

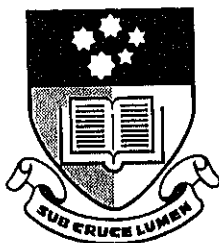
Smart VLSI Micro-Sensors for Velocity Estimation inspired by Insect Vision

by

Xuan Thong Nguyen B.E. (Hon. I), B.Sc.

Thesis submitted for the degree of

Doctor of Philosophy



The University of Adelaide

Faculty of Engineering

The Department of Electrical and Electronic Engineering

March 1996

Awarded 1996

Contents

<i>Abstract</i>	<i>viii</i>
<i>Statement of Originality</i>	<i>ix</i>
<i>Acknowledgements</i>	<i>x</i>
<i>Publications</i>	<i>xi</i>
<i>List of Abbreviations</i>	<i>xiii</i>
<i>List of Figures</i>	<i>xiv</i>
<i>List of Tables</i>	<i>xxii</i>
Chapter 1 Introduction	1
1.1 Background and Motivation	1
1.1.1 Insect Vision	1
1.1.2 Neuromorphic Engineering and Smart Sensors	2
1.1.3 Summary	4
1.2 Research Methodology	4
1.3 A Guided Tour of the Thesis	7
1.4 Contributions	9
1.5 Intended Audience	10
PART 1: ANALYSIS	11
Chapter 2 Visual Motion Computation - State of The Art	12
2.1 Introduction	12
2.2 Motion for structure and Motion Measurement	14
2.2.1 Gradient Models	15
2.2.2 Region-based Matching Models	17

2.2.3	Spatio-temporal Energy Models	18
2.2.4	Phase-Based Models	19
2.2.5	Limitations in Real-Time Applications	20
2.3	Purposive Vision and Motion Detection	22
2.4	From Insect Vision to Machine Vision	24
2.4.1	Task-Oriented Vision	24
2.4.2	Simple Circuitry and Automatic Gain Control	25
2.4.3	Principle of Parallel Analog Processing	25
2.4.4	Principle of Visuomotor Control	26
2.5	Summary	26
Chapter 3	The Insect Visual System and Motion Detection	28
3.1	Introduction	28
3.2	The Insect Visual System	29
3.2.1	The Lamina	29
3.2.2	The Medulla	31
3.2.3	The Lobula Complex	31
3.3	Principles of Visual Motion Detection: Elementary Motion Detector	32
3.4	Neural Models for Early Visual Motion Detection	34
3.4.1	Correlation Model	34
3.4.2	Variants of the Correlation Model	36
3.4.3	Shunting Lateral Inhibition Model	37
3.4.4	Spatial and Temporal Aliasing	38
3.5	Summary	39
Chapter 4	Template Model and Its Directional Selectivity	41
4.1	Introduction	41
4.2	Template Model	42
4.3	Temporal Aliasing	43

4.4	Spatial Aliasing	44
4.5	Directional Selectivity	44
4.5.1	Early Interpretation of Directional Selectivity	44
4.5.2	Observations	45
4.5.3	Derivation of Directional Selectivity Function	46
4.6	Directionally Motion-Sensitive Templates	49
4.6.1	Templates Sensitive to Coherent Motion	50
4.6.2	DMST Characterisation	52
4.6.3	Other Templates Possessing a Non-Zero Determinant	54
4.7	Response to Moving Objects	57
4.7.1	Examples	60
4.8	Discussion	60
Chapter 5 Feature Representation of Motion Trajectories		61
5.1	Introduction	61
5.2	Spatio-Temporal Approach	62
5.3	Motion as Orientation in the Spatio-Temporal Domain	63
5.4	Motion Paths under the Template Model	64
5.5	Feature Representation of Motion Trajectories	66
5.5.1	Templates on Motion Trajectories	66
5.5.2	Scheme 1: Staircase Motion Trajectories using DMSTs and PCTs	69
5.5.3	Scheme 2: Motion Trajectories using DMSTs as Features	70
5.5.4	Advantages	71
5.6	Simulation Results	72
5.6.1	One Moving Object	72
5.6.2	Two Moving Objects Occluding One Another	77
5.6.2.1	Crossing	77
5.6.2.2	Overtaking	79

