the ability to combine many different cues in a powerful way (limited spatial cues were used as part of the decision criteria in Section 9.5.6, however no independent spatial processing is being done). The Delaunay graph structure does seem to be a useful starting point for extracting this kind of cue and processing both spatial and motion cues at the same time does seem to have the potential to produce a robust system. The spatial cues of interest would be perceptual spatial structures of the form discussed in Chapter 7.

9.9.3 Alternative design options and applications

The second half of this thesis has investigated a unique method of processing the output of elementary motion detectors. These detectors were designed to provide an indication of direction and to provide adaptive properties. The processing methods have made only limited use of the directional information provided by the detectors, in the form of penalties in the matching process and in the initialisation of the matching process. However the most important information used by the segmentation processing is structural information. It seems likely that the same processing could be successful when using the response of temporal bandpass filters instead of motion detectors. Obviously a more complex process would be required to estimate point correspondence because no directional information would be available. Similar penalty criteria could probably be used in the correspondence estimation process. The reduction in the complexity of the front end system would have to outweigh the increase in complexity of the point tracking process.

There are other potentially interesting applications for some of the techniques discussed here. As mentioned earlier, it is likely that the Voronoi thresholding scheme and associated Delaunay graph structure could be useful in extracting perceptual spatial structures like smooth curves from local edge detection responses. The ability to extract such information could be useful for low bandwidth video systems.

The motion detectors used for the preprocessing system are only oriented to detect horizontal motion. This is obviously going to be inadequate in situations where significant vertical motion is expected. The problem of combining horizontal and vertical motion detector information has not been investigated, but there are a number of options:

- Perform completely independent processing on horizontal and vertical information and combine the results of the segmentation processing.
- Use the two orthogonal detectors to produce a vector description of the motion and use the extra information in more sophisticated tracking schemes and penalty terms.

It is also important to note that the information relating to the perceptual importance of moving points has been made available, but is not displayed in any of the tests in this chapter. The information about the perceptual importance of the tracks of individual salient points is only likely to be relevant when attempting to segment very small objects. This work has addressed the problem of segmenting larger objects, where the relationships between salient points are more important. The error and uncertainty information about each point is used by the spatial interaction processing and could be used to isolate single important points, so the system does satisfy the criteria outlined in Section 9.3.

The computationally intensive parts of the system are the Voronoi thresholding, Delaunay graph formation and spatial interaction steps. All of these steps can be performed in a parallel fashion on appropriate hardware, although there is no real reason to assume that custom digital serial hardware would not be sufficient. Although the steps can be considered as a highly parallel computation, the structures and representations used do not appear to be plausible for neural implementation.

9.10 Conclusion

This chapter has presented a technique to perform segmentation based upon short range motion information and results from both real and artificial image sequences have been shown. The technique demonstrates the potential of treating segmentation as the fundamental goal of the early visual processing system and introduces a number of ideas and representations that could be useful in other forms of image processing.

The technique was successful in demonstrating that explicitly representing relationships between simple features (locally important motion detector responses) and using decision criteria based upon simple models of human perception can be successful in segmenting moving objects in real scenes. Accurate and sophisticated tracking schemes are therefore not an essential part of an artificial early visual system.

Chapter 10

Conclusions and further work

10.1 Summary

This thesis was divided into two parts. The first part addressed the problem of designing local motion detectors. Chapter 2 reviewed the use of motion information by biological systems and described some of the techniques that can be used to extract motion information from the environment. Chapter 3 described the criteria for useful local motion detectors and showed that existing architectures are unsatisfactory. Chapter 4 introduced several biologically inspired, adaptive filter elements that can be used to design useful versions of the existing motion detection architectures. Chapter 5 described a new local motion detection system, called the *directionally sensitive local inhibitory motion detector* (DSLIMD), that can be used as an adaptive early visual processing layer. The DSLIMD also mimics many of the behaviours observed in biological motion detection neurons.

The second part of the thesis develops a technique for performing scene segmentation using the responses of local motion detectors. Existing techniques for velocity estimation and segmentation were reviewed in Chapter 6 and the segmentation problem was reformulated. Chapter 7 introduced the notions of perceptual importance and perceptual motion structures, which are the fundamental ideas behind the new segmentation system. Chapter 8 described a new thresholding technique, called Voronoi thresholding, that produces a graph representation of images in a scene independent fashion. The segmentation system that utilises this graph structure is described in Chapter 9. The results of the processing on real and artificial scenes are also shown.

10.2 Future Work

The motion detectors designed in the first half of the thesis are intended to perform the first layer of visual sensing and therefore need to be implemented in hardware. The possible VLSI implementations for the different architectures need to be explored to determine whether the combination of adaptive and computational properties provide real benefits in terms of performance and silicon area. It is likely that the steady state version of the shunting inhibitory based systems will be simpler to implement, and floating gate technology offers some circuits that could be useful for this.

The postprocessing algorithm for segmentation developed in this thesis demonstrates the usefulness of some new ideas but is unlikely to be useful in its current form. The explicit representation of relationships between simple features proved to be a useful mechanism in segmentation and should be investigated further. Tracking of relationships between more complex and robust features could prove to be simpler and therefore more useful for a real system. Representations other than the Delaunay graph may also be more realistic. The system must be modified to make a compact and real time version practical. Such a system would also help to answer the important question of whether the segmentation results are useful to an artificial system.

The other important computational task that must be investigated before artificial vision systems approach the flexibility and robustness of biological systems is data fusion. Biological visual systems seamlessly combine many cues and sources of information to produce the required information in a reliable way. The processes used are not well understood but are likely to be very valuable to designers of many kinds of artificial information processing systems.

10.3 Closing Comments

Biological visual systems process information in ways we do not fully understand. The aim throughout this thesis has been to develop components and techniques that help to improve the robustness and flexibility of artificial vision systems using lessons learnt from biological systems. The matching of adaptive mechanisms to the computations in the early visual process were attempting to make more efficient use of hardware. The segmentation techniques attempted to use a minimal amount of *a priori* knowledge and when *a priori* knowledge was required it was derived from a simple understanding of human perception. It is hoped that as the technology and understanding of biological vision systems improve, the lessons learnt

during this thesis can help make the implementation of practical vision systems a reality.

Appendix A

Modelling noise performance of simple motion detectors

A.1 Introduction

This Appendix presents an analysis of the noise performance of three biologically inspired motion detection architectures. These are the Reichardt, or Correlation detector, the feedforward shunting inhibitory detector, and the feedback shunting inhibitory detector. The Correlation model is probably the most popular and was developed by Hassentein and Reichardt [Hassentein and Reichardt 56] as a result of behavioural experiments on insects. Since then neurophysical experiments have identified inhibitory mechanisms as being responsible for motion detection in mamals and perhaps even in insects. This has lead to a number of models employing shunting inhibitory neurons being proposed. These models can mimic the same behavioural effects as the correlation model, but have the advantage of demonstrating adaptive properties with mean luminance [Bouzerdoum and Pinter 89, Bouzerdoum 91].

The performance of the different architectures in the presence of noise has been largely ignored. A simple analysis of the 3 architectures is presented here.

A.2 Analysis

A.2.1 Feedforward Inhibition

The systems based on shunting inhibition will be analysed using perturbation expansions. The result of the perturbation expansion describes a linearised version of the system about the mean luminance value. This linearised system consists of a cascade of linear filters. The

filter stages are separated by nonlinear interactions between signals. The following analysis for the feedforward inhibitory detector (Figure A.1) is a summary from [Bouzerdoum 91].

Shunting inhibition is described by the following nonlinear differential equation.

$$\dot{m}_i = L_i(t) - am_i(t) - m_i(t) \sum_i k_i f(x_k) \tag{A.1}$$

where x_k is the inhibitory input, and k_i is the weight.

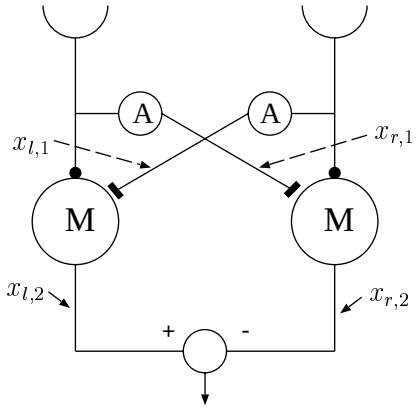


Figure A.1: Feedforward Inhibitory detector

The feedforward inhibition motion detection system shown in Figure A.1 is described by the following set of equations

$$\begin{aligned} \dot{x}_{\xi,1} &= L_p(t) - a_1 x_{\xi,1} \\ \dot{x}_{\xi,2} &= L_q(t) - a_2 x_{\xi,1} - kf(x_{\xi,1}) x_{\xi,2} \end{aligned} \tag{A.2}$$

where $\xi = l, r$, depending on whether the left or right subunit is being described. If $\xi = r$, $p = 1, q = 2$, otherwise $p = 2, q = 1$. The first part of the equation describes the linear delay filter (A in the figure) while the second part describes the shunting inhibitory neuron (M in the figure). The perturbation expansion of the system gives

$$\begin{aligned} x_{\xi,1} &= y_0 + cy_{\xi,0} \\ x_{\xi,2} &= z_0 + cz_{\xi,1} + c^2 z_{\xi,2} + \cdots \end{aligned} \tag{A.3}$$

for input of the form

$$L_i(t) = L_0 + c\ell_i(t)$$

Differentiating the result of the perturbation expansion, substituting for $f(x_{\xi,1})$ its Taylor series and equating coefficients of c gives the following system of equations.

$$\begin{aligned}
\dot{y}_{\xi,1} &= \ell_p(t) - a_1 y_{\xi,1} \\
\dot{z}_{\xi,1} &= \ell_q(t) - kf'(y_0) y_{\xi,1} z_0 - \alpha z_{\xi,1} \\
\dot{z}_{\xi,2} &= -\frac{kf''(y_0)}{2!} y^2 \xi, 1 z_0 - kf'(y_0) y_{\xi,1} z_{\xi,1} - \alpha z_{\xi,2} \\
&\vdots \\
\dot{z}_{\xi,n} &= -k \sum_{j=1}^{n} \frac{f^j(y_0)}{j!} y^j_{\xi,1} z_{\xi,n-j} - \alpha z_{\xi,n}
\end{aligned} \tag{A.4}$$

where $y_0 = L_0/a_1, z_0 = L_0/\alpha$ and $\alpha = a_2 + kf(y_0)$.

This system of equations describes a cascade of linear filters. For the purposes of noise analysis a simple system can be used. This linearised system is shown in Figure A.2.

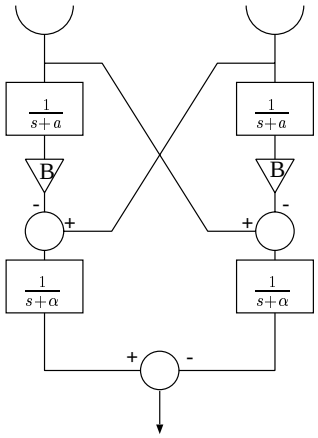


Figure A.2: First order approximation of Feedforward Inhibitory Detector

The transfer function describing the path from a single input to the output is given by

$$H_{total} = H_2 + BH_1H_2 \tag{A.5}$$

where $H_1(s) = 1/(s+a)$ is the transfer function of the linear delay filter, $H_2(s) = 1/(s+\alpha)$ is the transfer function of the first order approximation of the shunting neuron, and $B = kf'(y_0)z_0$ is a gain factor.