5.7	Examples from Real-Time Experiments	81
5.8	Summary	83
Chapter 6	Velocity Estimation	84
6.1	Introduction	84
6.2	Approaches for Velocity Estimation	85
6.3	Tracking Algorithms for Velocity Estimation	86
6.3.1	Motivation	86
6.3.1.1	Efficiency in Processing Time	86
6.3.1.2	Simplicity in Hardware Realization	86
6.3.2	General Description	86
6.3.3	Search Space	88
6.3.4	Stair-Step Tracking (SST) Algorithm	89
6.3.5	Backward Tracking (BT) Algorithm	91
6.3.6	Forward Tracking (FT) Algorithm	93
6.4	Simulation Results	94
6.4.1	Uniform Motion	95
6.4.1.1	Error Analysis	95
6.4.1.2	Results	95
6.4.2	Non-Uniform Motion	100
6.4.2.1	Error Analysis	100
6.4.2.2	Results	103
6.4.3	Two Moving Objects Occluding One Another	108
6.5	A Comparison	112
6.5.1	Robustness	112
6.5.2	Searching Efficiency and Strategy	113
6.5.3	Hardware Architecture Complexity	114
6.6	Conclusion	115

PART 2:	IMPLEMENTATION	117
Chapter 7	Smart Micro-Sensor I	118
7.1	Introduction	118
7.2	General Description	119
7.2.1	Functional Architecture	119
7.2.2	Choice of Technology	120
7.3	VLSI Implementation	121
7.3.1	System Architecture	121
7.3.2	Timing	122
7.3.3	Intensity Change Detection Stage	123
7.3.4	Template Interpretation Stage	126
7.3.5	Velocity and Position Estimation Stage	126
7.4	Testing	129
7.4.1	Testing Procedure	129
7.4.2	Interpretation of Template Formulation	130
7.4.3	Real-Time Experiments	132
7.5	Summary	137
Chapter 8	Smart Micro-Sensor II	138
8.1	Introduction	138
8.2	General Description	139
8.3	Timing	140
8.4	Front-End Processor	141
8.5	Template Processor	143
8.6	Velocity Estimation Processor	145
8.6.1	Processor Controller	147
8.6.2	Matching Module	147
8.6.3	Distance-Travelled Module	149

8.6.4	Time-Taken Module	151
8.7	Comparison between the Designs of the two Sensors	152
8.8	Testing	154
8.8.1	Front-End Processor	154
8.8.2	Template and Velocity Estimation Processors	157
8.8.3	Velocity Estimation	158
8.8.3.1	Apparatus	158
8.8.3.2	Results	159
8.9	Summary	162
Chapter 9	Conclusions	163
9.1	Thesis Summary	163
9.2	Ongoing Works and Investigations	165
Appendix A	Optical Interface	166
A.1	Introduction	166
A.2	Panoramic Machine Vision	167
A.3	Requirements for An Optical Interface	167
A.4	Gradient Index Lens	168
A.4.1	Basic Imaging Properties of GRIN-Rod Lenses	168
A.4.2	Aberrations of GRIN-Rod Lenses	170
A.5	Optical Resolution	170
A.6	Physical Layout	171
A.7	Summary	173
Appendix B	Software Simulation	174
Appendix C	Responses of Directionally Motion-Sensitive and Position Conjugate Templates	177

Appendix D Hough Transform for Velocity Estimation	181
D.1 Fundamentals of Hough Transform	181
D.2 Velocity Estimation using Hough Transform	183
Appendix E Quantization Errors	185
Bibliography	188

Abstract

Motion detection is a challenging problem in machine vision. If motion is successfully detected, the obtained information can be potentially utilised in many real-time applications. However, existing techniques for motion detection, like optical flow computation, and optical motion detectors, such as CCD camera, are far from being useful in real-time applications due to some major difficulties such as the computational bottleneck. To improve the real-time applications of visual motion detectors a new approach called *smart sensing* has been pursued where VLSI smart micro-sensors have been designed for the specific task of motion detection.

In this thesis, the smart sensing paradigm is exploited in two aspects. Firstly, insect vision principles have been applied to the main mechanism for motion detection. Secondly, advanced VLSI technologies have been employed for designing *smart micro-sensors* in which the imager and processor are integrated into one monolithic device.

A neural model for motion detection, named *template model*, which consists of interpreting local changes in intensity in order to infer directional motion information, has been studied and analysed. The analysis consolidates the validity of the model with the support of electrophysiological evidence from the study of insect vision, and of a mathematical formulation. This formulation and a heuristic investigation are together employed to identify responses due to real coherent motion, called *directionally motion-sensitive templates*. These templates can be used as features for representing motion trajectories in the spatio-temporal domain. To exploit the benefits of using the feature representation of motion trajectories without costly computational and hardware implementation phases, novel tracking algorithms to estimate the relative velocities of moving stimuli have been developed.

In an attempt to transfer knowledge on insect vision into hardware, particularly the template model, advanced VLSI technologies have been used to build a first concept-demonstrating and then a second test-facilitating sensor. Results obtained by testing the sensors in real time confirm that the sensors function correctly as expected from the simulation including the responses of the directionally motion-sensitive templates and the relative errors achieved from velocity estimation. However, further implementation works are needed to integrate tested modules into one device and to make it available to a wide range of applications.