Assuming that the noise at the receptors is white and independent, then the total noise power seen at the output of the system is given by

$$\sigma^2_{total} = (\sigma^2_1 + \sigma^2_2) \int_{-\infty}^{+\infty} \mid H_2 + BH_1H_2 \mid^2 d\omega$$

where σ_1^2 and σ_2^2 are the noise power spectral densities at the motion detector inputs. The result of the integration is

$$\sigma_{total}^2 = (\sigma_1^2 + \sigma_2^2)\frac{(a_1^2 + a_1\alpha + 2a_1B + B^2)}{2a_1\alpha(a_1 + \alpha)}$$

If the noise power spectral densities at adjacent inputs are the same ($\sigma_1 = \sigma_2 = \sigma_{input}$), then the total output power is

$$\sigma_{total}^2 = \sigma_{input}^2\frac{(a_1^2 + a_1\alpha + 2a_1B + B^2)}{a_1\alpha(a_1 + \alpha)} \tag{A.6}$$

The perturbation expansion gives more accurate results when the input variation is small in comparison to the mean input luminance.

A.2.2 Feedback Inhibition

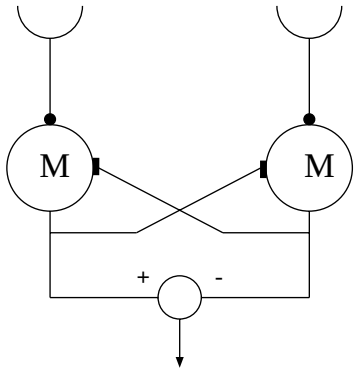


Figure A.3: Feedback Inhibitory Detector

The analysis of the feedback inhibitory detector shown in Figure A.3 is similar to that of the feedforward method, also employing a perturbation approach.

A perturbation expansion gives

$$m_i = z_0 + cz_{i,1} + c^2z_{i,2} + \ldots \tag{A.7}$$

Taking a 3rd order Taylor series expansion of $f(x_k)$ about z_0 gives

$$f(z_0 + cz_{i,1} + c^2z_{i,2}) = f(z_0) + f'(z_0)(cz_{i,1} + c^2z_{i,2}) + \frac{c^2z_{j,1}^2f''(z_0)}{2} + \ldots \tag{A.8}$$

Input Stimulus is of the form

$$L(t) = L_0 + c\ell_i(t) \tag{A.9}$$

Substituting equations A.7 and A.8 into A.1 and equating coefficients gives

$$\begin{aligned} \dot{z}_{l,1} &= \ell_l(t) - \alpha z_{l,1}(t) - B z_{r,1}(t) \\ \dot{z}_{l,2} &= -\alpha z_{l,2}(t) - B z_{r,2}(t) - \beta z_{l,1}(t) z_{r,1}(t) - \frac{k f''(z_0) z_{r,1}^2(t)}{2} \end{aligned} \tag{A.10}$$

and

$$\begin{aligned} \dot{z}_{r,1} &= \ell_r(t) - \alpha z_{r,1}(t) - B z_{l,1}(t) \\ \dot{z}_{r,2} &= -\alpha z_{r,2}(t) - B z_{l,2}(t) - \beta z_{l,1}(t) z_{r,1}(t) - \frac{k f''(z_0) z_{l,1}^2(t)}{2} \end{aligned} \tag{A.11}$$

where

$$\begin{aligned} \alpha &= a + k f(z_0) \\ z_0 &= \frac{L_0}{\alpha} \\ \beta &= k f'(z_0) \\ B &= \beta z_0 \end{aligned} \tag{A.12}$$

This equation also describes a system that is a cascade of linear filters, however the organisation is different to the previous case. Figure A.4 shows the first order approximation that will be used for the noise analysis.

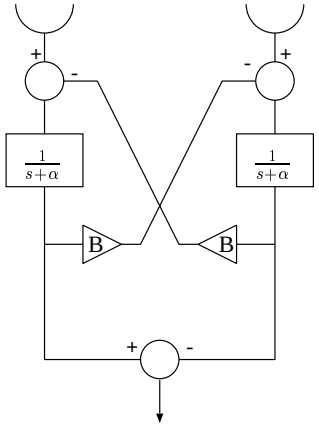


Figure A.4: First order approximation of Feedback Inhibitory Detector

The output noise power can be calculated in a similar way to that of the feedforward model. In this case the net transfer function is

$$H_{total}(s) = H_1(s) - H_2(s) \tag{A.13}$$

where

$$H_1(s) = \frac{\alpha + s}{(s+\alpha)^2 - B^2}$$

and

$$H_2(s) = \frac{-B}{(s+\alpha)^2 - B^2}$$

The output noise power is given by

$$\sigma^2_{total} = (\sigma^2_1 + \sigma^2_2) \int_{-\infty}^{+\infty} \mid H_{total} \mid^2 d\omega \tag{A.14}$$

resulting in

$$\sigma^2_{total} = \frac{\sigma^2_1 + \sigma^2_2}{2 \mid B - \alpha \mid}$$

If $\sigma_1 = \sigma_2 = \sigma_{input}$, then the total power is

$$\sigma^2_{total} = \frac{\sigma^2_{input}}{\mid B - \alpha \mid} \tag{A.15}$$

A.2.3 Correlation Model

The analysis of the correlation detector (Figure A.5) is relatively simple.

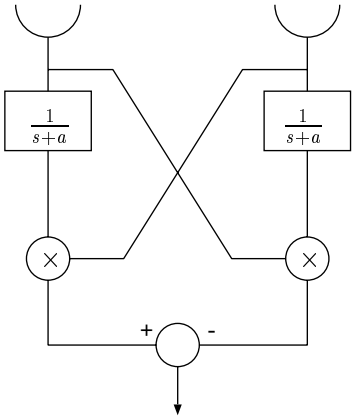


Figure A.5: Correlation Detector

$$\begin{aligned} x(t) &= (L_2(t) + n_2(t))(y_1(t) + v_1(t)) - (L_1(t) + n_1(t))(y_2(t) + v_2(t)) \\ x(t+\tau) &= (L_2(t+\tau) + n_2(t+\tau))(y_1(t+\tau) + v_1(t+\tau)) - \\ & \quad (L_1(t+\tau) + n_1(t+\tau))(y_2(t+\tau) + v_2(t+\tau)) \end{aligned} \tag{A.16}$$

where $L_1(t)$ and $L_2(t)$ are the input signals, $n_1(t)$ and $n_2(t)$ are the input noise signals and $v_1(t)$ and $v_2(t)$ are the noise outputs from the delay filters. $x(t)$ is the output of the detector and $y(t)$ is the response of the delay element.

$$
\begin{aligned}
E[x(t)x(t+\tau)] &= E[L_1(t)L_2(t+\tau)v_1(t)v_2(t+\tau) + \\
&\quad n_2(t)n_2(t+\tau)v_1(t)v_1(t+\tau) - \\
&\quad v_1(t)n_1(t+\tau)n_2(t)v_2(t+\tau) + \\
&\quad L_1(t)L_1(t+\tau)v_2(t)v_2(t+\tau) + \\
&\quad n_1(t)n_1(t+\tau)v_2(t)v_2(t+\tau) - \\
&\quad L_1(t+\tau)y_1(t)n_2(t)v_2(t+\tau) - \\
&\quad L_1(t)y_1(t+\tau)n_2(t+\tau)v_2(t+\tau) + \\
&\quad n_2(t)n_2(t+\tau)y_1(t)y_1(t+\tau) - \\
&\quad y_2(t)L_2(t+\tau)n_1(t)v_1(t+\tau) - \\
&\quad L_1(t)L_2(t+\tau)y_1(t+\tau)y_2(t) - \\
&\quad L_2(t)y_2(t+\tau)n_1(t+\tau)v_1(t) - \\
&\quad L_1(t+\tau)L_2(t)y_1(t)y_2(t+\tau) + \\
&\quad L_1(t)L_1(t+\tau)y_2(t)y_2(t+\tau) \\
&\quad y_2(t)y_2(t+\tau)n_1(t)n_1(t+\tau)] \\
&= E[y^2n_1(t)n_1(t+\tau) + y^2n_2(t)n_2(t+\tau) + \\
&\quad L^2v_1(t)v_1(t+\tau) + L^2v_2(t)v_2(t+\tau) + \\
&\quad n_1(t)n_1(t+\tau)v_2(t)v_2(t+\tau) + \\
&\quad n_2(t)n_2(t+\tau)v_1(t)v_1(t+\tau) - \\
&\quad 2n_1(t)v_1(t+\tau)n_2(t)v_t(t+\tau) - \\
&\quad 2Lyn_1(t)v_1(t+\tau) - 2Lyn_2(t)v_2(t+\tau)] \\
&= y^2\sigma_1^2 + y^2\sigma_2^2 + L^2\gamma_1^2 + L^2\gamma_2^2 + \sigma_1^2\gamma_1^2 + \sigma_2^2\gamma_1^2 - \\
&\quad 2E[n_1v_1]e[n_2v_2] - 2LyE[n_1v_1] - 2LyE[n_2v_2] \qquad \text{(A.17)}
\end{aligned}
$$

L is the mean luminance.

If $\sigma_1 = \sigma_2 = \sigma$ then

$$E[x(t)x(t+\tau)] = 2y^2\sigma^2 + 2L^2\gamma^2 + 2\sigma^2\gamma^2 - 2E^2[n_1v_1] - 4LyE[n_1v_1] \quad \text{(A.18)}$$

where σ^2 is the input noise power spectral density and γ is the noise power spectral density at the output of the delay filter.

In most situations the first two terms that involve the mean luminance dominate. Essentially this means that the multiplication is acting as a gain stage for the noise. Generally speaking the noise power term is less significant. At very low mean luminance values these terms may become more significant. The simple approximation for output noise power is

$$\sigma^2_{total} = 2L^2\sigma^2(\frac{1}{2a} + \frac{1}{a^2}) + \frac{\sigma^4}{a} \quad \text{(A.19)}$$

A.3 Noise comparisons

The most significant source of noise in the visual environment is shot noise. Shot noise has a square root dependence on the number of particles involved in a measurement. The models just described were used to compute the signal output power and changes in signal to noise ratio for the three motion detection systems in response to noise of this type are shown in Figure A.6. These graphs are using a very simple model of signal power and are therefore only useful to compare the relative performance of the detector systems. It is obvious from the graphs that the correlation model does not perform as well as the other two systems.

Note that the delay filters used in this analysis do not have a gain of 1, as was used elsewhere in the thesis. Both the correlation model and feedforward inhibitory model used the same filters and the analysis remains the same if the filters are of the form $a/(s+a)$.

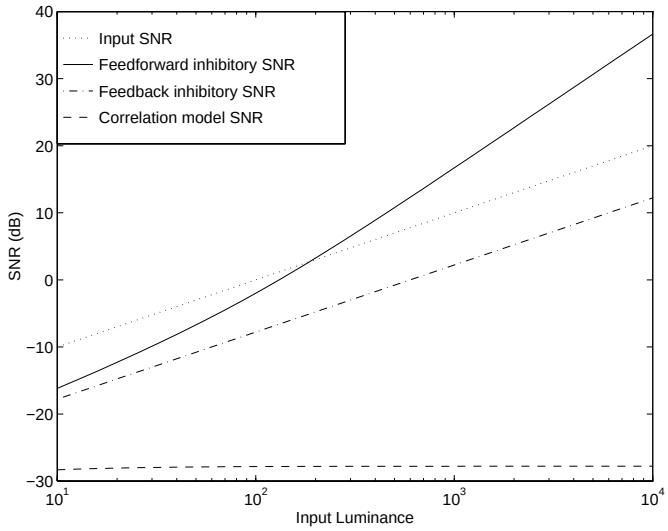


Figure A.6: Signal to noise ratios for different motion detector architectures. Note that the absolute magnitude of the input SNR is not meaningful. It is illustrated to demonstrate the dependence on mean luminance.

Appendix B

Bandpass filter linearisation

The perturbation expansion described in the previous appendix was also used to develop a linearised model of the adaptive bandpass filter described in Section 4.2. The filter is shown in Figure B.1. The linearisation was performed separately for each feedback subunit of the filter (Figure B.2).

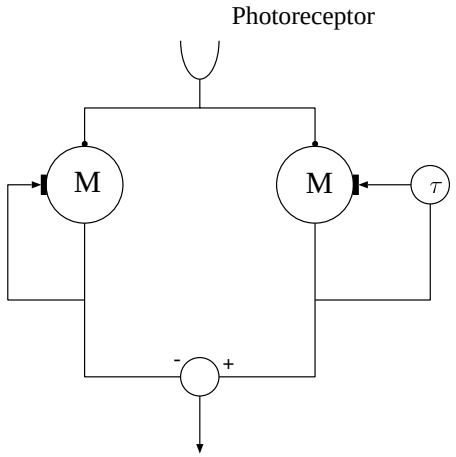


Figure B.1: Adaptive bandpass filter

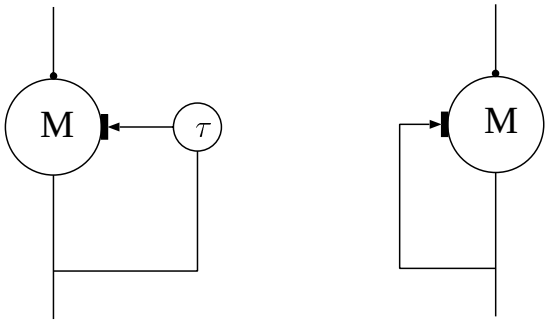
B.1 Without feedback delay

The perturbation expansion of the response is given by

$$x_\xi \;=\; z_0 + cz_1 + c^2 z_2 + \cdots \tag{B.1}$$

The DE describing the system is

$$\dot{m} = L_0 + c\ell_i(t) - ax + kxf(x) \tag{B.2}$$



(a) With feedback delay

(b) Without feedback delay

Figure B.2: Components of adaptive bandpass filter. τ is a linear delay element with a transfer function of the form $\frac{A}{s+A}$.

Substituting the Taylor series expansion for $f(x)$ and the perturbation expansion of x (Equation B.1) into Equation B.2 and equating coefficients gives.

$$\dot{z_1} = \ell - az_1 - kz_0z_1D(f) - kz1f(z_0) \tag{B.3}$$

$$\dot{z_2} = -az_2 - kz_0z_2D(f) - \frac{1}{2}kz_0z_1^2D_2(f) - kz_1^2D(f) - kz_2f(z_0) \tag{B.4}$$

The first order filter approximation is therefore

$$\begin{aligned} H_1(s) &= \frac{1}{a+X+B} \\ X &= a + kz_0 \\ B &= kz_0 \\ 0 &= L_0 - az_0 - kz_0^2 \end{aligned} \tag{B.5}$$

B.2 With feedback delay

The perturbation expansions of the response of the neuron and the delay element are given by

$$x_\xi = z_0 + cz_1 + c^2z_2 + \cdots \tag{B.6}$$

$$y_\xi = m_0 + cm_1 + c^2m_2 + \cdots \tag{B.7}$$

The DE describing the system is

$$\dot{m} = L_0 + c\ell_i(t) - ax + kxf(y) \tag{B.8}$$

$$\dot{y} = A(x - y)$$

Substituting the Taylor series expansion for $f(y)$ and the perturbation expansions into Equations B.8 and equating coefficients gives

$$\dot{z_1} = \ell - az_1 - kz_0 m_1 D(f) - kz_1 f(m_0) \quad \text{(B.9)}$$

$$\dot{z_2} = -az_2 - kz_0 m_2 D(f) - \frac{1}{2} kz_0 m_1^2 D_2(f) - kz_1 m_1 D(f) - kz_2 f(m_0) \quad \text{(B.10)}$$

The linear filter approximation is given by

$$H_2(s) = \frac{s + A}{s^2 + s(A + X) + A(X + B)} \quad \text{(B.11)}$$

B.3 Net Response

The transfer function of the system is given by the difference between these two approximations

$$\begin{aligned} H_{total}(s) &= \frac{sB}{(s^2 + s(A + X) + A(X + B))(s + X + B)} \quad \text{(B.12)} \\ X &= a + kz_0 \\ B &= kz_0 \\ 0 &= L_0 - az_0 - kz_0^2 \end{aligned}$$

This is a third order bandpass filter with parameters that are dependent on the mean luminance. The same analysis can also be used to describe the SUSTAINED neuron.

Appendix C

Simulation Techniques

All of the results described in this thesis were obtained using numerical simulations. This appendix will briefly discuss some of the techniques used.

The motion detectors and adaptive components described in Chapter 4 are constructed from linear filters and inhibitory neurons and can therefore be described by differential equations. Some of these equations are nonlinear. The differential equations describing the systems were solved numerically using a version of the adaptive Runge-Kutta technique described in [Press et al. 88]. The code was written in Ada and used floating point precision for all calculations. A timestep of 1ms was used for simulations in Chapter 4. All input and output data was stored in Matlab files and Matlab was used for visualisation purposes.

In the second part of the thesis that explores segmentation all calculations used fixed point numbers. The type used was defined in Ada as follows

```
type Fixed_Point is delta 2.0 **(-8) range -1023.0 .. +1023.0;
```

This was done to test the effect of limited precision. The motion detection layer was implemented differently to the previous set of simulations. The delay elements were implemented as digital filters and only the steady state versions of the inhibitory neurons were used. These simplifications demonstrated the practicality of the steady state version of the detector and improved the execution speed.

The remainder of the segmentation processing used conventional data structures. In the Voronoi thresholding the nearest salient point to each neighbour was represented as a pointer. A publicly available Ada lists package was used as the basis for all other data structures. This was not the most computationally efficient way of structuring the code, but was relatively simple.

The input image sequences were stored as 8 bit grayscale images in pgm format. (pgm is

an extremely simple format which stores each image pixel as an 8 bit binary value). These images were converted to fixed point format when read by the processing software. The fixed point results were converted to floating point matlab format for storeage and manipulation.

Bibliography

[Adelson and Bergen 85] E.H. Adelson & J.R. Bergen, "Spatiotemporal energy models for the perception of motion," *Journal of the Optical Society of America*, Vol. 2, No. 2, pp. 284–299, 1985. 14

[Adiv 85] G. Adiv, "Determining three-dimensional motion and structure from optical flow generated by several moving objects," *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 4, pp. 384–401, 1985. 80

[Ahuja and Tuceryan 89] N. Ahuja & M. Tuceryan, "Extraction of early perceptual structure in dot patterns : Integrating region, boundary, and component gestalt," *Computer Vision, Graphics, and Image Processing*, Vol. 48, pp. 304–356, 1989. 86, 95

[Anstis 80] S.M. Anstis, "The perception of apparent motion," *Phil. Trans. R. Soc. Lond. B.*, Vol. 290, pp. 153–168, 1980. 67

[Arnett 72] D.W. Arnett, "Spatial and temporal integration properties of units in the first optic ganglion of dipterans," *Journal of Neurophysiology*, Vol. 35, pp. 429–444, 1972. 13, 22, 37, 46

[Barlow and Levick 65] H.B. Barlow & W.R. Levick, "The mechanism of directionally selective units in the rabbit's retina," *J. Physiol., Lond*, Vol. 178, pp. 477–504, 1965. 11

[Beare and Bouzerdoum 96] R.J. Beare & A. Bouzerdoum, "A simple model of the SUSTAINED neuron," . In ANZIIS 96, 1996. 13, 46

[Beare et al. 95] R.J. Beare, A. Blanksby & A. Bouzerdoum, "Low level visual motion processing using local motion detectors," . In ICNN 95, volume 1, pp. 1–6, 1995. 30

[Blake et al. 93] A. Blake, R. Curwen & A. Zisserman, "A framework for spatio-temporal control in the tracking of visual contours," *International Journal of Computer Vision*, 1993. 80

[Bouzerdoum and Pinter 89] A. Bouzerdoum & R. B. Pinter, "Image motion processing in biological and computer vision systems," *SPIE Vol. 1199 Visual Communications and Image Processing IV*, pp. 1229–1240, 1989. 151

[Bouzerdoum and Pinter 93] A. Bouzerdoum & R. B. Pinter, "A neural network model for motion detection in the fly visual system," . In World Congress on Neural Networks, 1993. 12

[Bouzerdoum 91] A. Bouzerdoum. *Nonlinear lateral inhibitory neural networks analysis and application to motion detection*. PhD thesis, University of Washington, 1991. 152

[Bouzerdoum 93] A. Bouzerdoum, "The elementary movement detection mechanism in insect vision," *Phil. Trans. R. Soc. Lond. B*, Vol. 339, pp. 375–384, 1993. 39, 69

[Braddick 74] O. Braddick, "A short-range process in apparent motion," *Vision Research*, Vol. 14, pp. 137–151, 1974. 5

[Deriche and Faugeras 90] R. Deriche & O.D. Faugeras, "Tracking line segments," *Image and Vision Computing*, Vol. 8, No. 4, pp. 261–270, November 1990. 19, 77, 77

[Eckert 80] H. Eckert, "Functional properties of the H1-neurone in the third optic ganglion of the blowfly, *phaenicia*," *J. Comp. Physiol.*, Vol. 135, pp. 29–39, 1980. 70

[Egelhaaf and Borst 89] M. Egelhaaf & A. Borst, "Transient and steady-state response properties of movement detectors," *J. Opt. Soc. Am.*, Vol. 6, No. 1, pp. 116–127, 1989. 70

[Egelhaaf and Reichardt 87] M. Egelhaaf & W. E. Reichardt, "Dynamic response properties of movement detectors: Theoretical analysis and electrophysical investigation in the visual system of the fly," *Biol Cybern*, Vol. 56, pp. 69–87, 1987. 22

[Fleet and Jepson 89] D. J. Fleet & A. D. Jepson, "Hierarchical construction of orientation and velocity sensitive filters," *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 11, No. 3, pp. 315–325, 1989.

[Fodor and Pylyshyn 81] J.A. Fodor & Z.W. Pylyshyn, "How direct is visual perception? Some reflections on Gibson's "Ecological Approach"," *Cognition*, Vol. 9, pp. 139–196, 1981. 8

[Franceschini et al. 89] N. Franceschini, A. Riehle & A. Le Nestour, "Directionally sensitive motion detection by insect neurons," . In D. G. Stavenga & R. C. Hardie, editeurs, Facets of Vision, pp. 360–389. Berlin: Springer-Verlag, 1989. 22, 37

[Gibson 79] J. J. Gibson. The Ecological Approach to Visual Perception. Houghton Mifflin, Boston, 1979. 7

[Goodman 60] L.J. Goodman, "The landing responses of insects. the landing response of the fly *lucilia sericata*, and other calliphorinae.," *J. Exp. Biol.*, Vol. 37, pp. 854–878, 1960. 6

[Gruss et al. 91] A. Gruss, L.R. Carely & T. Kanade, "Integrated sensor and range-finding analog signal processor," *Journal of Solid State Circuits*, Vol. 26, No. 3, pp. 184–191, March 1991. 32

[Hartline and Ratliff 74] H.K. Hartline & F. Ratliff. Studies in excitation and inhibition in the retina. New York: Rockefeller University Press, 1974. 12

[Hassentein and Reichardt 56] B. Hassentein & W. Reichardt, "Functional structure of a mechanism of perception of optical movement," . In Proc. Int. Cong. Cybern, Namur, pp. 797–801, 1956. 9, 151

[Heeger 87a] D. Heeger, "Optical flow from spatiotemporal filters," . In Proceedings 1st International Conference on Computer Vision, pp. 181–190, 1987. 76

[Heeger 87b] D. J. Heeger, "Analyzing object motion based on optical flow," . In Proc 1th Int Conf Comp Vis, Lond, pp. 171–180, 1987. 16

[Helmholtz 24] H. L. F. Helmholtz. Treatise on physiological optics. [Rochester] : Optical Society of America, 1924. 5

[Heuer 93] H. Heuer, "Estimates of time to contact based on changing size and changing target vergence," *Perception*, Vol. 22, pp. 549–563, 1993. 6

[Hildreth 84] E. C. Hildreth, "The computation of the velocity field," *Proc. R. Soc. Lond. B.*, Vol. 221, No. 189-220, 1984. 79

[Horiuchi and Koch 96] T. Horiuchi & C. Koch, "Analog VLSI circuits for visual motion-based adaption of post-saccadic drift," . In MicroNeuro96, 1996. 50

[Horn and Schunck 81] B. K. P. Horn & B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, Vol. 17, pp. 185–203, 1981. 17, 75

[Horridge and Marcelja 90] G. A. Horridge & L. Marcelja, "Responses of the H1 neuron of the fly to jumped edges," *Phil. Trans. R. Soc. Lond. B*, Vol. 327, pp. 65–73, 1990.

[Horridge 91] G. A. Horridge, "Ratios of template responses as the basis for semivision," *Phil. Trans. R. Soc. Lond. B*, Vol. 331, No. 189-197, 1991. 18

[Isard and Blake 96] M. Isard & A. Blake, "Contour tracking by stochastic propogation of conditional density," . In Proc. European Conf on Computer Vision, pp. 343–356, 1996. 145

[Jain and Binford 91] R.C. Jain & T.O. Binford, "Ignorance, myopia, and naivete in computer vision systems," *CVGIP: Image Understanding*, Vol. 53, No. 1, pp. 112–117, 1991. 82

[Julesz 71] B. Julesz. Foundations of cyclopean perception. University of Chicago Press, 1971. 81

[Kouzani et al. 95] A. Z. Kouzani, A. Bouzerdoum & M. Liebelt, "Fuzzy motion estimation," *Australian Journal of Intelligent Information Processing Systems*, Vol. 2, No. 3, pp. 46–55, 1995. 19

[Kramer et al. 95] J. Kramer, R. Saroeshkar & C. Koch, "An analog VLSI velocity sensor," . In IEEE Int. Symp. Circuits and Systems, volume 1, pp. 413–416, 1995. 19

[Laughlin 87] S. Laughlin, "Form and function in retinal processing," *TINS*, Vol. 10, No. 11, pp. 478–483, 1987. 47, 52

[Laughlin 89] S. Laughlin, "The role of sensory adaption in the retina," *J. exp. Biol.*, Vol. 146, pp. 39–62, 1989.

[Moini et al. 93] A. Moini, A. Bouzerdoum, A. Yakovleff, D. Abbot, Kim O., K. Eshragian & R. E. Bogner, "An analog implementation of early visual processing in insects," . In International Symposium on VLSI technology,, pp. 283–287, 1993.

[Moini et al. 97] A. Moini, B. Bouzerdoum & K. Eshraghian, "A current mode implementation of shunting inhibition," . In ISCAS'97, June 9-12, Hong Kong, 1997. 42

[Moini 97] A. Moini. Vision chips, or seeing silicon. http://www.eleceng.adelaide.edu.au /Groups/GAAS/Bugeye/visionchips/index.html, 1997. 22

[Murray and Buxton 87] D.W. Murray & B.F. Buxton, "Scene segmentation from visual motion using global optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, No. 2, pp. 220–228, March 1987. 80

[Nagel and Enkelmann 85] H. H. Nagel & W. Enkelmann, "An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 565–593, 1985. 76

[Nakayama 85] K. Nakayama, "Biological image processing: a review," *Vision Res.*, Vol. 25, No. 5, pp. 625–660, 1985. 18

[Nguyen et al. 96] X. T. Nguyen, A. Bouzerdoum & R. E. Bogner, "Backward tracking of motion trajectories for velocity estimation," . In 1996 Australian and New Zealand Conference on Intelligent Information Systems (ANZIIS-96), pp. 338–341, November 1996. 18

[Nguyen 96] X.T. Nguyen. *Smart VLSI Micro-Sensors for Velocity Estimation inspired by Insect Vision*. PhD thesis, The University of Adelaide, 1996. 78

[Öğmen and Gagné 90a] H. Öğmen & S. Gagné, "Neural models for sustained and on-off units of insect lamina," *Biological Cybernetics*, Vol. 63, pp. 51–60, 1990. 37

[Öğmen and Gagné 90b] H. Öğmen & S. Gagné, "Neural network architectures for motion perception and elementary motion detection in the fly visual system," *Neural Networks*, Vol. 3, pp. 487–505, 1990. 12

[Pinter et al. 90] R.B. Pinter, D. Osorio & M.V. Srinivasan, "Shift of edge-taxis to scototaxis depends on mean luminance and is predicted by a matched filter theory on the responses of fly lamina LMC cells," *Visual Neuroscience*, Vol. 4, pp. 579–584, 1990.

[Pinter 83] R.B. Pinter, "The electrophysiological basis for linear and for nonlinear product term lateral inhibition and the consequences for wide field textured stimuli," *Journal of Theoretical Biology*, Vol. 105, pp. 233–243, 1983. 12

[Poggio et al. 85] T. Poggio, V. Torre & C. Koch, "Computational vision and regularization theory," *Nature*, Vol. 317, No. 26, pp. 314–319, 1985.

[Press et al. 88] W. Press, B. Flannery, S. Teukolsky & W. Vetterling. Numerical recipies in C. The art of scientific computing. Cambridge University Press, 1988. 163

[Reichardt and Poggio 79] W. E. Reichardt & T. Poggio, "Figure-Ground Discrimination by relative movement in the visual system of the fly. Part 1: Experimental Results," *Biol Cybern*, Vol. 35, pp. 81–100, 1979. 80

[Reichardt et al. 83] W. E. Reichardt, T. Poggio & K. Hausen, "Figure-ground discrimination by relative movement in the visual system of the fly," *Biol Cybern*, Vol. 46 (Suppl), pp. 1–30, 1983.

[Reichardt et al. 88] W. E. Reichardt, R. W. Schlogl & M. Egelhaaf, "Movement detectors provide sufficient information for the local computation of 2-D velocity field," *Naturwissenschaften*, Vol. 75, pp. 313–315, 1988. 79

[Reichardt 61] W. E. Reichardt, "Autocorrelation, a principle for the evaluation of sensory information by the nervous system," *Sensory Communication*, pp. 303–317, 1961. 27

[Schmid and Bulthoff 88] A. Schmid & H. Bulthoff, "Using neuropharmacology to distinguish between excitatory and inhibitory movement detection mechanisms in the fly *calliphora erythrocephala*," *Biol. Cybern.*, Vol. 59, pp. 71–80, 1988. 11

[Sha'ashua 88] A. Sha'ashua. Structural saliency : the detection of globally salient structures using a locally connected network. Master's thesis, Weizmann Institute of Science, 1988. 86

[Smith and Brady 95] S.M. Smith & J.M. Brady, "ASSET-2: Real-time motion segmentation and shape tracking," *IEEE transactions on pattern analysis and machine intelligence*, Vol. 18, No. 8, pp. 814–820, 1995. 77

[Smith 92] S. Smith. *Feature based image sequence understanding*. PhD thesis, Department of Engineering Science, University of Oxford, 1992. 117, 119

[Snippe and Koenderink 94] H. P. Snippe & J. J. Koenderink, "Extraction of optical velocity by use of multi-input Reichardt detectors," *J. Opt. Soc. Am*, Vol. 11, No. 4, pp. 1222–1236, 1994. 17

[Srinivasan et al. 82] M. V. Srinivasan, S. B. Laughlin & A. Dubs, "Predictive coding: a fresh view of inhibition in the retina," *Proc. R. Soc. Lond. B*, Vol. 216, pp. 427–459, 1982. 25

[Srinivasan et al. 90] M. V. Srinivasan, R. B. Pinter & D. Osorio, "Matched filtering in the visual system of the fly: large monopolar cells of the lamina are optimized to detect moving edges and blobs," *Proc. R. Soc. Lond. B*, Vol. 240, pp. 279–293, 1990.

[Srinivasan 76] M. V. Srinivasan, "A proposed mechanism for multiplication of neural signals," *Biological Cybernetics*, Vol. 21, pp. 227–236, 1976. 37, 52

[Srinivasan 77] M.V. Srinivasan, "A visually evoked roll response in the housefly," *Journal of Comparative Physiology*, Vol. 119, pp. 1–14, 1977. 8

[Srinivasan 90] M. V. Srinivasan, "Generalized gradient schemes for the measurement of two-dimensional image motion," *Biol. Cybern.*, Vol. 63, pp. 421–431, 1990. 18, 76, 79

[Ullman 92] S. Ullman, "Low-level aspects of segmentation and recognition," *Phil. Trans. R. Soc. Lond. B*, Vol. 337, pp. 371–379, 1992.

[van Hateren 92] J.H. van Hateren, "Theoretical predictions of spatiotemporal receptive fields of fly LMCs, and experimental validation," *J. Comp. Physiol A*, Vol. 171, pp. 157–170, 1992. 26, 45

[van Santen and Sperling 85] J. P. H. van Santen & G. Sperling, "Elaborated Reichardt Detectors," *J. Opt Soc Am*, Vol. 2, No. 2, pp. 300–321, 1985. 13, 17

[Watson and Ahumada 85] A. B. Watson & A. J. Ahumada, "Model of human visual-motion sensing," *J. Opt Soc Am*, Vol. 2, No. 2, pp. 322–342, 1985. 14

[Zihl et al. 83] J. Zihl, D. von Cramon & N. Mai, "Selective disturbance of movement vision after bilateral brain damage," *Brain*, Vol. 106, pp. 313–340, 1983. 8